Eclasess.com

Training & Placement

Azure Data Engineering

- Data Factory
- Data Lake
- Data Bricks-PySpark
- Synapse DW
- Azure SQL

Address: Manjeera square, Block-201, 2nd Flore, By Ln, opposite to Prime Hospital, Ameerpet, Hyderabad
Call: 7997457228/9391463856

ECLASESS.COM

INDEX

INTRODUCTION TO MICROSOFT AZURE

- Introduction to Microsoft Azure
- what is Cloud computing
- what are services offering Microsoft azure
- Advantages of cloud computing
- Types of cloud deployments
- Types of cloud Services
- Microsoft Azure vs AWS
- How to Create Azure Account
- Azure Architecture & Services
- How to Create Resources and Resource Groups in Microsoft Azure
- Azure Resource Manager vs. classic deployment
- How to create Virtual Machine

What are Storage Accounts in Azure

- Blob
- File
- Queues
- Tables

Azure SQL Databases

- What is Azure SQL
- Different deployments of SQL in Azure
- How to create Azure SQL Server
- How to Create Database in Azure SQL Server
- How to Query Azure SQL Server from On-premises SSMS and Cloud
- What is Poly base in Azure SQL
- What is External tables
- What are limitations compared On-premises to Cloud SQL Server

ECLASESS.COM

- How to Load Data from On-premises to Cloud SQL using data factory
- Migration of Existing Data from on premises to SQL Azure

What is Azure Data ware House

- Compare SQL Database vs Azure DW
- How to Create Azure DW
- How to cross query data in DW
- How to Query Azure SQL DW from On-premises SSMS and Cloud
- How to Load Data to Azure DW using data Factory from Azure SQL
- How to Load Data from on premises to Cloud SQL DW using data factory

Azure Analysis Services

- What is Azure Analysis Services Tabular Model
- How to create Azure Analysis Service Tabular model project
- What is Workspace database
- Import Data
- Direct Query Mode
- Tables and Columns
- Relationships
- Calculations
- Measures
- KPIs
- Hierarchies
- Partitions
- Perspectives
- Translations
- Roles
- Deployment of Tabular Project
- Migration of Tabular Project

ECLASESS.COM

Azure Data Factory

- Azure Data Factory vs SSIS
- Linked Services
- Data Sets
- Pipelines
- Parameters
- Copy Data
- Monitor Manage
- Author and Deploy
- Different kinds of integration runtimes
- How to create pipelines from template
- How to configure different Integration Runtimes
- Azure Integration runtime
- Azure Self Hosted Integration Runtime
- SSIS Integration runtime.

Move & transform

- Copy Data

General activities

- Append Variable
- Execute Pipeline
- Execute SSIS Package
- Get Metadata
- Lookup
- Stored procedure
- Set Variable
- Delete
- Wait

ECLASESS.COM

- · Until

Iteration & Conditionals

- · Filter
- · For Each
- · If Condition
- · Until
- What are ARM Templates using for Deployment
- How to Deploy Data Factory Pipelines DEV, Test, Prod environment
- Version Control and Code Repository in Git Hub
- Debugging and Monitoring Pipelines
- Error handling and Logging error records
- Lift and Shift SSIS packages in to Azure Data Factory

Data Lake

- What is Data Lake
- How to store Data into data lake
- How to Query data from Data Lake to SQL Server.
- Introduction to Azure Data Lake U-SQL Batch Job
- Data Lake Analytics
- What is HD-insight
- U SQL

Data bricks

- What is Data bricks
- Configure data bricks
- Read data from data bricks by creating clusters

Power BI

- What is power BI
- How to create Reports in Power BI desktop Using Azure Data sources
- Publish Reports into power BI cloud

ECLASESS.COM

What is cloud?

Cloud computing is renting and accessing resources through internet browser, like storage space or CPU, software's and hardware's running on another company's computers. You only pay for what you use. The company providing these services is referred to as a cloud provider.

Some example providers are Microsoft, Amazon, and Google.

The cloud provider is responsible for the physical hardware required to execute your work, and for keeping it up-to-date. The computing services offered tend to vary by cloud provider. However, typically they include:

- **Compute power** - such as Linux servers or web applications
- **Storage** - such as files and databases
- **Networking** - such as secure connections between the cloud provider and your company
- **Analytics** - such as visualizing telemetry and performance data

Every business has different needs and requirements. Cloud computing is **flexible** and **cost-efficient**, which can be beneficial to every business, whether it's a small start-up or a large enterprise.

What is on premises?

Setting up all applications, infrastructures and hardware's everything in your premises servers is called on premises.

An on premise setup requires in-house server hardware, software licenses, integration capabilities, and IT employees on hand to support and manage potential issues that may arise

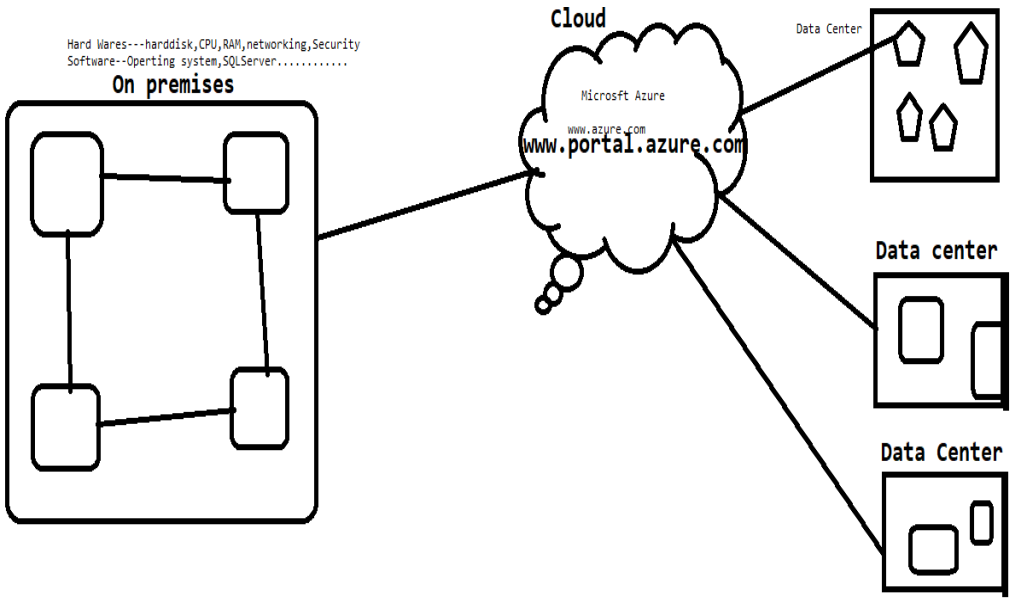
On-premise software requires that an enterprise purchases a license or a copy of the software to use it. Because the software itself is licensed and the entire instance of software resides within an organization's premises.

Disadvantages:

- They are responsible for the ongoing costs of the server hardware, power consumption, and space
- Maintenance of applications
- Initial capital investment is required to purchase all servers and applications.
- They are responsible for any system upgrades and migrations.

ECLASESS.COM

What is on-premises and cloud?



Why we need to move from on premises to cloud?

We have 8 Reasons to Move On-Premises Software to the Cloud

1. **Drive IT Costs Down**

A cloud-based infrastructure reduces capital expenditures and recurring maintenance costs. It allows you to scale data storage up or down to meet customer demand. You no longer need to worry about large upfront software and hardware costs.

2. **Increase Security in the Cloud**

Advanced security capabilities like improved data center security, password management, rigorous VPN features.

3. **Reach Wider Markets**

Offering your products via the cloud lets you distribute to the global market for a fraction of the cost of traditional sales and marketing efforts. It sends a message to your customer base that you're committed to staying relevant.

ECLASES-

4. Accelerate Deployments

With your software in the cloud, performance no longer dependent on physical data centers. New software can be ready to launch in seconds rather than months. Users get immediate access to the full version of your software.

5. Increase Customer Satisfaction

Customers can use your product with minimal assistance. They get accelerated access to new features instead of having to wait for them to be rolled out. Cloud software also enables end users to work anywhere, from any device.

6. Simplify the Sales Process

The cloud deployment sales model reduces barriers to entry with pay-as-you-go pricing. Customers pay for what they need, when they need it. They get fast implementation without a big upfront licensing fee. SaaS applications also tend to have higher adoption rates because they are accessible via familiar web browsers.

7. Spend More Time on Innovation

A SaaS platform frees you from a never-ending stream of infrastructure upgrades so you can focus on innovating new products. In the cloud you can develop and test prototypes and release new applications and features on the fly without having to purchase costly new infrastructure.

- **Microsoft is Incentivizing Customers to Migrate to Azure**

Microsoft recently implemented an incentive program to help ISVs grow their business. If you deploy your applications on Azure and make them available on the App Source marketplace, you will receive access to Microsoft tools like Power Apps and Flow at no additional charge.

Microsoft will also send you the contact information of users that install your app through App Source. Those leads can go straight into your CRM so your sales team can follow up effectively.

ECLASESS.COM

How to Create Azure Account

Azure Account:

Azure Account is required that gets you to start access to Azure services and your Azure subscriptions.

Azure Subscriptions:

- Azure Subscriptions are used for billing information of services used in Azure Account.
- Many subscriptions can be created under one azure account.
- A company having multiple departments, they need different subscriptions to purchase services
- And implementing projects.
- IT makes more flexible at end of financial year, Company how much spent on each department
- For IT development.
- You can apply security policies and access to users on subscription level.

Azure Resource Group:

- Resource group is grouping of all resources and storing Meta data information about all resources created and mapped under this resource group.
- It is easy to deploy all resource mapping under one resource group.
- It is easy to provide access policies on resource group instead of individual resources
- It is easy to delete all resources under resource group at a time.

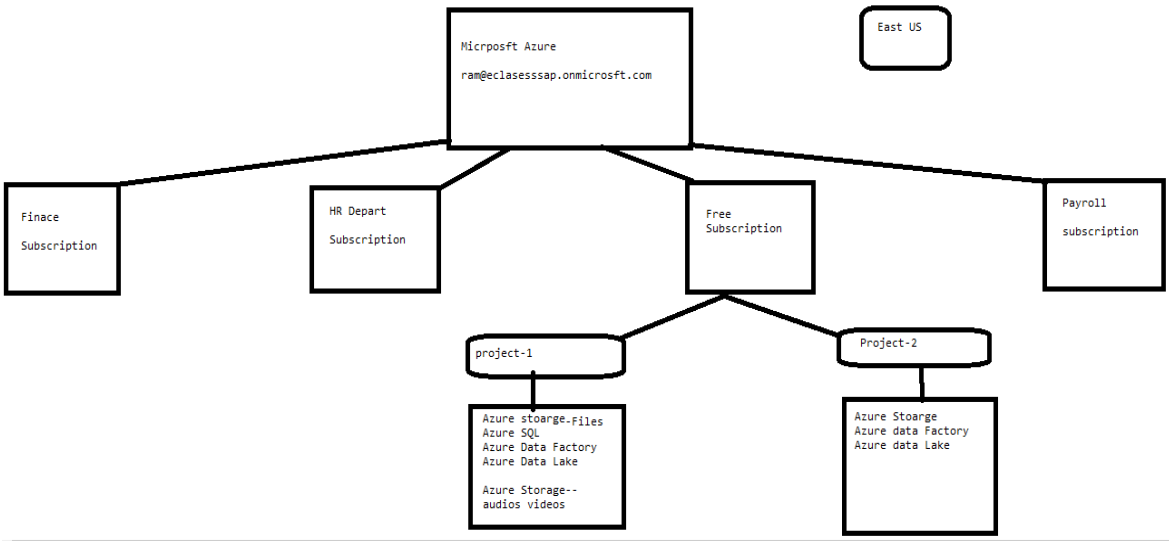
Azure Resources:

- Azure having so many services, after creating service under a resource group
- It is called as a resource.
- Whenever we are creating any resource needs to specify
- Subscription Name
- Resource Group Name
- Resource Name
- Location

ECLASESS.COM

EX Azure resource is: Azure Storage Account,

Azure Account□Subscriptions□Resource Group Manager□Resources



How to Create Azure Free Subscription.

Azure Free subscription benefits:



- Free access of 25+ services 1 year

- Free Access of 25+ more services life Long
- Providing 200 $ USD credits to utilize with in first 30 days for using other azure services.
- After 30 days your account free subscription will be disabled automatically
- Within 30 days they don't charge anything from your credit card.
- After 30 Days account will disabled, if you want to use same account needs to upgrade account to pay as go account. Then azure will start charging from your credit card.

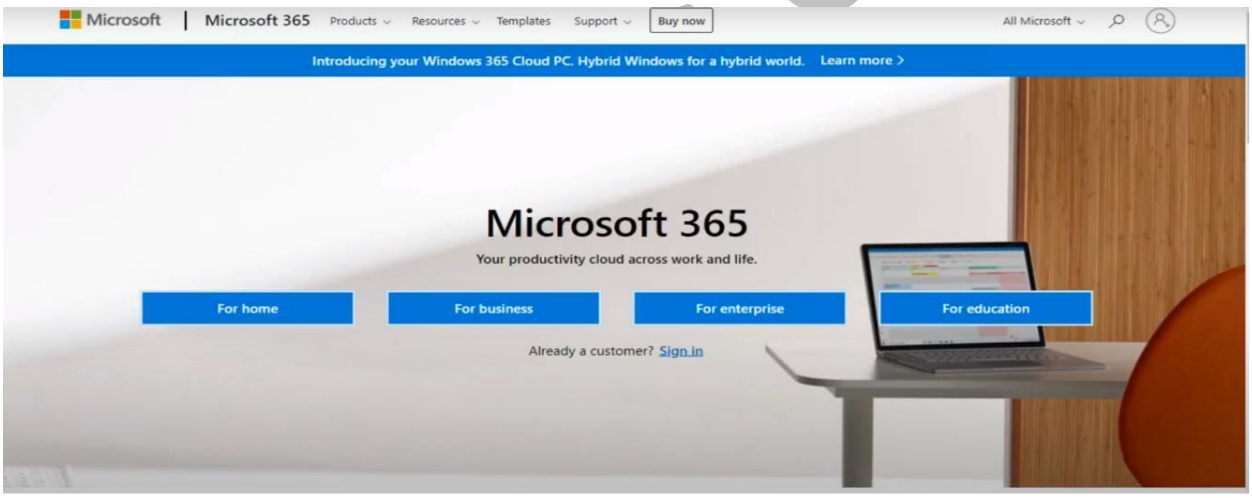
Prerequisites:

- Needs to create an Microsoft email in www.office365.com
  Ex: ram@eclasessdev.onmicrosoft.com

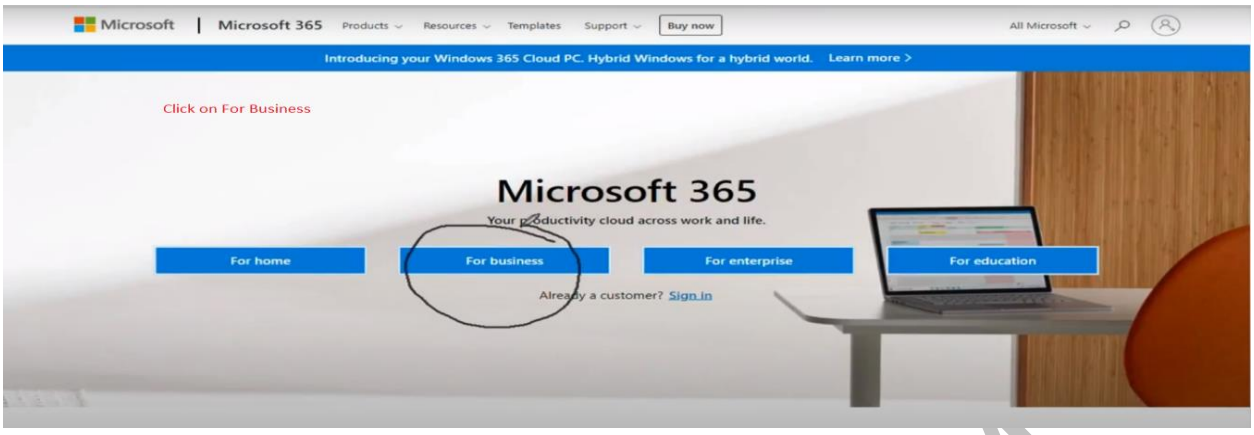
  Note:

  You can use Commercial mails also Gmail, Yahoo, Organization mails
  But few services like Azure Gateways Commercial Mails are not supporting
- Need credit card which ever not used for any free subscription earlier.

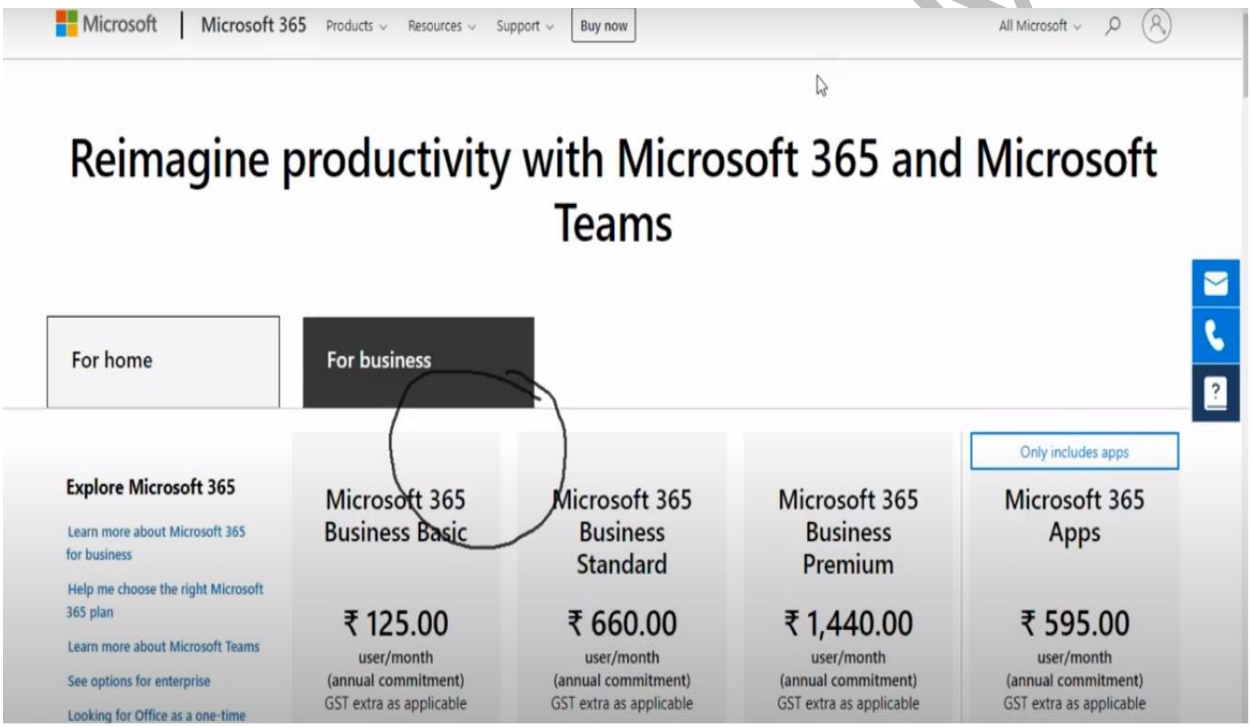
Step-1: Go to Web browser www.office365.com.



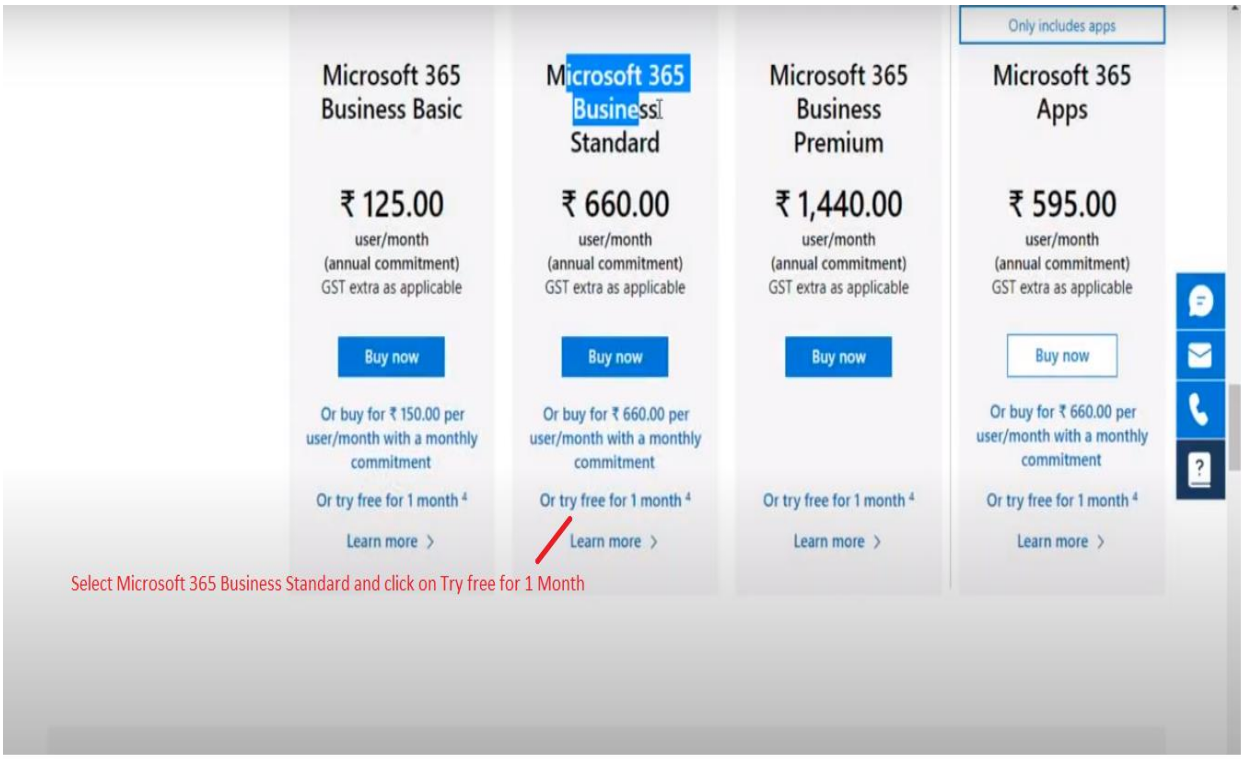
Step-2: Select For Business



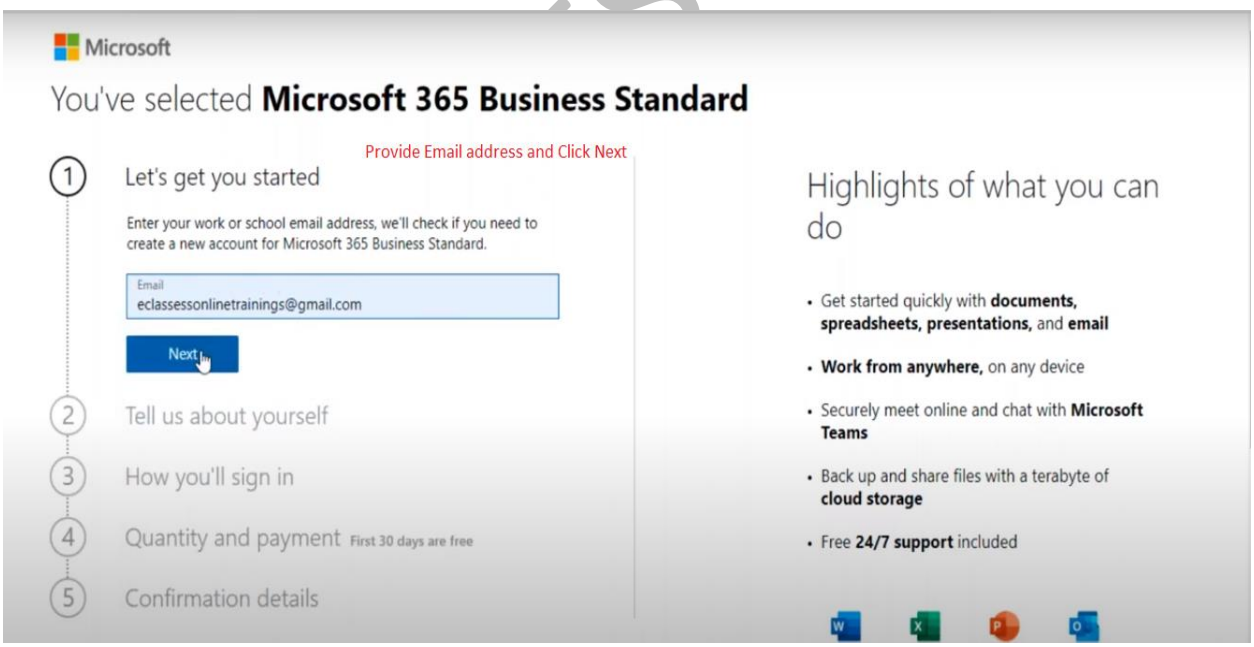
Step-3: Scroll down and go to Microsoft 365 Business standard



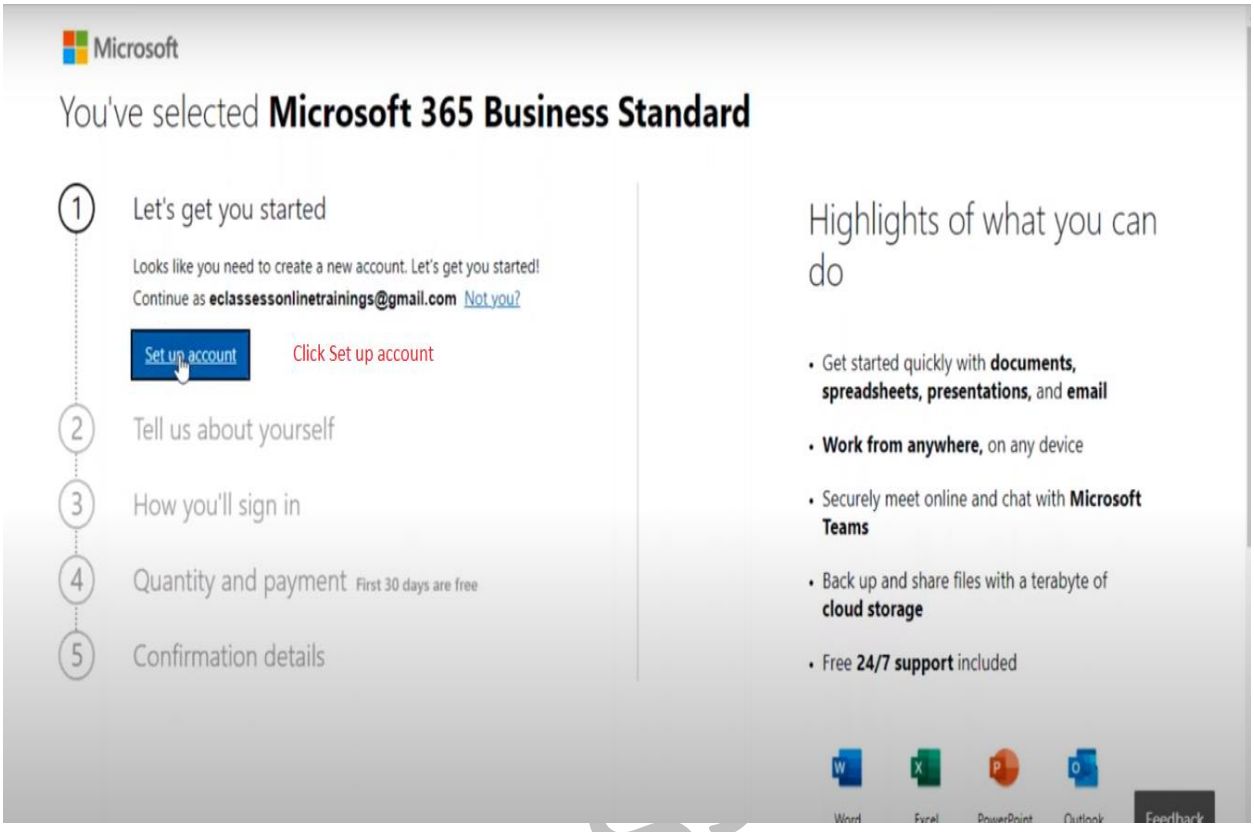
Step-4: Select Try free for 1 month under Microsoft 365 Business Standard.



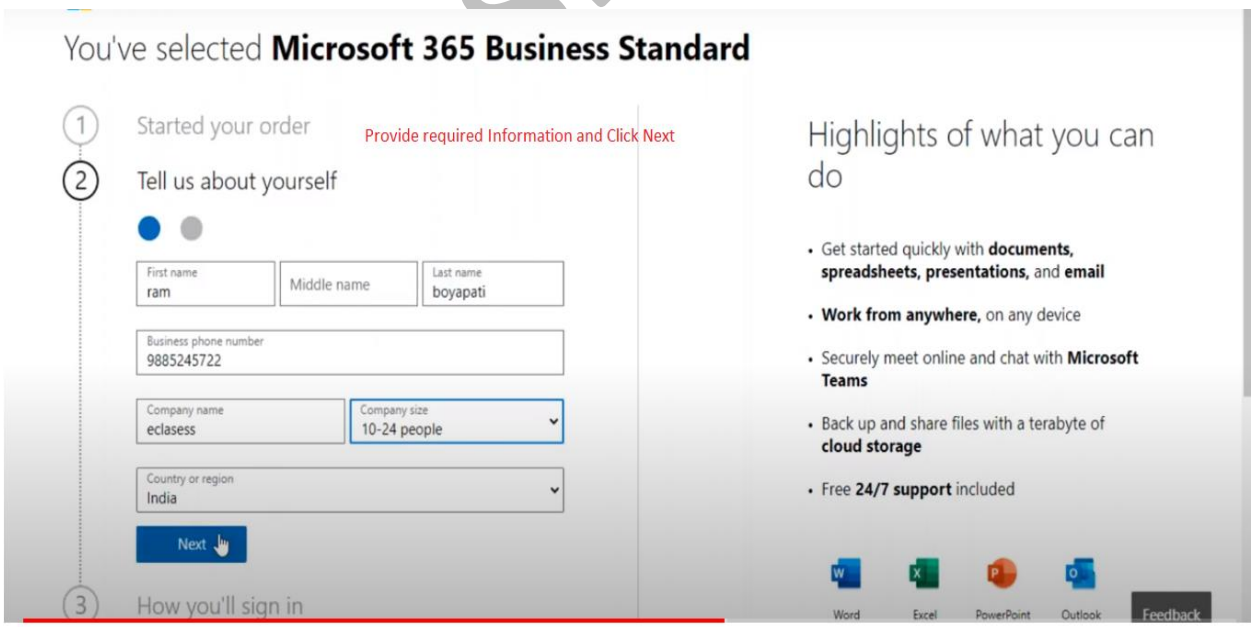
Step-5: Provide your personal Gmail address.



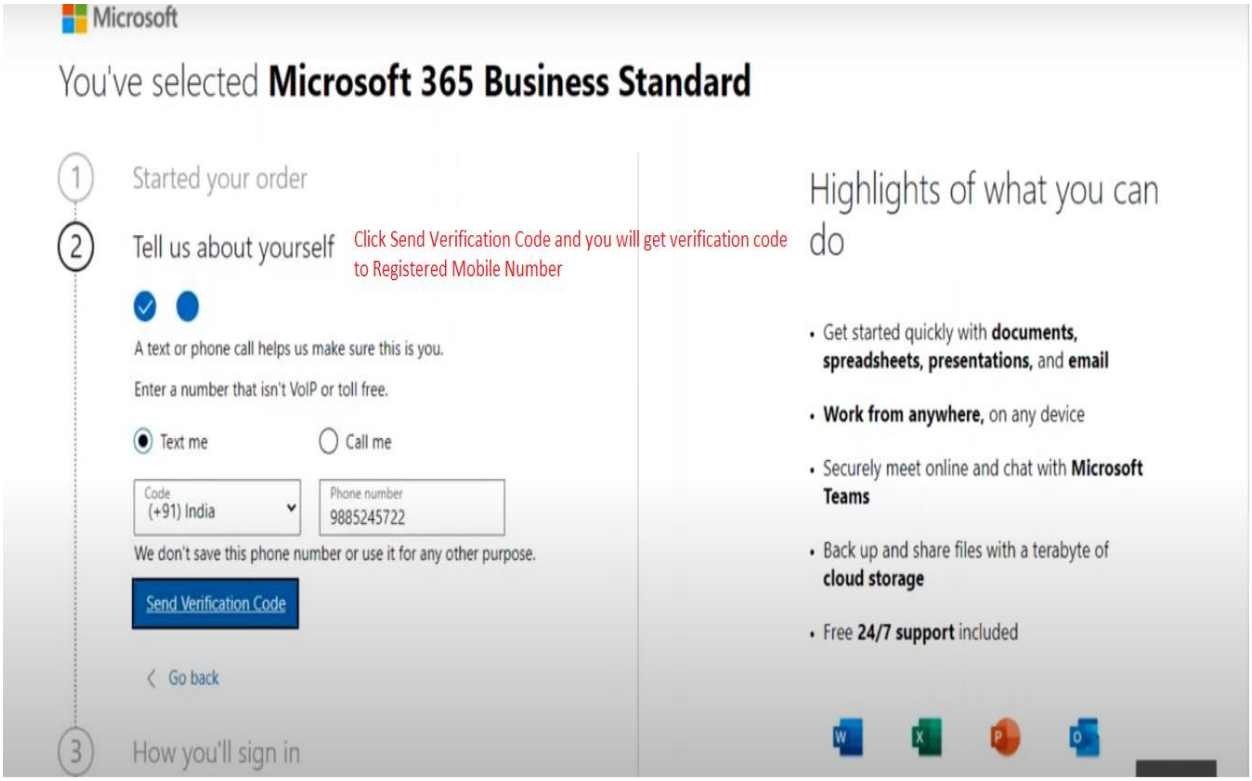
Step-6: Click on Set up Account.



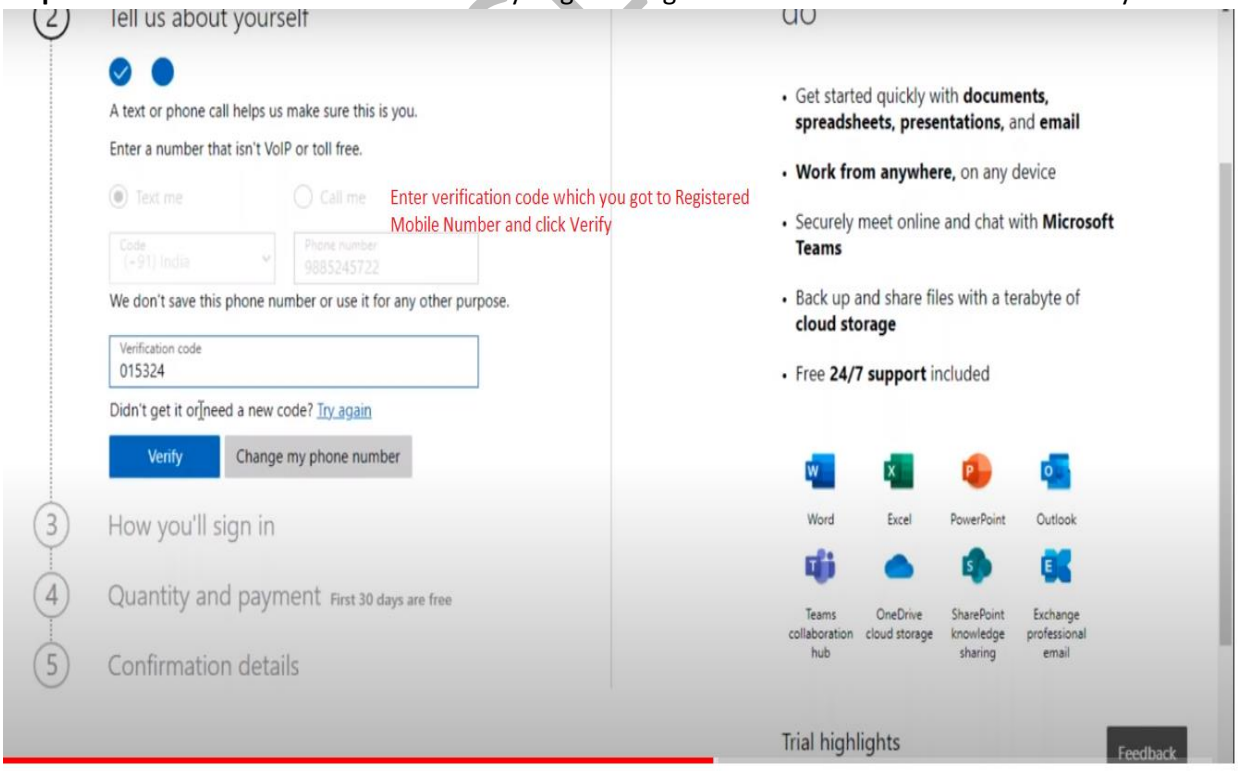
Step- 7: Provide required information and click next.



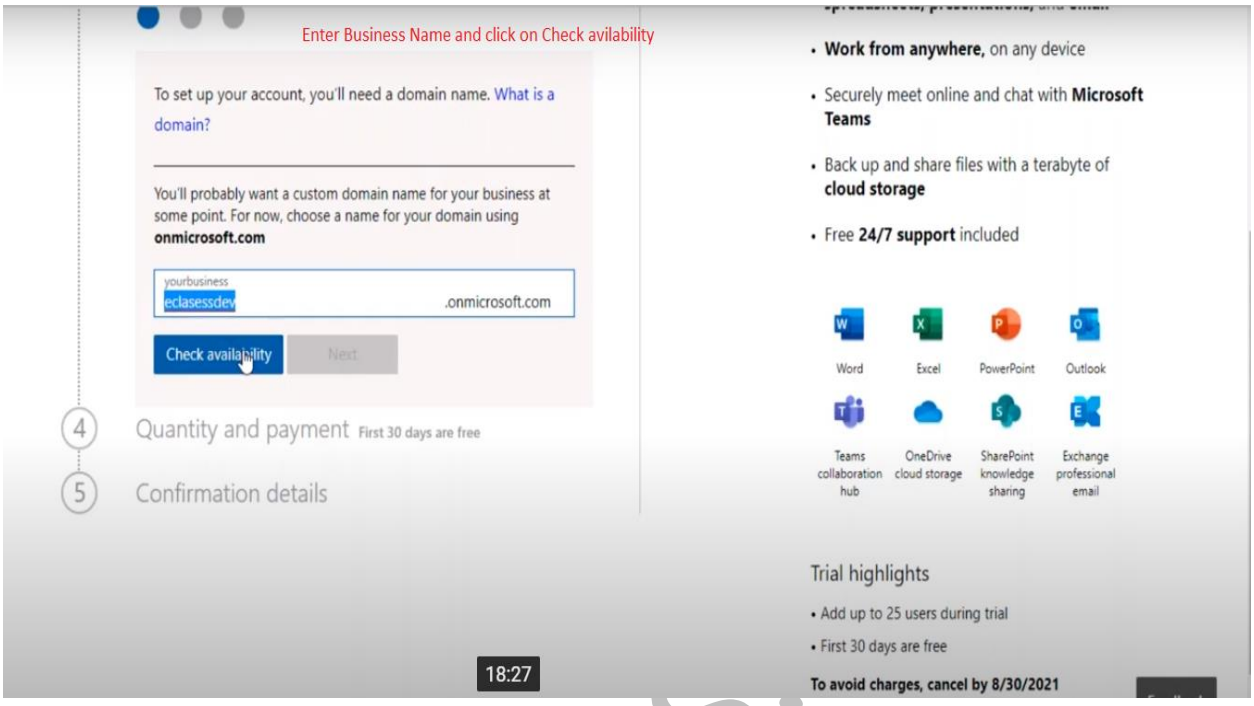
Step-8: Enter your mobile number to get OTP and click send verification code.



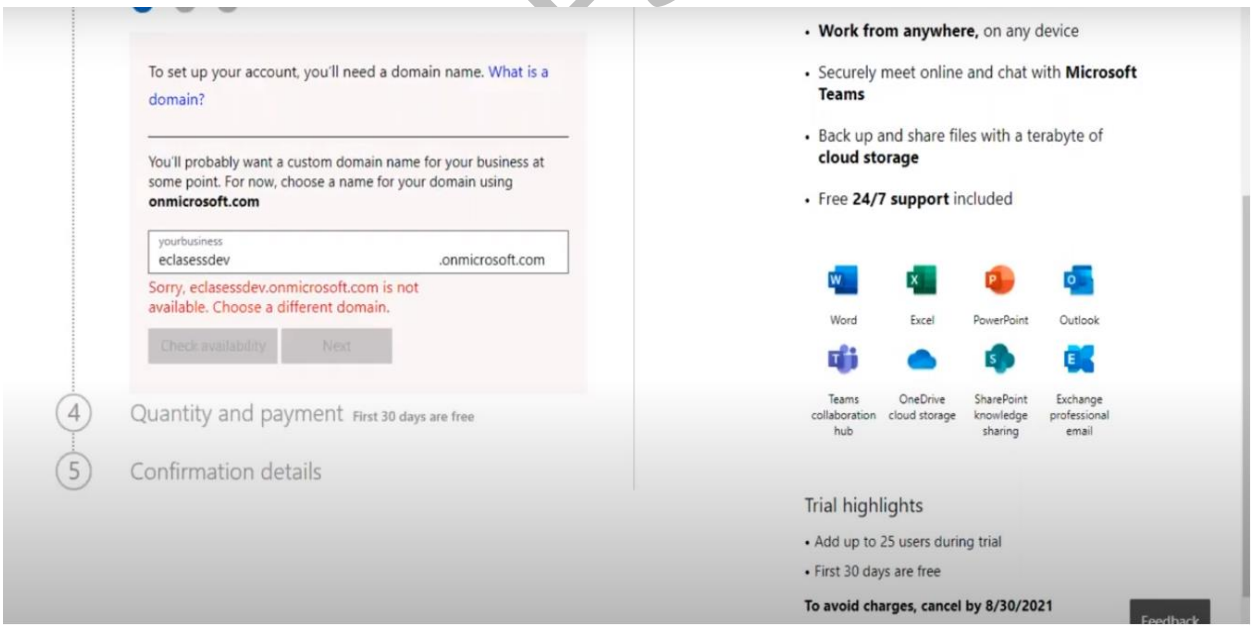
Step-9: Enter verification code which ever you got to registered mobile number and click verify.



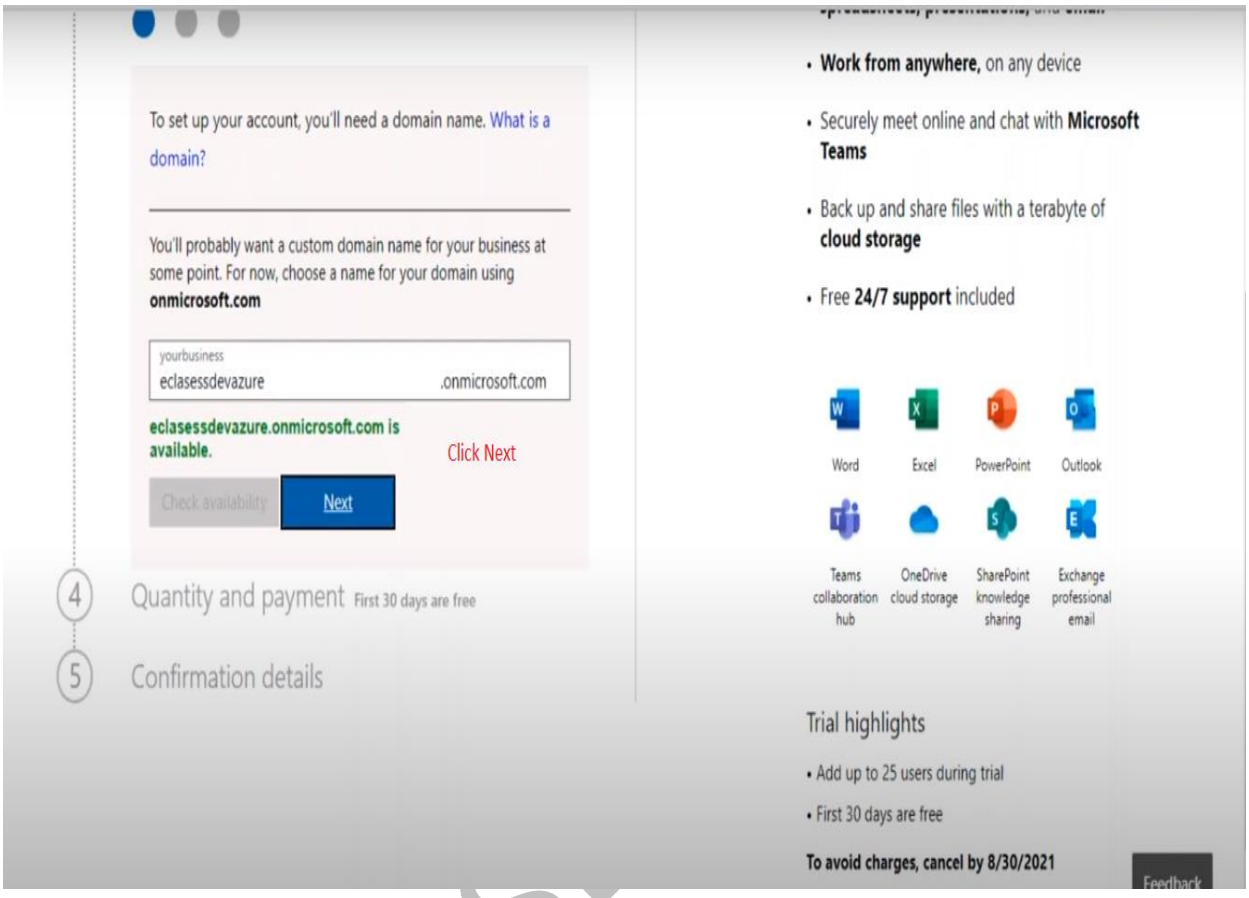
Step-10: Enter Business name and click on check availability.



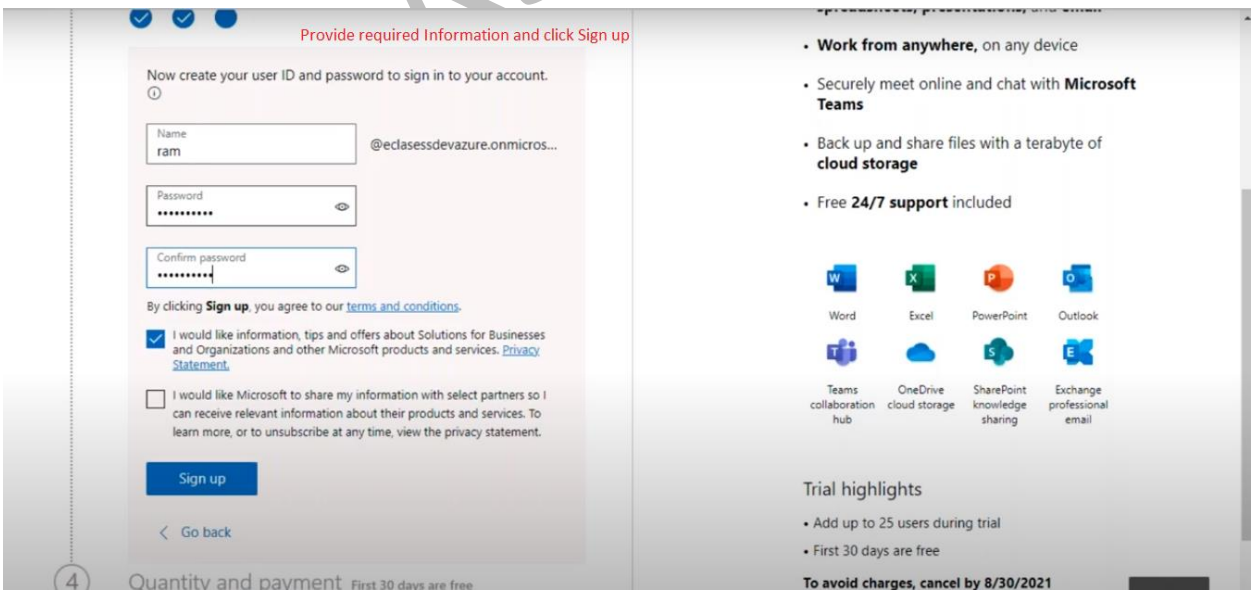
Step-11: Enter Business Name any name you can enter but needs to check availability.



Step 12: Click Next.

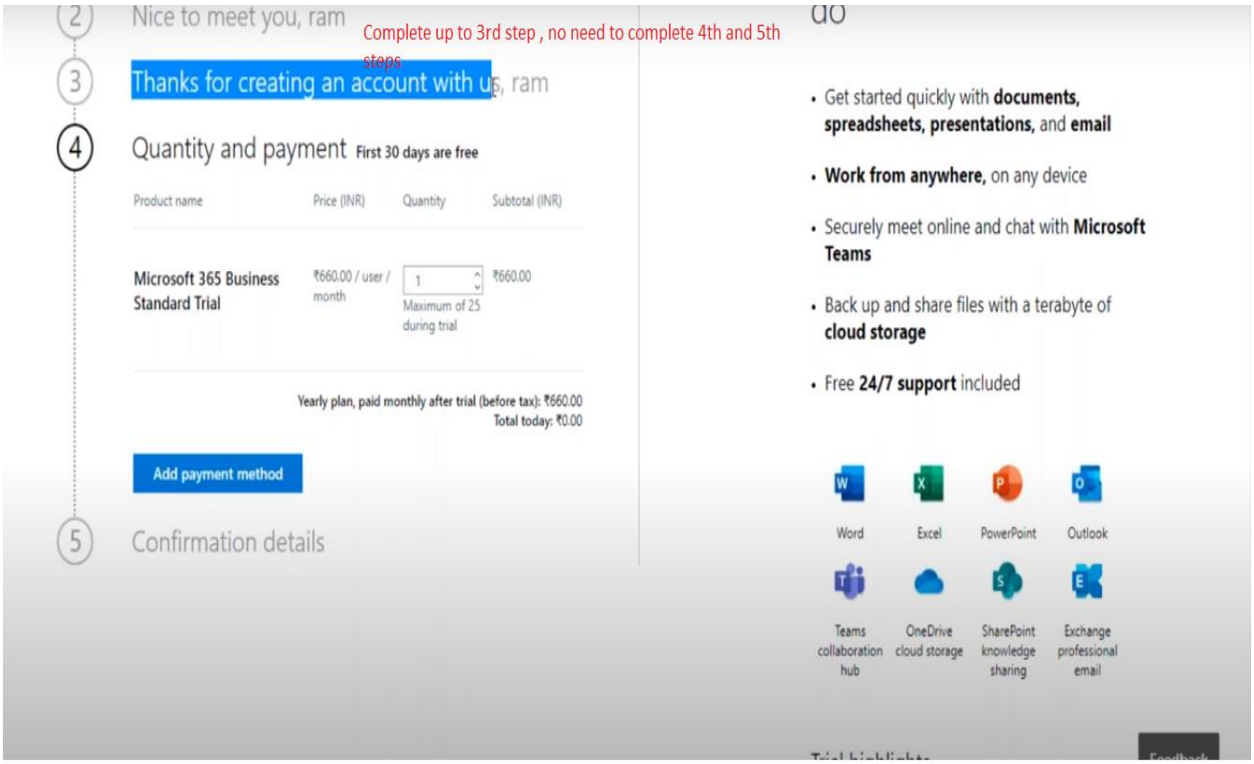


Step-13: Enter Name any Name and Set some password also then click sign up.



Step-14: No need to perform 4th step, finally your email has created.

ram@eclasessdev.onmicrosoft.com

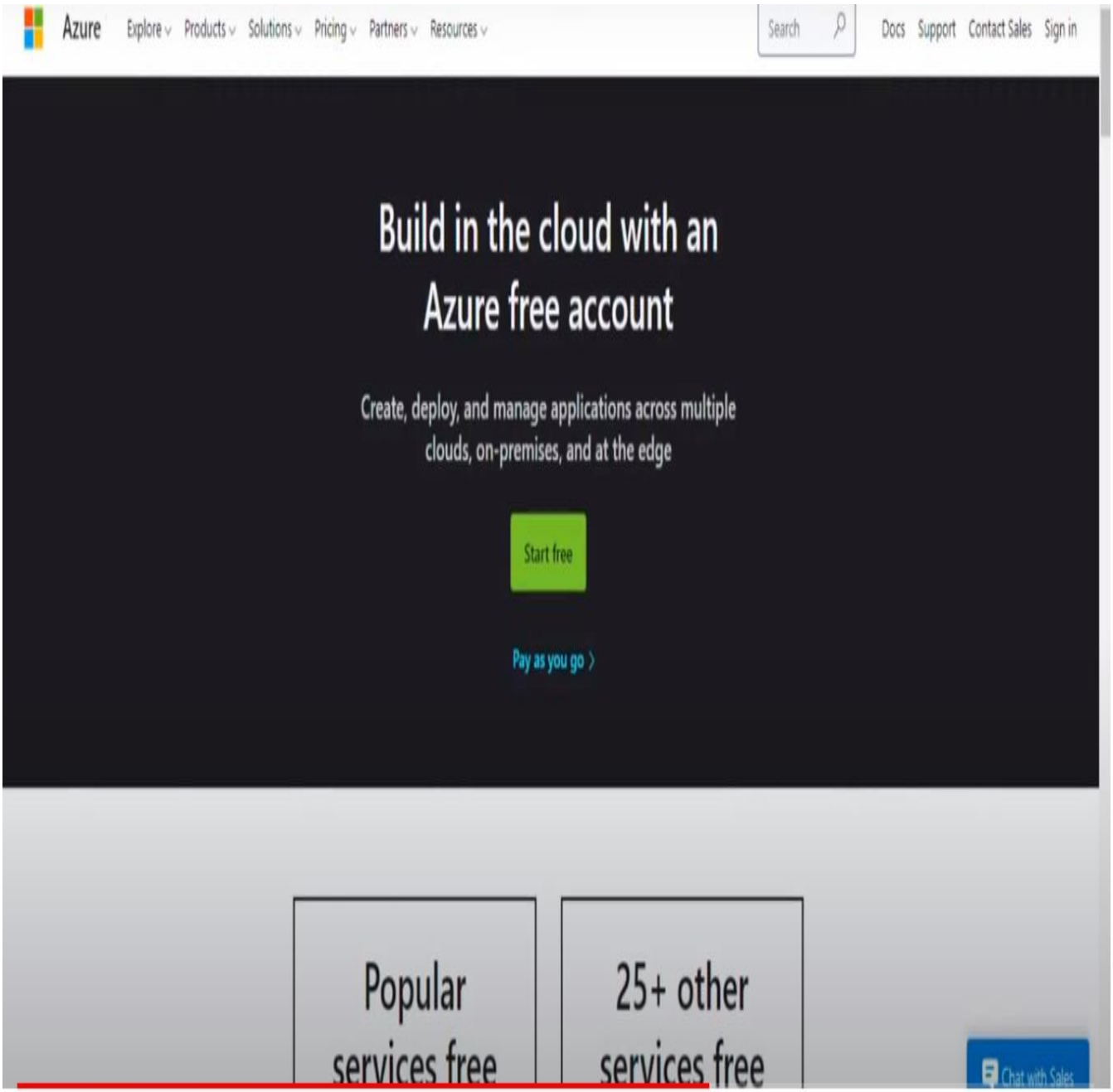


ECLASE

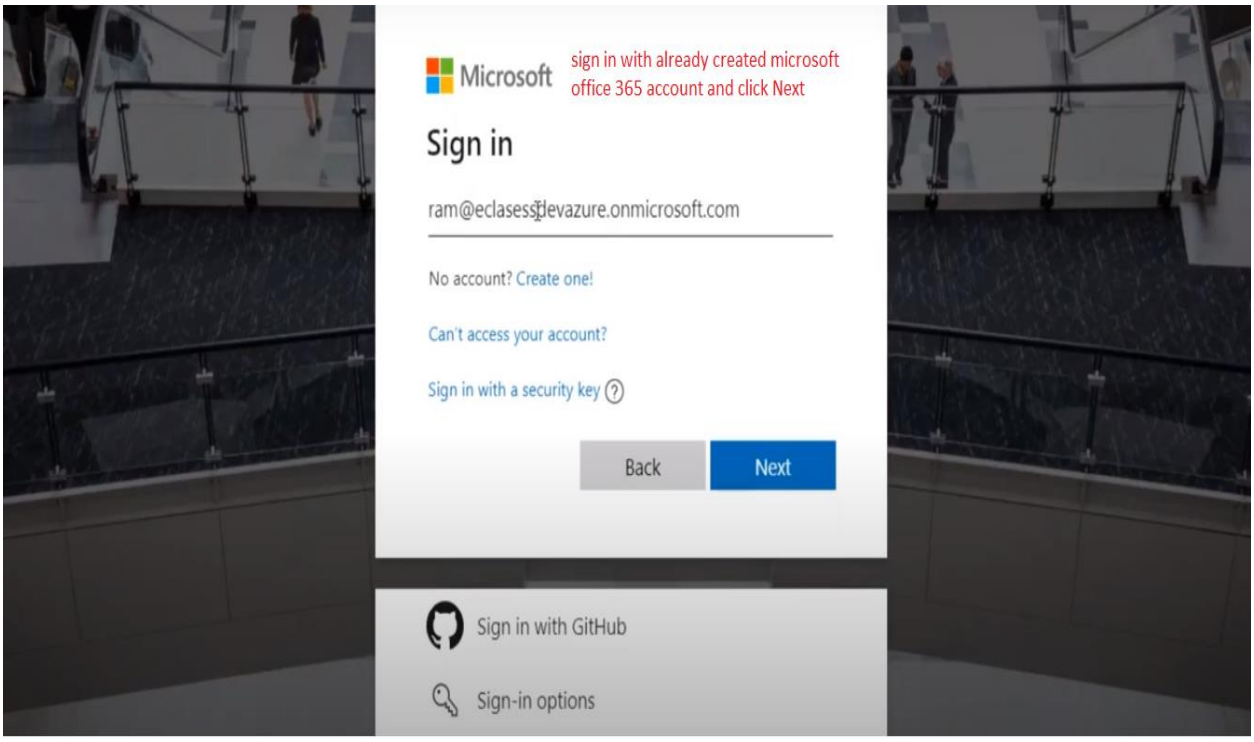
How to create Azure Free Subscription by using above Microsoft email address.

Step-1: Open below URL and click Start Free.

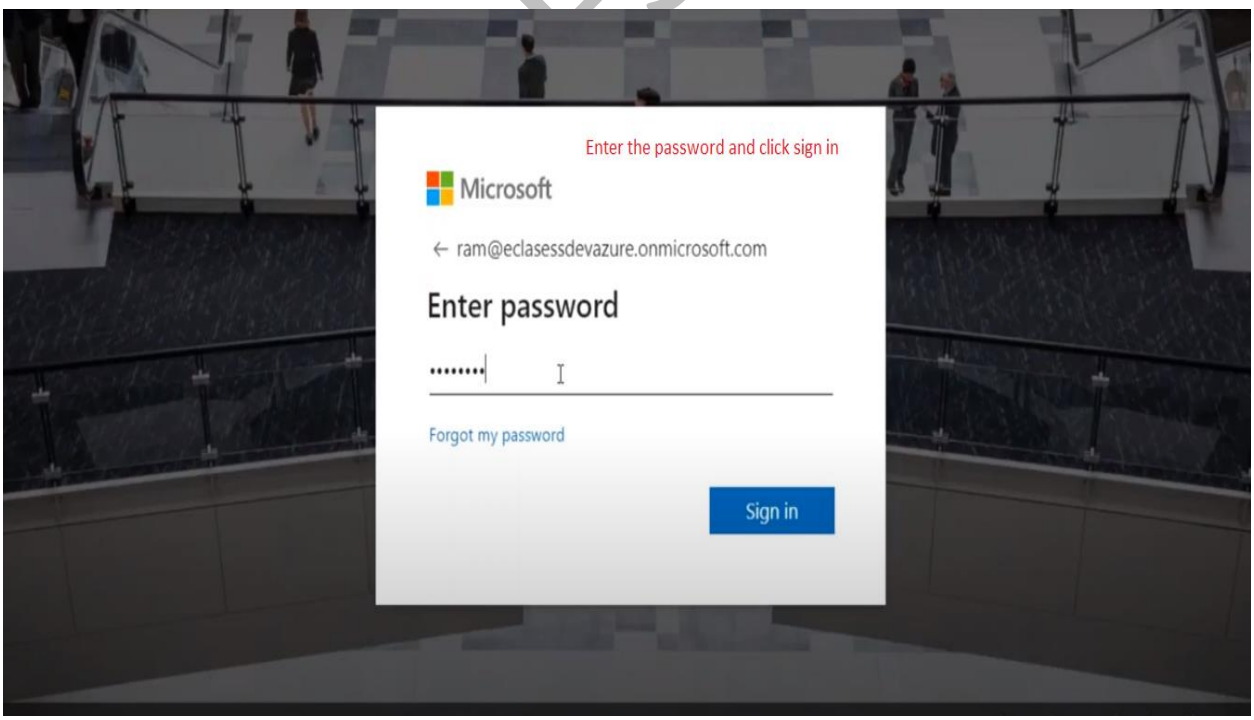
https://azure.microsoft.com/en-us/free/



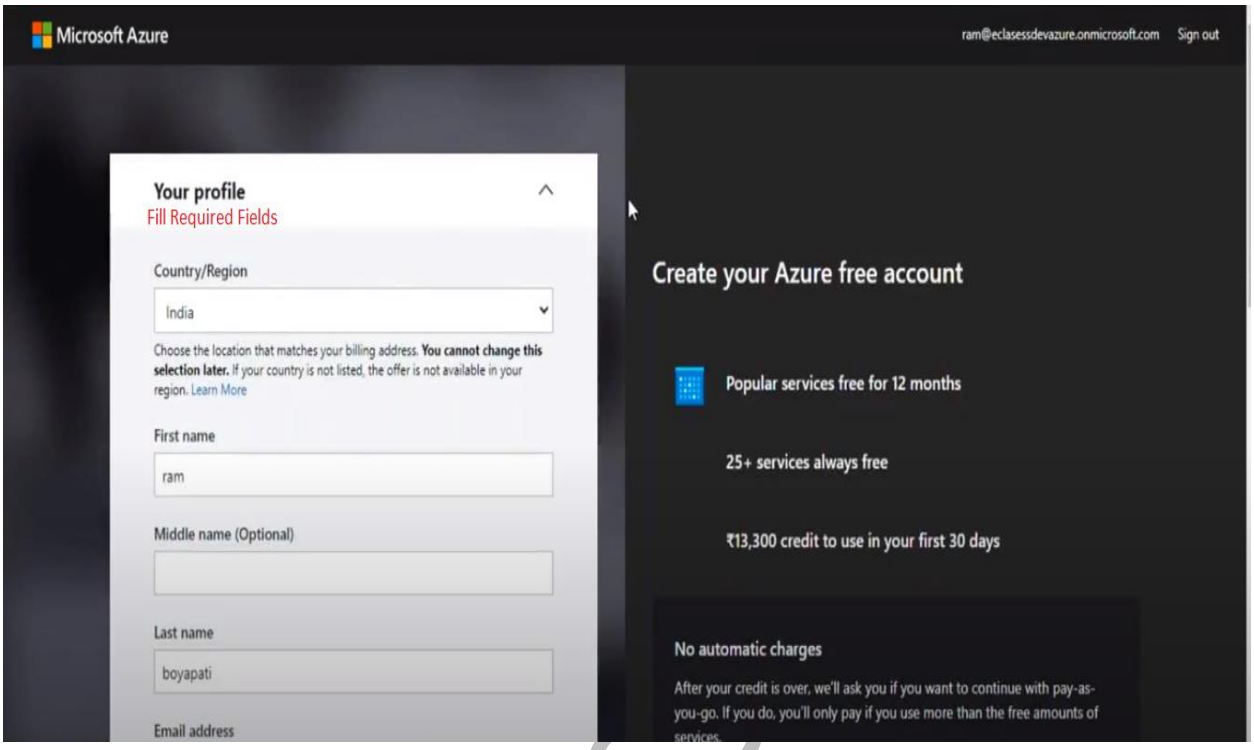
Step-2: Sign in with already created Microsoft office 365 account and click next.



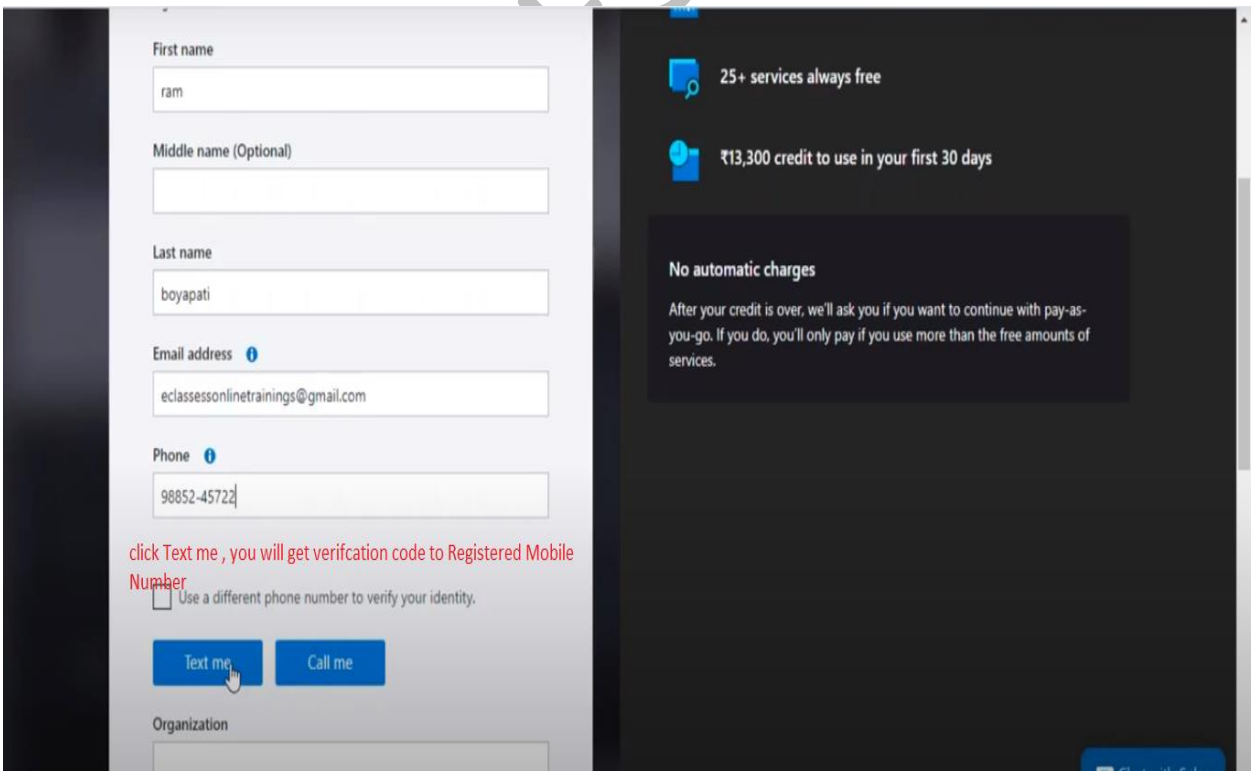
Step-3: Enter email password and Sign in.



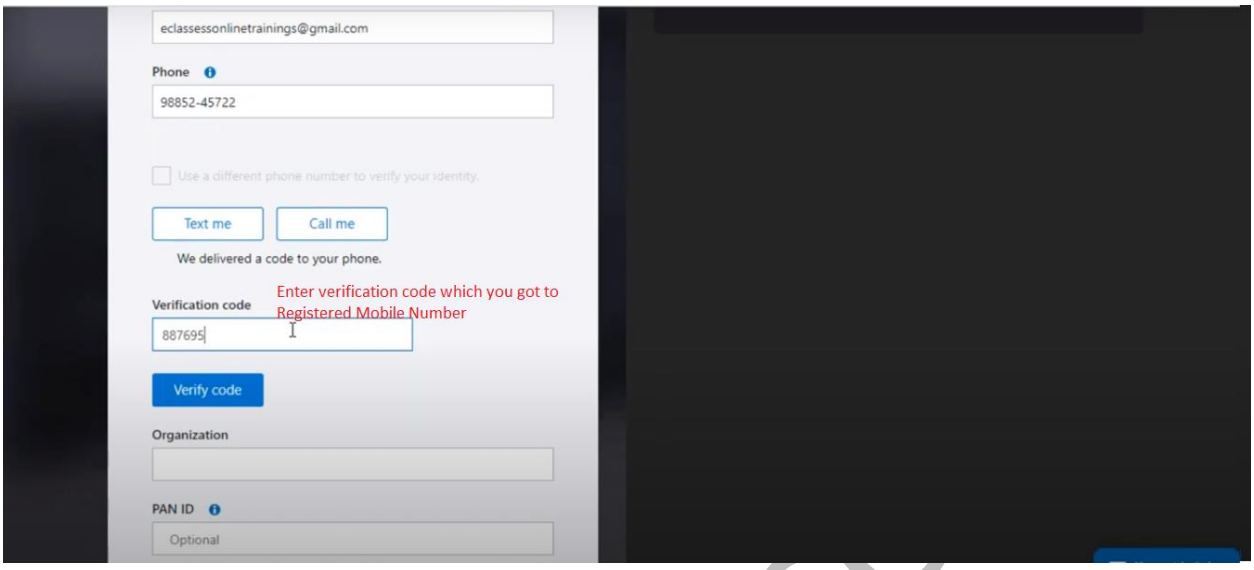
Step-4: Fill required fields.



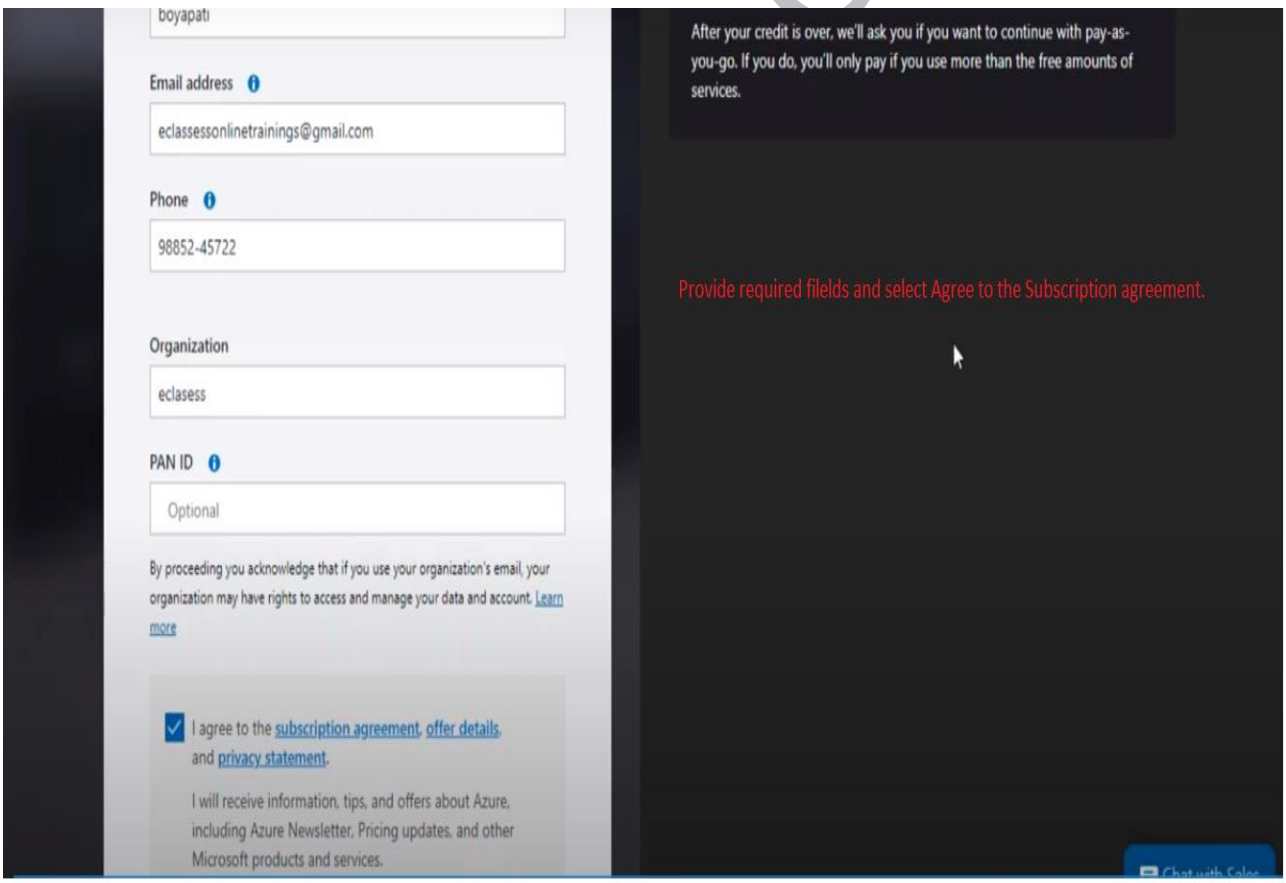
Step- 5: Give your mobile number and enter text me.



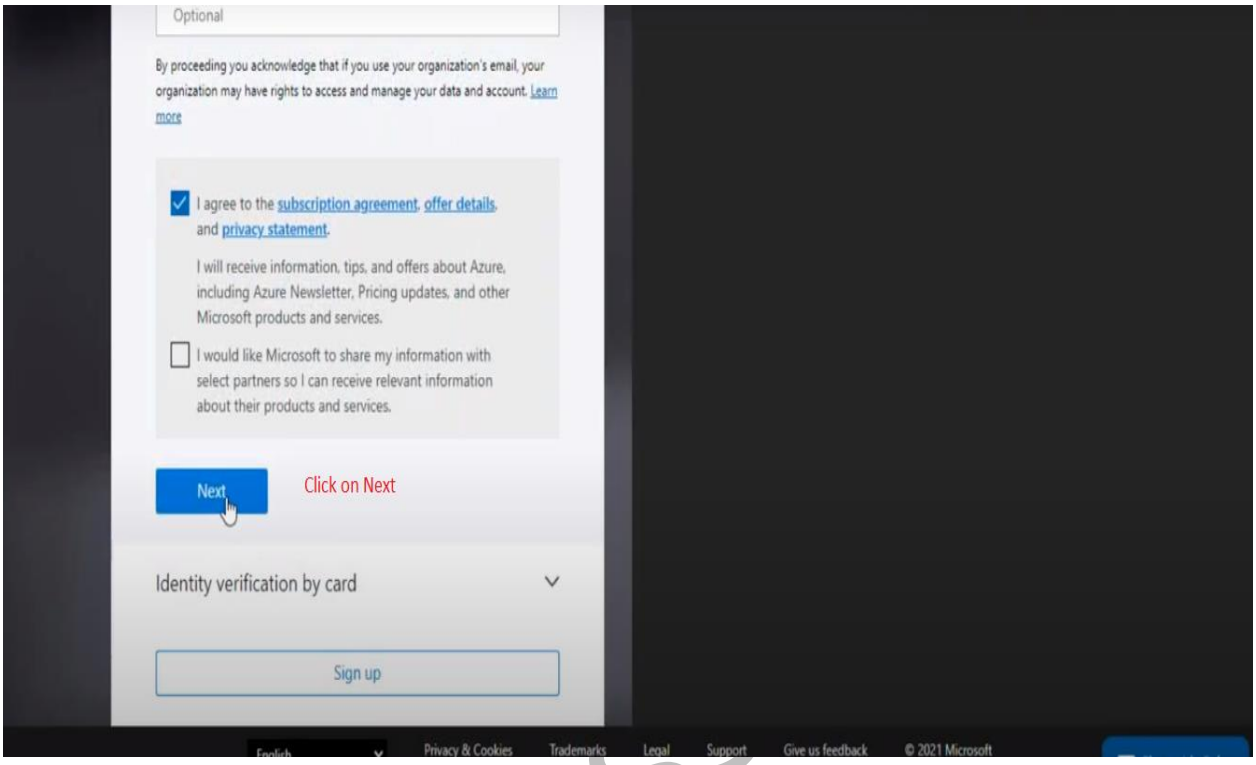
Step-6: Enter verification code received to your mobile and click on verify code.



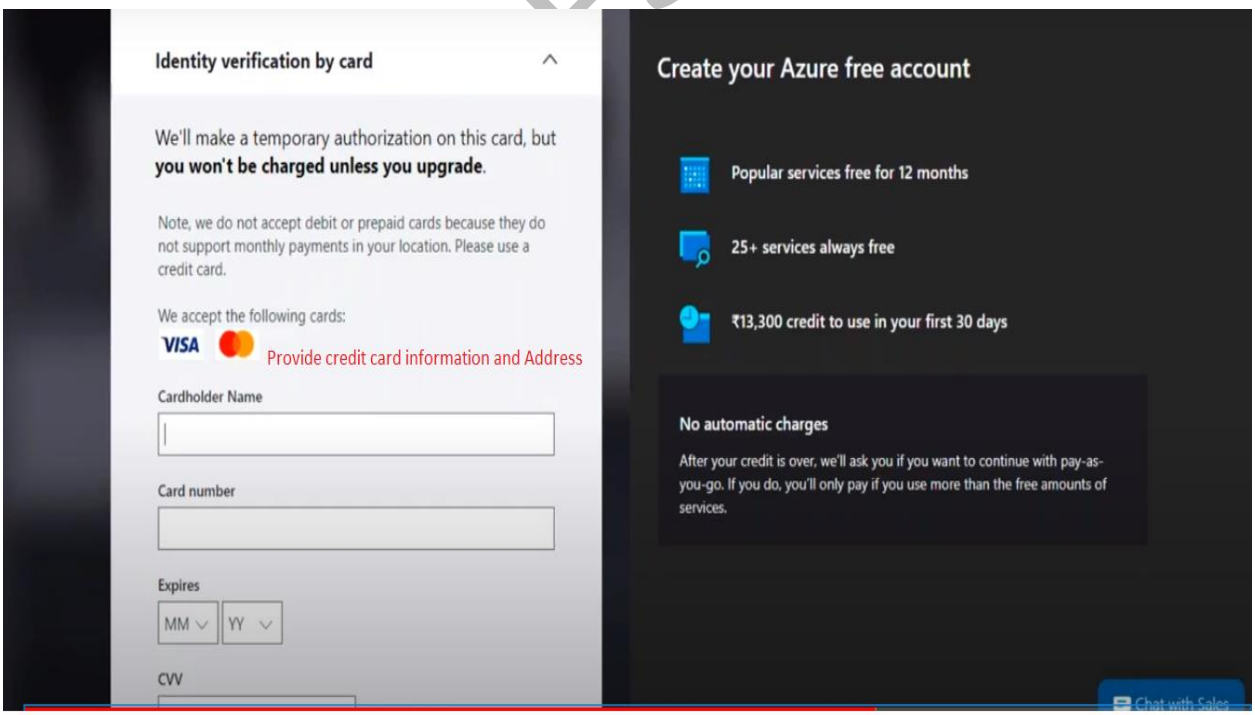
Step-7: Enter any Gmail address and organization Name any name and pan Id is optional.



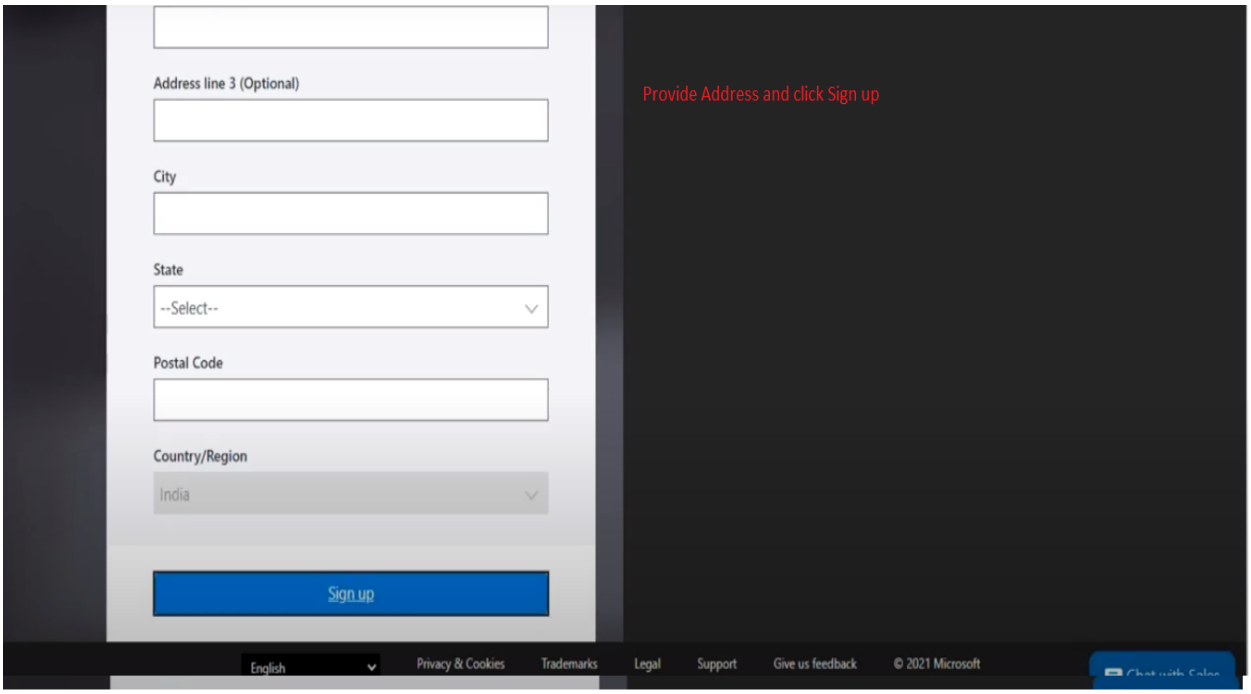
Step-8: select I agree and click next.



Step- 9: Provide credit cardholder name and card number Expire and cvv.

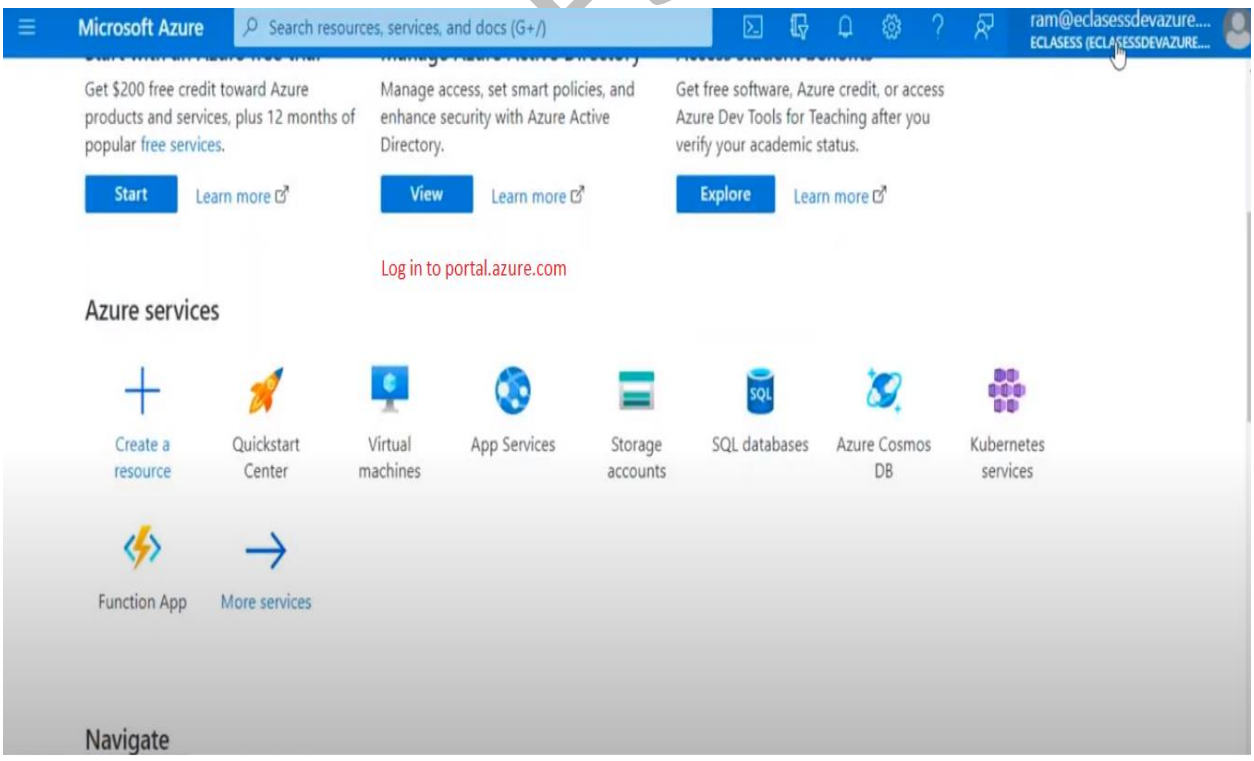


Step-10: provide credit card address details and click on sign up.

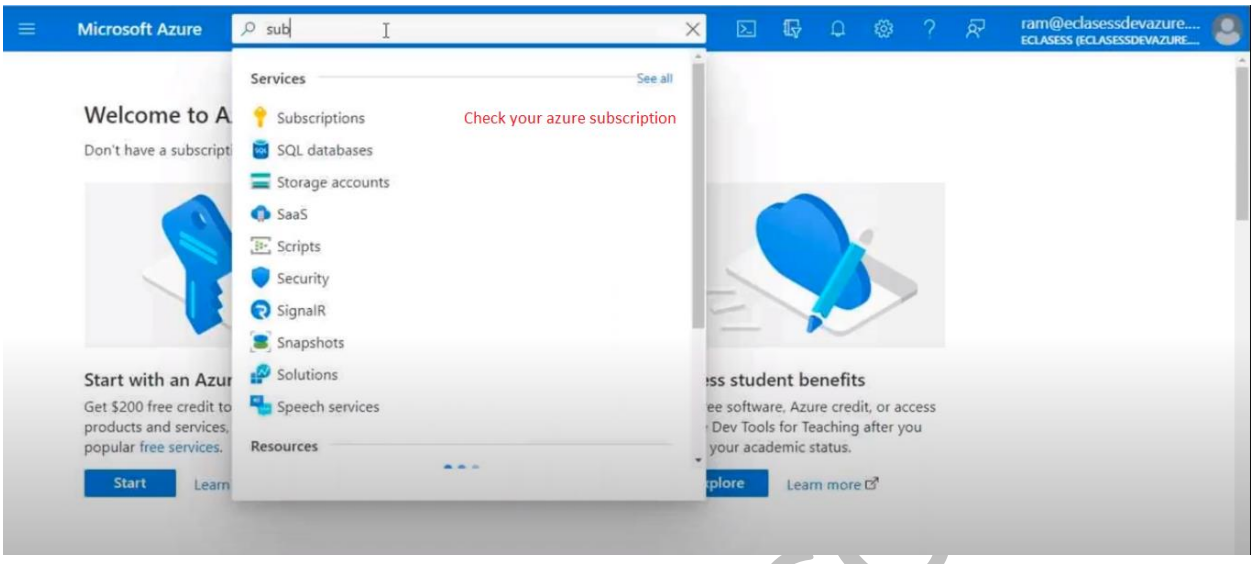


Step- 11: Login to Azure portal by clicking on below url by providing email and password.

https://portal.azure.com/



Step - 12: check free subscription created or not by searching for subscription.

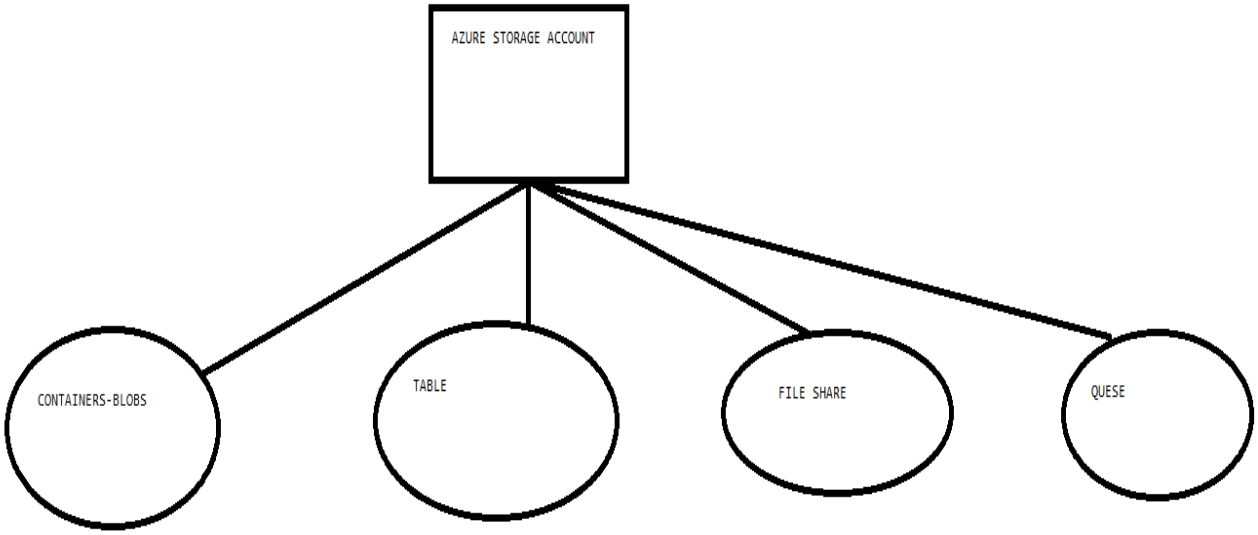


What is Azure Storage Accounts?

Azure storage account is cloud based storage solution to store any kind of data like structured, unstructured and semi structured.

Azure Storage Types

With an Azure Storage account which includes Blob, Table, Quese, and File share storage types.



Azure Blob Storage:

Blob Storage is basically general purpose storage for unstructured data that can include pictures, videos, music files, documents, raw data, and log data…along with their meta-data. Blobs are stored in a directory-like structure called a "**container**". Blob storage can also be used for data or device backup.

Blob Storage service comes with three types of blobs:

Block blobs: You can use block blobs for documents, image files, and video file storage

Append blobs: Append blobs are similar to block blobs, but are more often used for append operations like logging.

Page blobs: Page blobs are used for objects meant for frequent read-write operations. Page blobs are therefore used in Azure VMs to store OS and data disks.

Azure Table Storage:

Table storage, as the name indicates, is preferred for tabular data, which is ideal for key-value NoSQL data storage. Table Storage is massively scalable and extremely easy to use. Like other NoSQL data stores, it is schema-less and accessed via a REST API.

Azure File Storage:

Azure File Storage is meant for legacy applications. Azure VMs and services share their data via mounted file shares, while on premise applications access the files using the File Service REST API. Azure File Storage offers file shares in the cloud using the standard SMB protocol and supports both SMB 3.0 and SMB 2.1.

Azure Quese Storage:

The Queue Storage service is used to exchange messages between components either in the cloud or on premise (compare to Amazon's SQS). You can store large numbers of messages to be shared between independent components of applications and communicated asynchronously via HTTP or HTTPS. Typical use cases of Queue Storage include processing backlog messages or exchanging messages between Azure Web roles and Worker roles.

Types of Storage Accounts

The first decision to be made when creating Storage Accounts is the type. We have three types available**: General Purpose (v1), General Purpose (v2)**

The **General Purpose (v1)** has been available since the classic deployment model, which has been replaced for the ARM (Azure Resource Manager) in the last years. It does not have all the new features, and the cost, in general, is higher than its predecessor, General Purpose (v2).

ECLASESS.COM

The **General Purpose (v2)** is the new kid on the block. It has all the capabilities that General Purpose (v1) has. All existent and new features are being created to be used in this version. Using General Purpose (v2), we have access to define the access tier (hot or cold) at the account level.

Using either General Purpose v1 or v2, the new Storage Account will have all services available: Blobs, Files, Tables, and Queues. In the Overview page, we can also check the current performance, replication, and type of account as depicted in the image below

Performance

When using general purpose Storage Accounts, we have the option to decide between Standard and Premium storage.

Standard is the default option and is suitable for the vast majority of workloads. It uses magnetic media to store the data. Storage cost is less and performance also less compare to premium.

Premium offering provides high-performance storage and it is designed based on Solid State Drives (SSD) to store data (much faster). It is recommended for VM workloads (disks) and high-performance applications. Storage cost is more and performance also more.

Access tier (Cool vs Hot)

Cool Tier is suitable for infrequently accessing data. Cost is less and performance also less.

Hot Tier is suitable for frequently accessing data. Cost also high and performance also more.

Replication

Azure storage provides durability and high availability of all data stored in the platform. By default, we always will have replication in at least three places when using Azure Storage that is a requirement to maintain the service level agreement provided by Microsoft Azure.

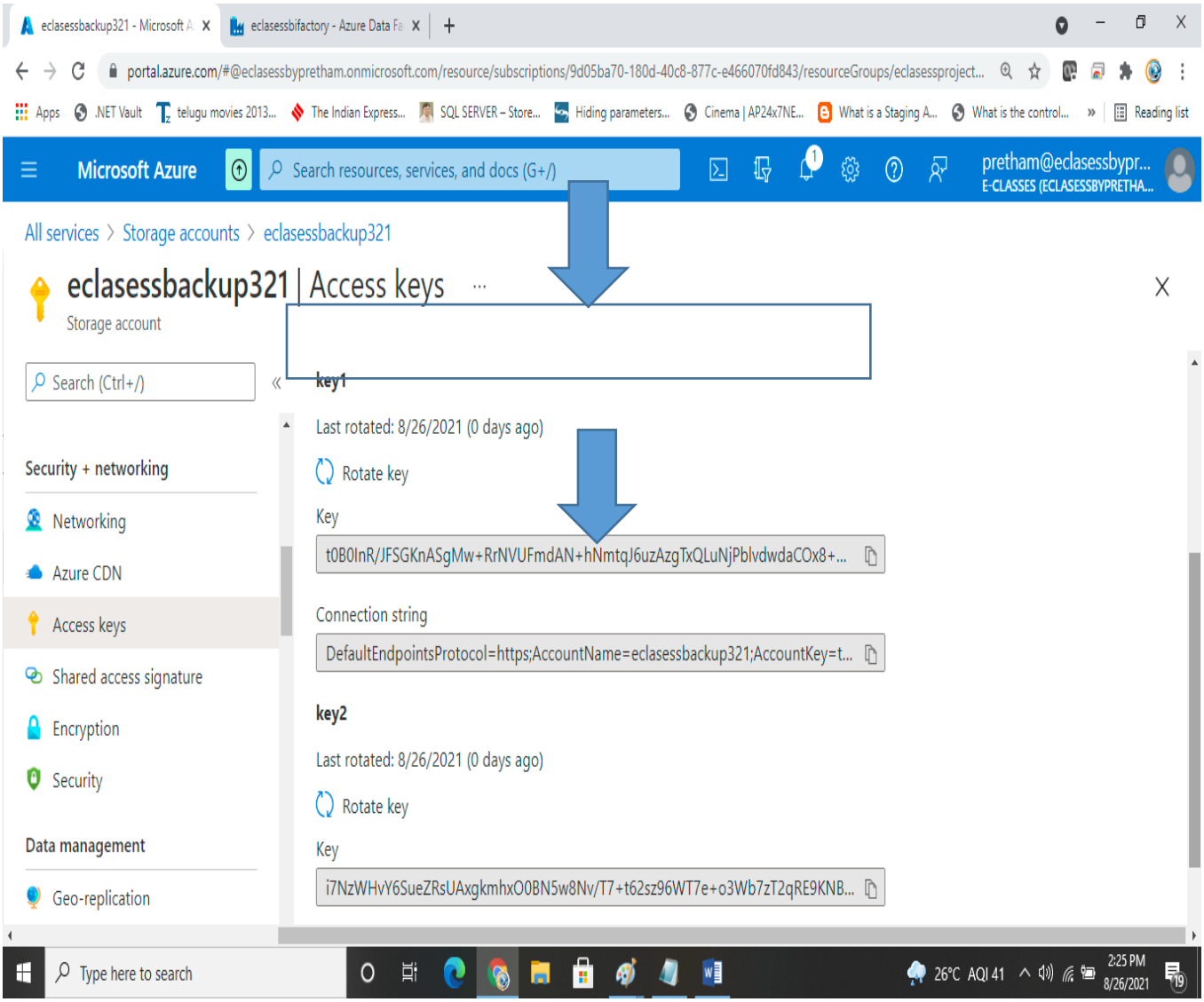
- **LRS** (Locally Redundant Storage): At least three copies of the data are kept in the same region, so a failure of a rack/server will not impact the data.
- **ZRS** (Zone Redundant Storage): Each Azure region is comprised of more than one datacenter. Using ZRS, three copies of the data will be spread among zones that have independent power, cooling, and networking capabilities.
- **GRS** (Geo-Redundant Storage): Besides your three regular copies, another set of three copies are stored in another region within the same geopolitical region. It uses asynchronous replication and the failover between regions is performed by Microsoft. The customer doesn't have control over it.
- **RA-GRS** (Read-Only Geo-Redundant Storage): This one is GRS with a twist. The cloud administrator can have read-only access to the data in the second region.

ECLASESS.COM

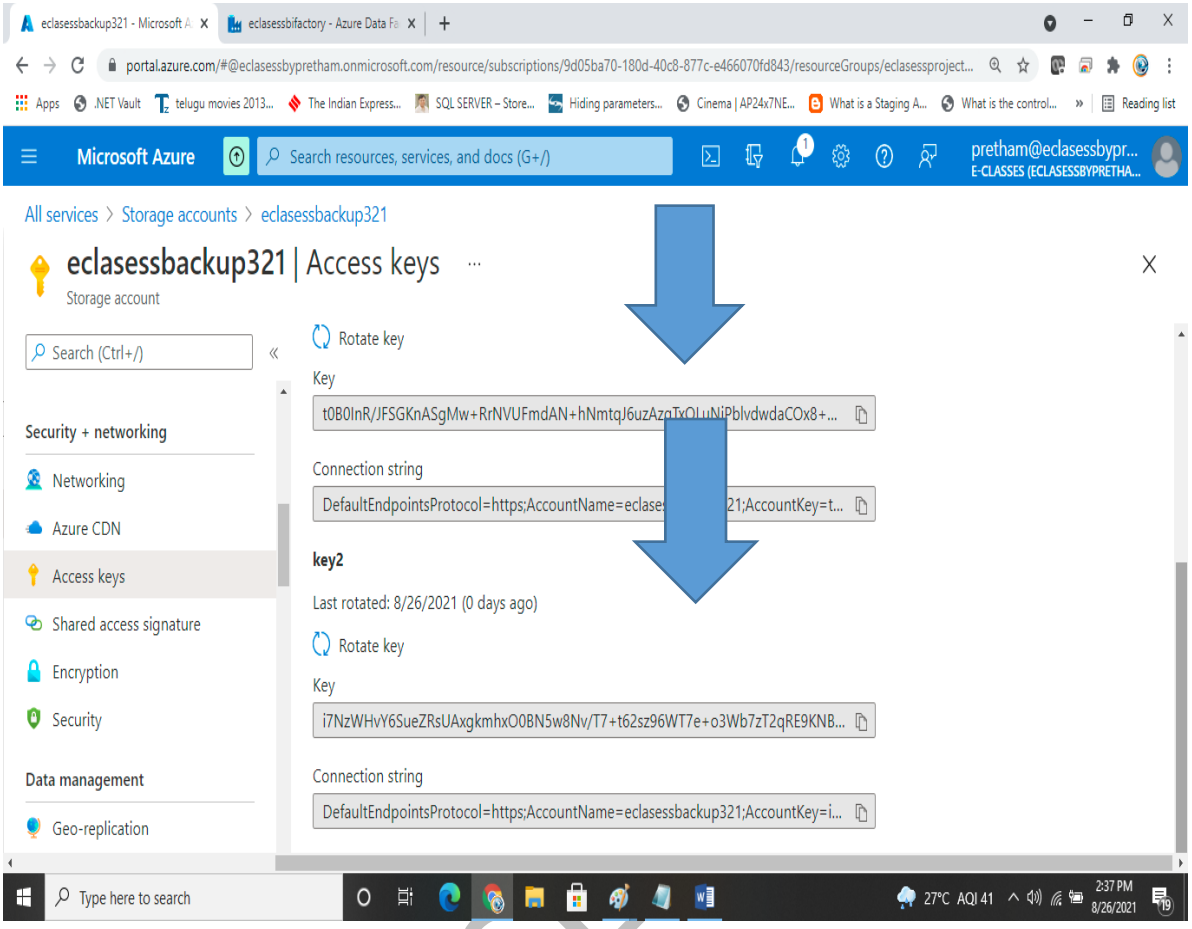
In order to Access information, it is just a matter of adding **-secondary** to the endpoint and use the same access keys to access the information.

- **Security in Azure Storage Account**

- **Access keys:**



Connection Strings:



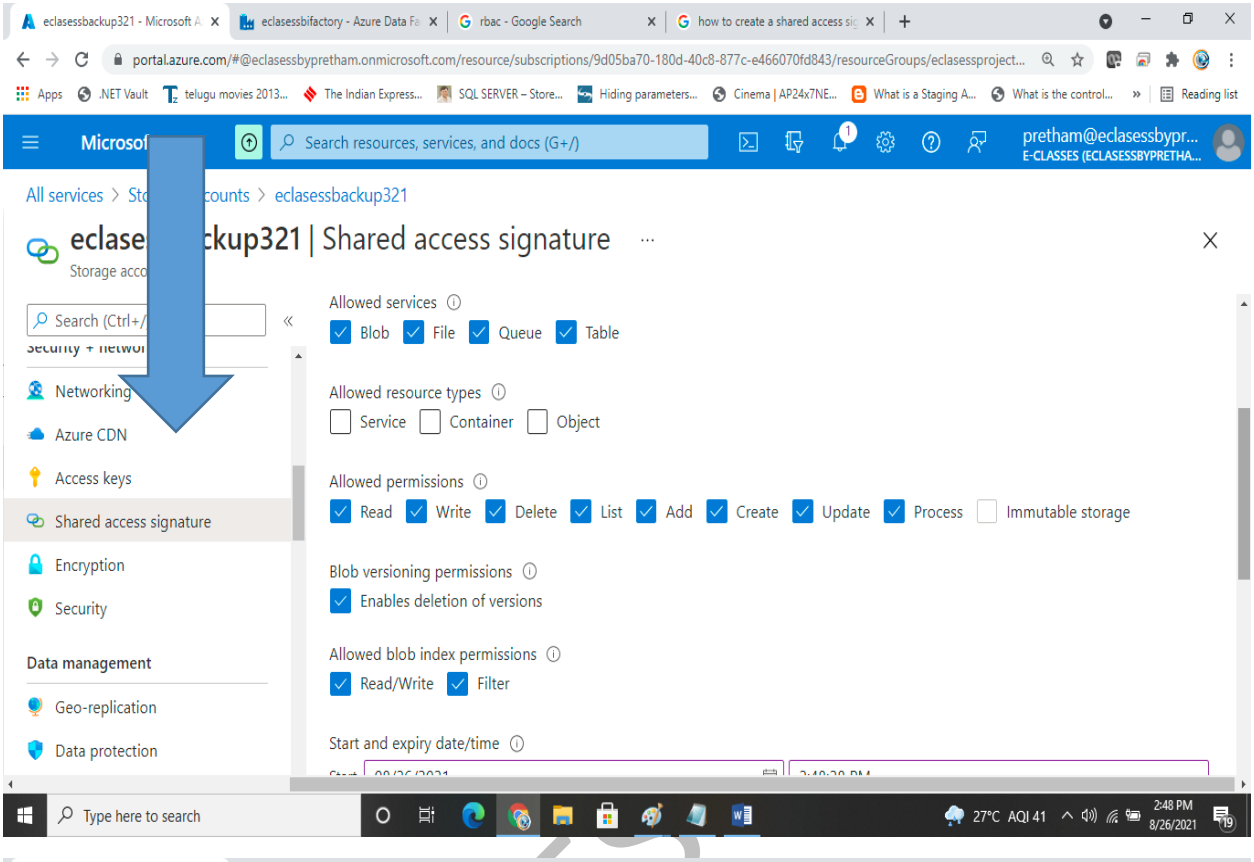
Shared Access Signatures: A shared access signature (SAS) is a URI that grants restricted access rights to Azure Storage resources.

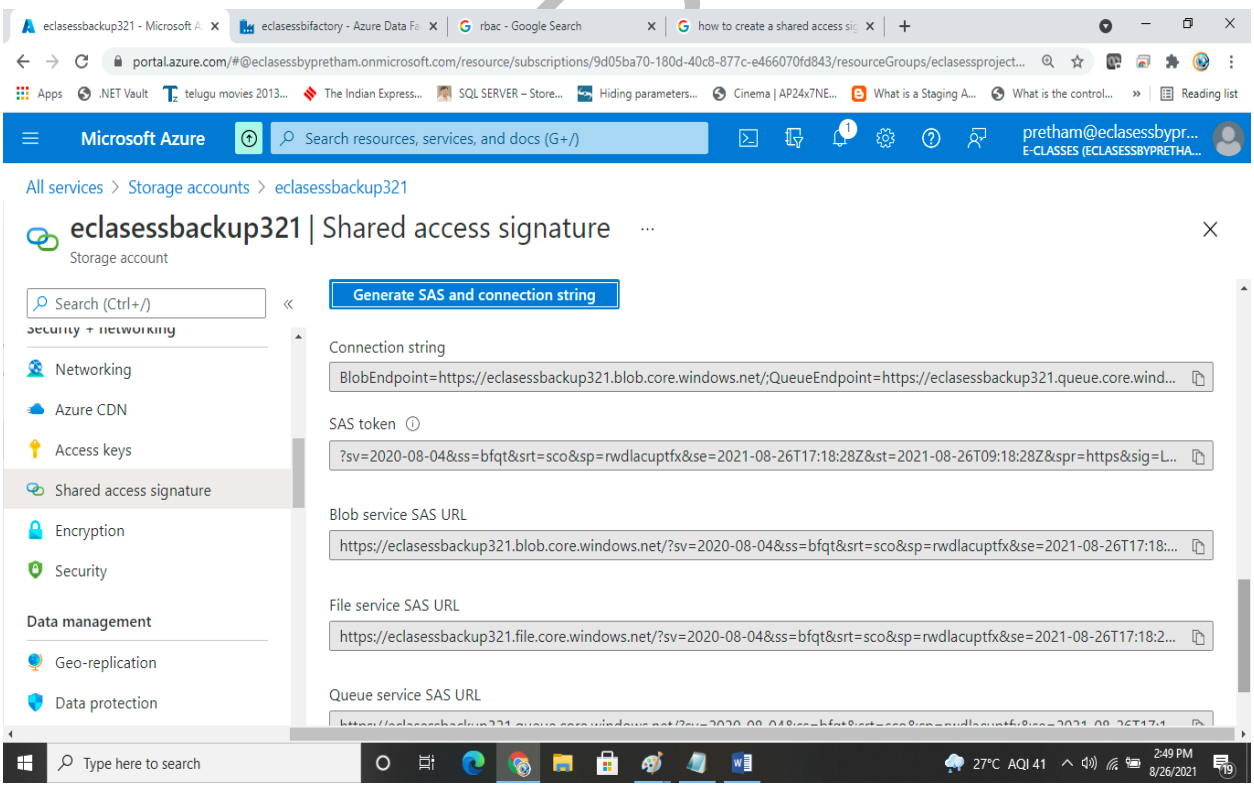
You can provide a shared access signature to clients who should not be trusted with your storage account key.

Advantages:

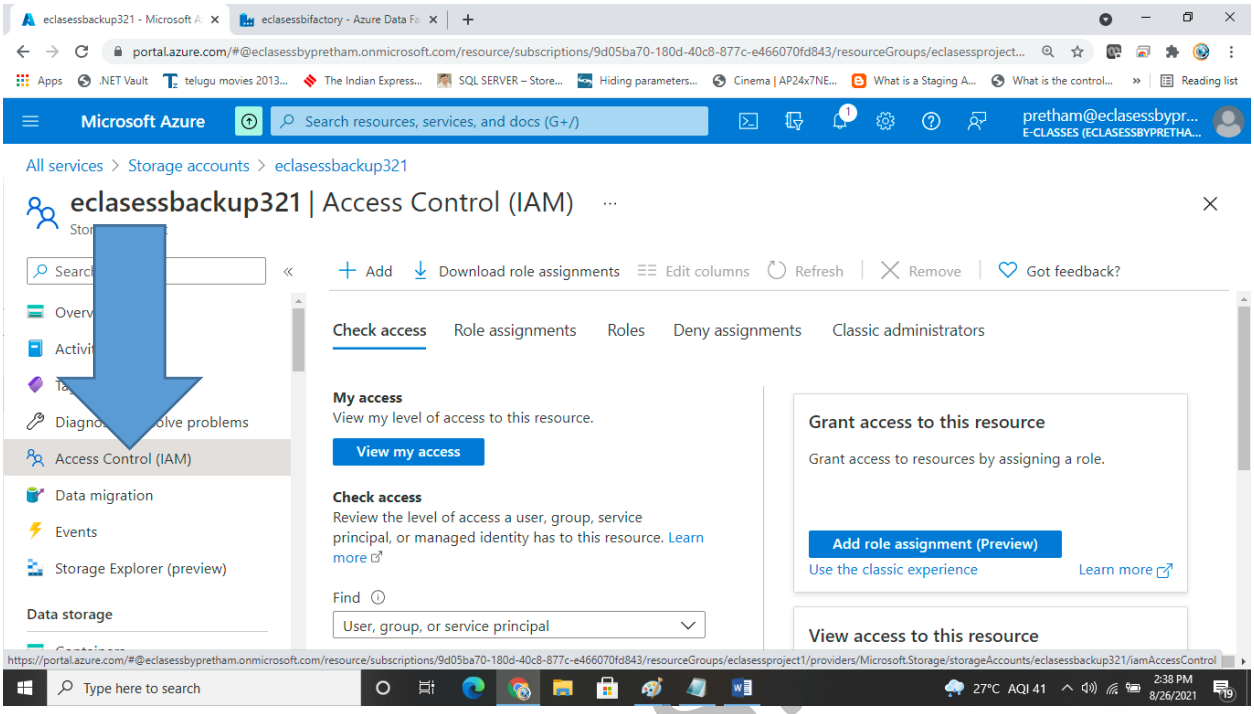
- Providing access to only particular resources
- Providing specific role
- Providing only particular IP system

ECLASESS

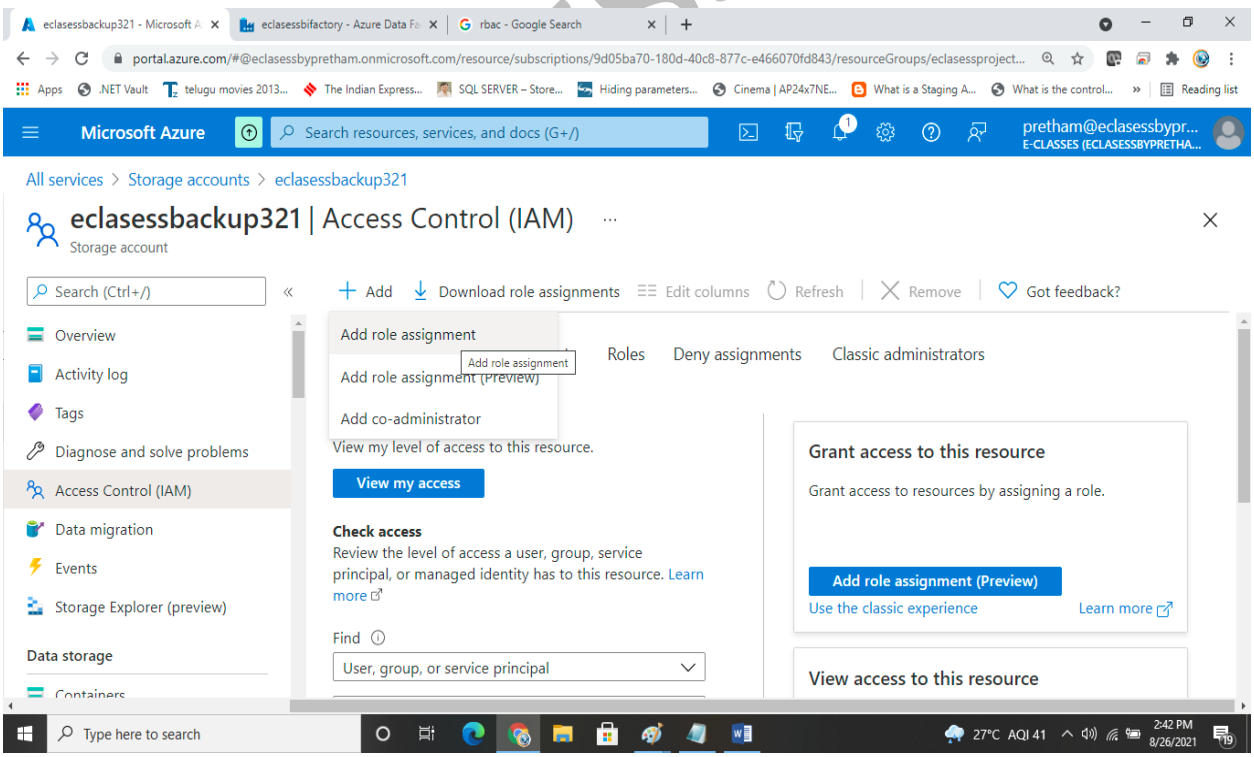

eclasessbackup321 - Microsoft A
eclasessbifactory - Azure Data Fa
rbac - Google Search
how to create a shared access sig
portal.azure.com/#@eclasessbypretham.onmicrosoft.com/resource/subscriptions/9d05ba70-180d-40c8-877c-e466070fd843/resourceGroups/eclasessproject...
Search resources, services, and docs (G+/)
pretham@eclasessbypr...
E-CLASSES (ECLASESSBYPRETHA...
All services > St counts > eclasessbackup321
eclase ckup321 | Shared access signature
Storage acco
Search (Ctrl+/
Networking
Azure CDN
Access keys
Shared access signature
Encryption
Security
Data management
Geo-replication
Data protection
Allowed services
Blob File Queue Table
Allowed resource types
Service Container Object
Allowed permissions
Read Write Delete List Add Create Update Process Immutable storage
Blob versioning permissions
Enables deletion of versions
Allowed blob index permissions
Read/Write Filter
Start and expiry date/time
Type here to search
27°C AQI 41
2:48 PM
8/26/2021


eclasessbackup321 - Microsoft A
eclasessbifactory - Azure Data Fa
rbac - Google Search
how to create a shared access sig
portal.azure.com/#@eclasessbypretham.onmicrosoft.com/resource/subscriptions/9d05ba70-180d-40c8-877c-e466070fd843/resourceGroups/eclasessproject...
Microsoft Azure
Search resources, services, and docs (G+/)
pretham@eclasessbypr...
E-CLASSES (ECLASESSBYPRETHA...
All services > Storage accounts > eclasessbackup321
eclasessbackup321 | Shared access signature
Storage account
Search (Ctrl+/)
Networking
Azure CDN
Access keys
Shared access signature
Encryption
Security
Data management
Geo-replication
Data protection
Generate SAS and connection string
Connection string
BlobEndpoint=https://eclasessbackup321.blob.core.windows.net/;QueueEndpoint=https://eclasessbackup321.queue.core.wind...
SAS token
?sv=2020-08-04&ss=bfqt&srt=sco&sp=rwdlacuptfx&se=2021-08-26T17:18:28Z&st=2021-08-26T09:18:28Z&spr=https&sig=L...
Blob service SAS URL
https://eclasessbackup321.blob.core.windows.net/?sv=2020-08-04&ss=bfqt&srt=sco&sp=rwdlacuptfx&se=2021-08-26T17:18:...
File service SAS URL
https://eclasessbackup321.file.core.windows.net/?sv=2020-08-04&ss=bfqt&srt=sco&sp=rwdlacuptfx&se=2021-08-26T17:18:2...
Queue service SAS URL
Type here to search
27°C AQI 41
2:49 PM
8/26/2021

RBAC (Role Based Access Control): Click on Access Control (IAM).



Click on add role assignment.

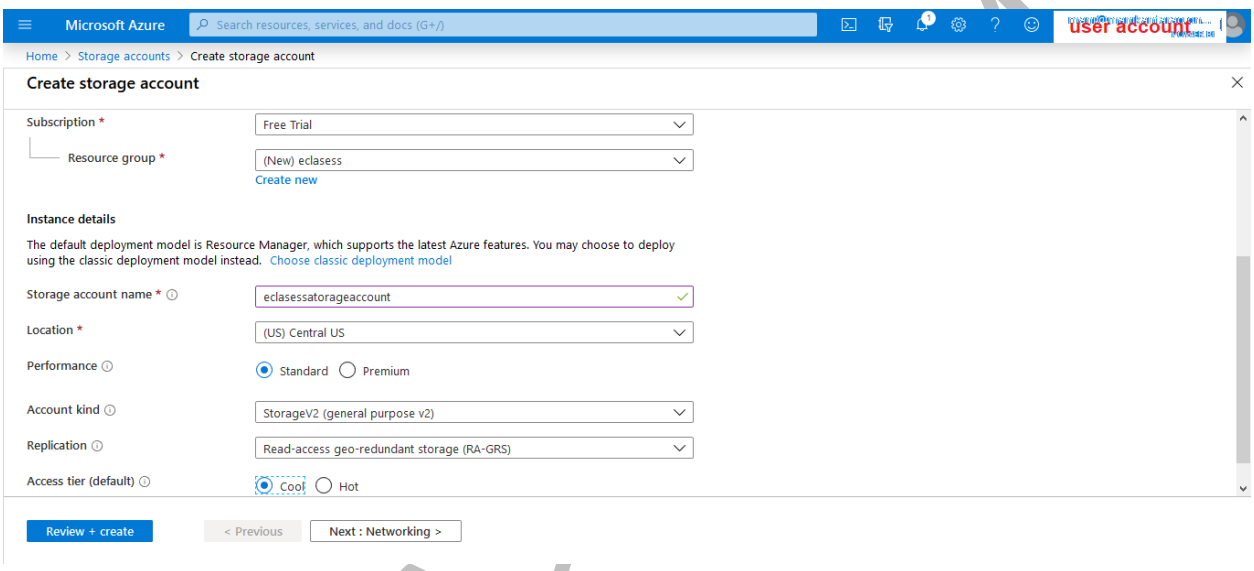


How to Create Azure Storage Account

Now you are ready to create a storage account.

Every storage account must belong to an Azure resource group. A resource group is a logical container for grouping your Azure services. When you create a storage account, you have the option to either create a new resource group, or use an existing resource group. This article shows how to create a new resource group.

A **general-purpose v2** storage account provides access to all of the Azure Storage services: blobs, files, queues, tables, and disks. The steps outlined here create a general-purpose v2 storage account, but the steps to create any type of storage account are similar.



1. On the Azure portal menu, select **all services**. In the list of resources, type **Storage Accounts**. As you begin typing, the list filters based on your input. Select **Storage Accounts**.

2. On the **Storage Accounts** window that appears, choose **Add**.

3. Select the subscription in which to create the storage account.

4. Under the **Resource group** field, select **Create new**. Enter a name for your new resource group, as shown in the following image.

5. Next, enter a name for your storage account. The name you choose must be unique across Azure. The name also must be between 3 and 24 characters in length, and can include numbers and lowercase letters only.

6. Select a location for your storage account, or use the default location.

7. Leave these fields set to their default values:

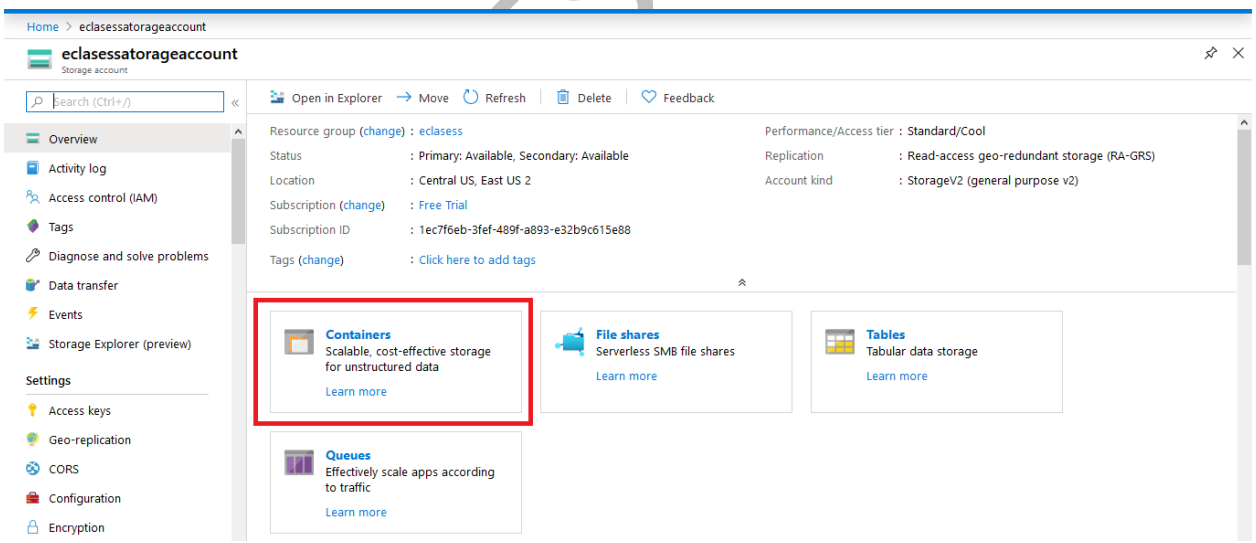
Field	Value
Deployment model	Resource Manager
Performance	Standard
Account kind	StorageV2 (general-purpose v2)
Replication	Read-access geo-redundant storage (RA-GRS)
Access tier	Hot

8. Select **Review + Create** to review your storage account settings and create the account.
9. Select **Create**.

How to create Container Account:

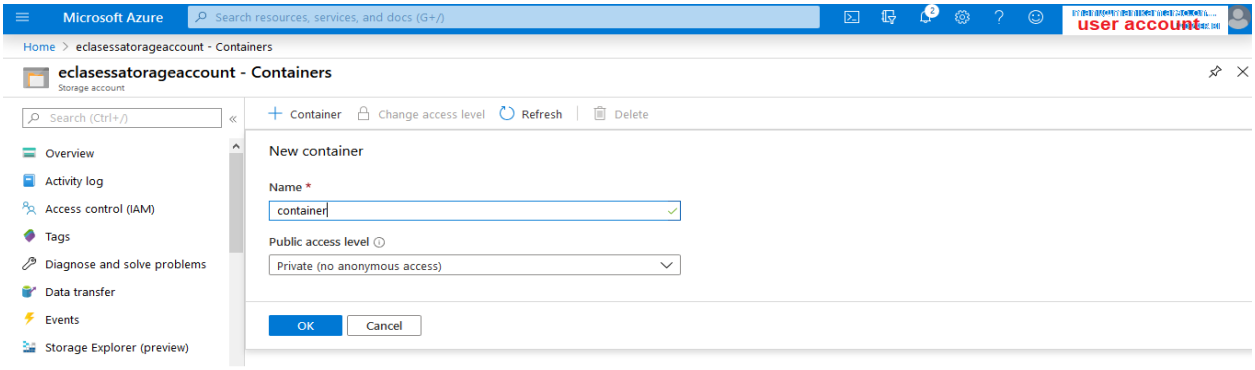
To create a container in the Azure portal, follow these steps:

1. Navigate to your new storage account in the Azure portal.
2. In the left menu for the storage account, scroll to the **Blob service** section, then select **Containers**.
3. Select the **+ Container** button.



4. Type a name for your new container. The container name must be lowercase, must start with a letter or number, and can include only letters, numbers, and the dash (-) character. For more information about container and blob names, see Naming and referencing containers, blobs, and metadata.
5. Set the level of public access to the container. The default level is **Private (no anonymous access)**.

6. Select **OK** to create the container.



Upload a block blob

Block blobs consist of blocks of data assembled to make a blob. Most scenarios using Blob storage employ block blobs. Block blobs are ideal for storing text and binary data in the cloud, like files, images, and videos. This quick start shows how to work with block blobs.

To upload a block blob to your new container in the Azure portal, follow these steps:

1. In the Azure portal, navigate to the container you created in the previous section.
2. Select the container to show a list of blobs it contains. Since this container is new, it won't yet contain any blobs.
3. Select the **Upload** button to upload a blob to the container.
4. Browse your local file system to find a file to upload as a block blob, and select **Upload**.



5. Select the **Authentication type**. The default is **SAS**.

6. Upload as many blobs as you like in this way. You'll see that the new blobs are now listed within the container.

Download a block blob

You can download a block blob to display in the browser or save to your local file system. To download a block blob, follow these steps:

1. Navigate to the list of blobs that you uploaded in the previous section.
2. Right-click the blob you want to download, and select **Download**.

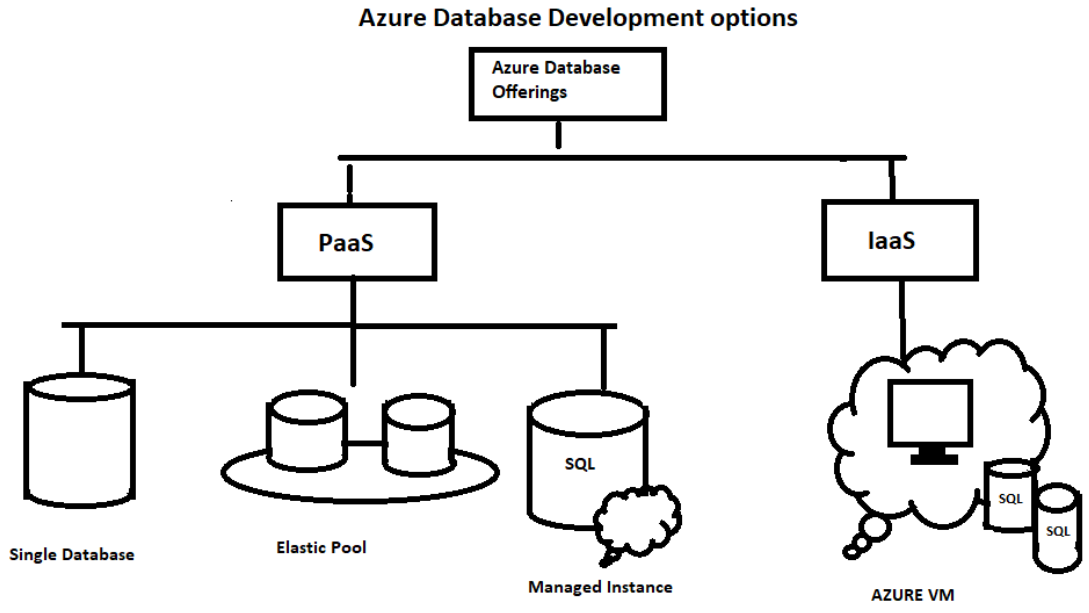
Clean up resources

To remove the resources you created in this quick start, you can delete the container. All blobs in the container will also be deleted.

To delete the container:

1. In the Azure portal, navigate to the list of containers in your storage account.
2. Select the container to delete.
3. Select the **More** button (**...**), and select **Delete**.
4. Confirm that you want to delete the container.

Azure SQL



Azure SQL part covering deployment models and utilization of SQL related services inside Azure cloud.

PAAS- Platform as a service further has 3 deployment modes.

- **Azure SQL Database**
  Single Database
  Elastic Pool
- **Azure SQL Managed Instance**
- **Azure SQL in virtual Machine**
- **Single Database** Single isolated database that is perfect for application need single database. It has its own dedicated compute memory, and storage. Without requiring any hardware or software installation.

There is no flexibility to choose right version, always latest version of SQL can be chosen.
This service will not get instance level features.
The performance of service can be defined based on Storage and DTU (Data Transaction Units), V-cores.

- Long-term backup retention enables us to keep backups for up to 10 years
- Geo-replication provides to create readable secondary databases in different data center locations
- Automatic tuning is an autonomous artificial intelligence-based performance tuning option that fixes the performance problems automatically
- Business continuity
- High-availability
- Scaling database resources feature is the most powerful feature that provides scaling the database resources either up or down
- Automated backups

- **Elastic Pool**: Collection of single databases with shared set of resources CPU Memory, IO. Elastic pool can be created against logical database server.
  Fixed amount of resources can be shared among multiple databases
- **Azure SQL Managed Instance**: Set of databases that can used together. Easy migration of on premises database application to cloud. Getting all Instance level features. This Dedicated instance of SQL Server running in your virtual network.

IAAS- Infra Structure as a service further has one deployment mode.

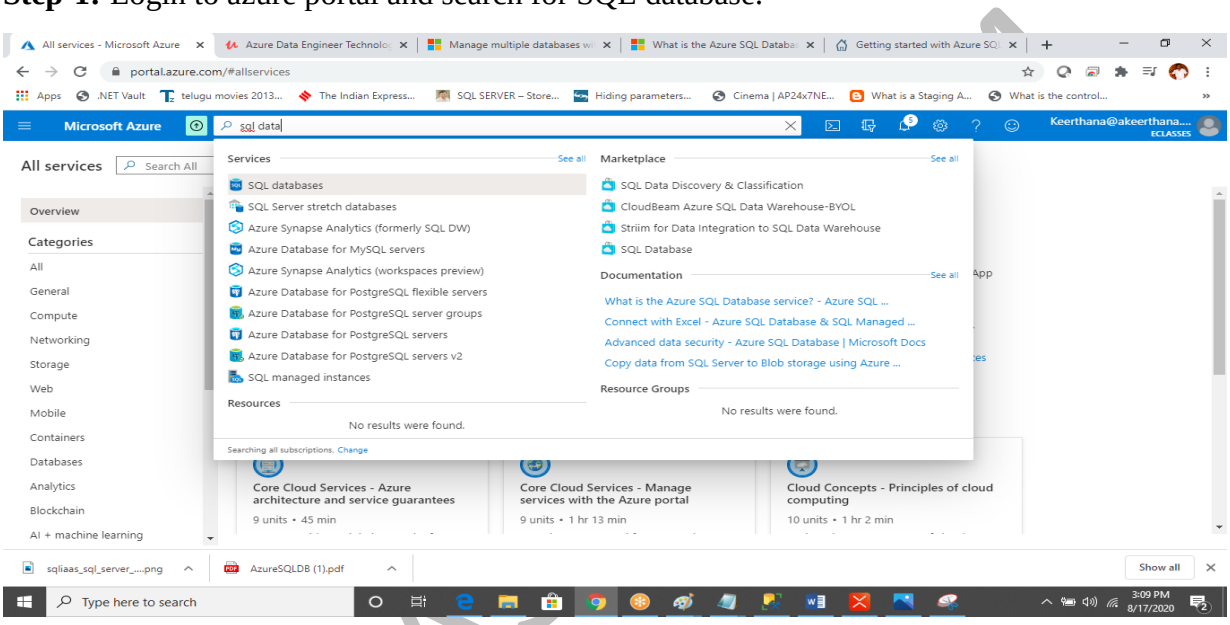
- **SQL Server in Fully Managed Virtual Machine**:

- Installing SQL Server with windows or Linux Operating system platform inside Virtual Machine with dedicated compute storage without managing hardware.
  This is 100 % similar like on premises instance based SQL installation.
  Flexibility to choose right version of SQL server according to on premises.

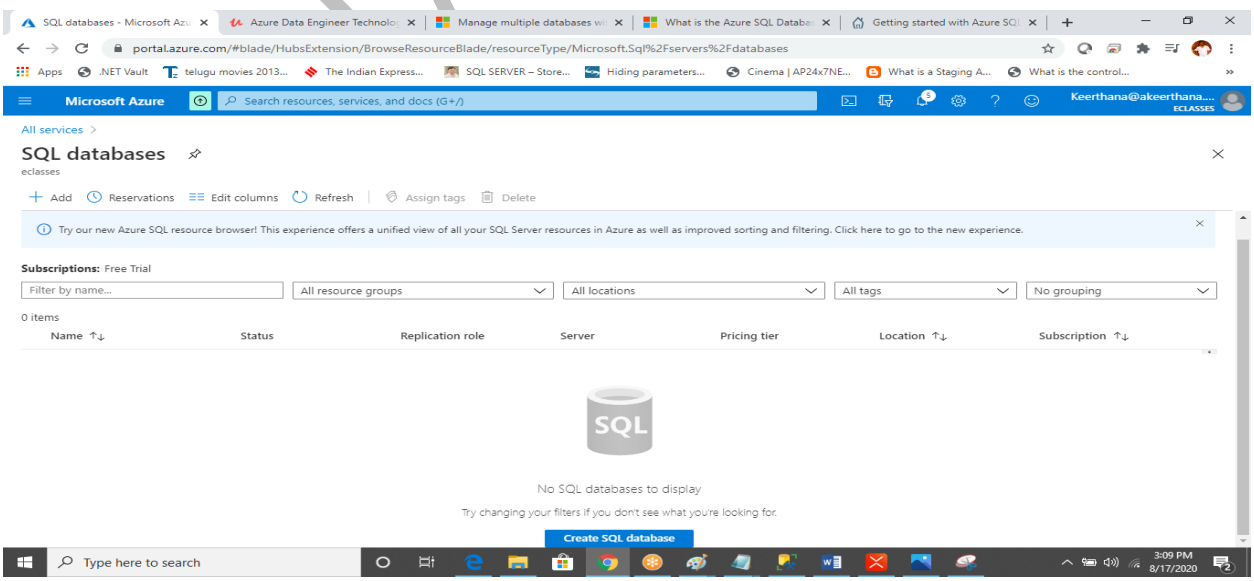
Microsoft Azure is not responsible for below Activates.
Maintaining Operating Systems
Maintain SQL Server Backups and Restore
Maintain High Availability
Maintain Disaster Recovery
Maintain Performance
Maintain Security

How to create Azure SQL Server

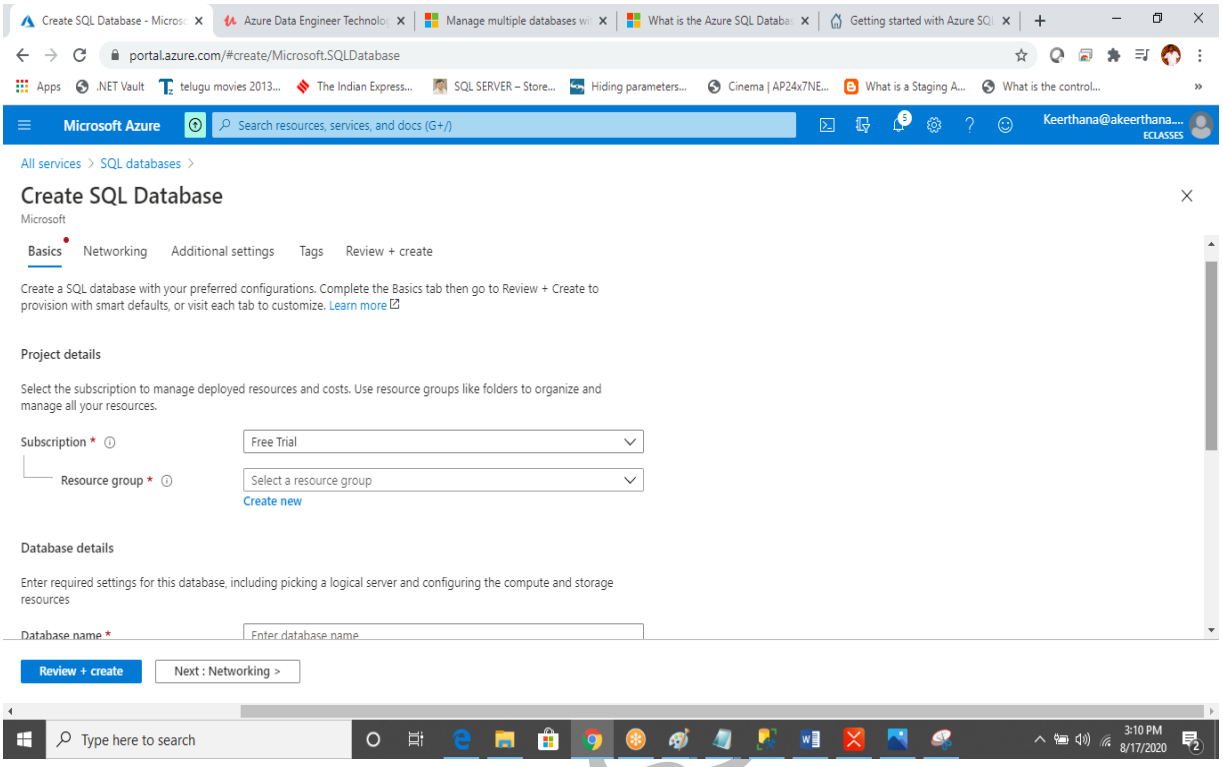
Step-1: Login to azure portal and search for SQL database.



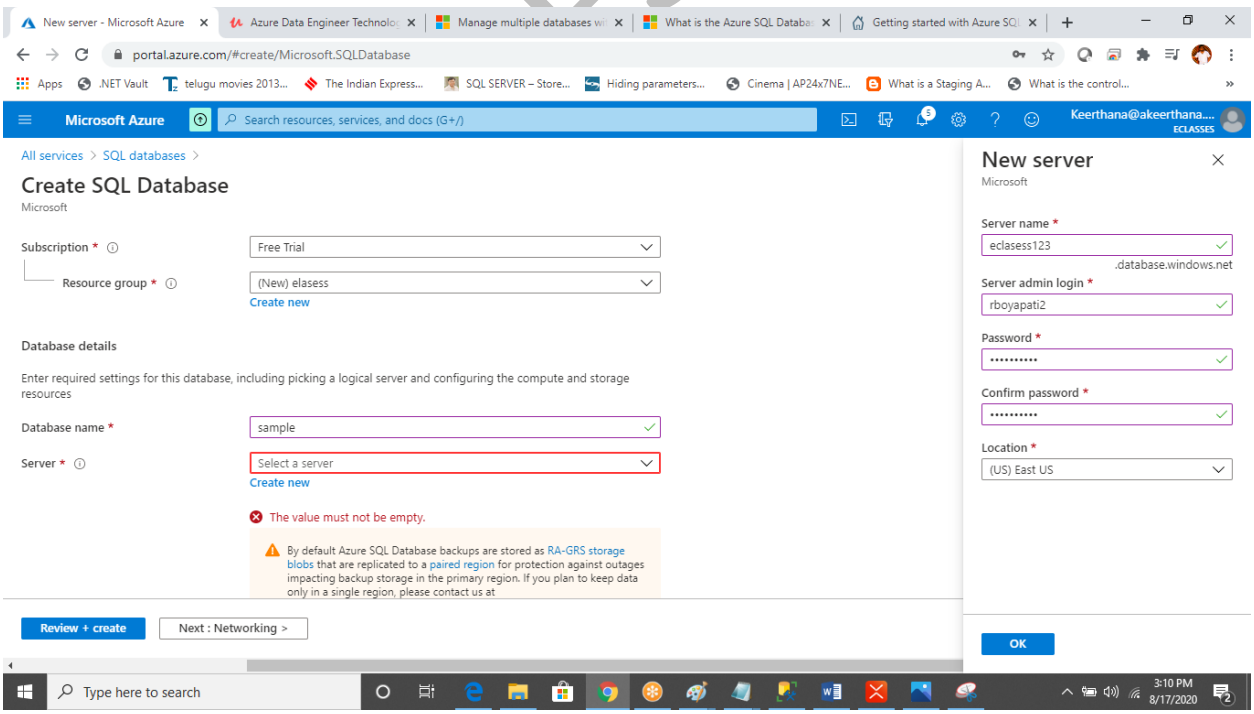
Step-2: Click on Add Azure SQL database.



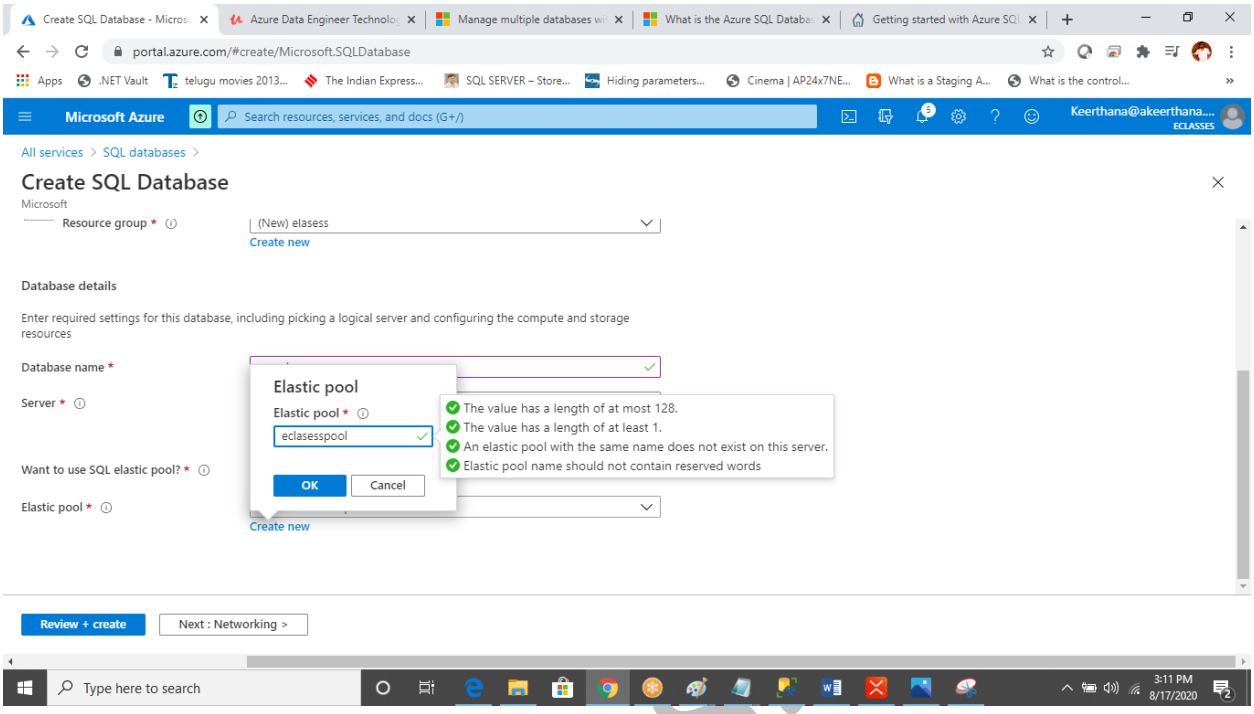
Step-3: Provide subscription and resource group.



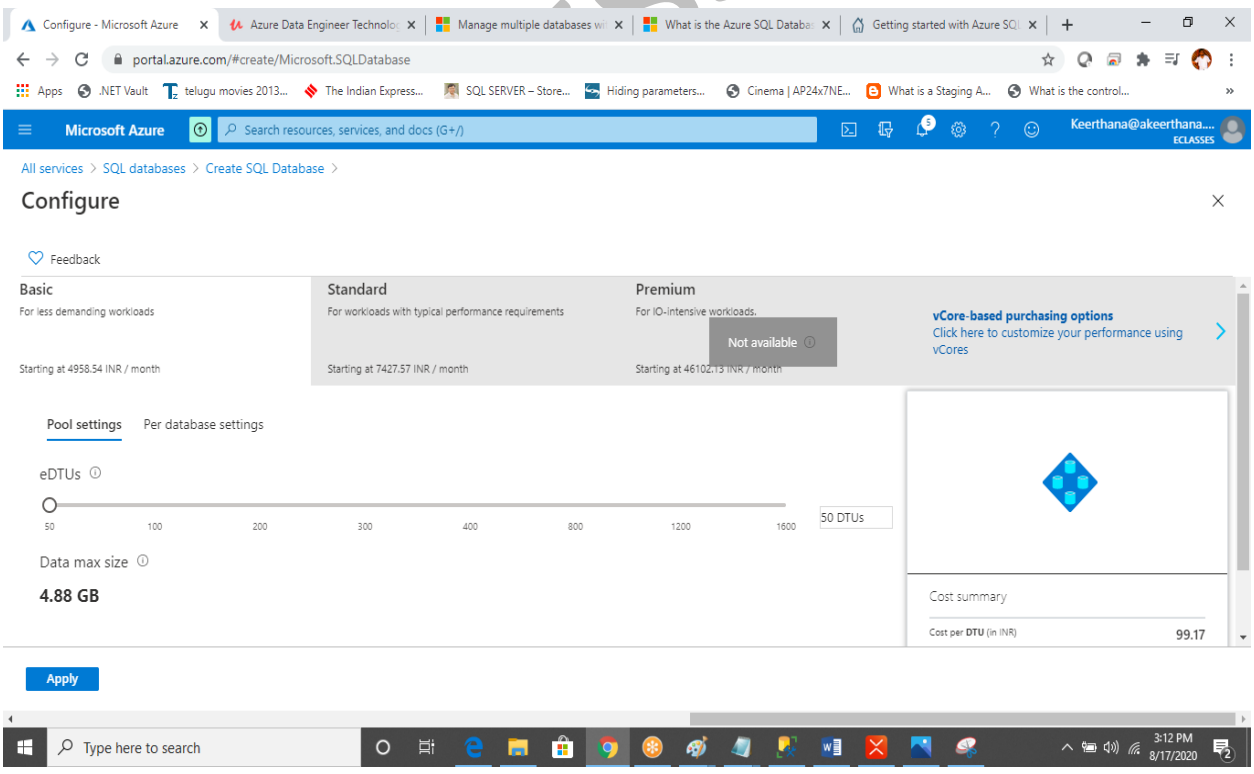
Step-4: Enter database Name and server Name login name password.



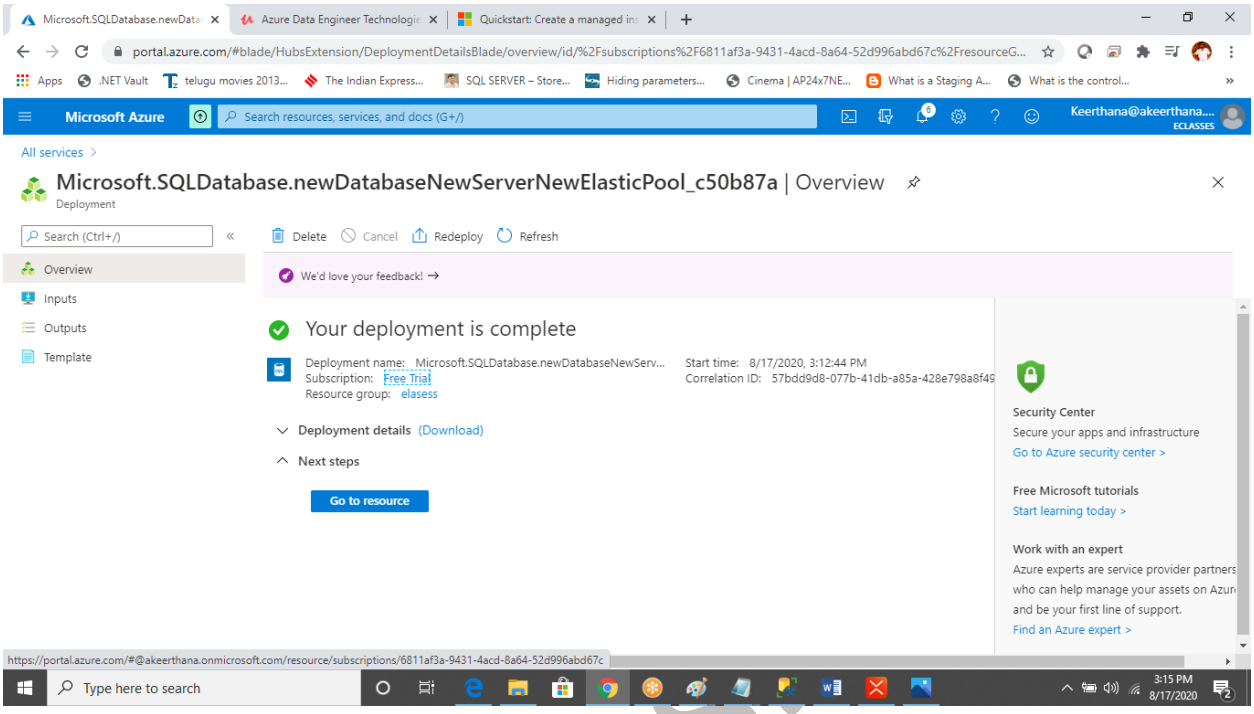
Step-5: Create Elastic pool.



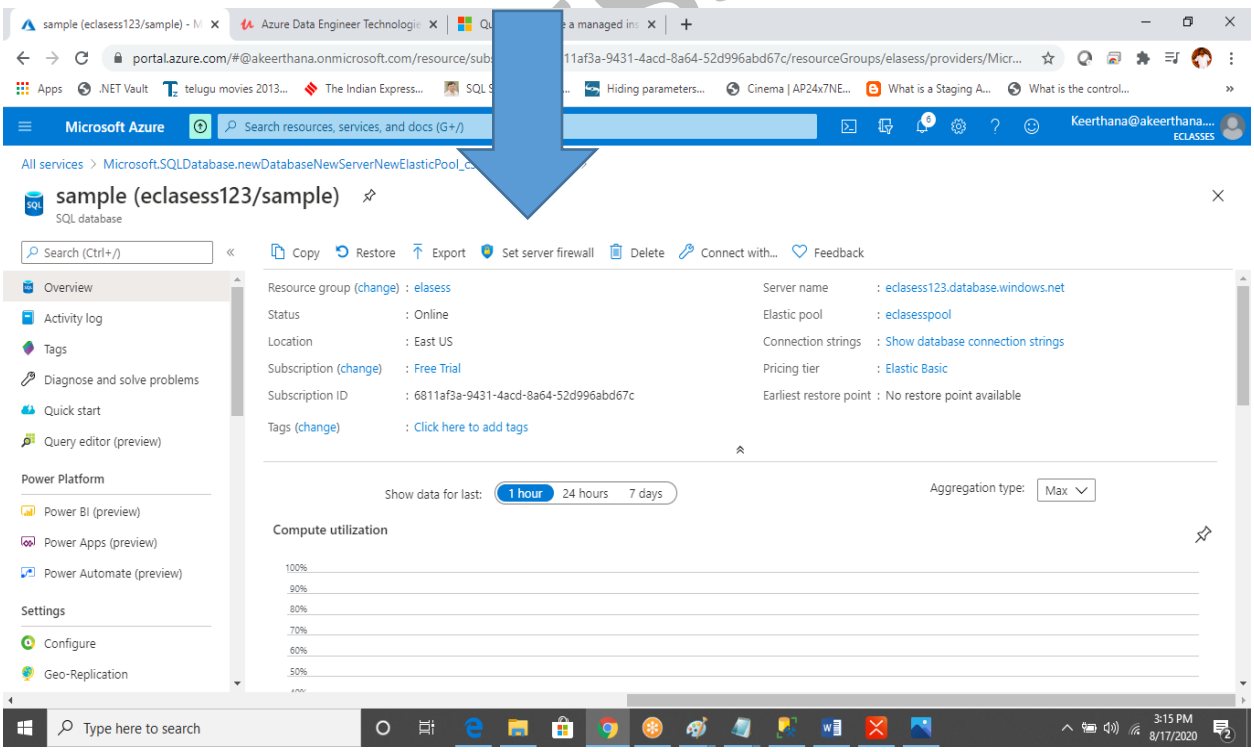
Step-6: Select Compute + Storage DTU or Vcore by click on configure elastic pool.



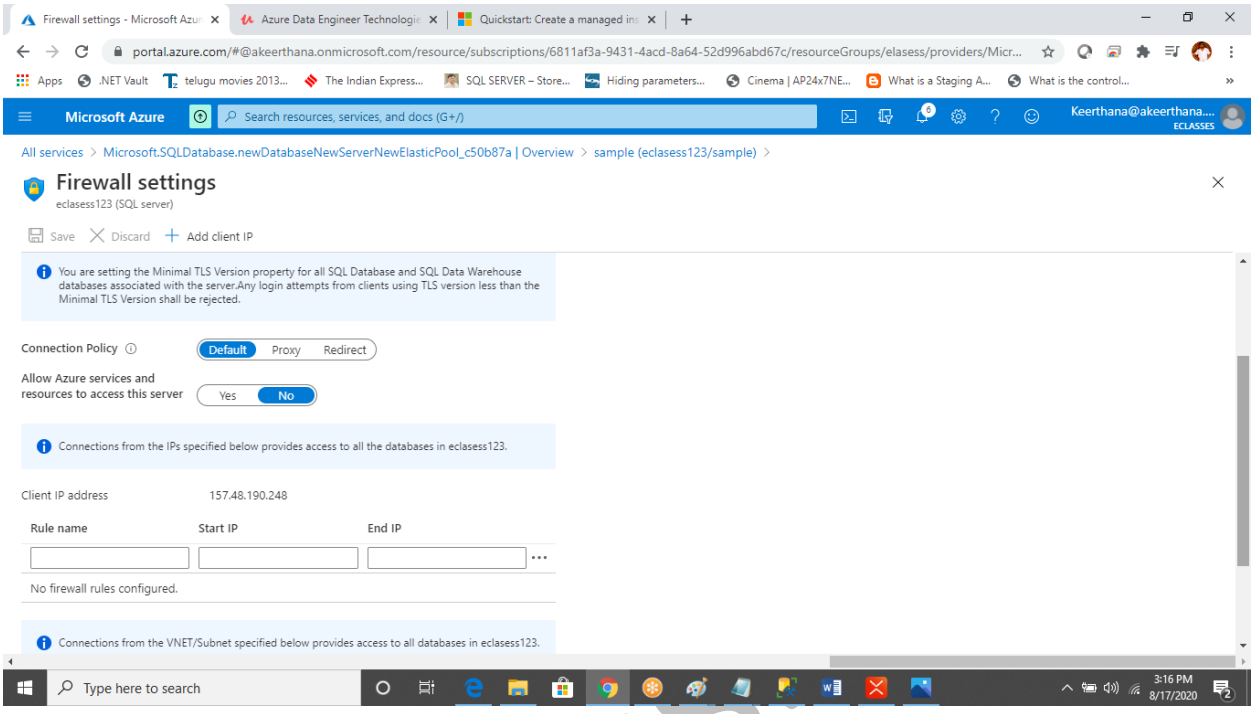
Step -7: Click on got resource.



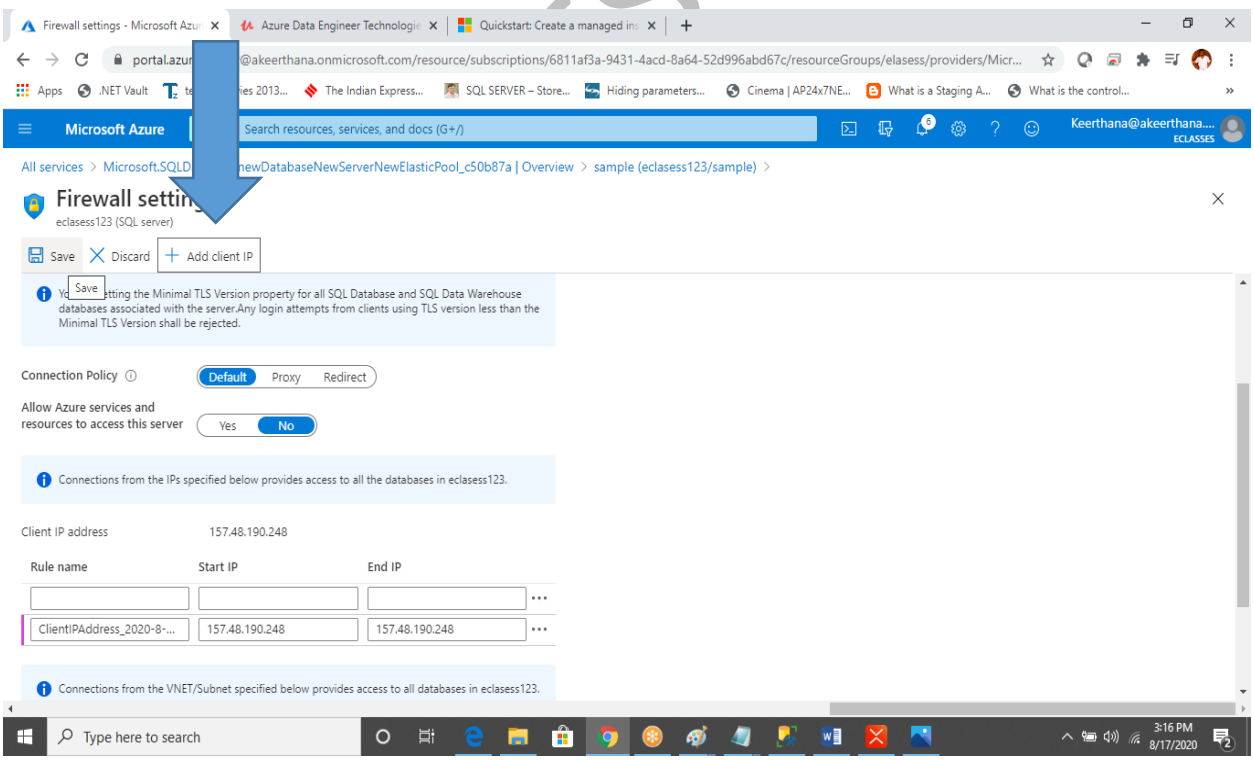
Step-8: Click on set server firewall.



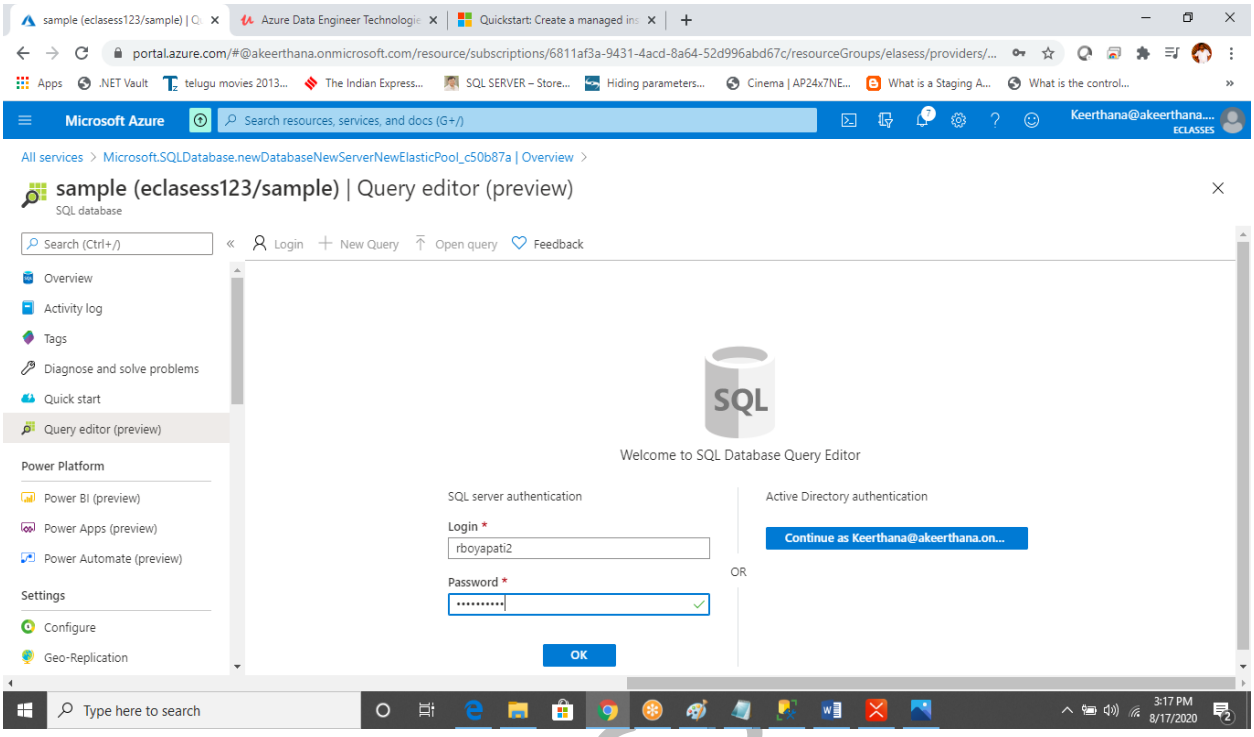
Step-9: click on add client IP



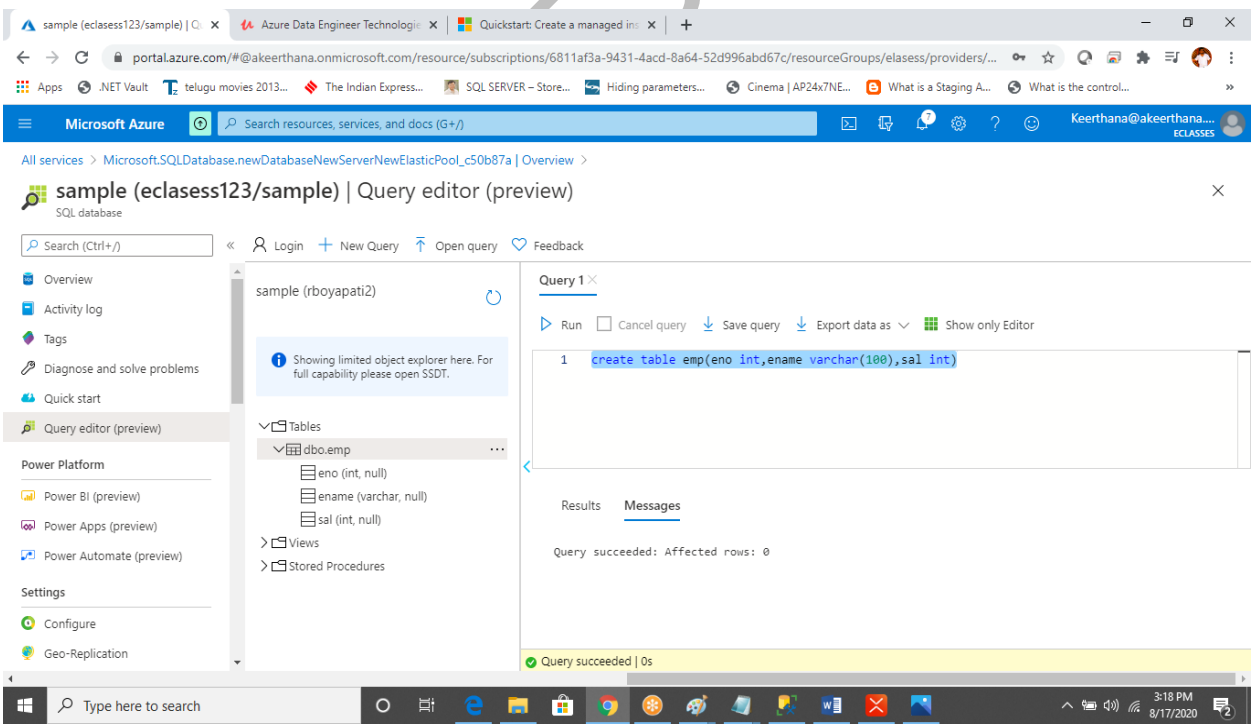
Step-10: Client IP has added**.**



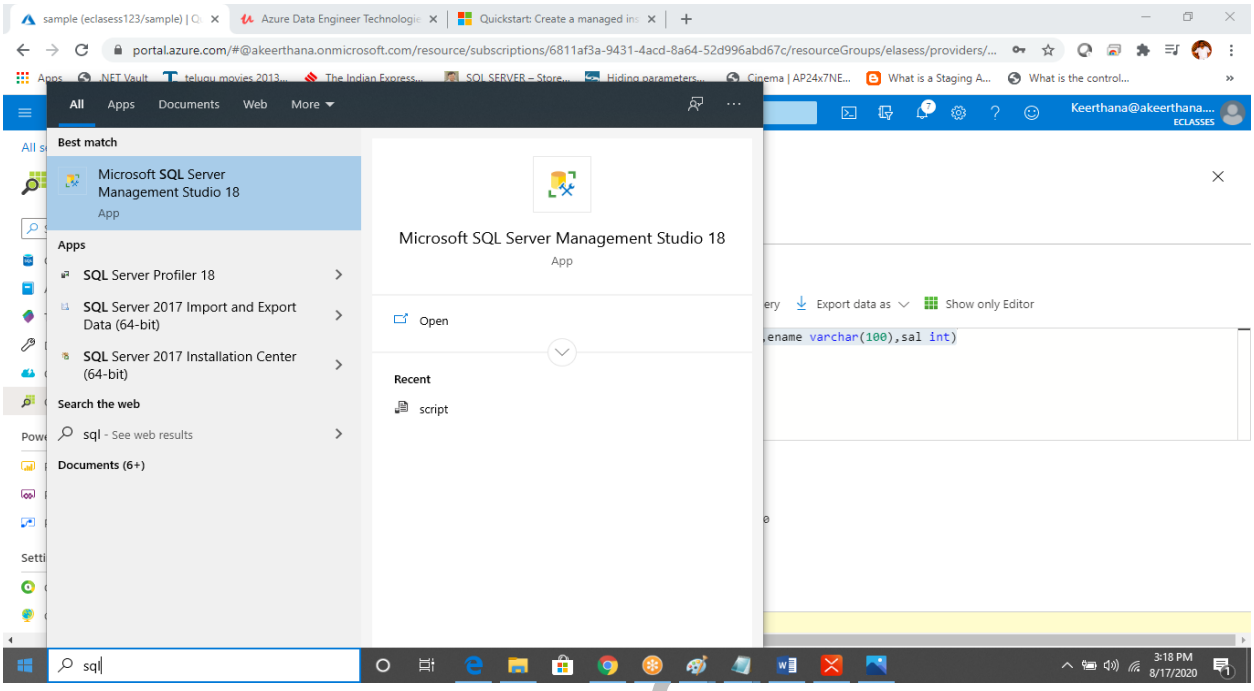
Step-11: Connect SQL server using Query Editor by entering username and password.



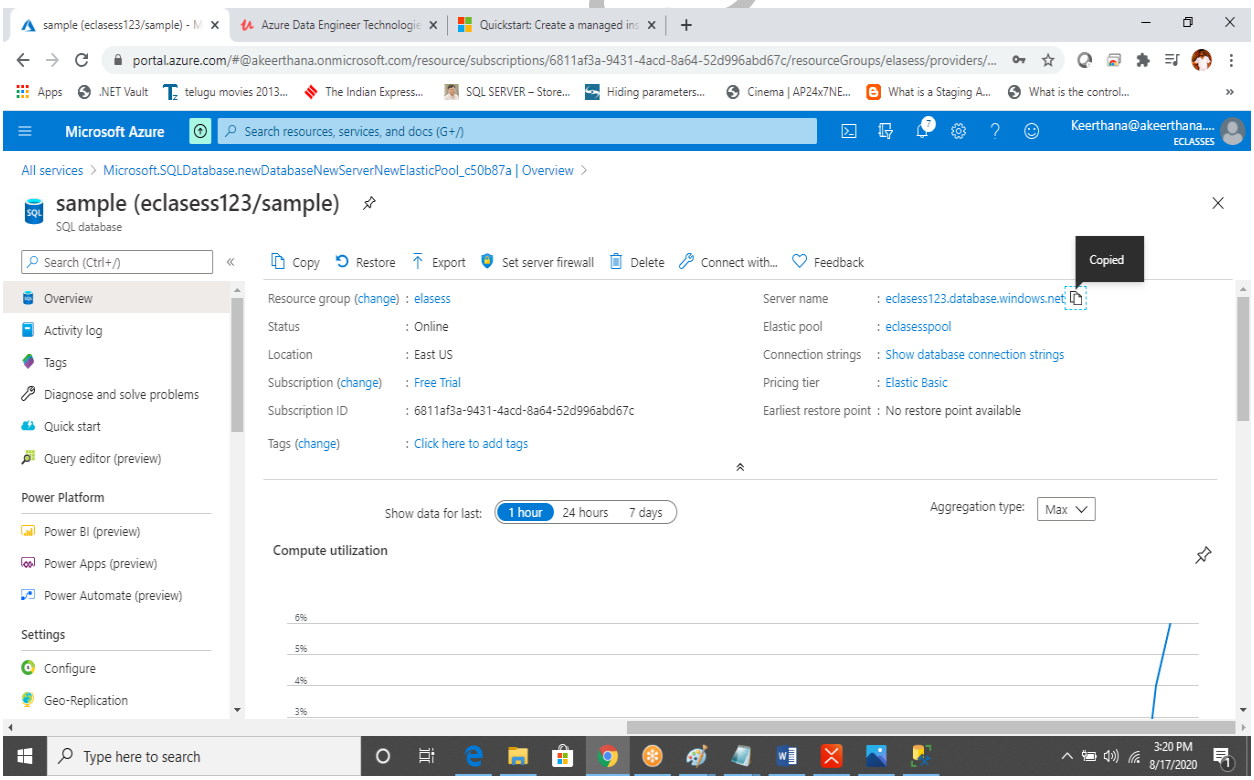
Step-12: Create table by using below script.



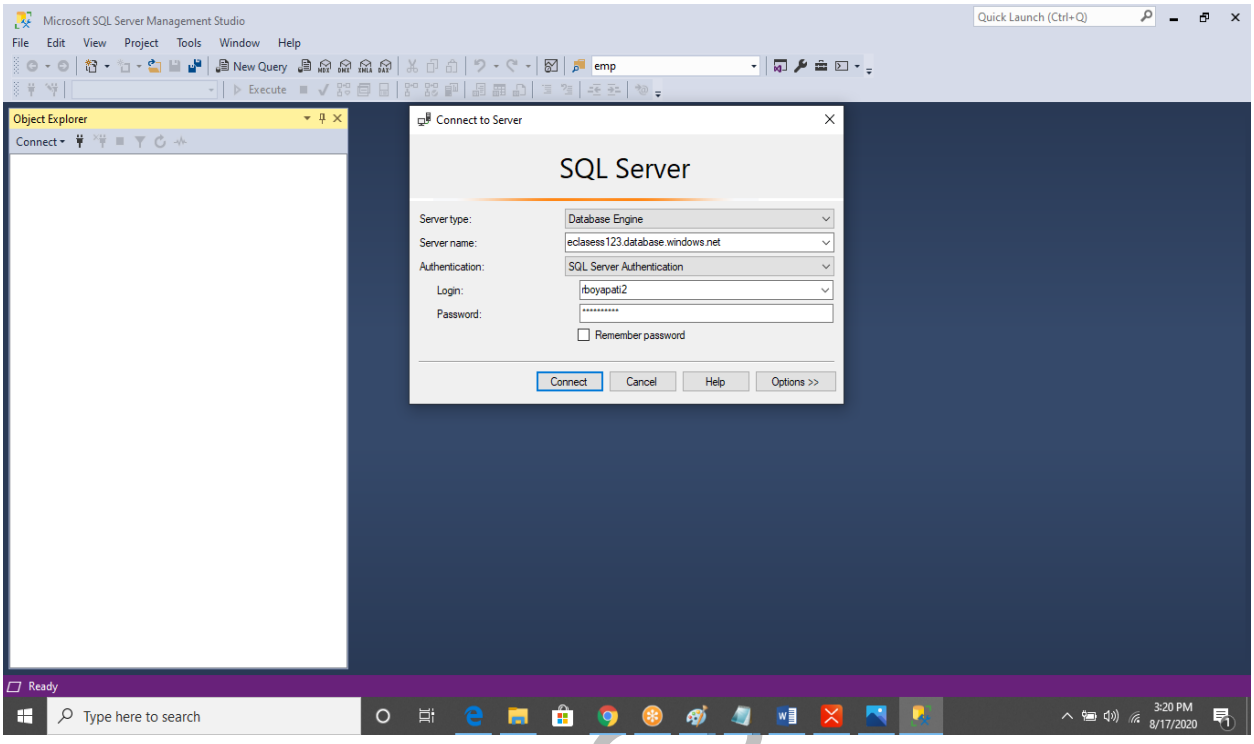
Step-13: How to connect Azure SQL database from on premises SQL Server Management studio.



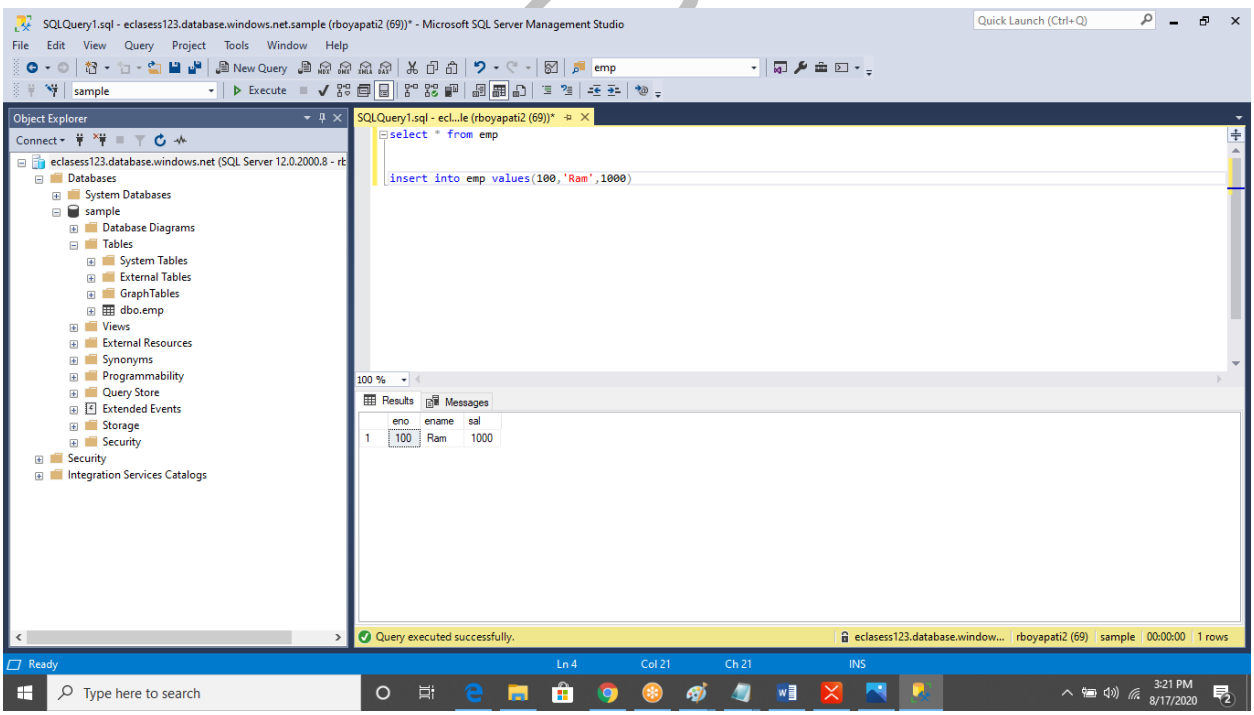
Step-14: Copy Server Name in Azure SQL database.



Step-15: Provide Server name and username and password.

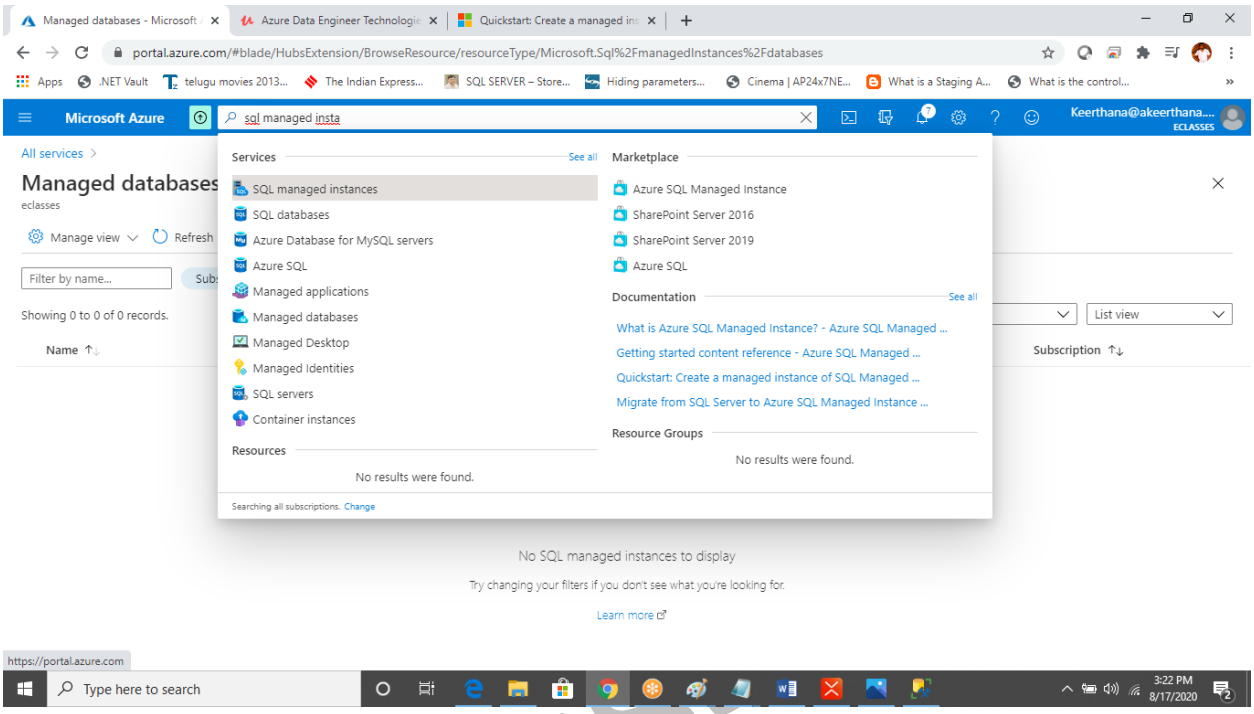


Step-16: insert some sample data into table by writing below statements.

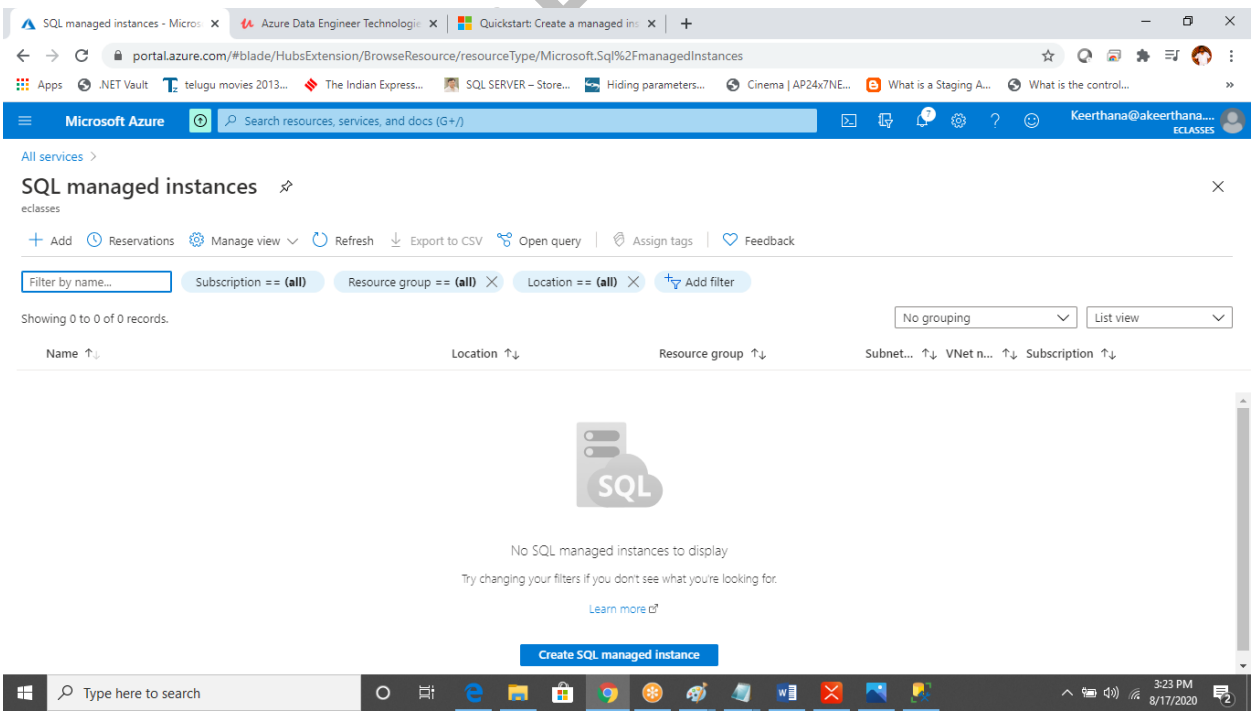


How to Create Managed Instance SQL in Azure:

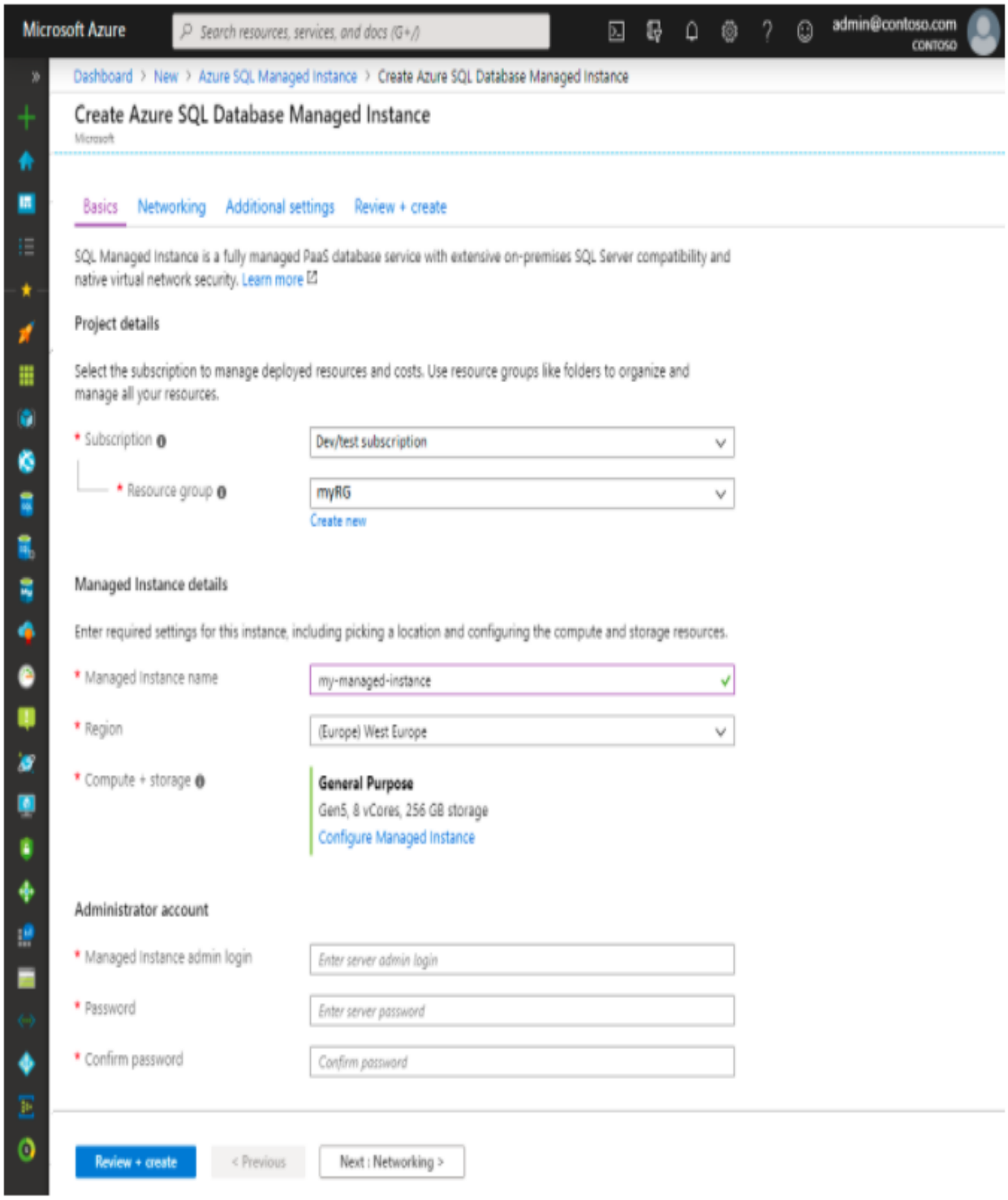
Step-1: login to azure portal and search for SQL managed instance

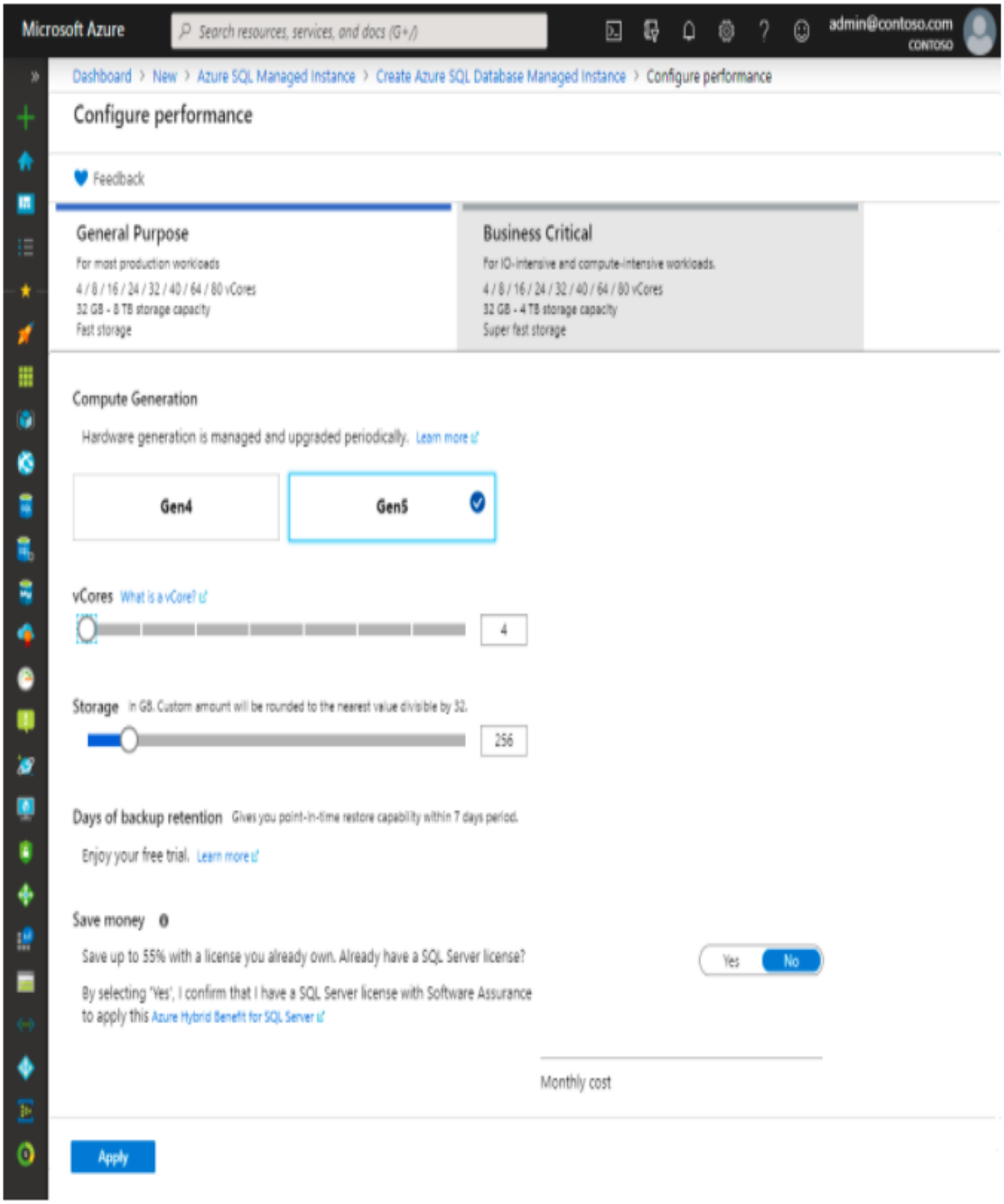


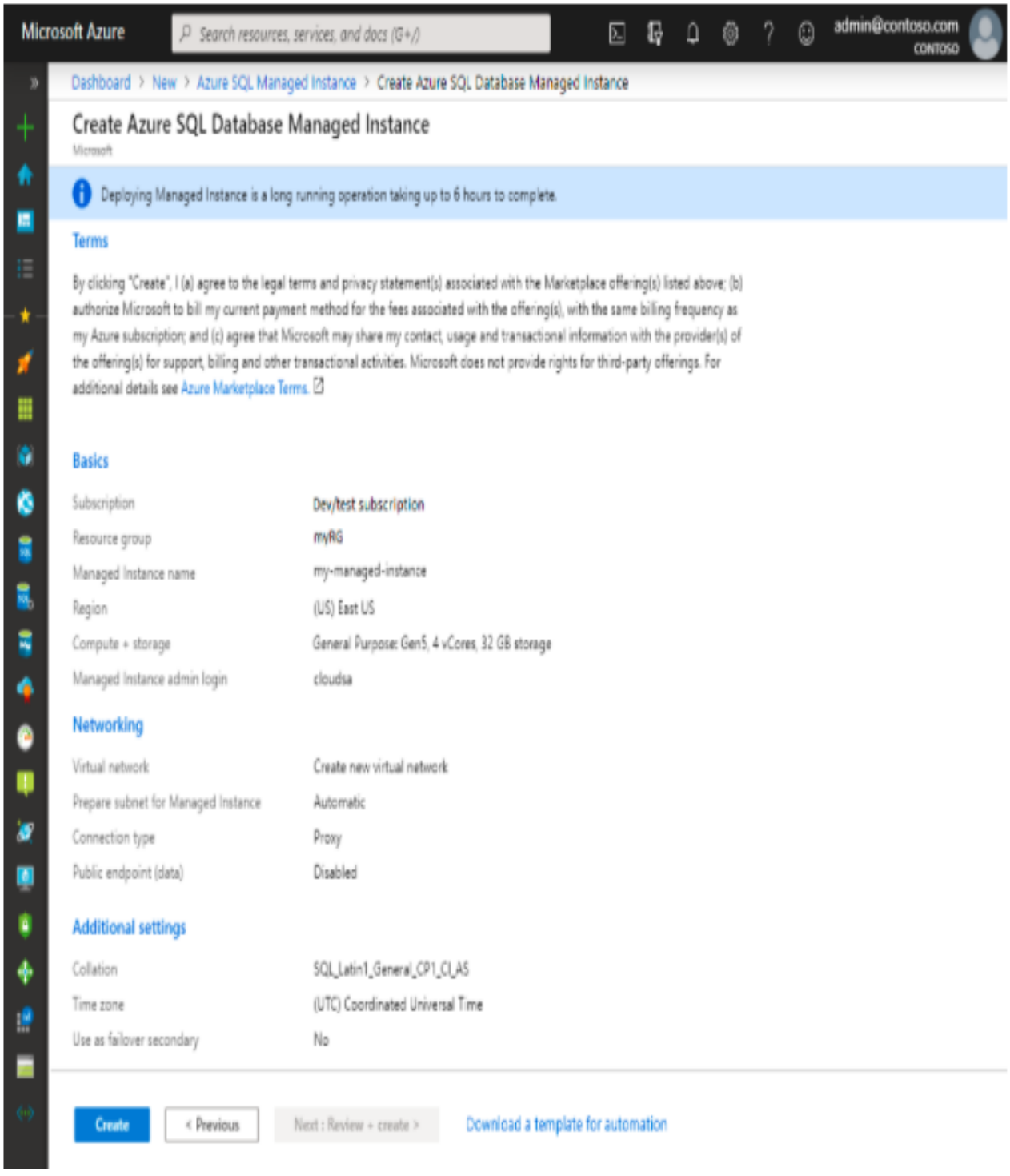
Step-2: Click add SQL Managed instance.



Step-3: provide all features subscription, resource group, managed instance name(any name), region, compute and storage. Enter username and set some password.

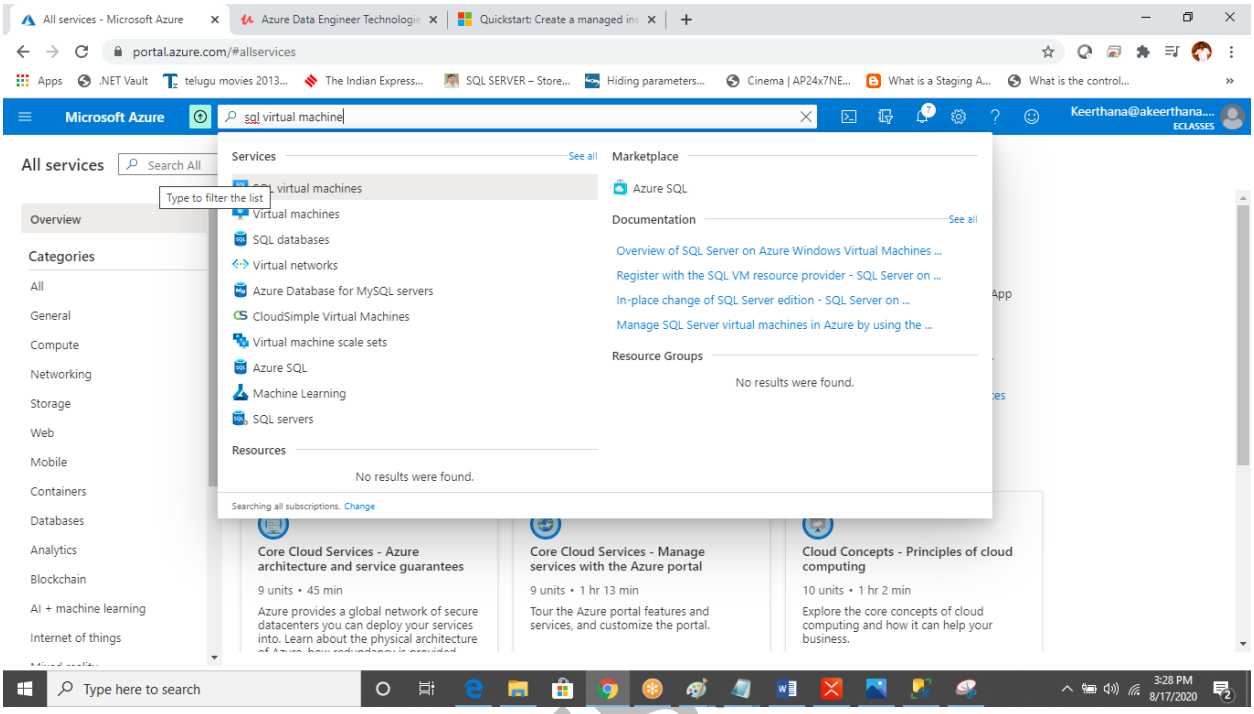



Microsoft Azure
Search resources, services, and docs (G+/)
admin@contoso.com
CONTOSO
Dashboard > New > Azure SQL Managed Instance > Create Azure SQL Database Managed Instance > Configure performance
Configure performance
Feedback
General Purpose
For most production workloads
4 / 8 / 16 / 24 / 32 / 40 / 64 / 80 vCores
32 GB - 8 TB storage capacity
Fast storage
Business Critical
For IO-intensive and compute-intensive workloads.
4 / 8 / 16 / 24 / 32 / 40 / 64 / 80 vCores
32 GB - 4 TB storage capacity
Super fast storage
Compute Generation
Hardware generation is managed and upgraded periodically. Learn more
Gen4
Gen5
vCores What is a vCore?
4
Storage In GB. Custom amount will be rounded to the nearest value divisible by 32.
256
Days of backup retention Gives you point-in-time restore capability within 7 days period.
Enjoy your free trial. Learn more
Save money
Save up to 55% with a license you already own. Already have a SQL Server license?
Yes
No
By selecting 'Yes', I confirm that I have a SQL Server license with Software Assurance to apply this Azure Hybrid Benefit for SQL Server
Monthly cost
Apply

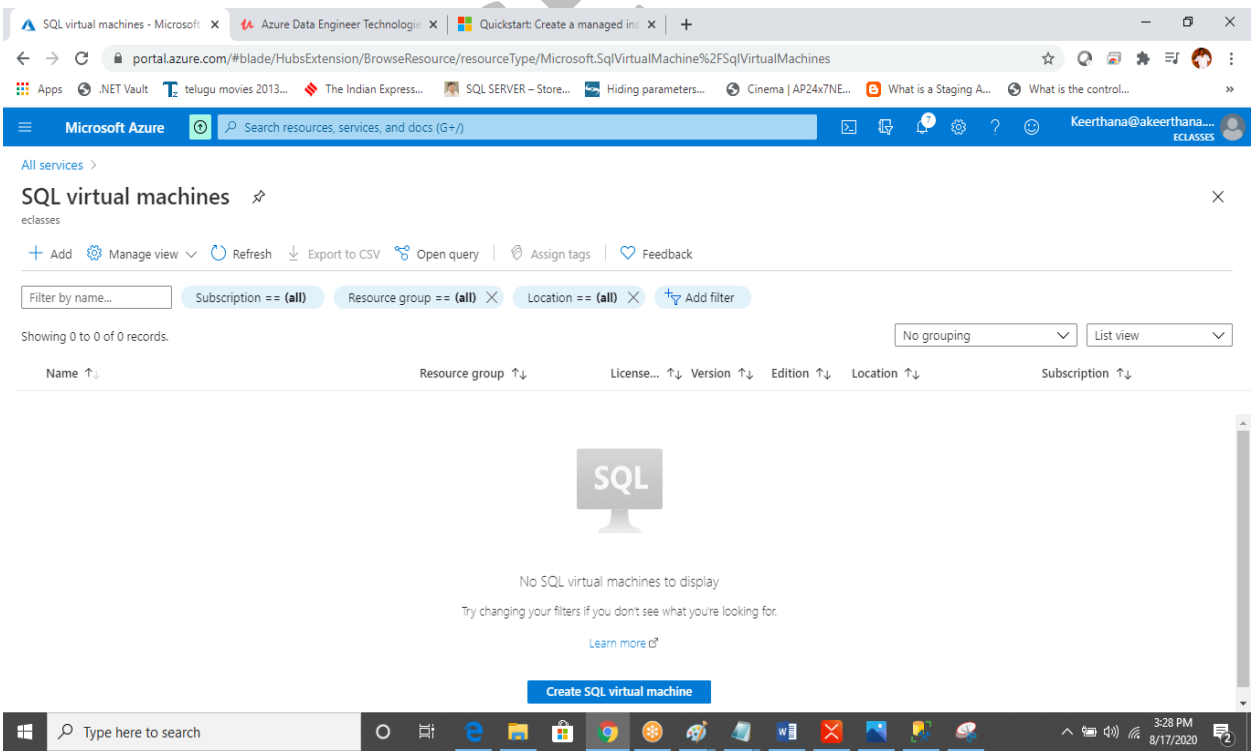

Microsoft Azure
Search resources, services, and docs (G+/)
admin@contoso.com
CONTOSO
Dashboard > New > Azure SQL Managed Instance > Create Azure SQL Database Managed Instance
Create Azure SQL Database Managed Instance
Microsoft
Deploying Managed Instance is a long running operation taking up to 6 hours to complete.
Terms
By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. For additional details see Azure Marketplace Terms.
Basics
Subscription Dev/test subscription
Resource group myRG
Managed Instance name my-managed-instance
Region (US) East US
Compute + storage General Purpose: Gen5, 4 vCores, 32 GB storage
Managed Instance admin login cloudsa
Networking
Virtual network Create new virtual network
Prepare subnet for Managed Instance Automatic
Connection type Proxy
Public endpoint (data) Disabled
Additional settings
Collation SQL_Latin1_General_CP1_CI_AS
Time zone (UTC) Coordinated Universal Time
Use as failover secondary No
Create
< Previous
Next : Review + create >
Download a template for automation

How to create SQL Virtual Machine:

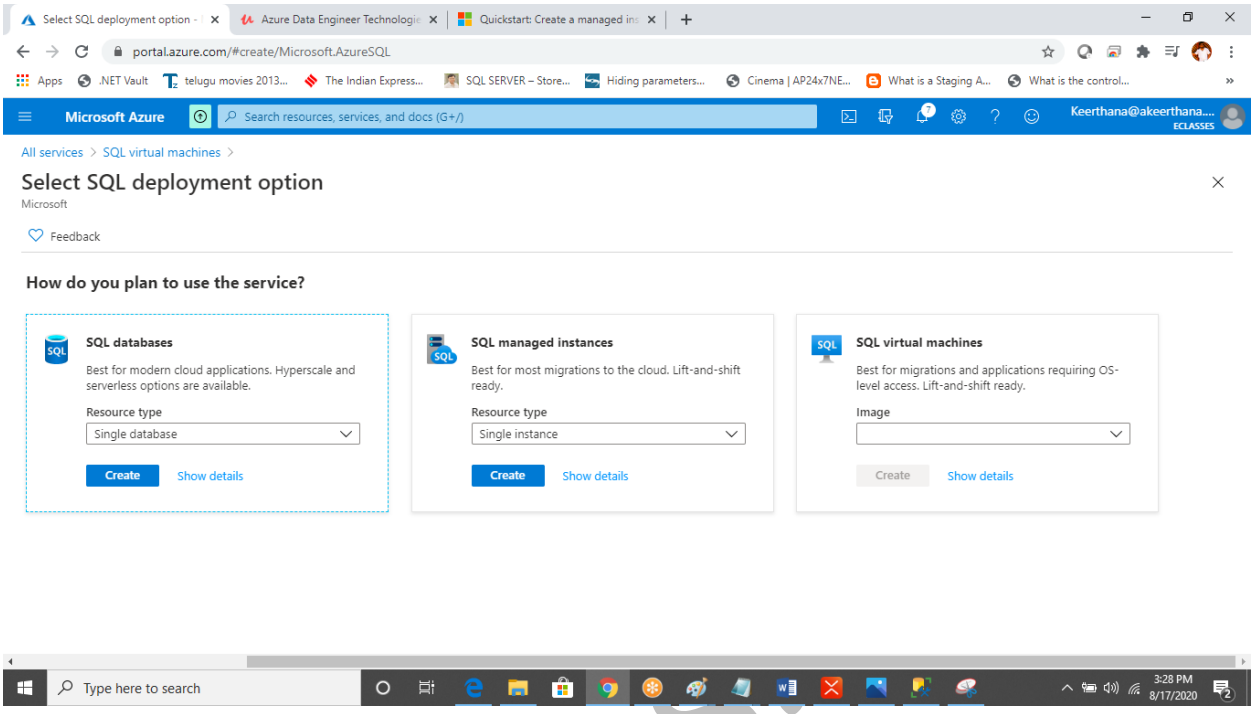
Step-1: Login to Azure portal and search for SQL virtual Machine.



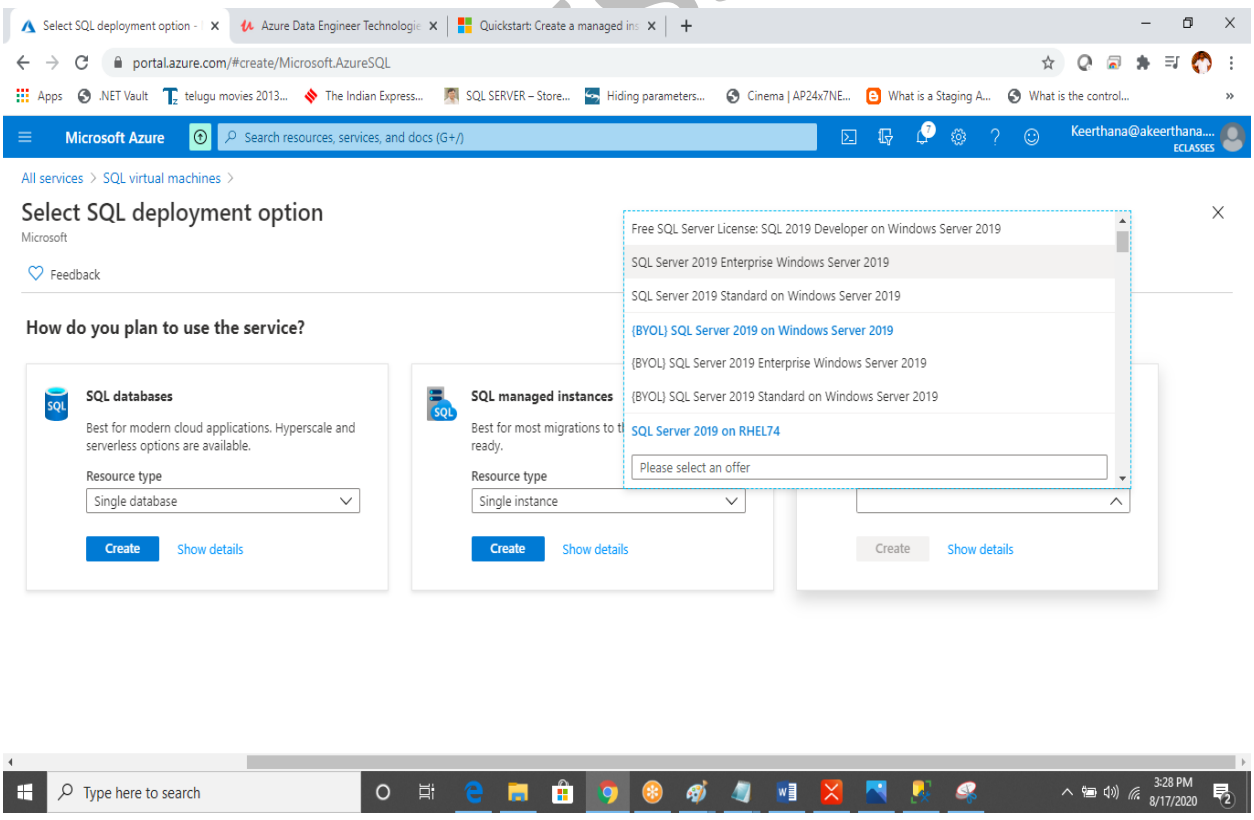
Step-2: Click add to create SQL virtual machine.



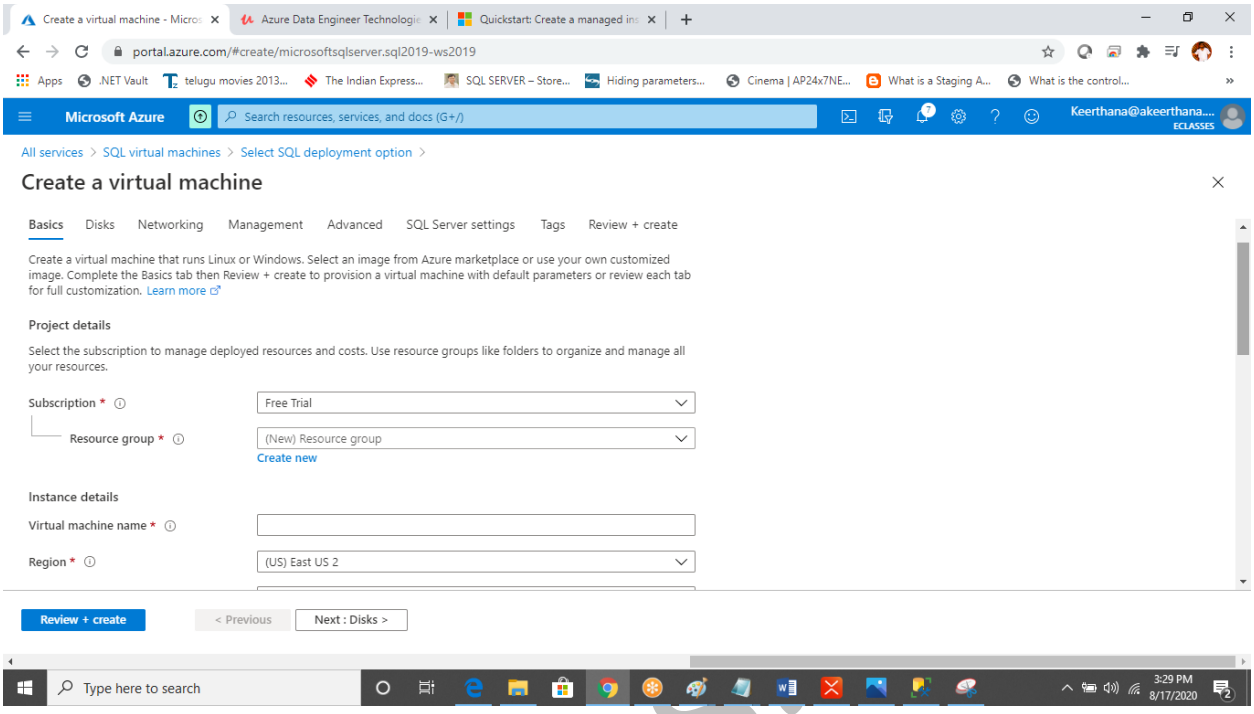
Step-3: Go to Select SQL virtual machine.



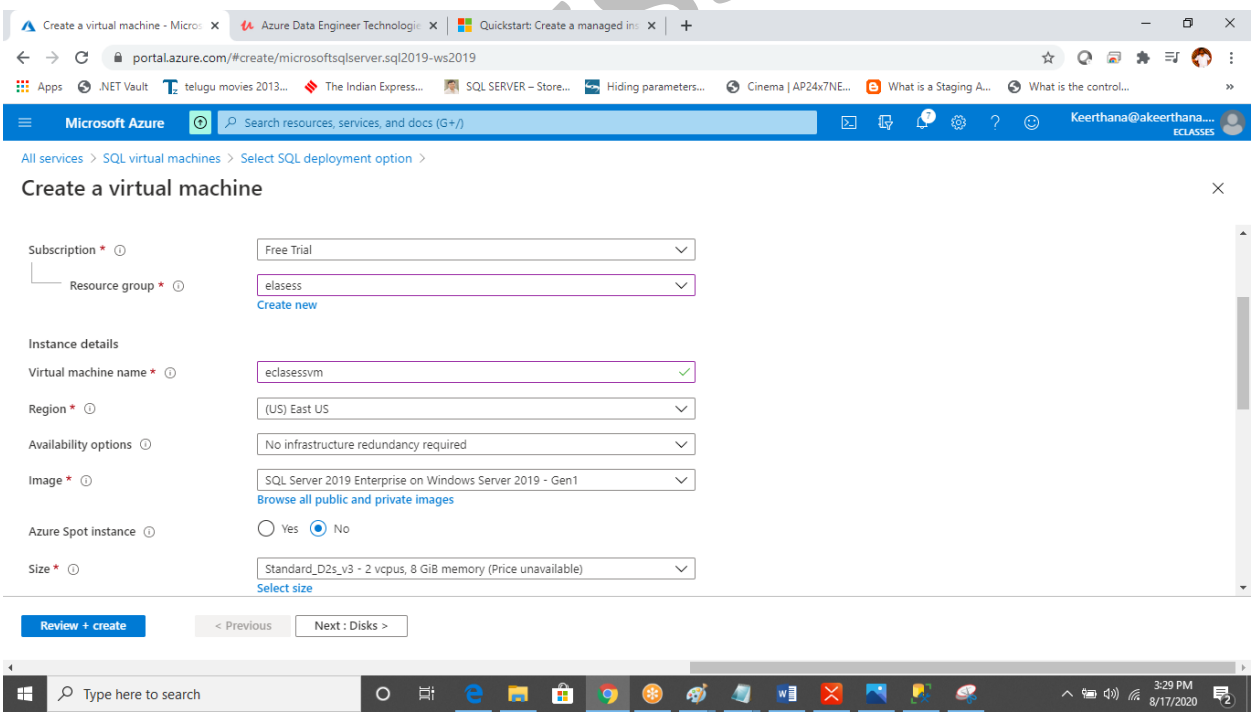
Step-4: Select image as SQL Server 2019 standard on windows server 2019 and click on create.



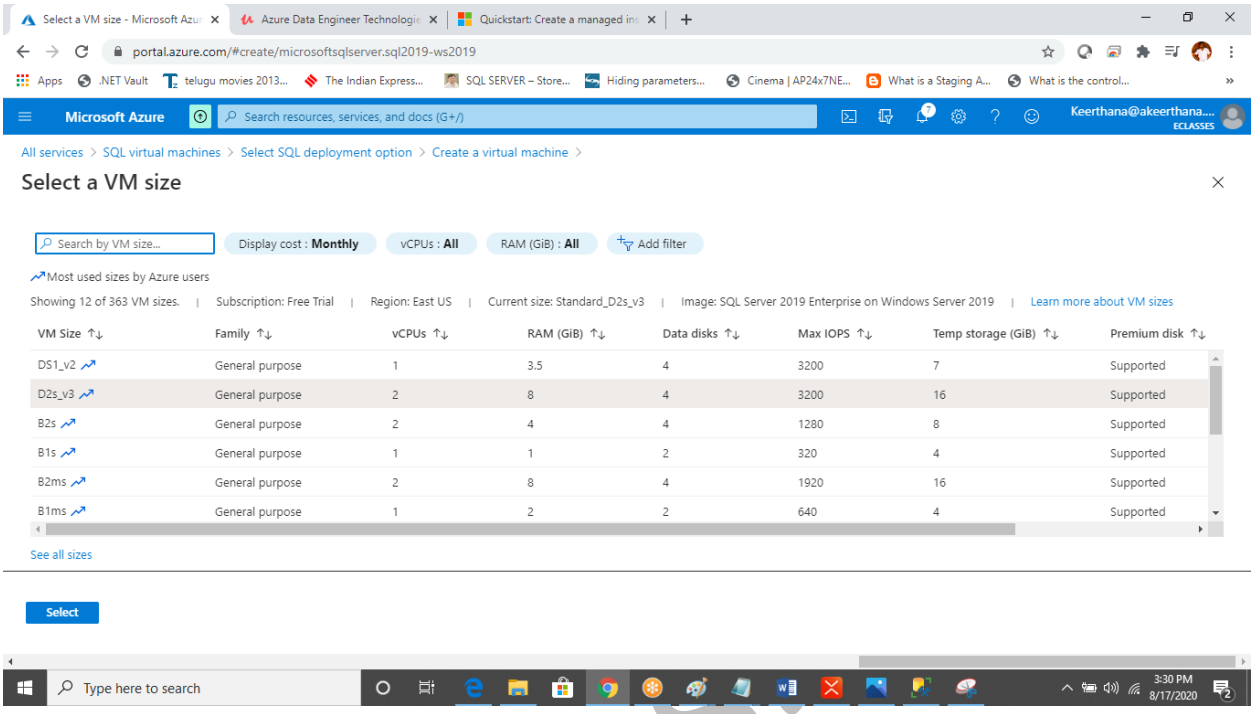
Step-5: provide Subscription, Resource group.



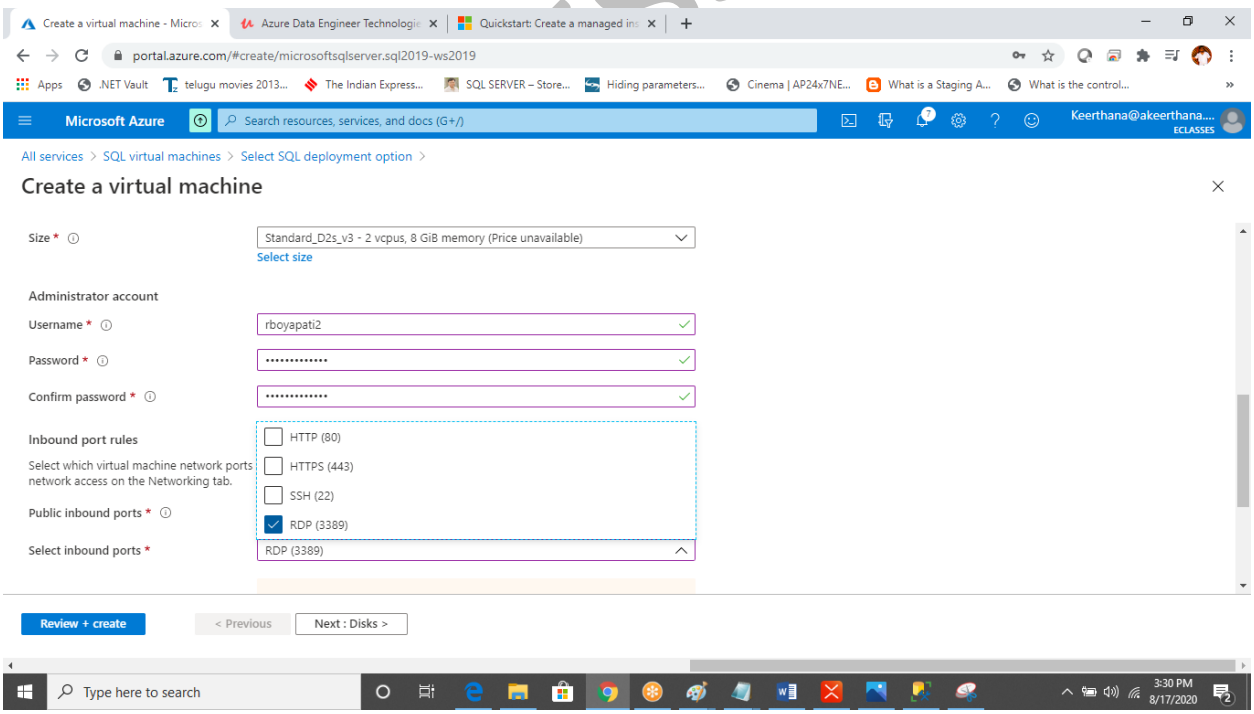
Step-6: provide virtual machine name (any name).



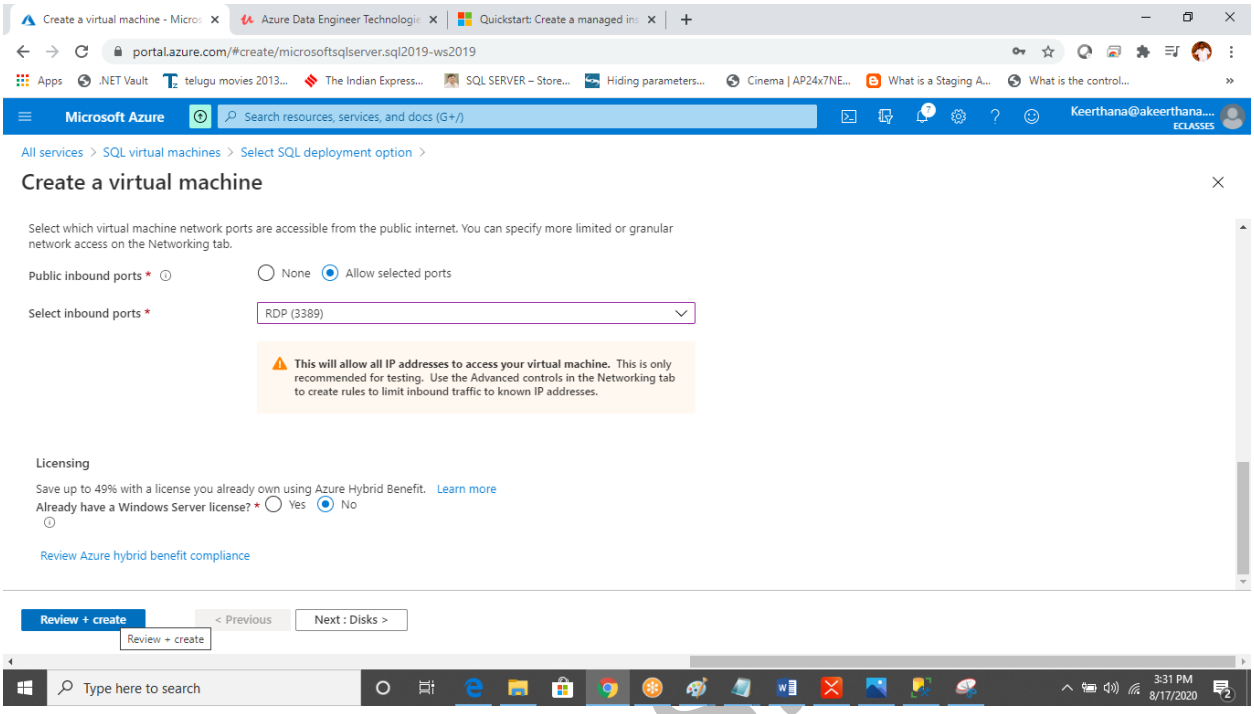
Step-7: Provide Virtual machine Size.



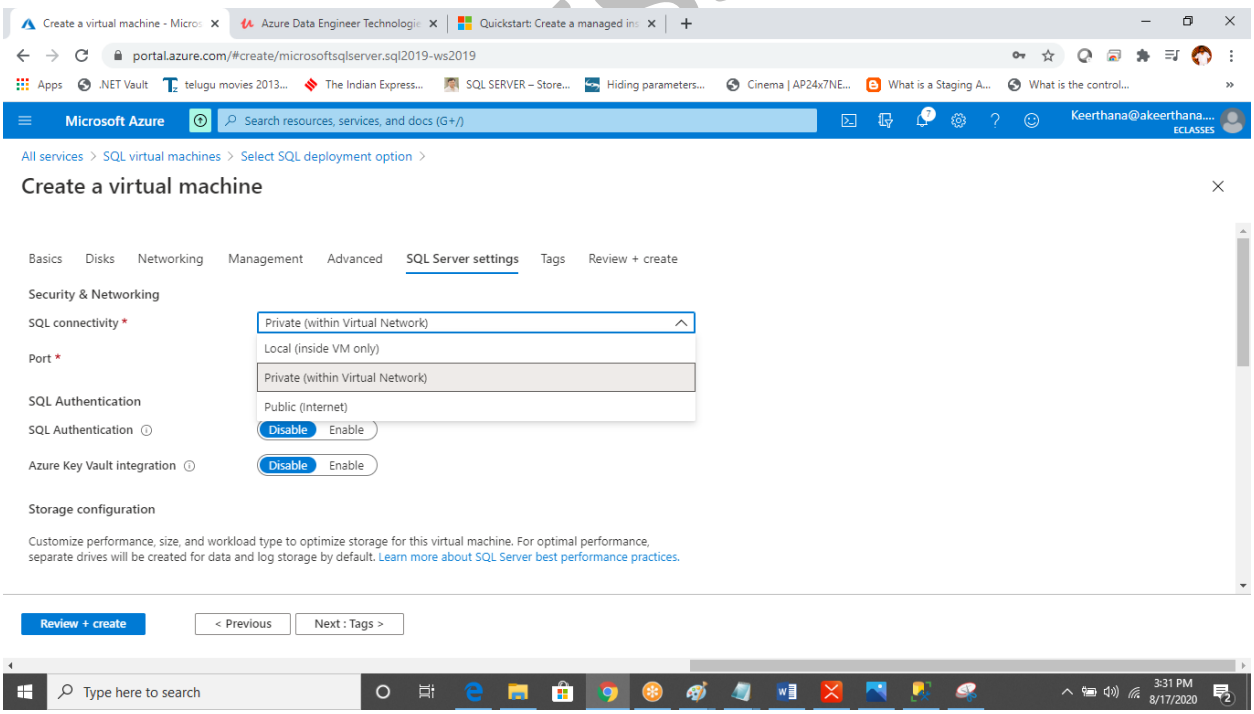
Step -8: Provide username and password.



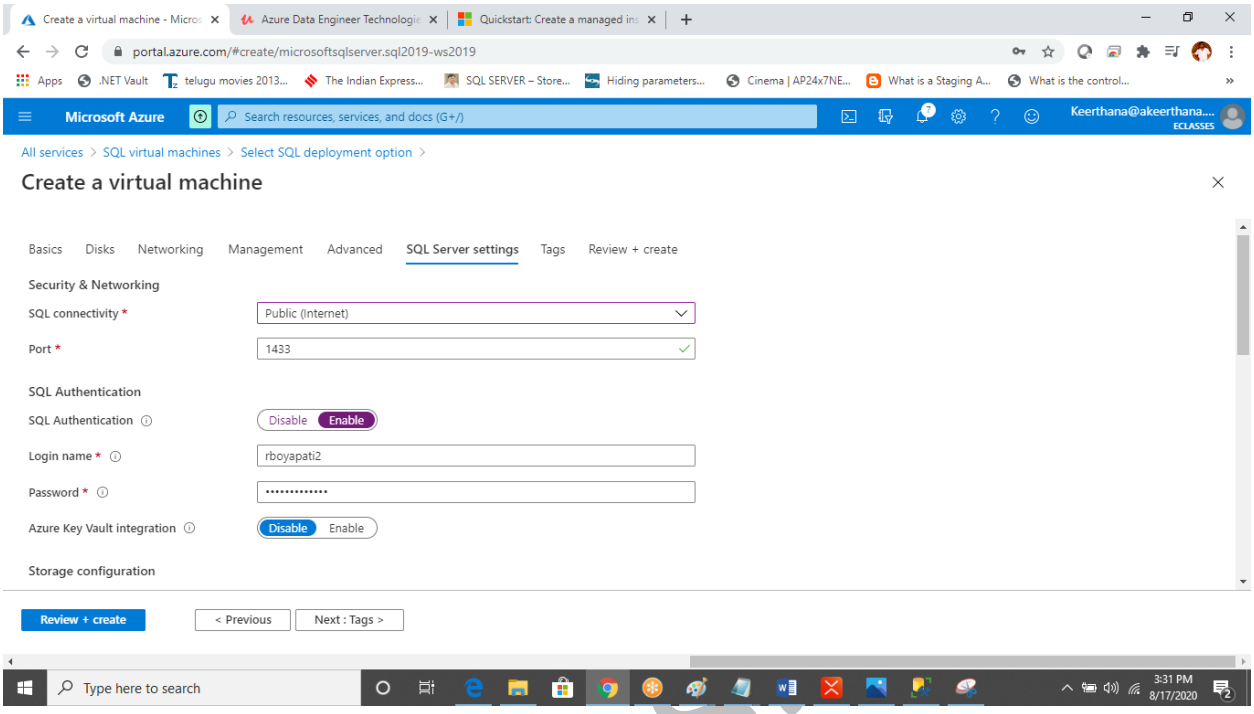
Step -9: Allow selected ports.



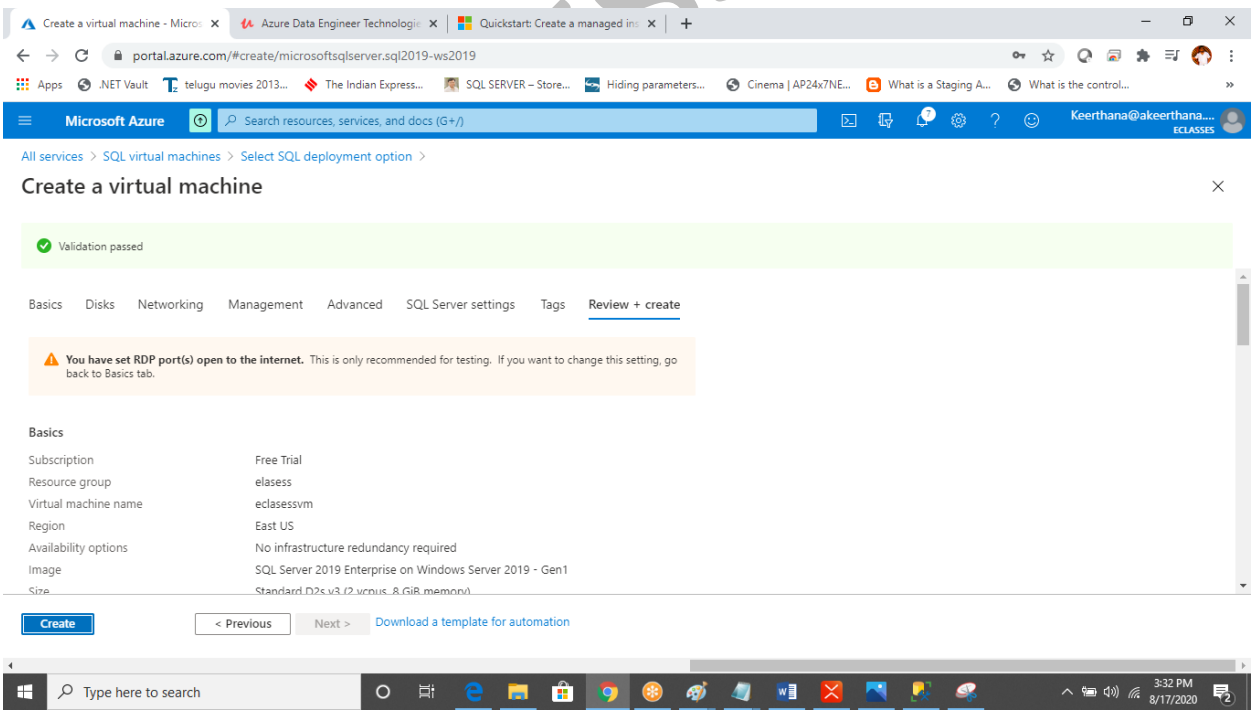
Step-10: Select SQL Connectivity Public (Internet).



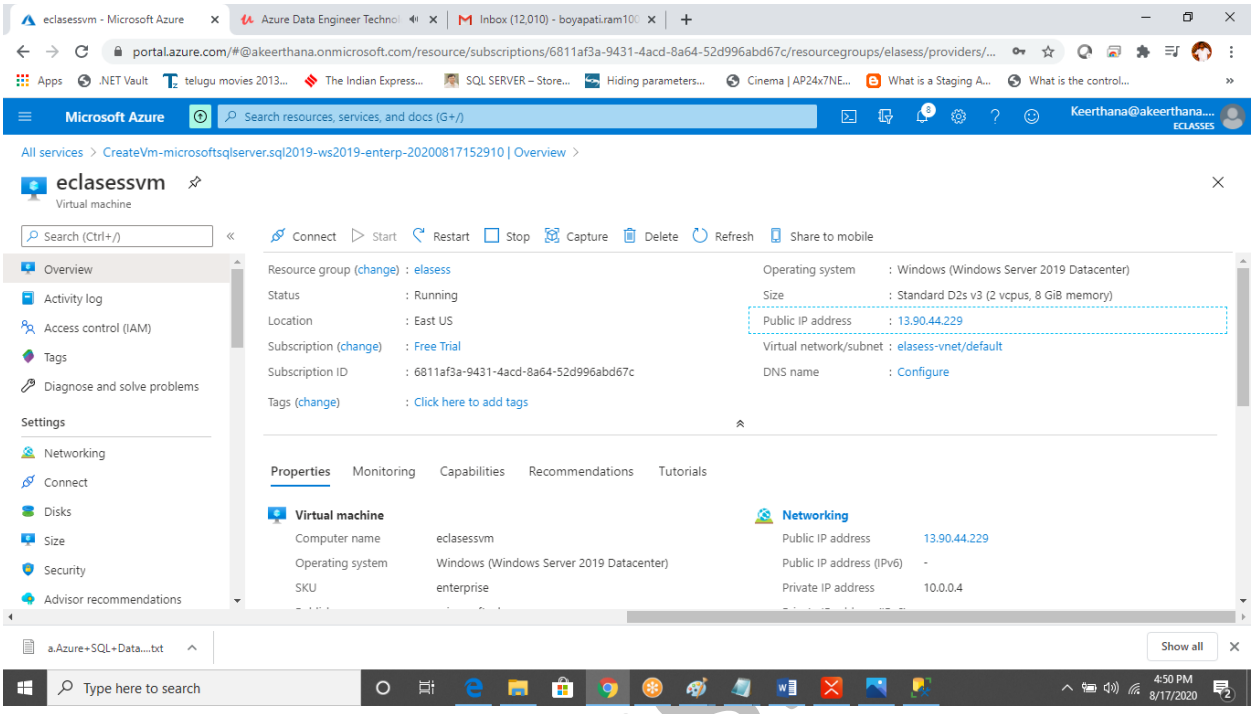
Step-11: Select SQL Authentication Enabled and click on review + create.



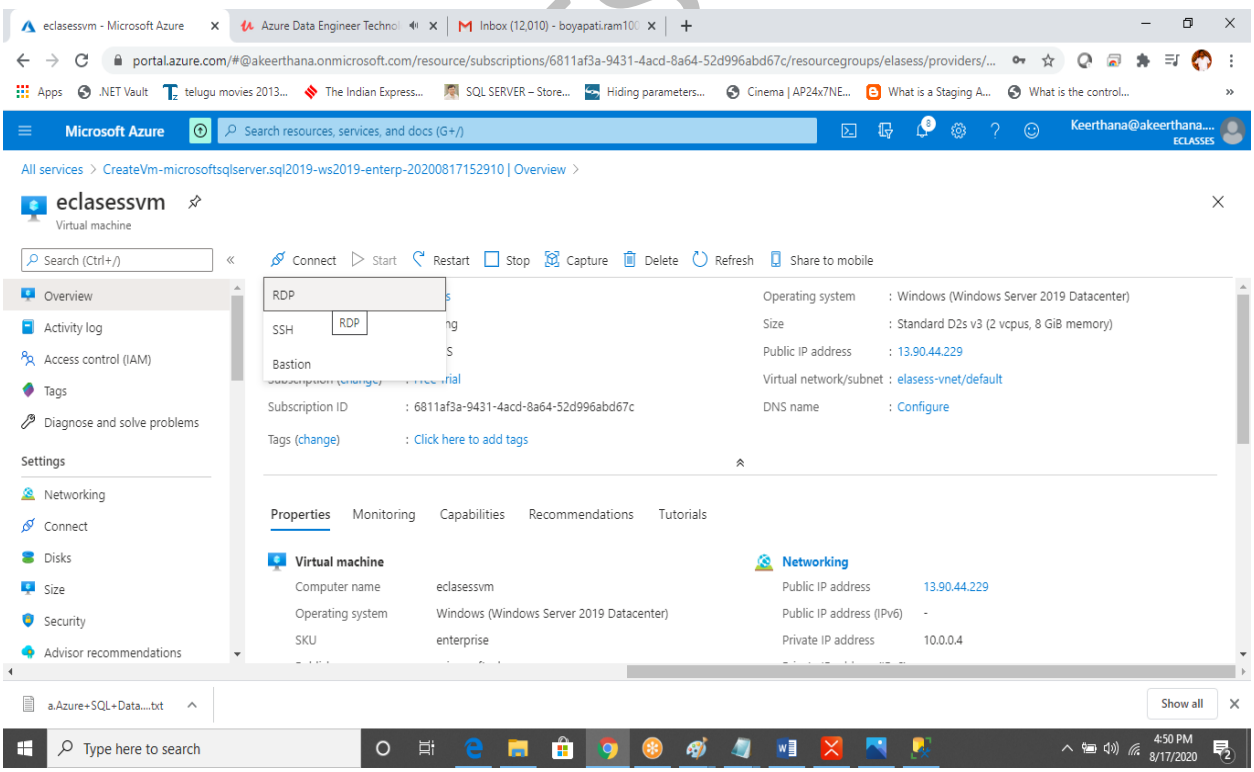
Step-12: Then Click on Create.



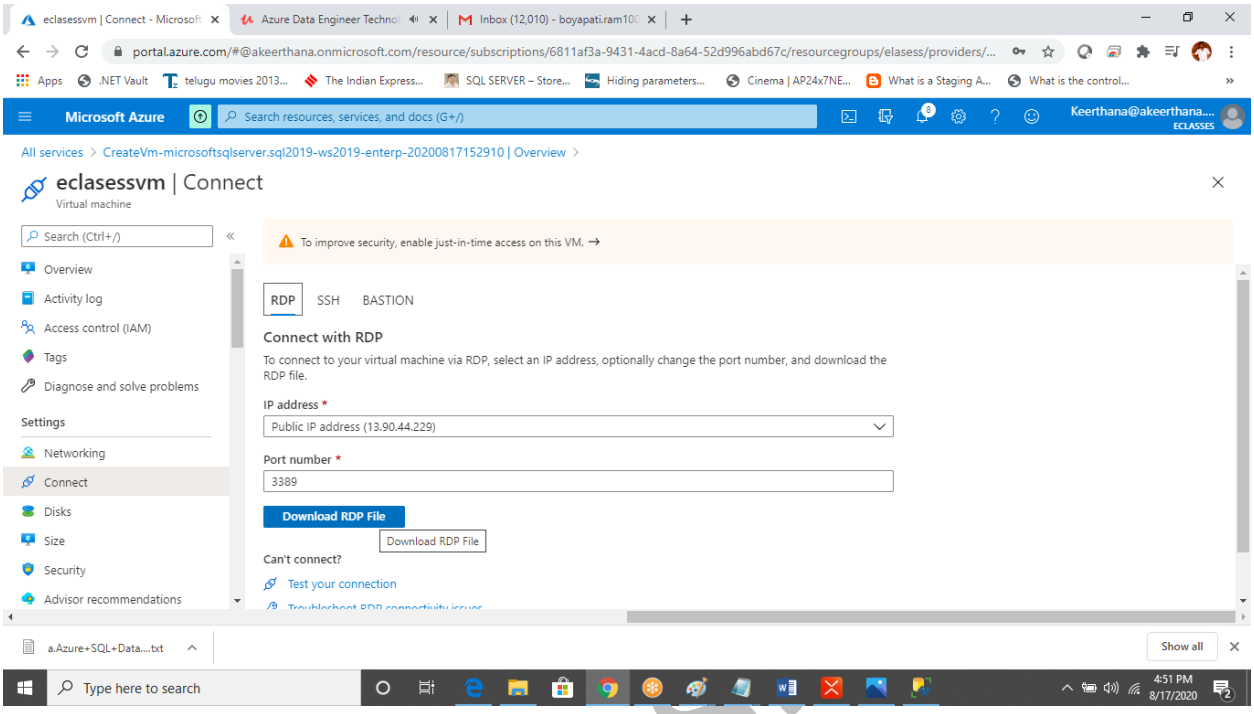
How to connect to virtual machine Created from local Computer.



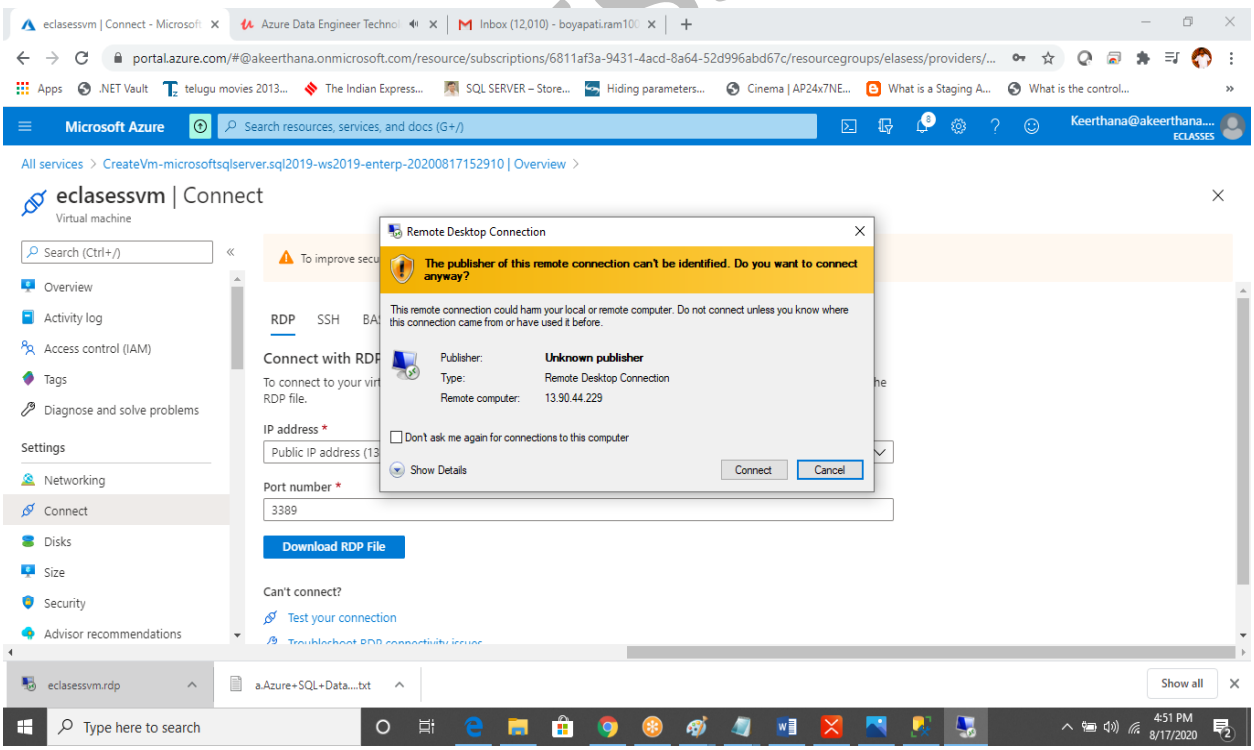
Step-1: Click on Connect and select RDP.



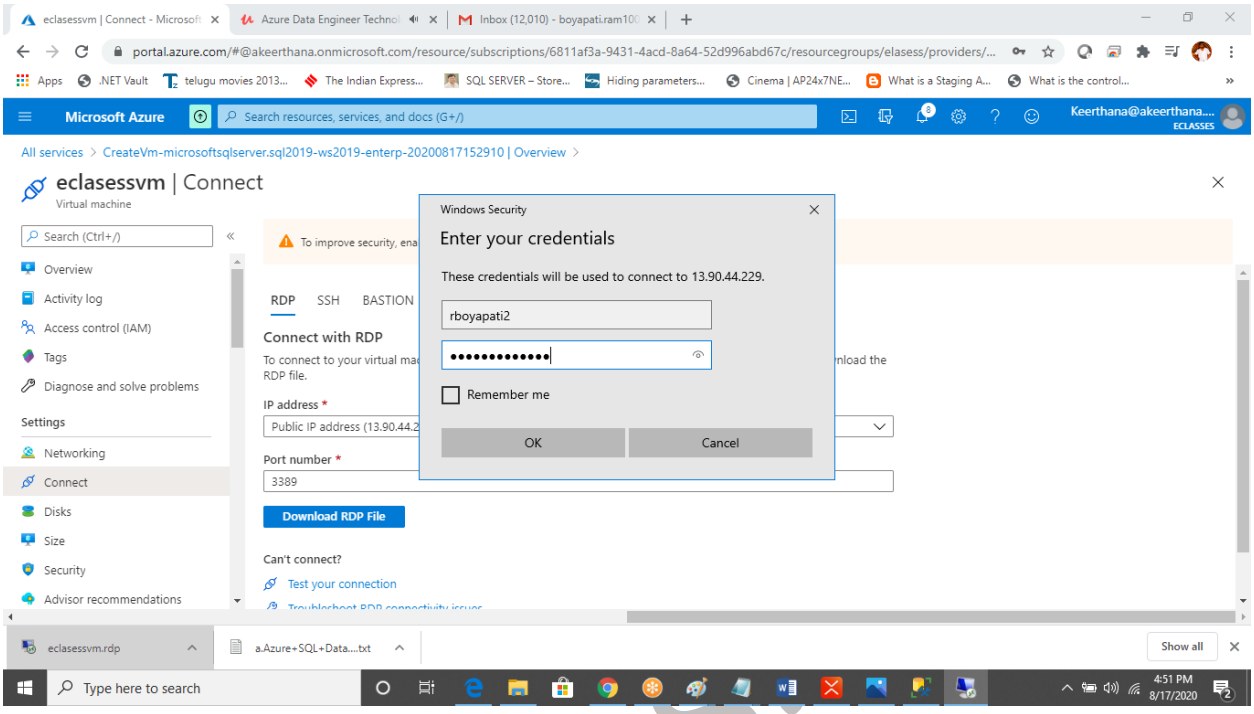
Step-2: Download RDP File.



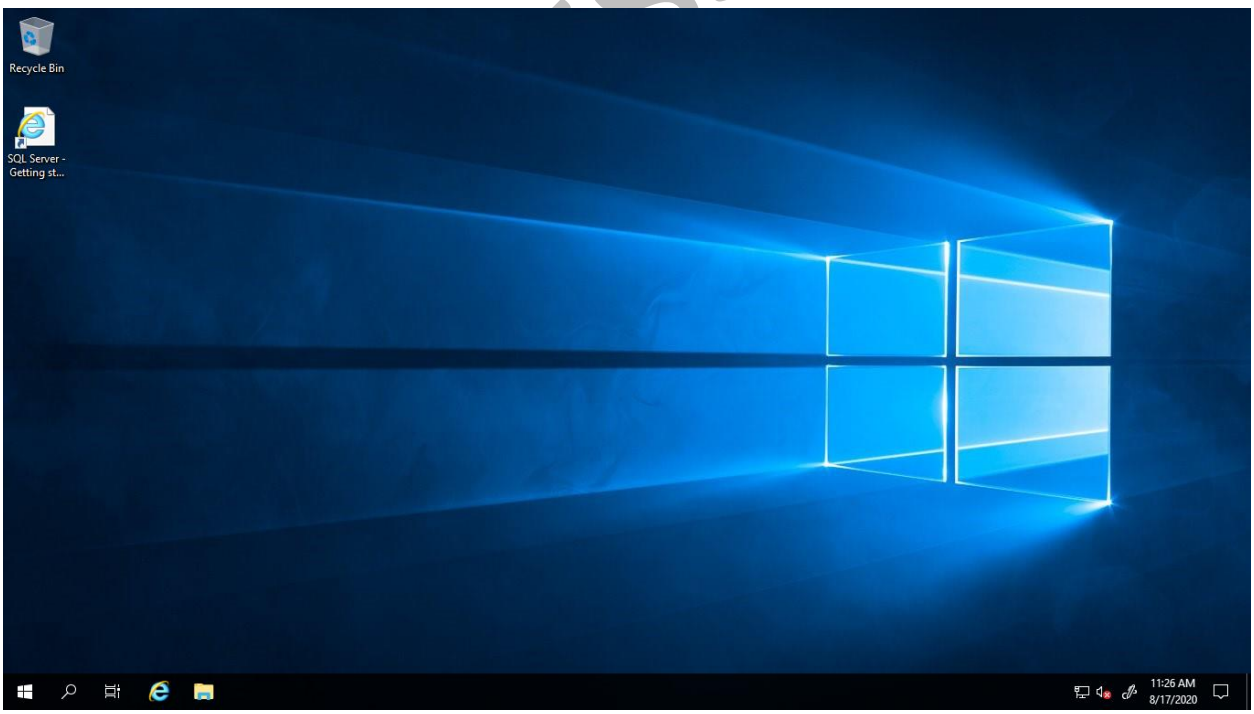
Step-3: Click on downloaded file under downloads folder and click connect.



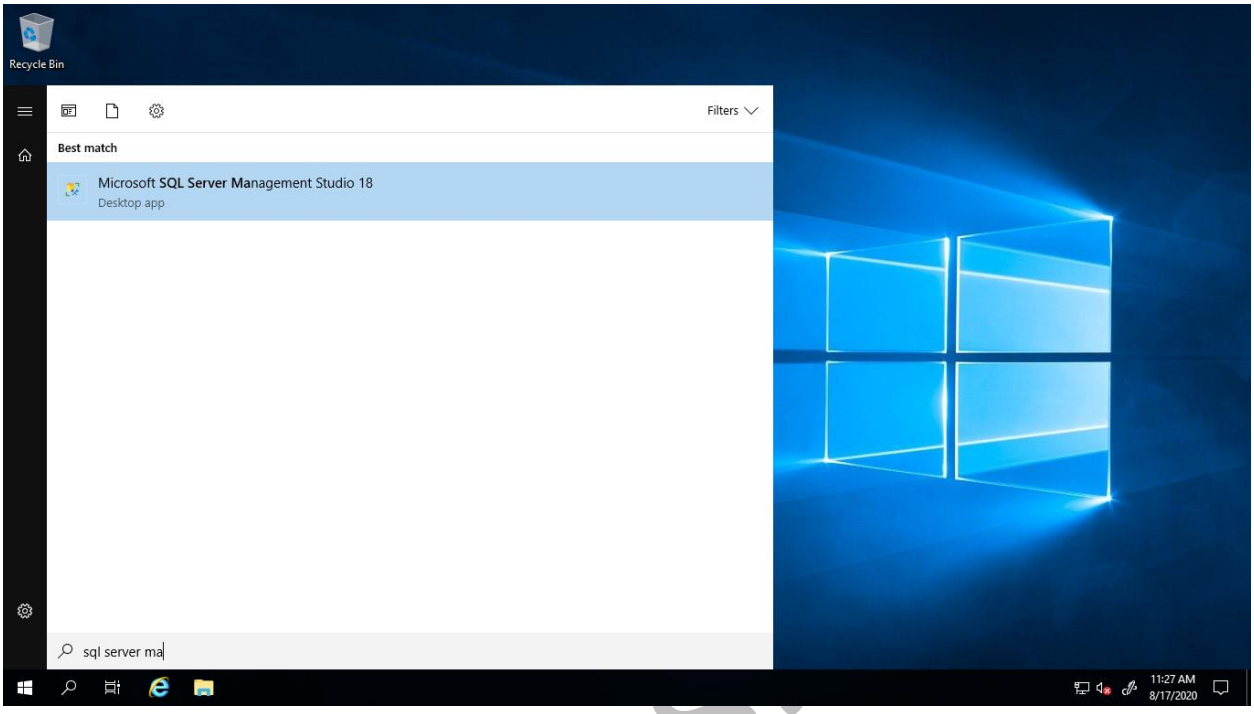
Step-4: Enter Username and password and click ok



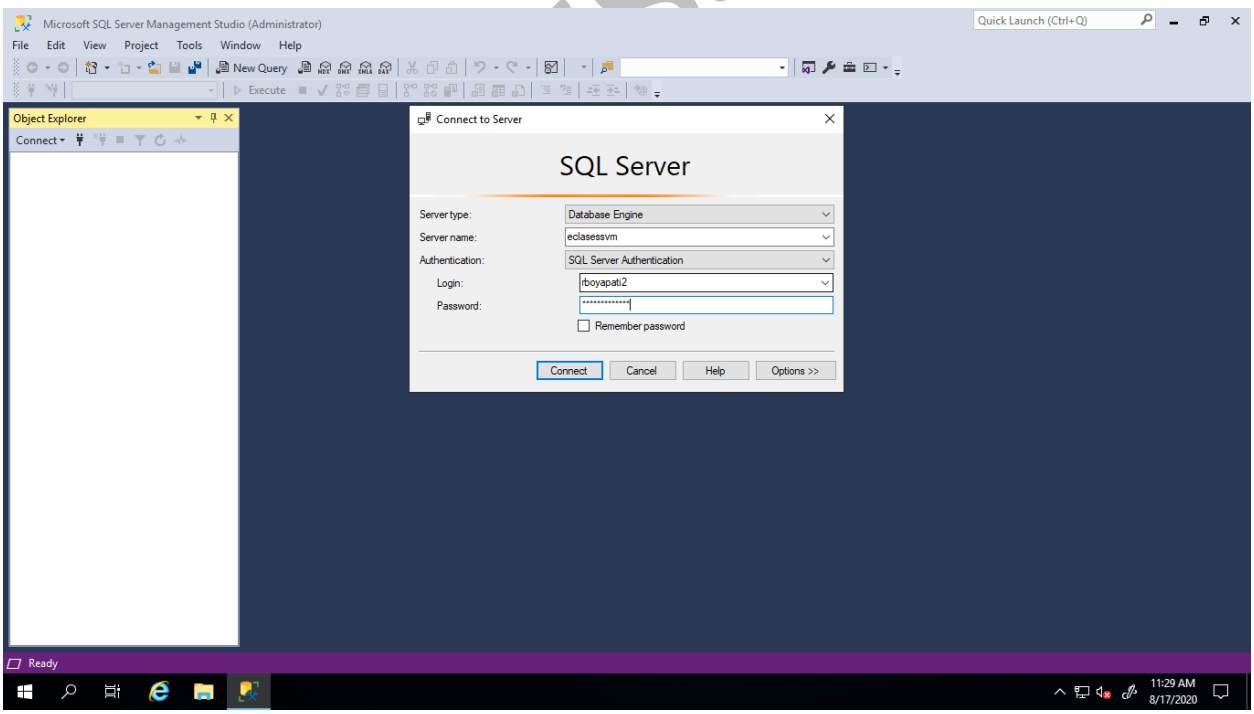
Step-5: You can see virtual machine.

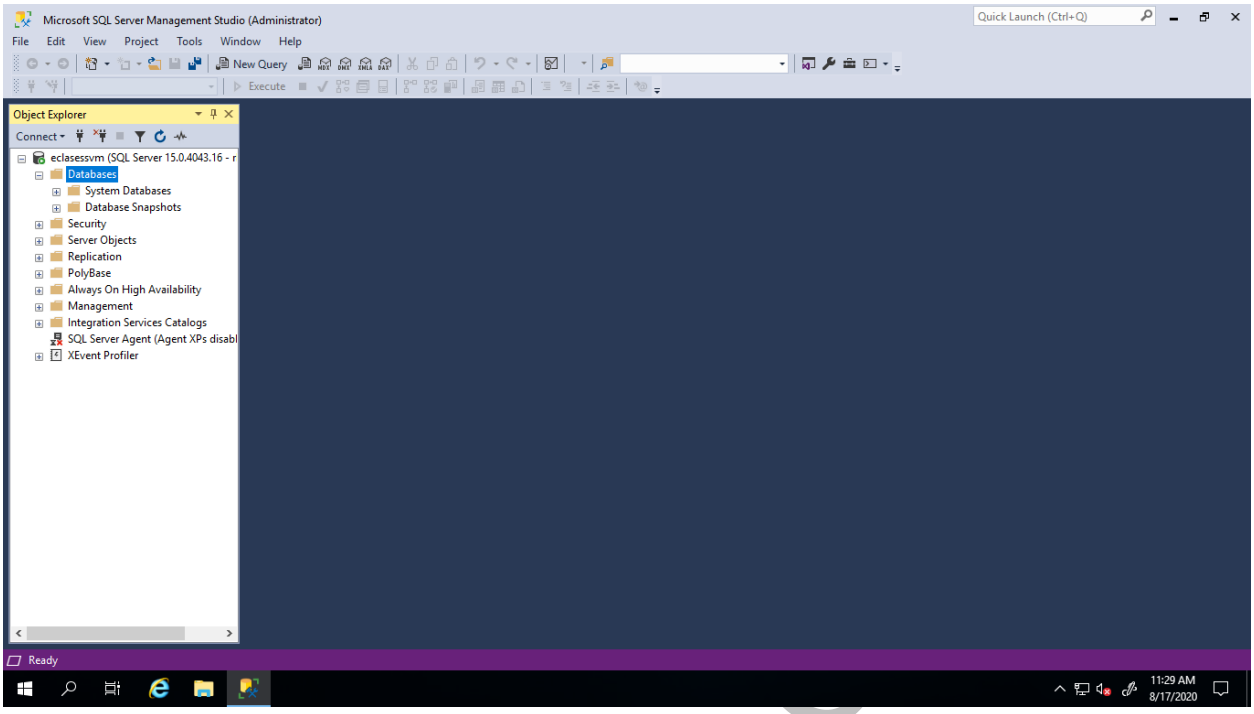


Step-5: Goto Start and search for SQL management Studio.



Step-6: select Authentication and Enter username password.

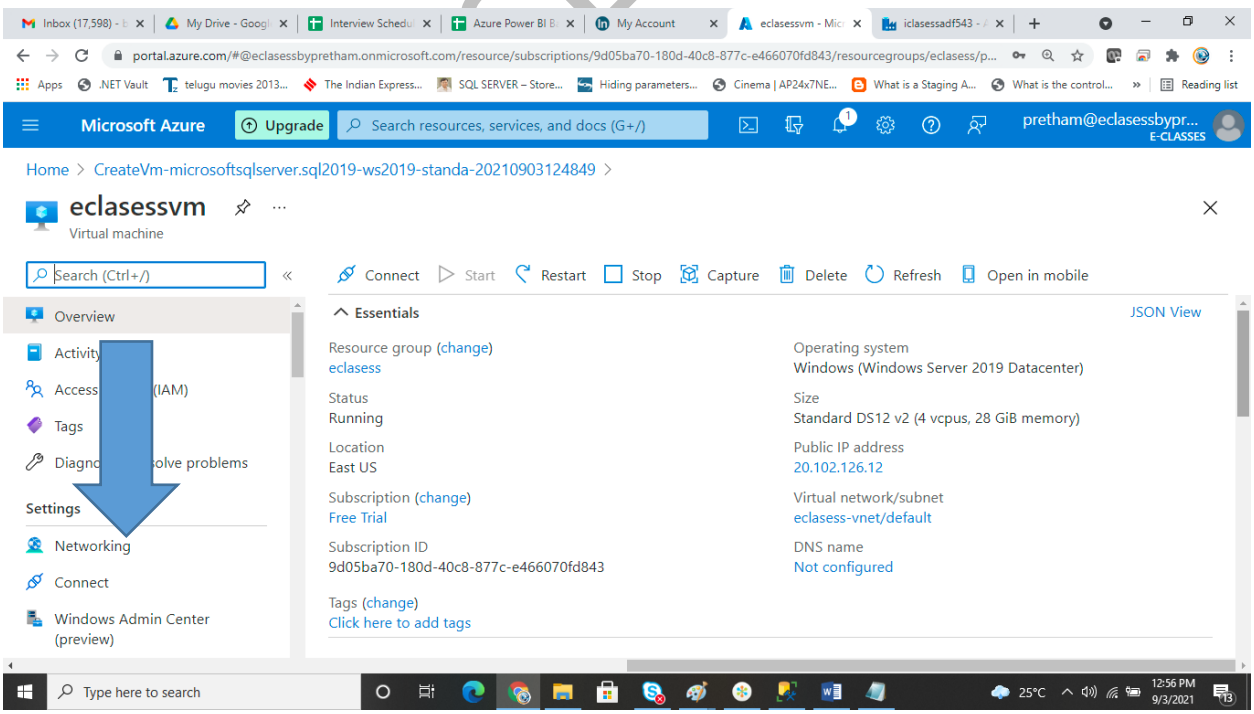




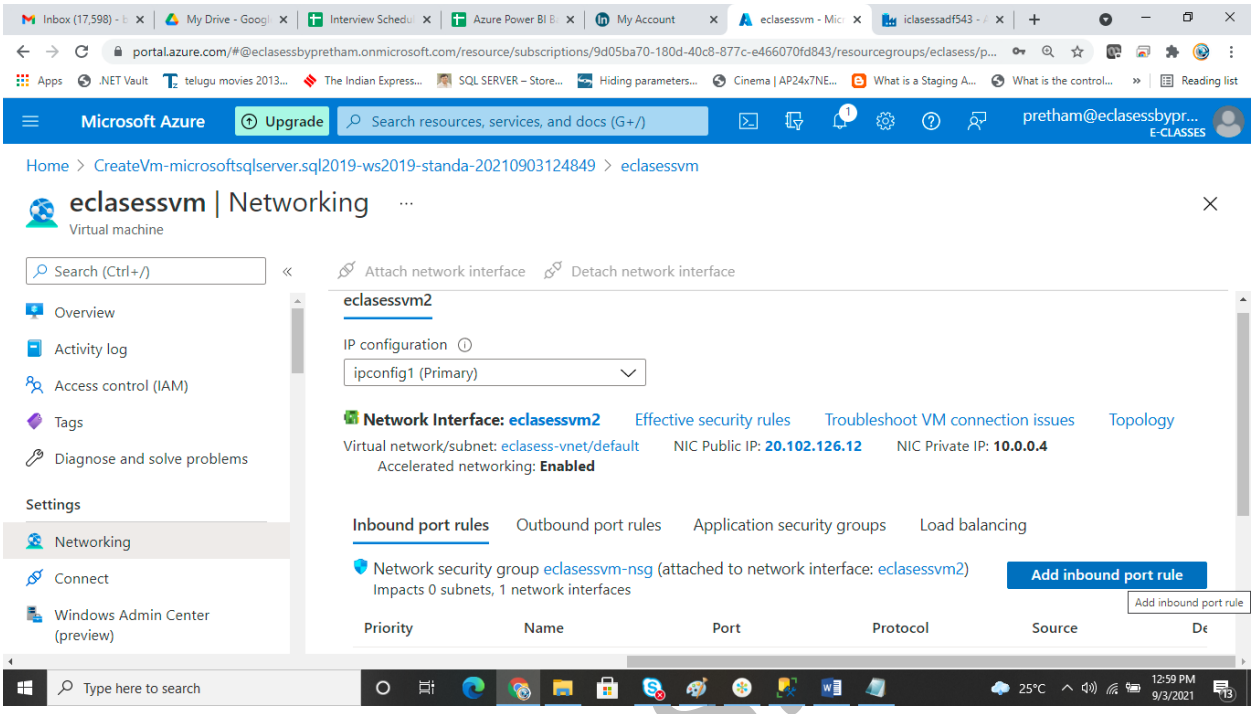
How to connect SQL Server Installed in Virtual Machine from On-Premises SQL Management Studio.

Step-1: Login to azure account got to SQL Virtual Machine

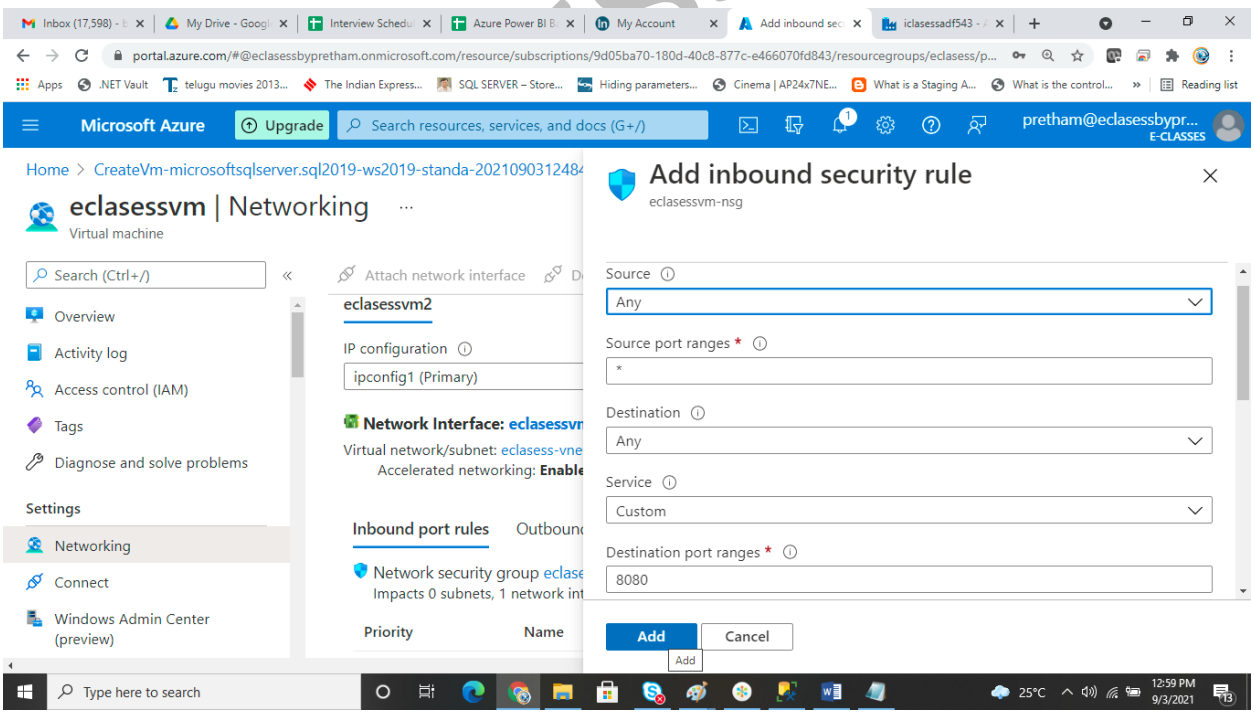
Click on Networking.



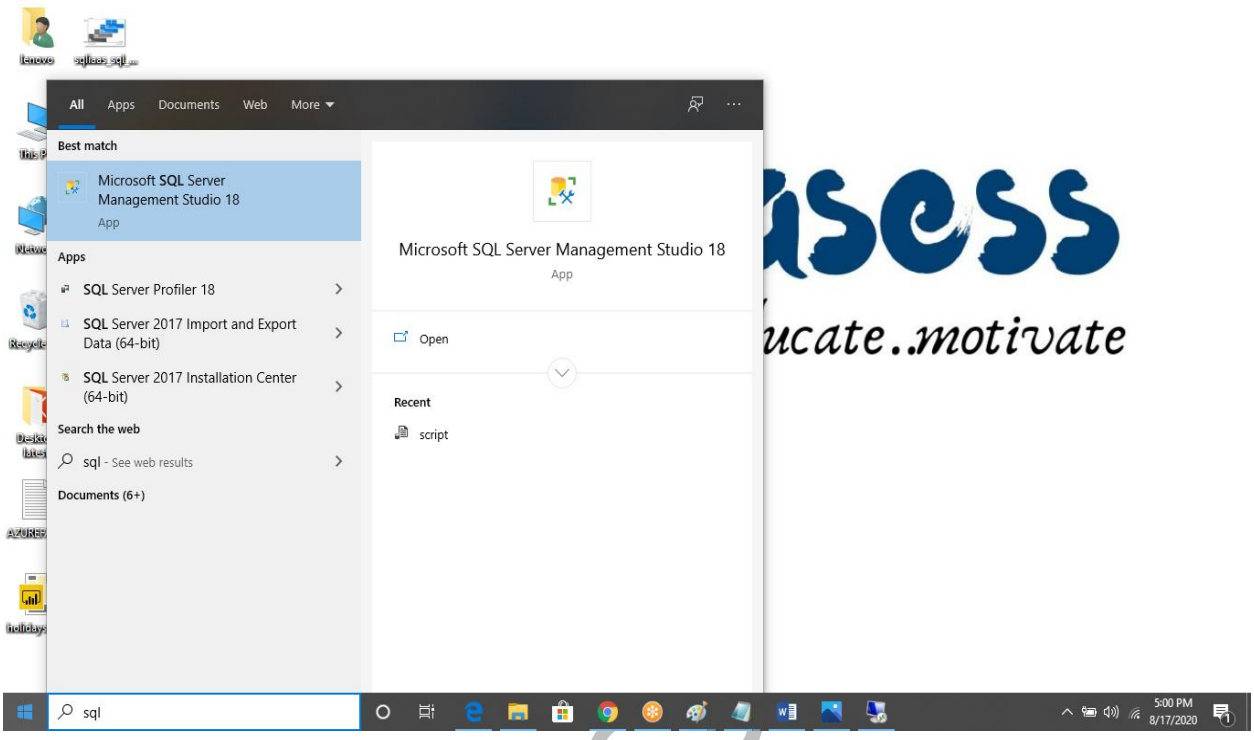
Step-2: Click on Inbound Port Rules.



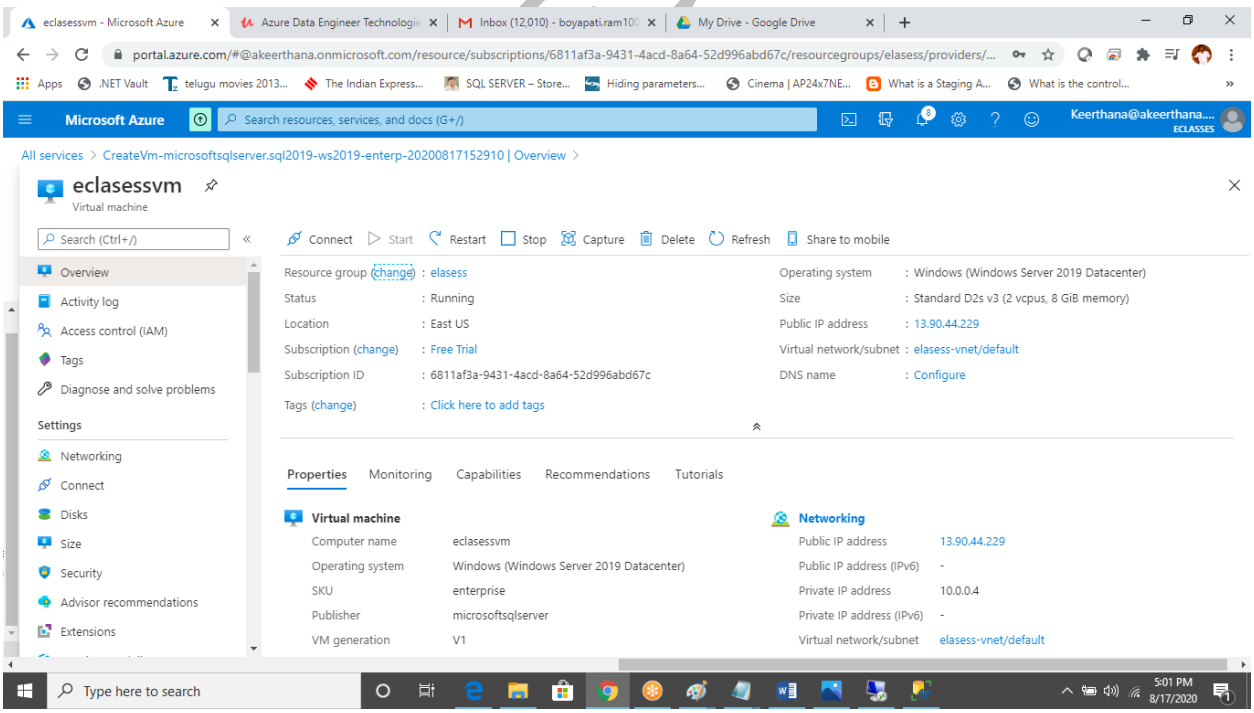
Step-3: Click on Add.



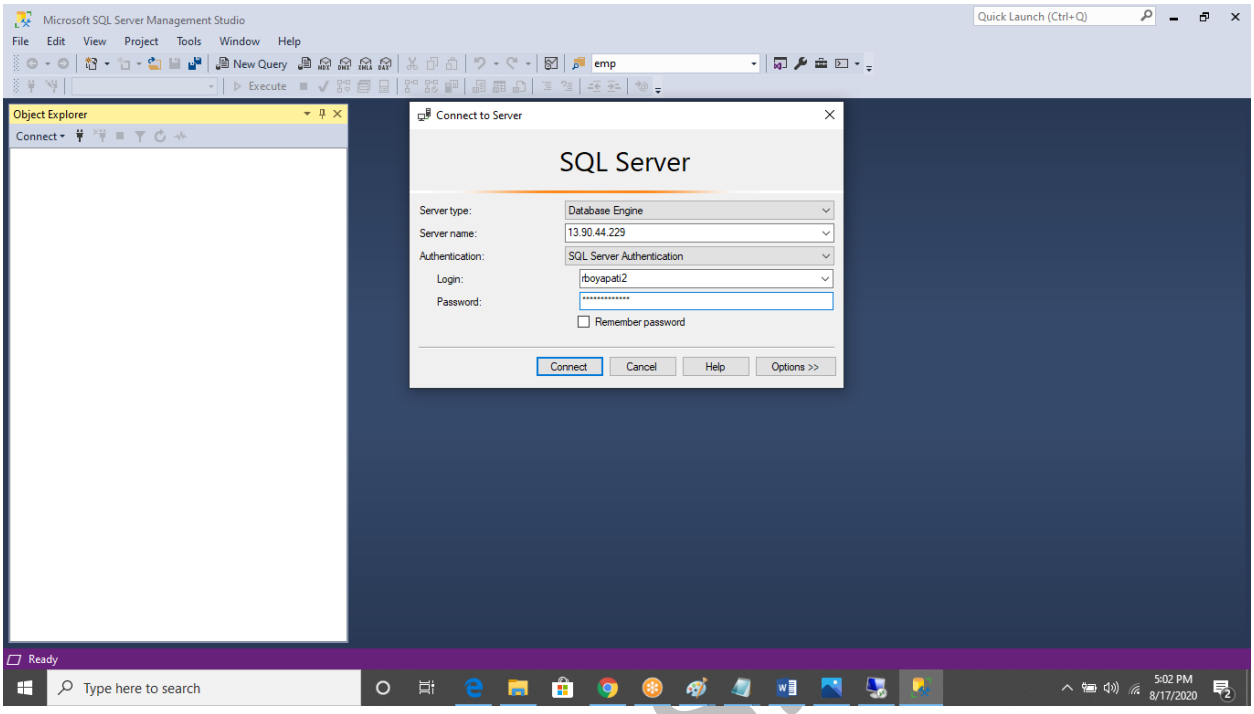
Step-4: Go to Local Computer open SQL management studio.



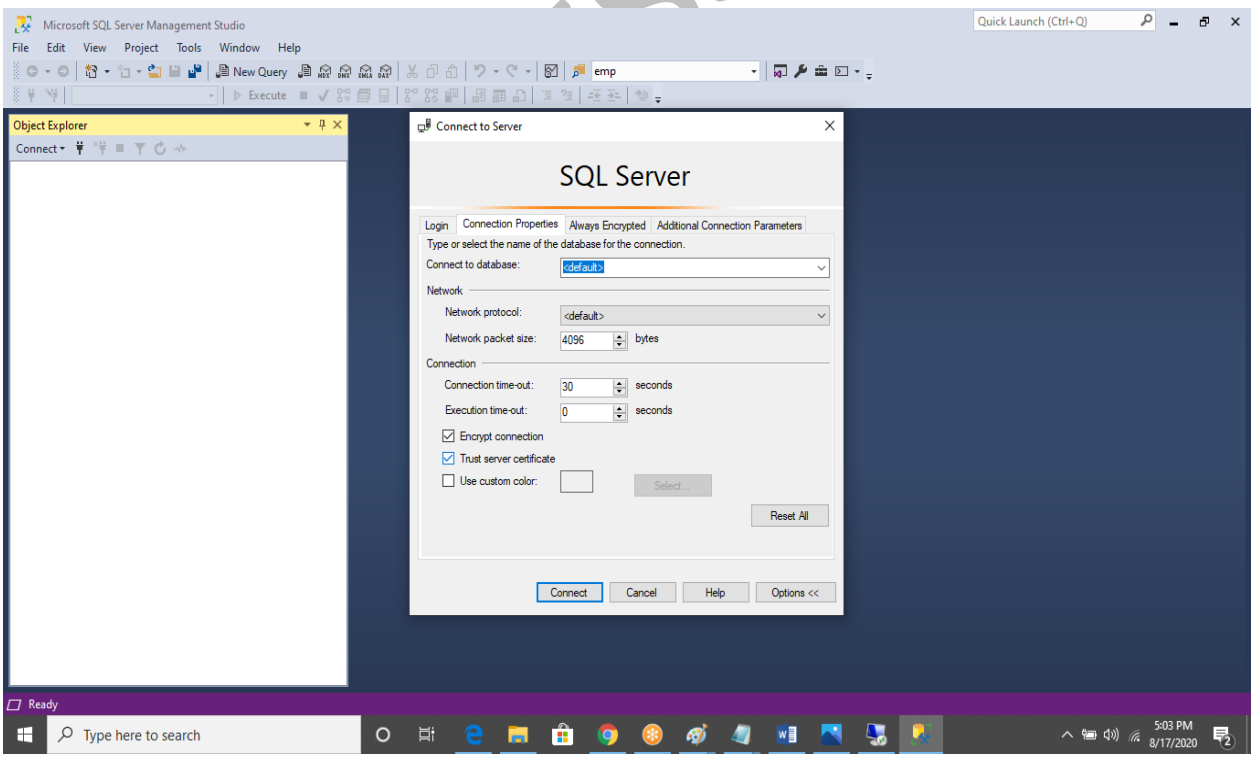
Step-5: Copy Public IP Address from Azure VM.



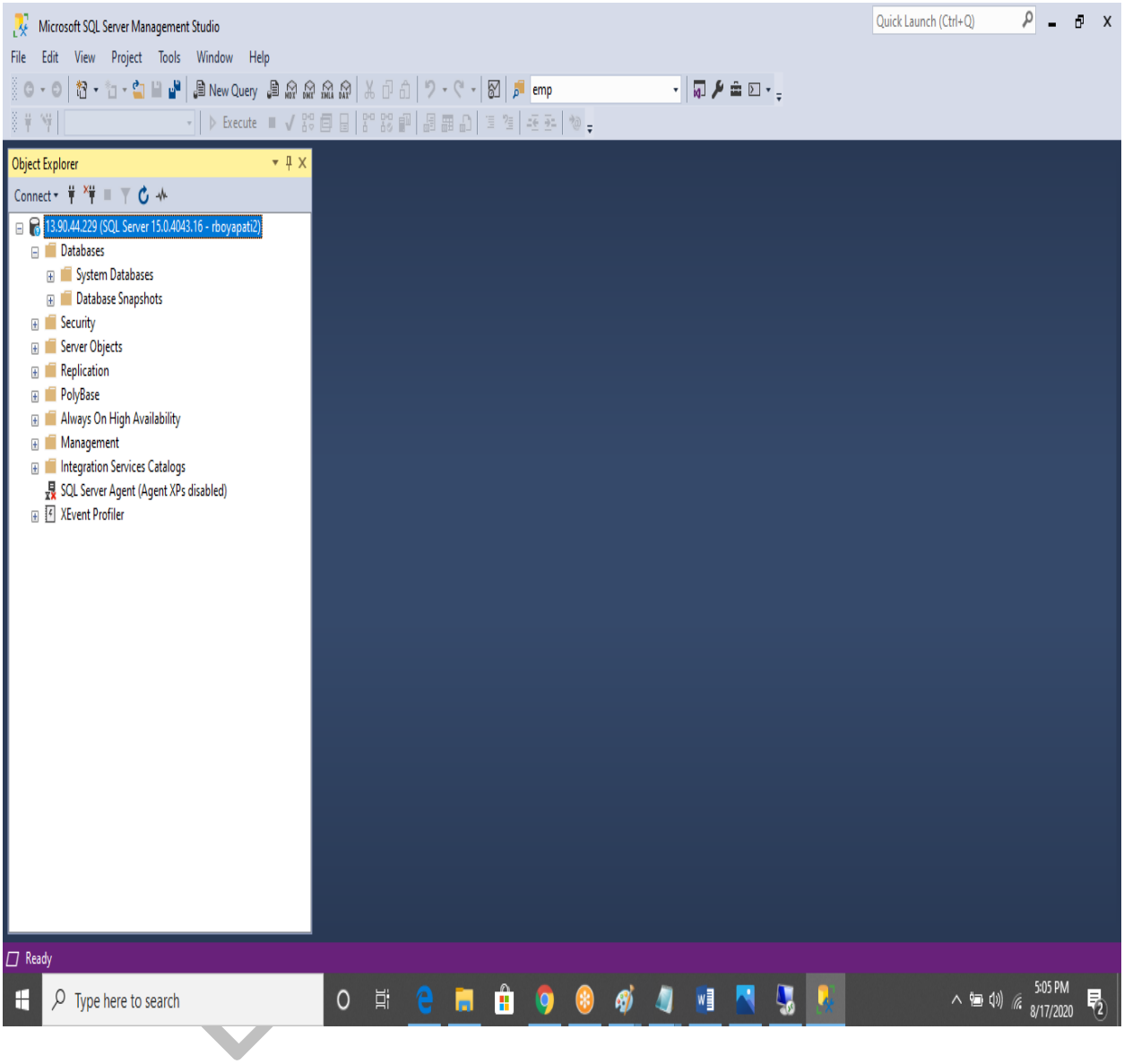
Step-6: Enter IP Address and username and password.



Step-7: Enable Encrypt connection and trust server certificate and click connect.



Step-8: You can see SQL Server.



Poly base: Querying the data from external data sources like blob, Data Lake into SQL synapse Data warehouse by creating external data source and external tables is called Poly base.

External tables just referring data available in files.

Query data stored in Hadoop from a SQL Server instance or PDW.

Users are storing data in cost-effective distributed and scalable systems, such as Hadoop.

Poly Base makes it easy to query the data by using T-SQL.

Query data stored in Azure blob storage.

Azure blob storage is a convenient place to store data for use by Azure services. PolyBase makes it easy to access the data by using T-SQL.

Import data from Hadoop, Azure blob storage, or Azure Data Lake Store.

Leverage the speed of Microsoft SQL's column store technology and analysis capabilities by importing data from Hadoop, Azure blob storage, or Azure Data Lake Store into relational tables. There is no need for a separate ETL or import tool. Supported by SQL Server 2014 onwards,

Cross query table existing in one database to another database using polybase external Table.

Prerequisites to implement below scenario:

Create Two Azure SQL database with name Source and Target

Create one EMP table with below script in Source Database

```
Create table EMP
        (
        Eno int, Ename varchar (100), Sal int
        )
```

Implement below polybase steps in target database.

Step-1: creating master key

Create master key encryption by password='SQl@12345'

Step-2: creating database scoped credential

```
Create database scoped credential Polybasecred
  With
        Identity='eclasess', -----SQL Server username
        Secret='Appuro@100'----SQL Server Password
```

ECLASESS.COM

Step-3: Creating external data source

```
Create external data source polybds
  With
    (
          Type=rdbms,
          Location='eclasesssql123.database.windows.net', ---Azure SQL database Name
          Database_name='source',
          Credential=polybasecred
          )
```

Step – 4 : Creating external table

```
Create external table empext
          (
          Eno int, Ename varchar (100), Sal int
     )
 with
 (
          data_source=polybds,
          Schema_name='dbo',
          Object_name='emp'
             )
```

Query data From External table.

```
Select * from empext
```

How to Drop External table

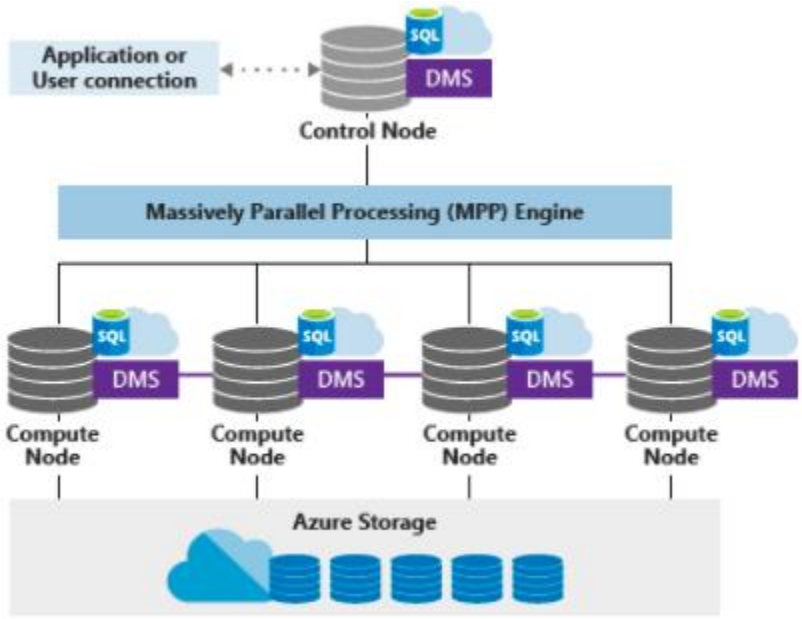
```
Drop external table empext
```

Note: We can't perform DML operations like Insert, delete update operations on external tables.

ECLASESS.COM

Synapse Data Warehouse (Dedicated SQL pools)

Dedicated SQL pool (formerly SQL Synapse DW) uses a node-based architecture.



Applications connect and issue T-SQL commands to a Control node.

The Control node hosts the MPP (Massive Parallel processing) distributed query engine, which optimizes queries for parallel processing,

And then passes operations to compute nodes to do their work in parallel.

The Compute nodes store all user data in Azure Storage and run the parallel queries.

The Data Movement Service (DMS) is a system-level internal service that moves data across, the nodes as necessary to run queries in parallel and return accurate results.

Key Terminologies in Synapse DWH Architecture

Control Node: it is taking requests from applications and host MPP engine to distribute and optimize quires for parallel processing

MPP Engine: Massive parallel processing of T- SQL quries by splitting into multiple smaller tasks.

Compute Nodes: Providing Infra stricter to execute each task on each compute node

DMS: Data Movement Service to move data across nodes.

Distribution Methods: Based on distribution method created on tables, data will be distributes accross nodes.

There are three different kind of Distribution method available.

1. **Round Robin**

   Data will be distributed equally or randomly across nodes

   Recommended for less volume of data tables like Dimension Tables

   Disadvantage is data needs to be sorted at time of combining from all nodes.

```
CREATE TABLE [dbo].[DIMCUSTOMER]
(
   [CustID] int NOT NULL,
   [Name] int NOT NULL,
   [Dob] int NOT NULL,
   )
WITH
(
   DISTRIBUTION = ROUND_ROBIN,
   CLUSTERED COLUMNSTORE INDEX
);
```

2. **Hash**

   Data will be distributed based on column specified in hash. Same kind of product ID's will be shared to one compute node.

   Hash (product id)

   Advantage is no need of sorting while combining of data from nodes

   Mostly recommended for large volume of tables like Fact tables.

```
CREATE TABLE [dbo].[FactInternetSales]
(  [ProductKey]          int         NOT NULL
,  [OrderDateKey]        int         NOT NULL
,  [SalesAmount]         money       NOT NULL
)
WITH
(  CLUSTERED COLUMNSTORE INDEX
, DISTRIBUTION = HASH ([Product Key])
);
```

ECLASESS.COM

3. **Replication**

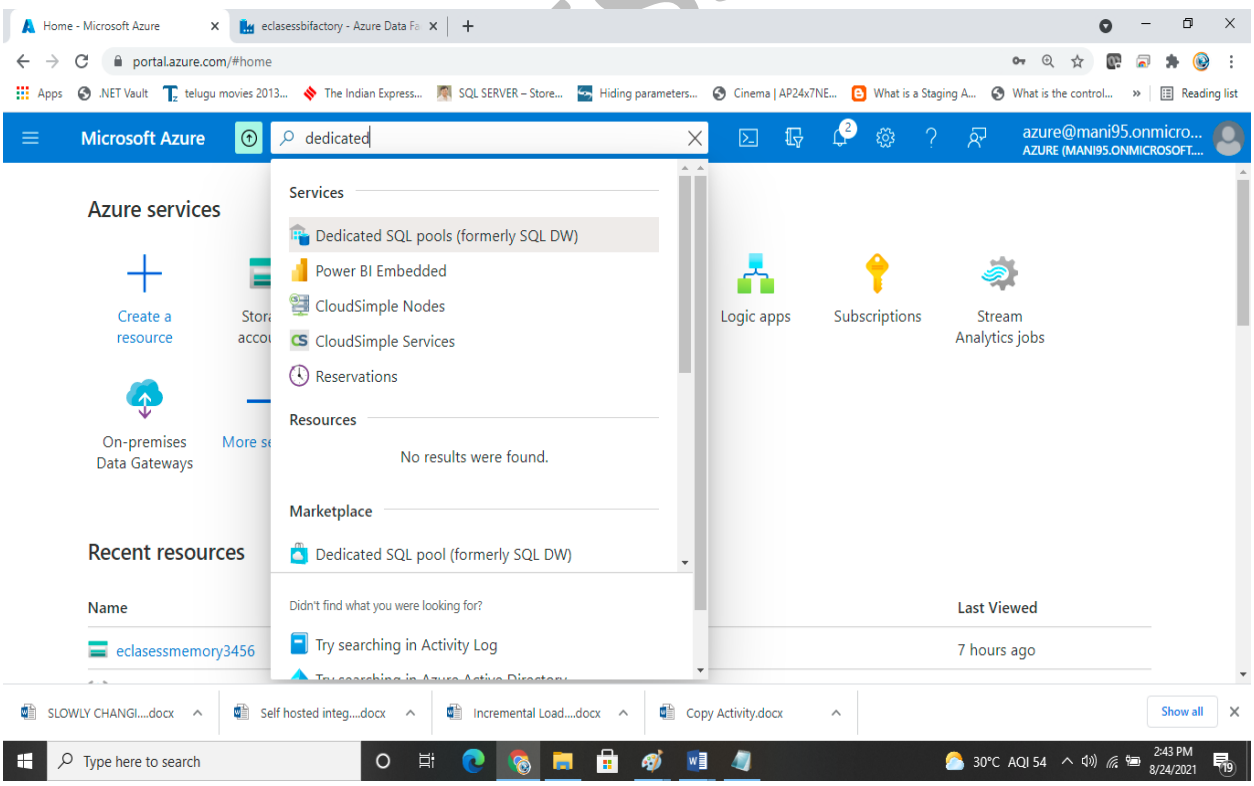
All data will be shared among all nodes.

Recommended for small Look Tables

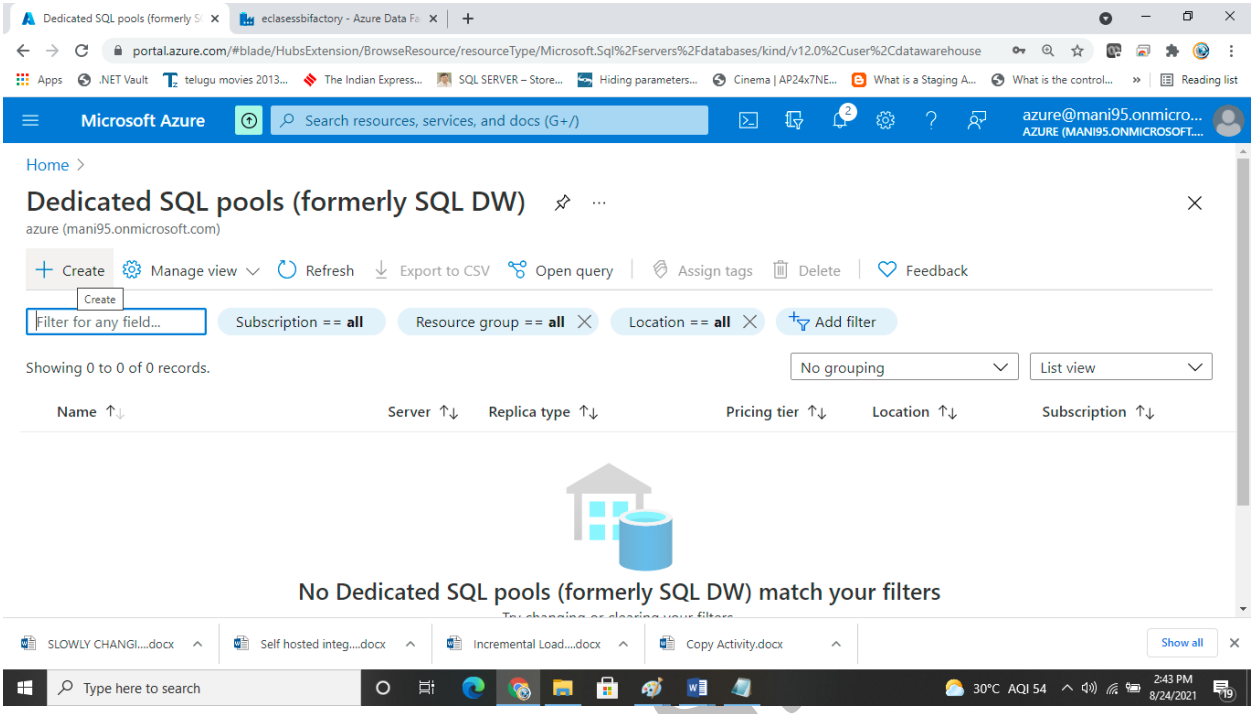
```
CREATE TABLE [dbo].[CITYLOOKUP]
(  [CityId]          int         NOT NULL
,  [CityName]        Varchar(100)        NOT NULL
)
WITH
(  CLUSTERED COLUMNSTORE INDEX
, DISTRIBUTION = REPLICATION
)
;
```

How to Create Dedicated SQL Pools (Formally SQL DW):

Step-1: Select Dedicated SQL Pools (Formally SQL DW).



Step-2: Click on Create

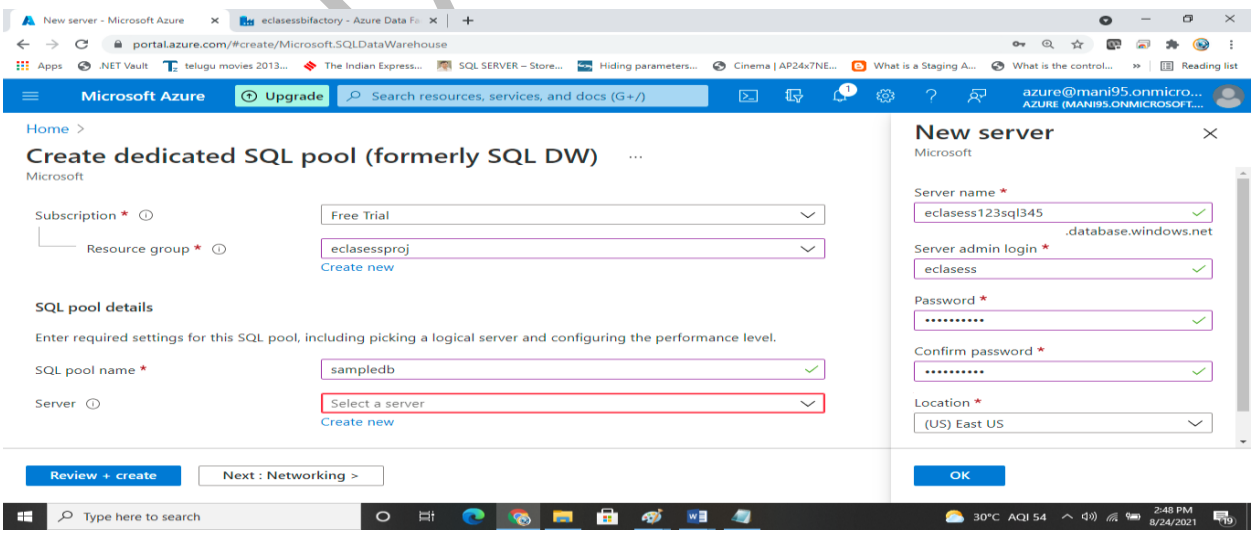


Step-3: Select Subscription

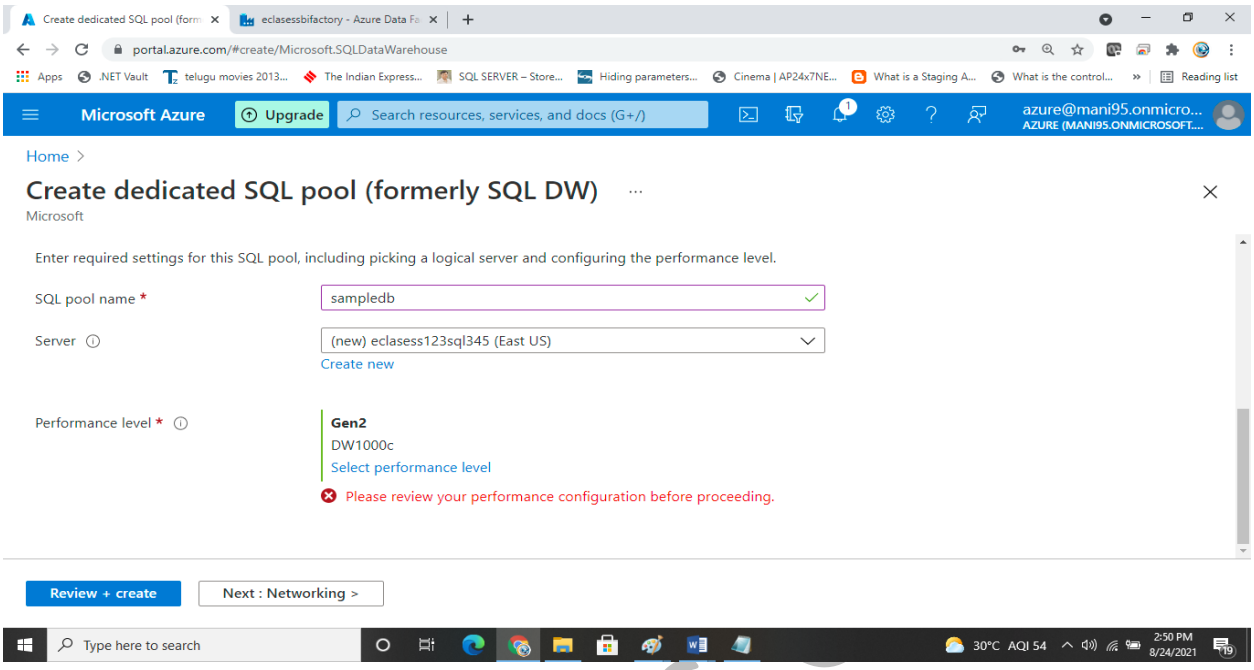
Select Resource Group

Enter SQL Pool Name as Sample DB

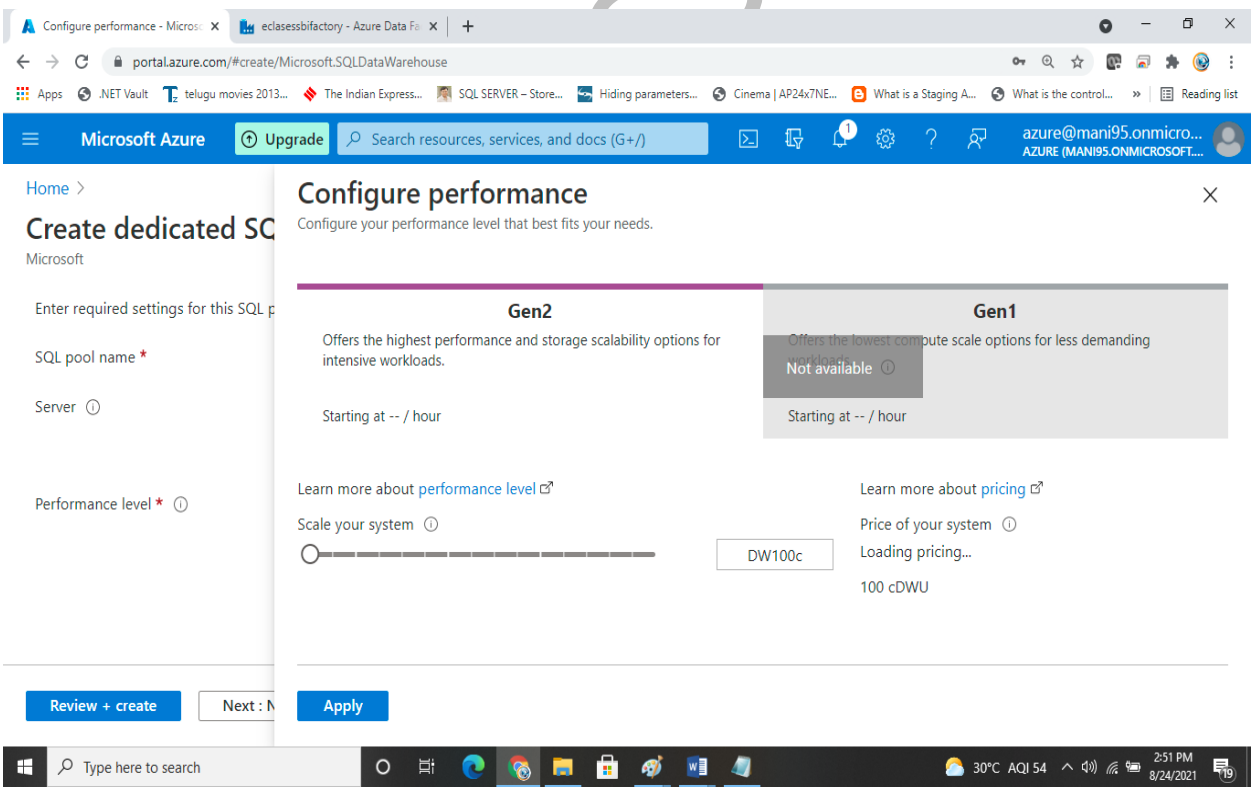
Enter Server Name and Username and password.



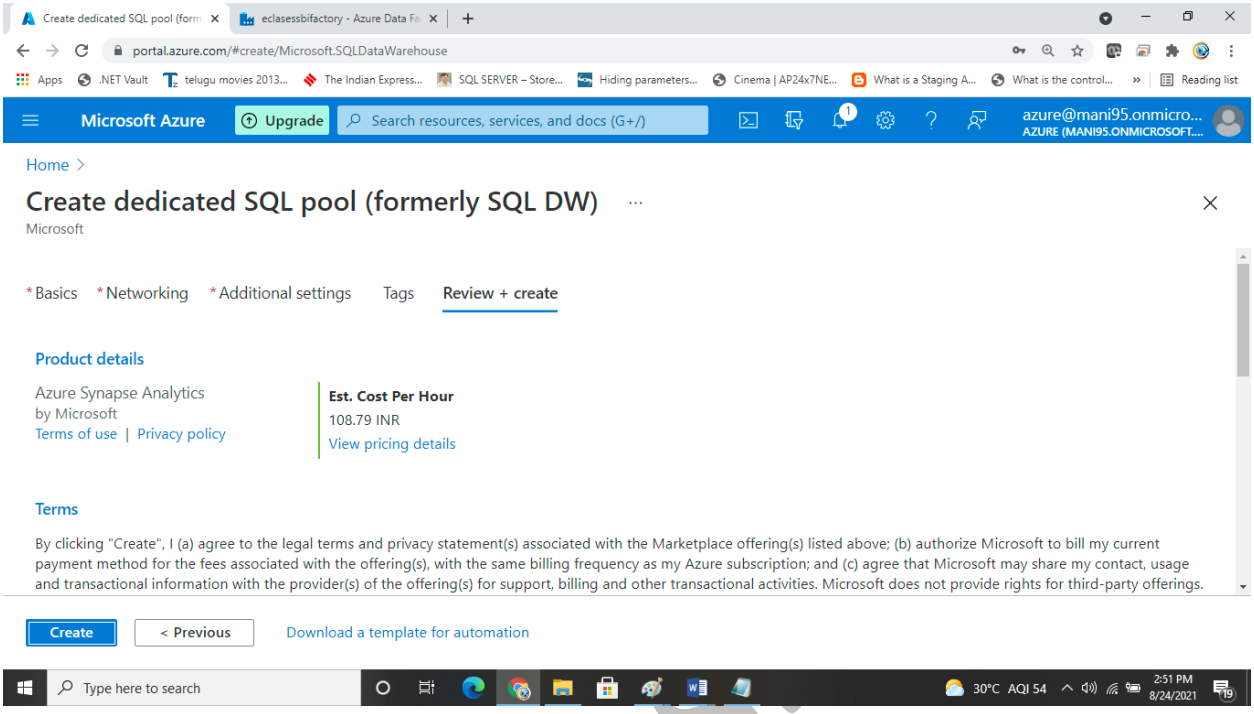
Step-4: Select performance level of DWU by clicking

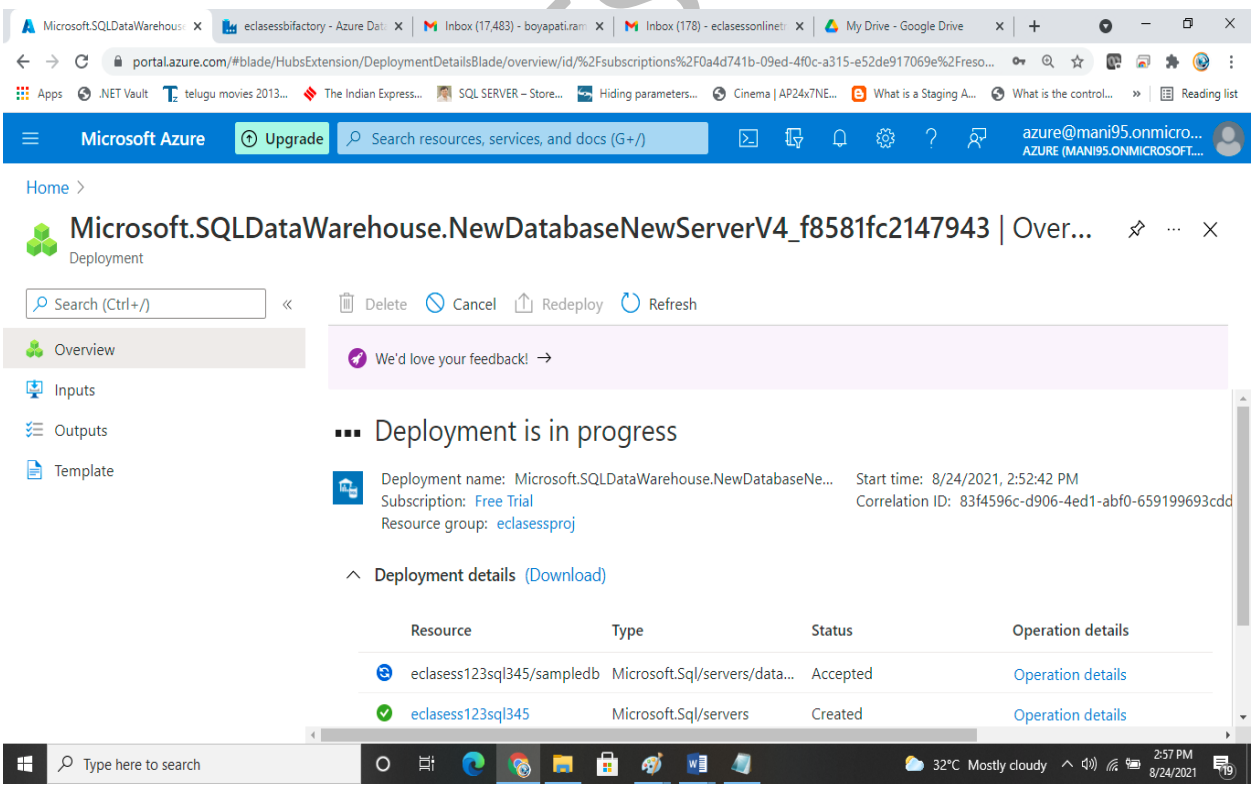


Step-5: Adjust scale your system and click on Review + Create

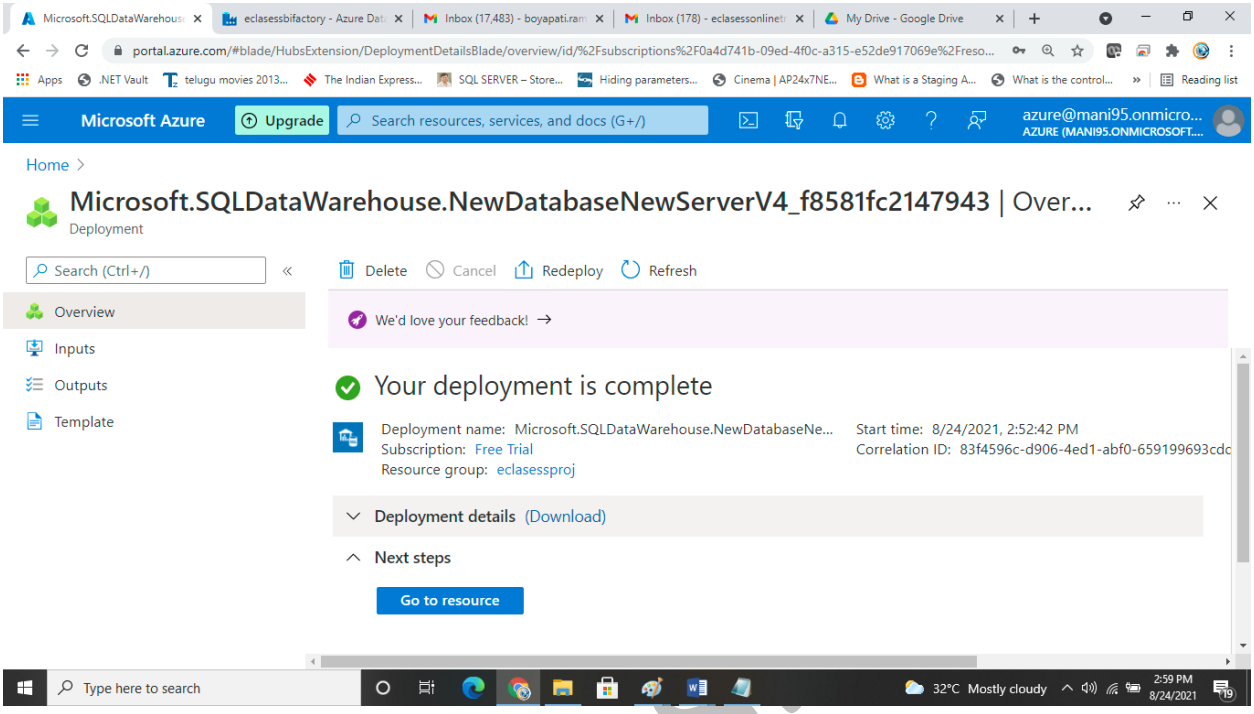


Step-6: Click on Create

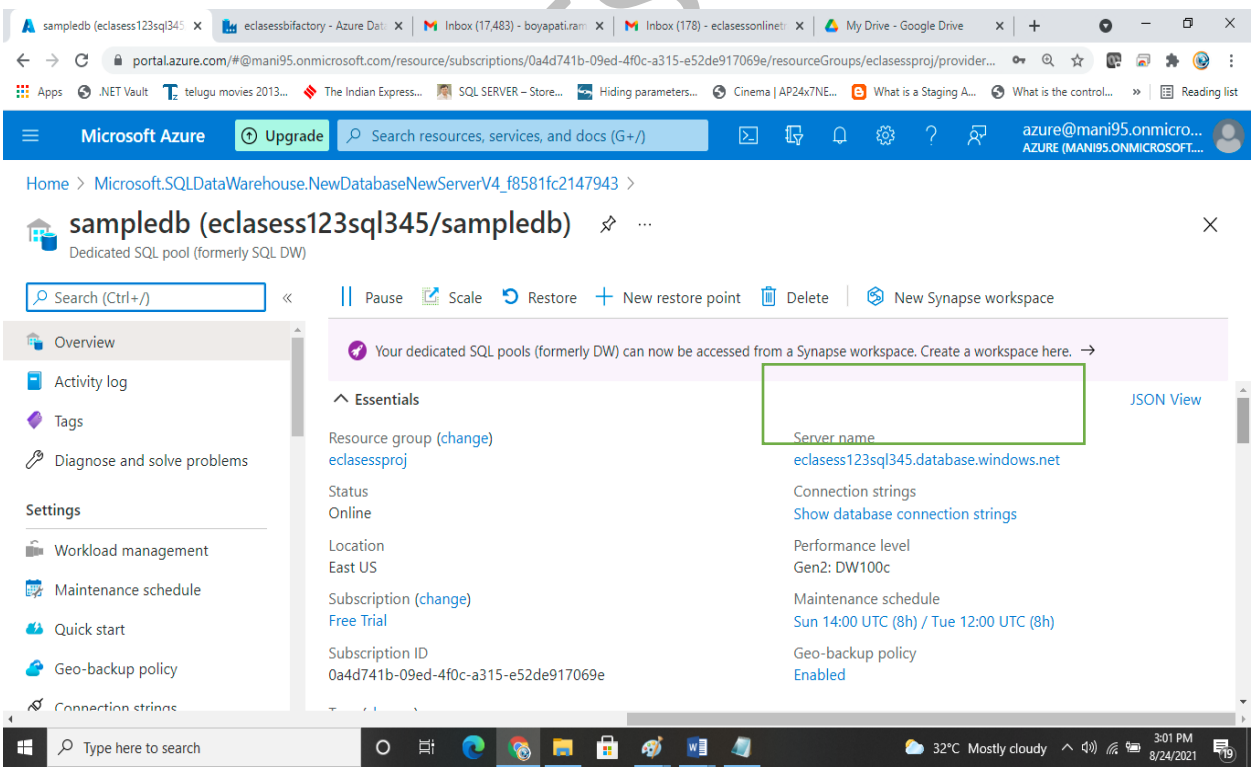




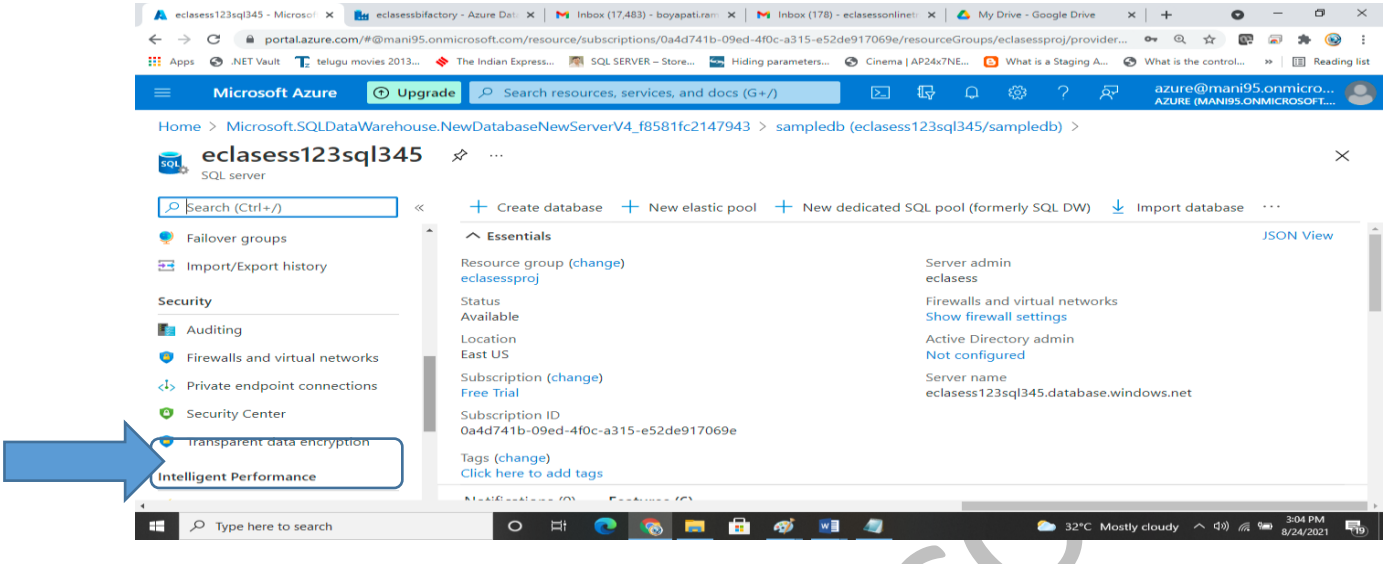
Step-7: Click on Goto Resource



Step-8: Click on below server name



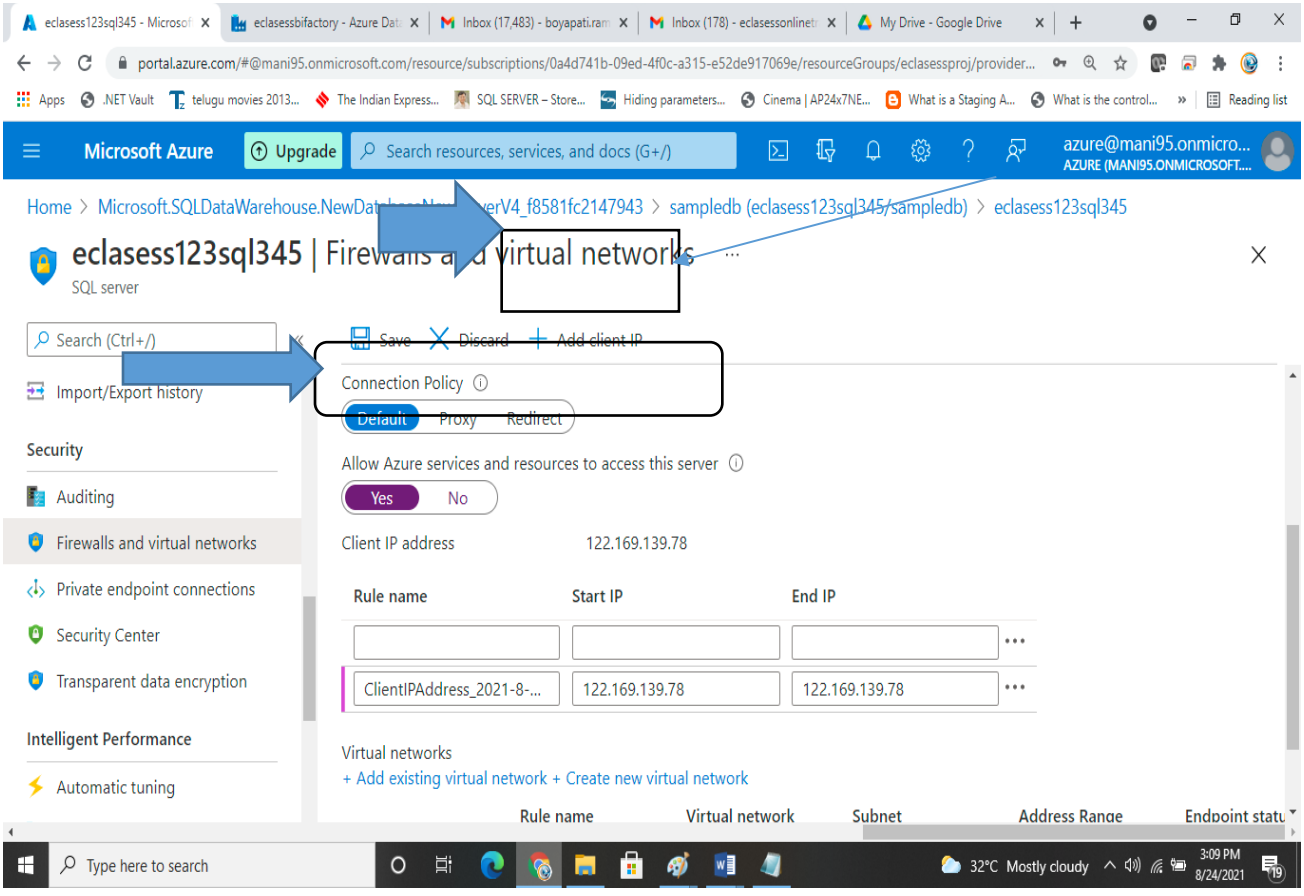
Step-9: Go to Firewalls and Virtual Networks.



Step-10: Click on add client IP address

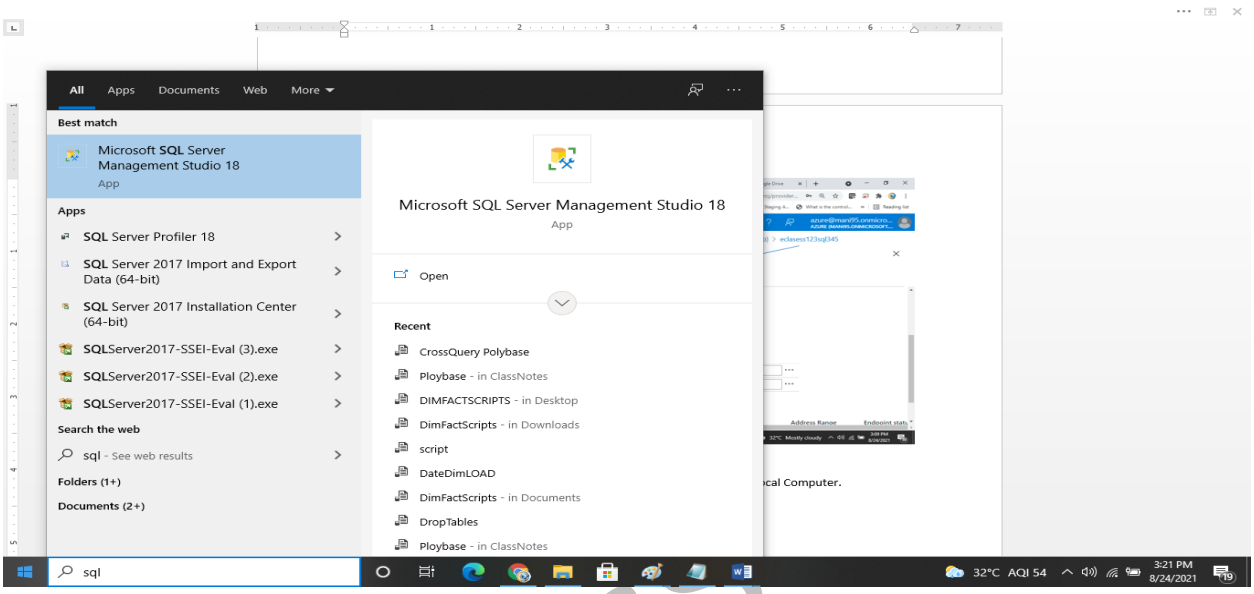
Click Yes Allow Azure services and resources to access this server

Click on Save.

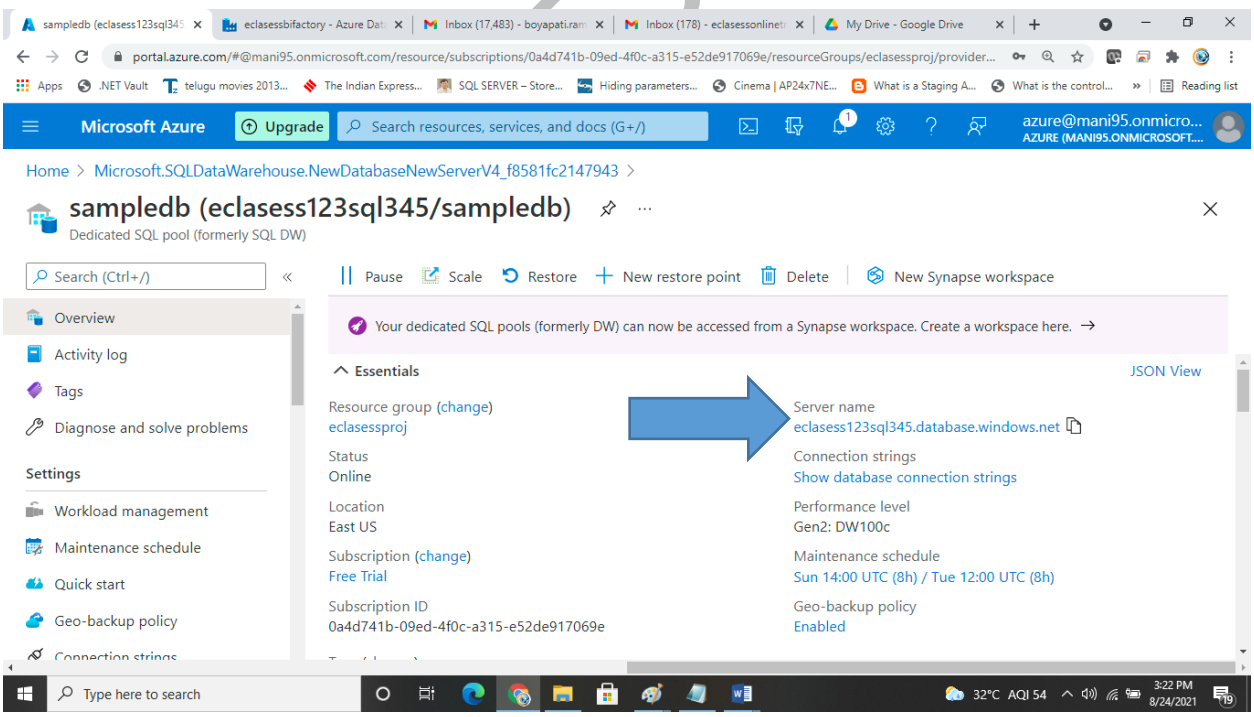


How to connect to SQL Synapse DW pool using SQL management Studio in Local Computer.

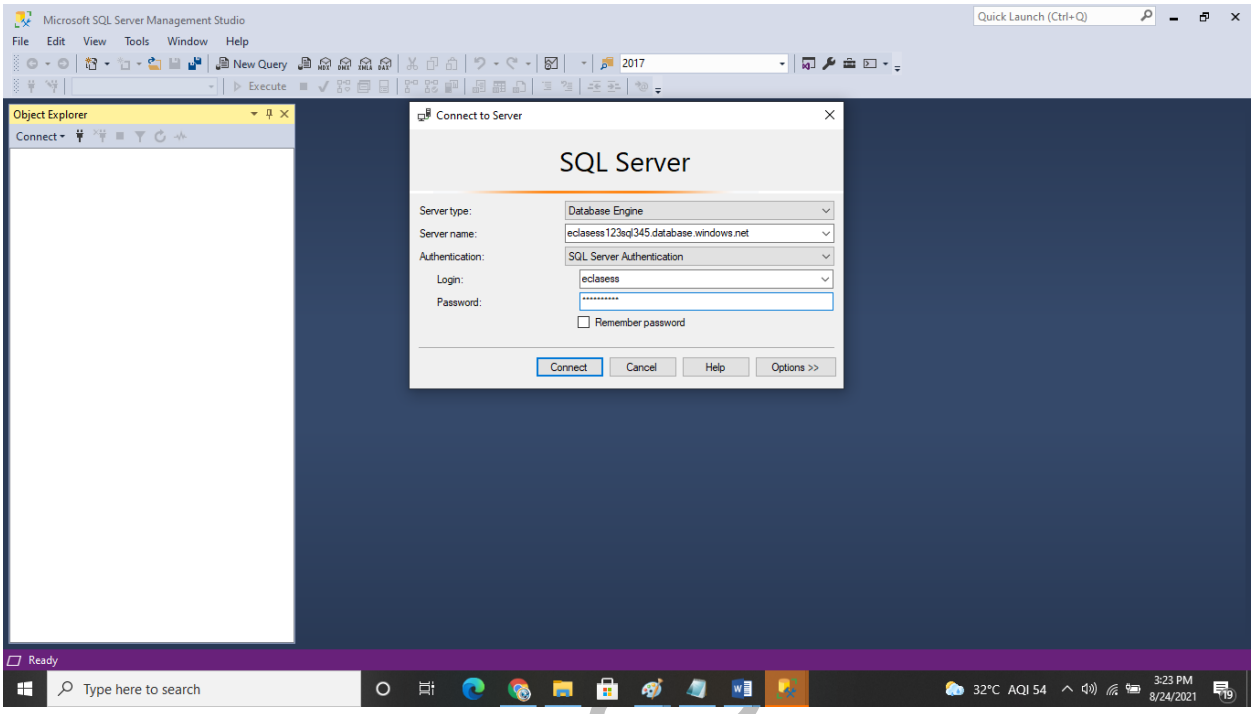
Step-1: Got Local computer start Select SQL Server Management Studio.



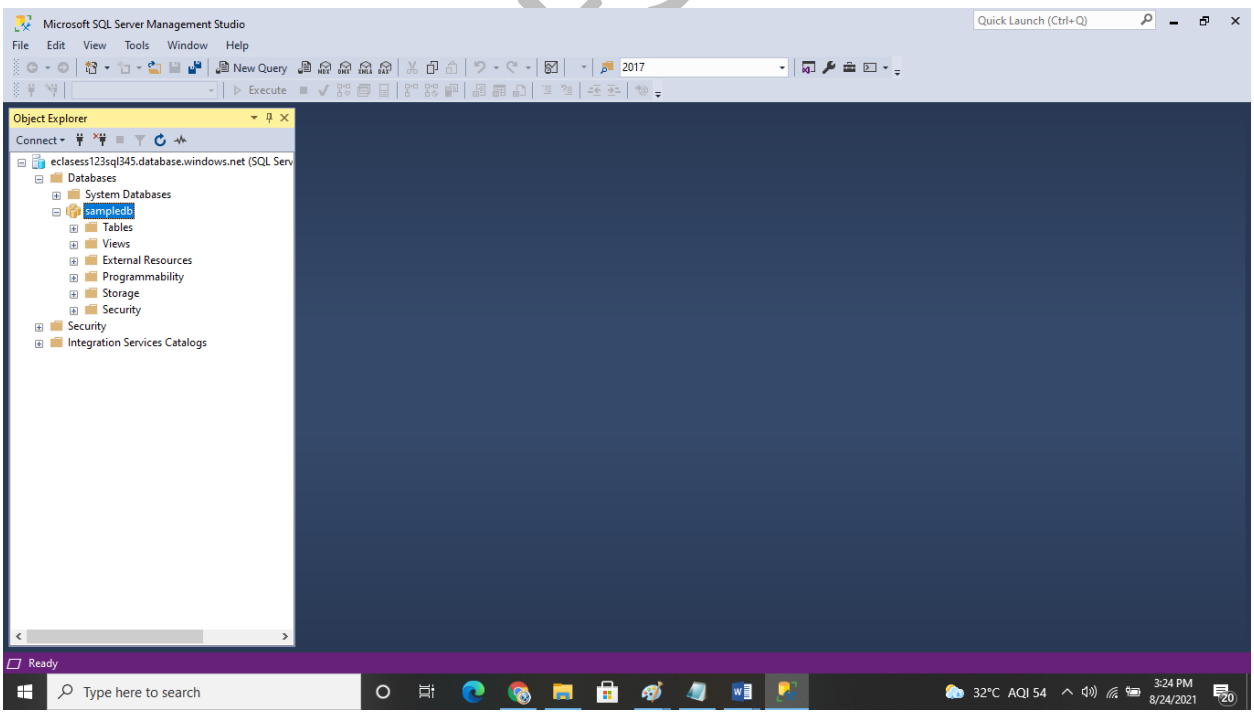
Step-2 Copy Server name from Azure Synapse DW Pool.



Step-3: Open SSMS and Enter Copied Server Name and Select Authentication SQL Enter Username and Password. Finally click on connect.



Step-4 : See database Sample DB



How to Query Data from blob storage to Synapse DW using Poly base

- **Prerequisites to implement:**
- **Create SQL Synapse Data warehouse formally (Dedicated SQL pools)**
- **Create Storage Account**
- **Create blob Container**
- **Upload some sample files (empbang.txt, emphyd.txt…….)**

Step-1: Login to Synapse DW database using SQL Management Studio. Please run below scripts

Step-2: Create master key

```
CREATE MASTER KEY ENCRYPTION by PASSWORD='Appuroram@100'
```

Step-3: create data source credentials stored inside database

```
CREATE DATABASE SCOPED CREDENTIAL AZURESTORAGECRED
WITH
IDENTITY='RAM',
SECRET='VX/dv2SRoUw0iKzcM6KGMVZHKNM7vid7pvV1BMx7oWTwKAZhNHWeI6LgIw99WglifDA20rVrx56gihUYu5aEbA==' /* copy Access key of Storage Account */
```

Step-4: create external data source

```
CREATE EXTERNAL DATA SOURCE AZURESTORAGE /* Container name & Storage Account name*/
WITH
 (
   TYPE=HADOOP,   ---BLOB OR DATALAKE
   LOCATION='WASBS: //employee@eclasessstorage.blob.core.windows.net',
   CREDENTIAL=AZURESTORAGECRED
)
```

Step-5: create external file format

ECLASESS.COM

```
CREATE EXTERNAL FILE FORMAT EMPFILE
WITH
 (
  FORMAT_TYPE=delimitedtext,
  format_options(field_terminator=',',FIRST_ROW = 2)
 )
```

Step-6: create external table

```
Create external table EMP
(
Eno int,
Ename varchar (100),
Sal int
) with (
      data_source=AZURESTORAGE,
      file_format=EMPFIL,
      Location='/emp/'
)
Select * from [dbo]. [EMP]
```

ECLASESS.COM

Azure Data Lake

Azure Data Lake is repository to store any kind and volume of data in its native format and enormous processing power and the ability to handle large size

It can store Structured, semi structured and unstructured data without defining any schema at time of loading data into Data Lake.

This is more suitable for storing and performing big data analytics.

The data lake has designed on top of Hadoop HDFS Frame work.

There are two different versions of Data lake Storages.

Data lake Gen1 Storage:

Azure Data Lake Storage Gen1 is an enterprise-wide hyper-scale repository for big data analytic workloads. Azure Data Lake enables you to capture data of any size, type, and ingestion speed in one single place for operational and exploratory analytics.

Data Lake Storage Gen1 is an Apache Hadoop file system that's compatible with Hadoop Distributed File System (HDFS), and works with the Hadoop ecosystem

Data lake Gen2 Storage:

Azure Data Lake Storage Gen2 is a set of capabilities dedicated to big data analytics, built on Azure Blob storage.

Data Lake Storage Gen2 = Azure Blob storage + Azure Data Lake Storage Gen1

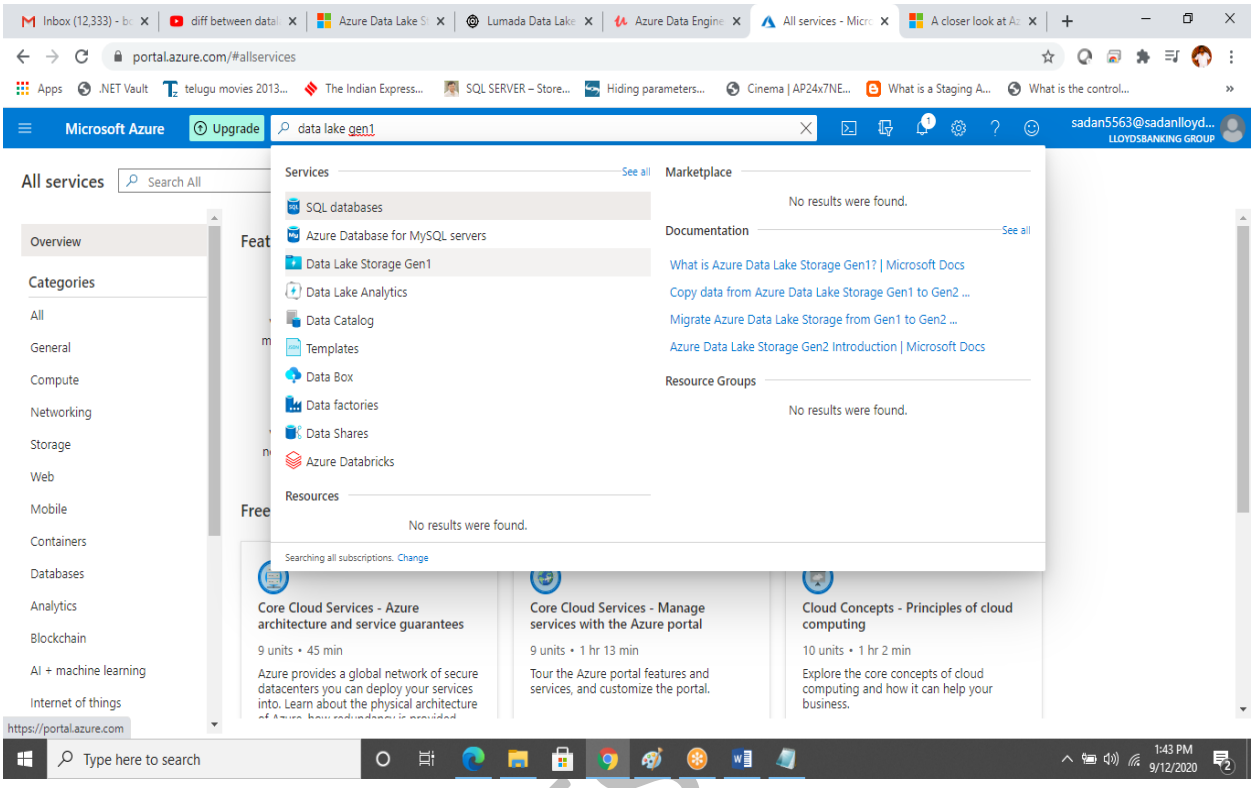
What is Diff between Blob storage and Data Lake Storage?

Azure Blob Storage	Azure Data Lake Storage
General purpose Storage	Optimized storage for Big data analytics purpose
Object based model with container	Hierarchical name space model
Not Compatible with Hadoop	compatible with Hadoop
Data Access can be done using Access keys and Share Access Signatures	Data Access Can be done through Active Directory Authentication
No limit on Account, file sizes or number of files	Limit on **account** and file size
Less Cost of storage	Cost is expensive

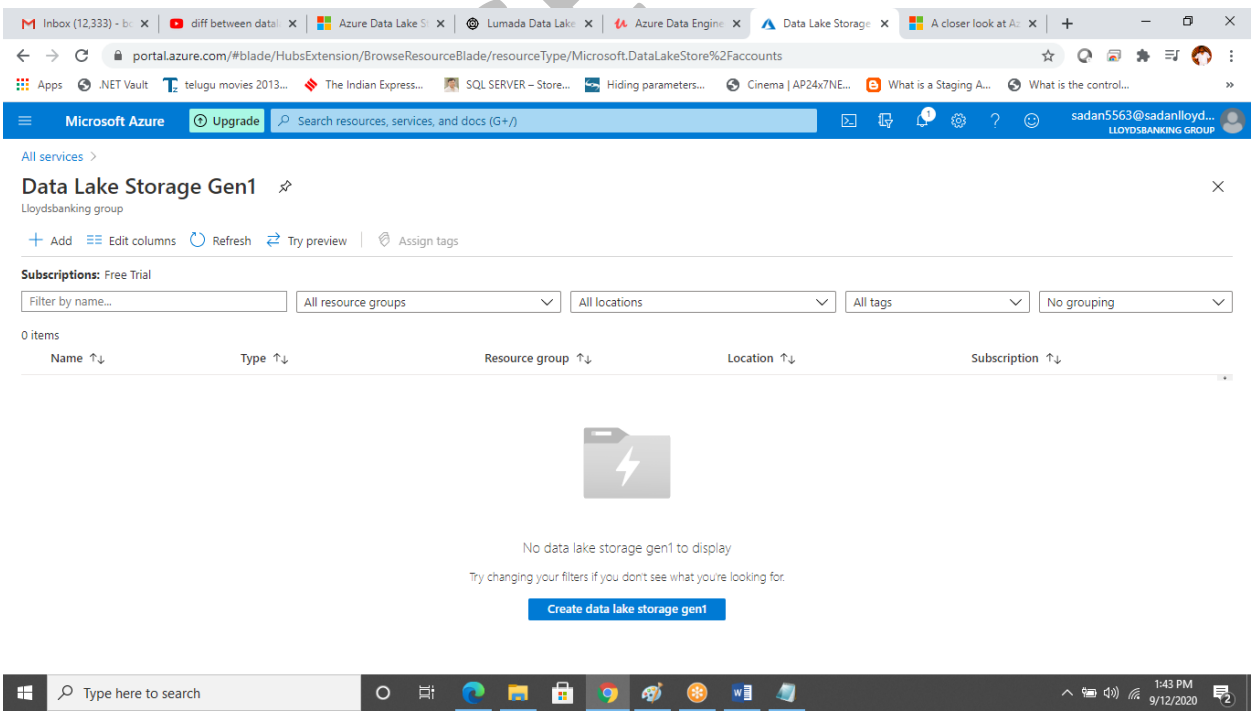
How to create Azure Data lake Storage Gen1 and upload Data.

ECLASESS.COM

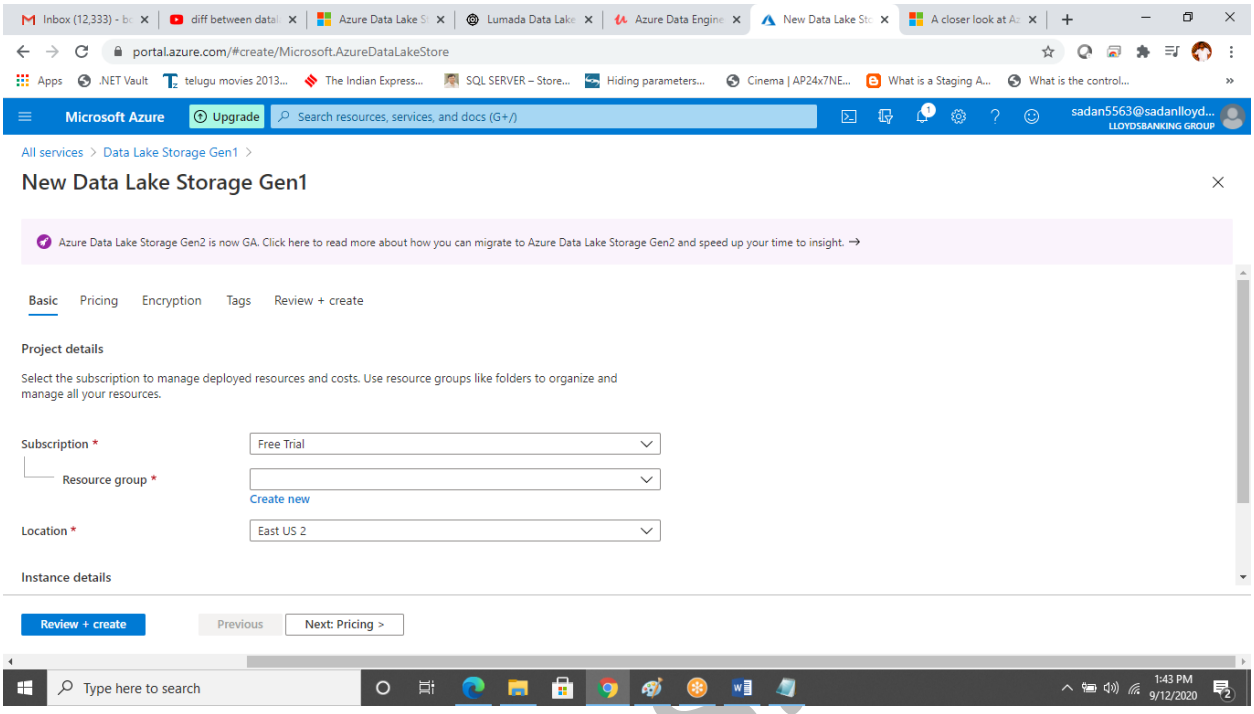
Step 1: Login to portal and search for Data Lake Storage Gen1.



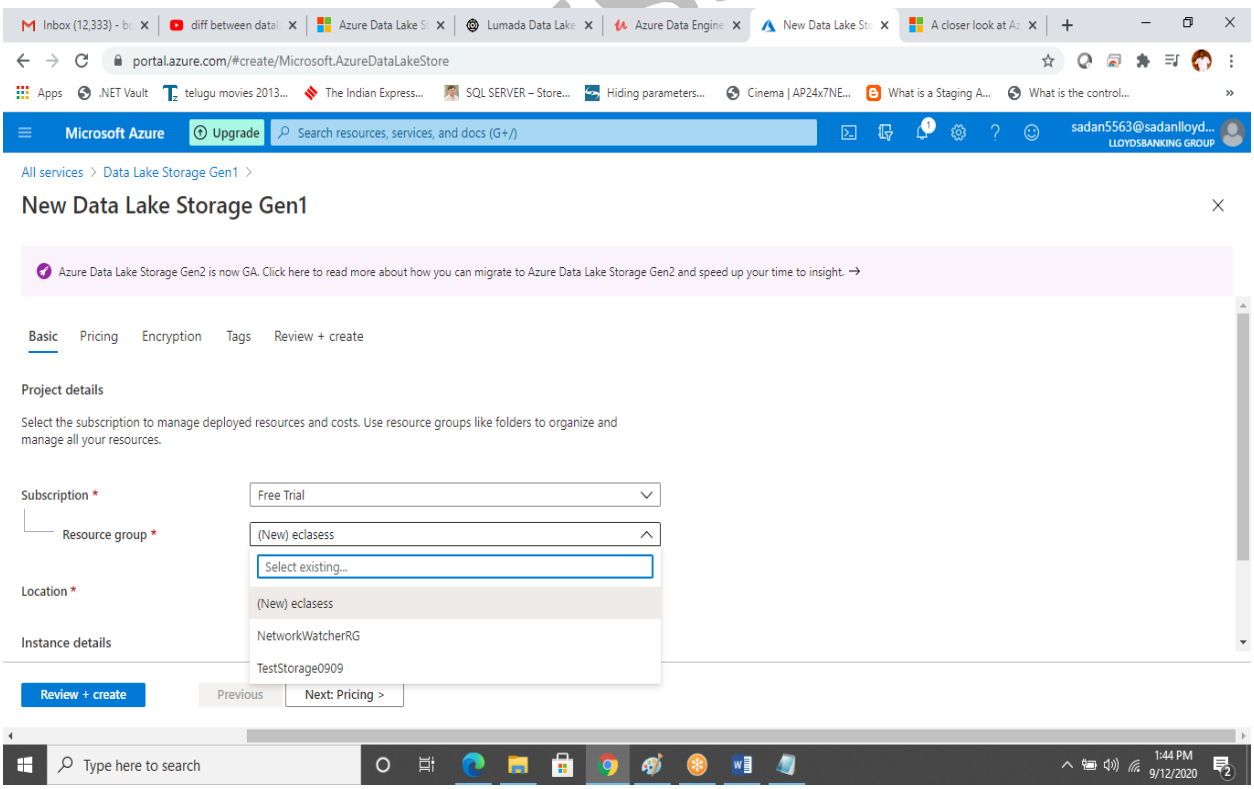
Step 2: Add Data lake Storage gen1.



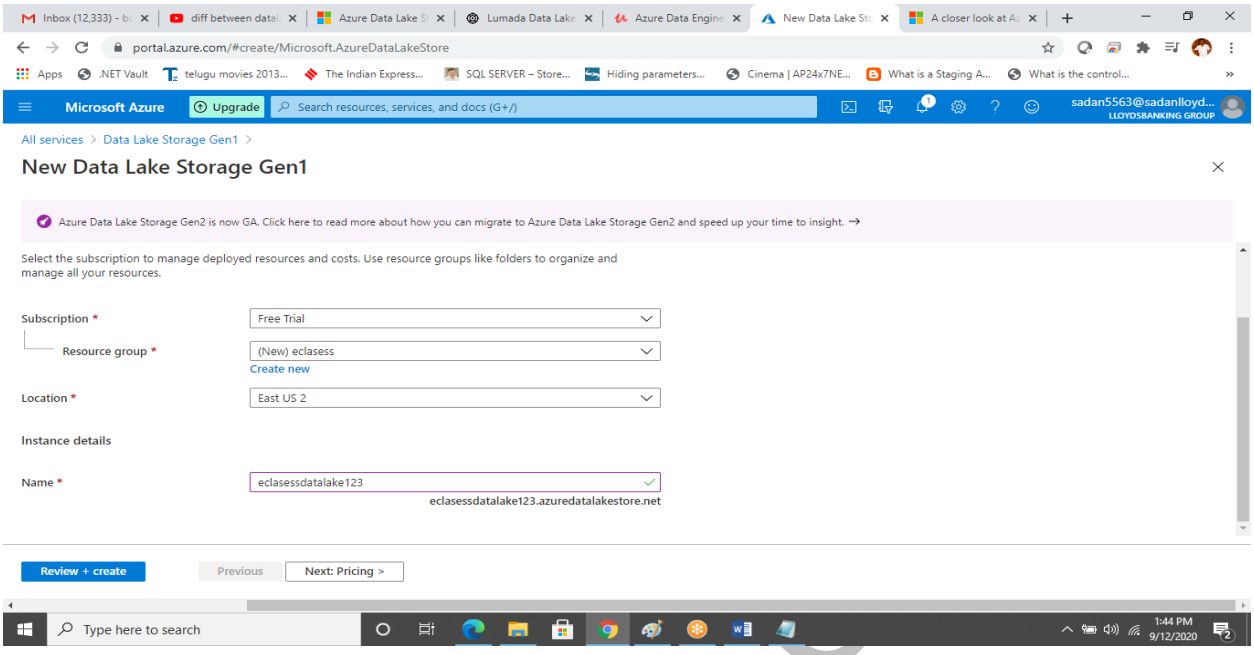
Step 3: Provide Subscription, resource Group, Location.



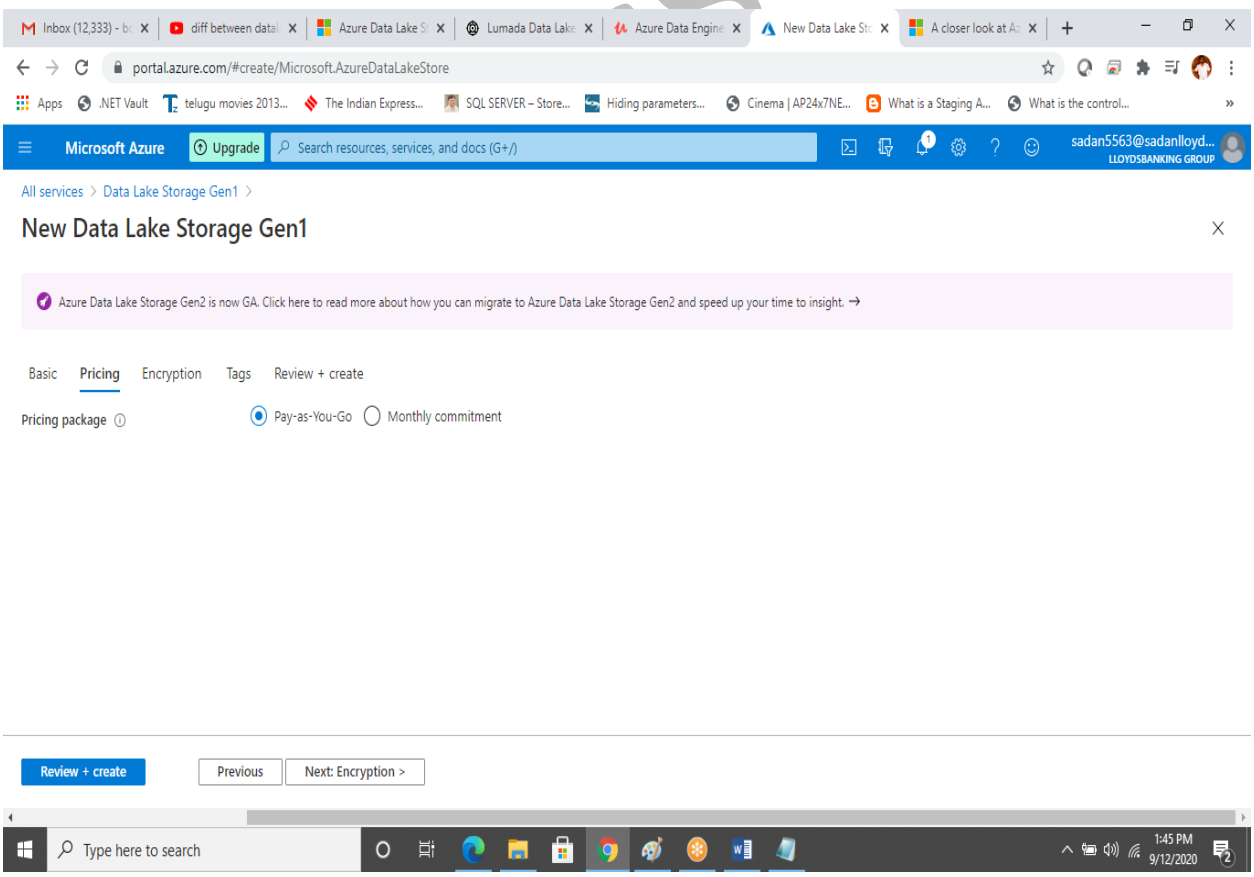
Step 4: Provide resource group name under list.



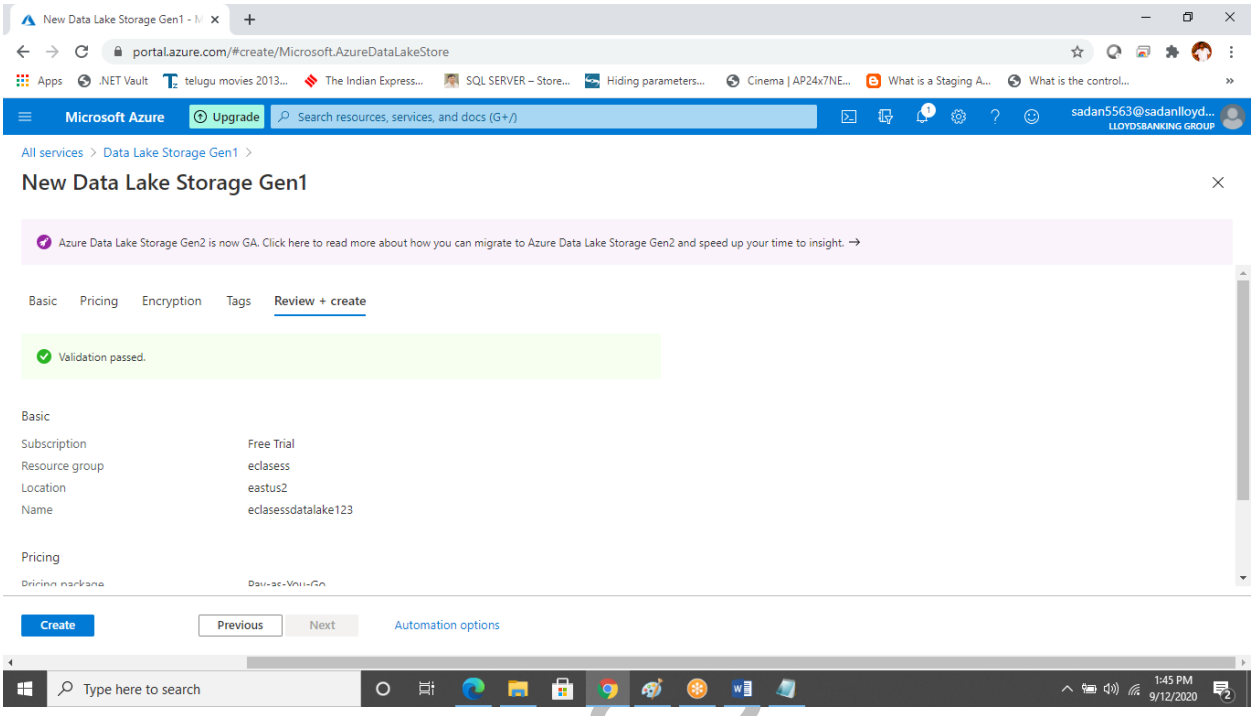
Step 5: provide Name of data Lake any name you can give.



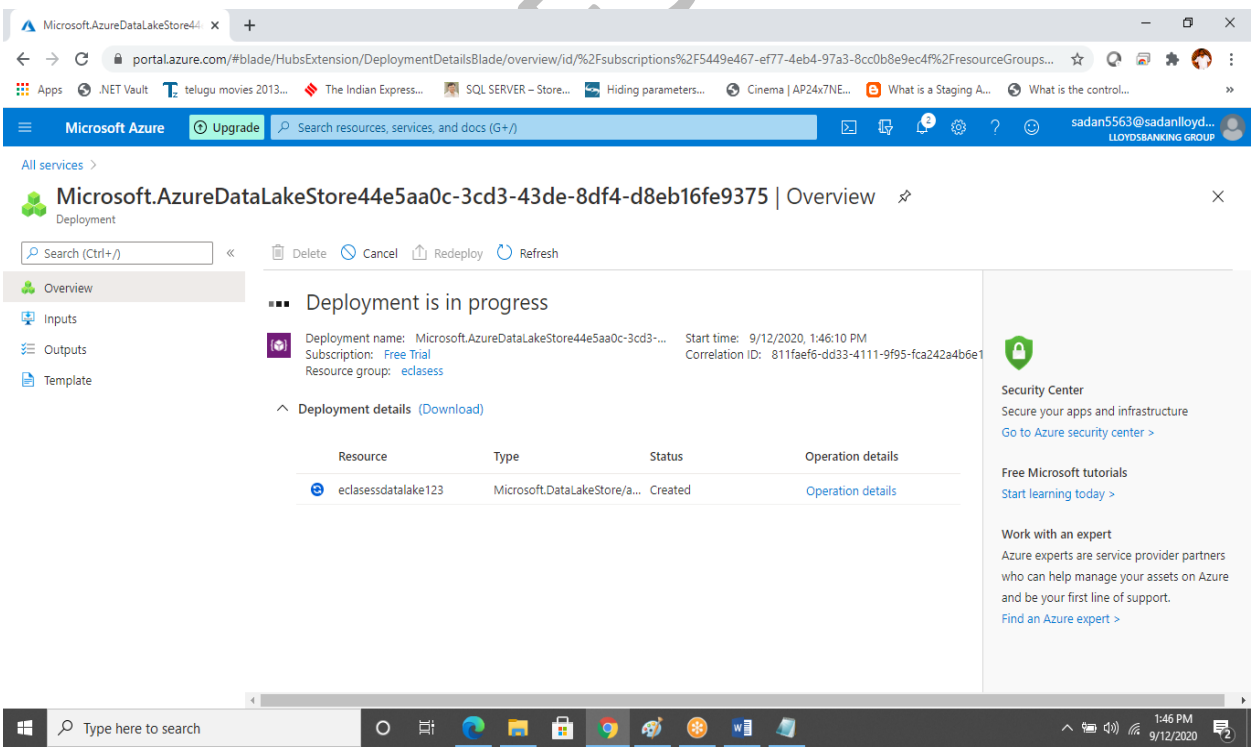
Step 6: go to the pricing and select pay as you go.



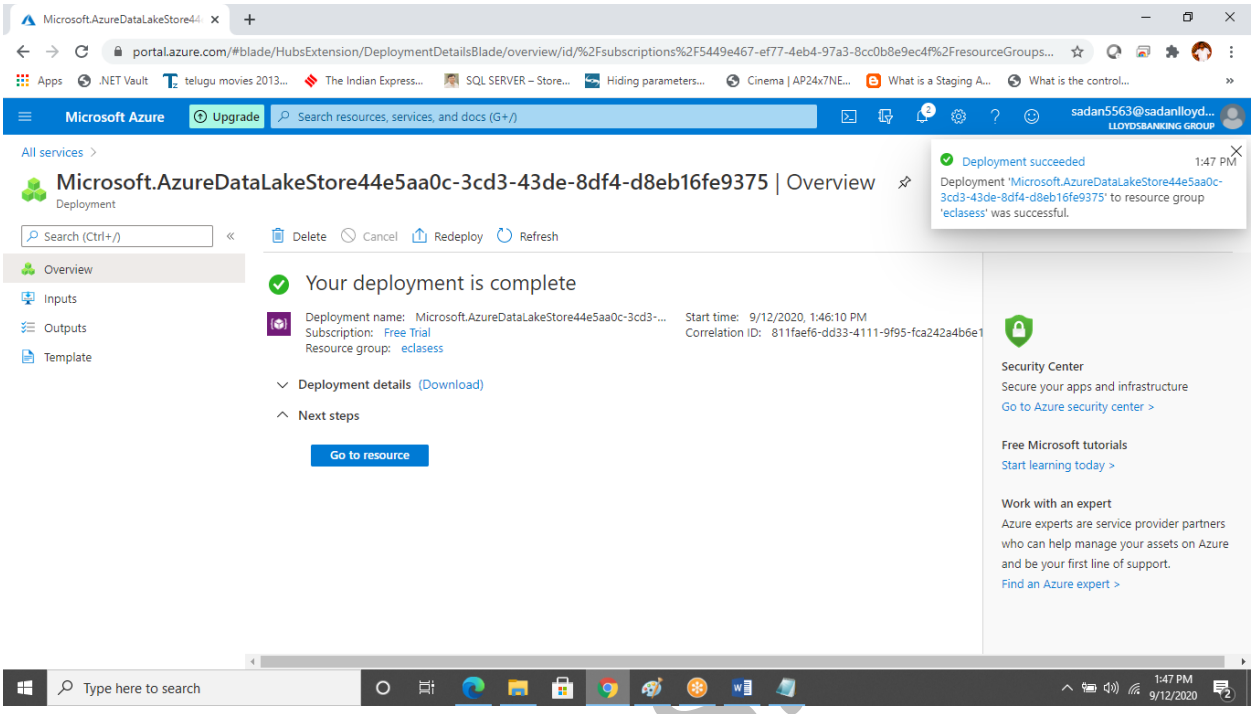
Step 7: click on Review and Create.



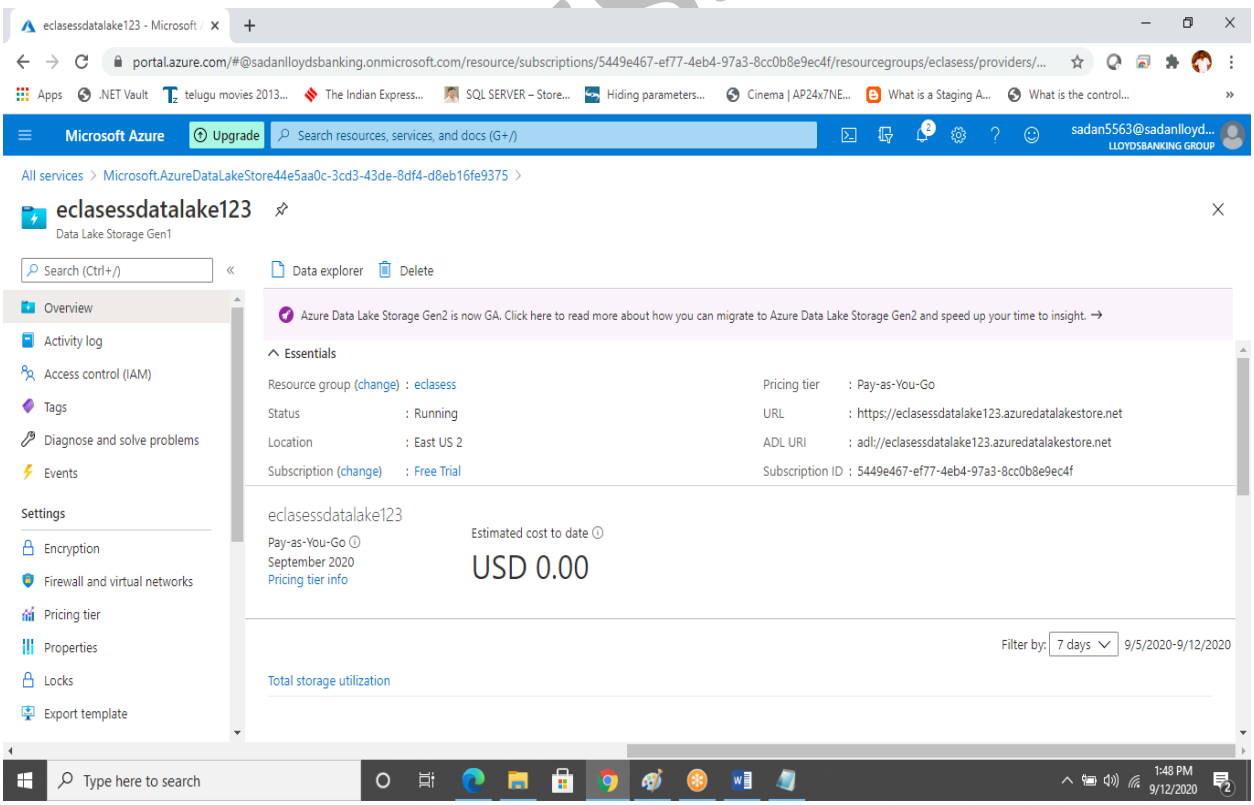
Step 8:



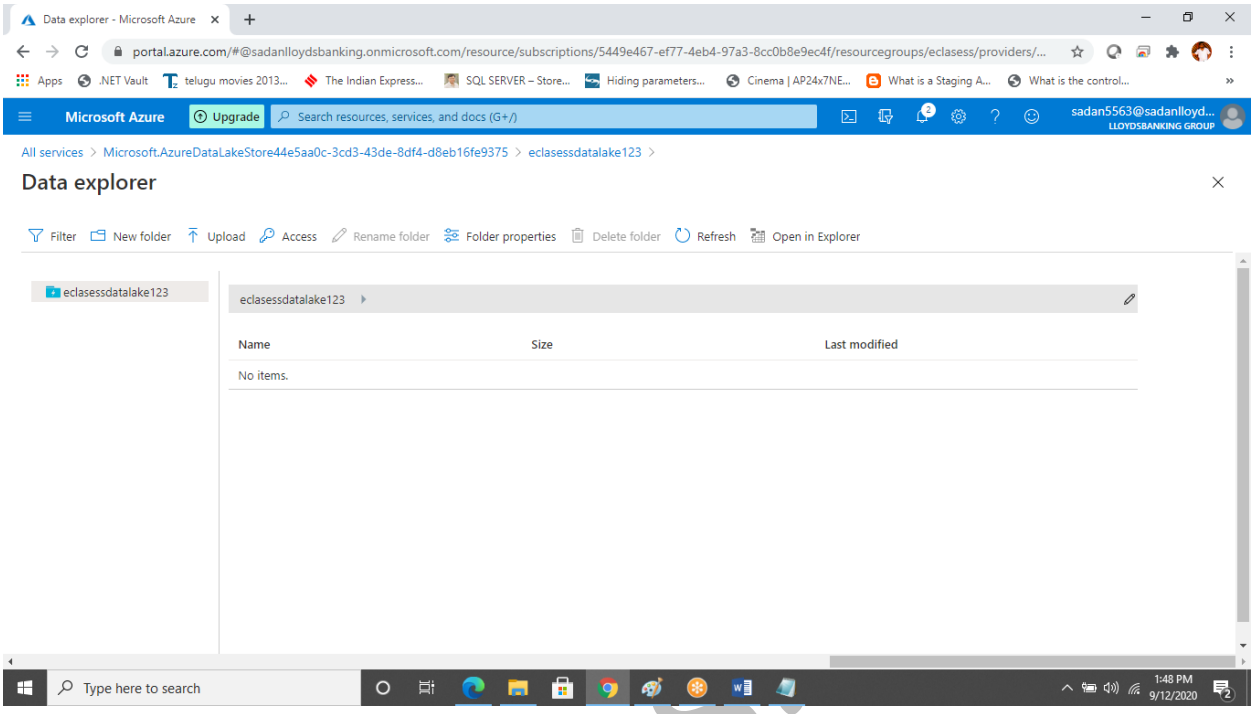
Step 9: Deployment has completed and click on Go to Resource.



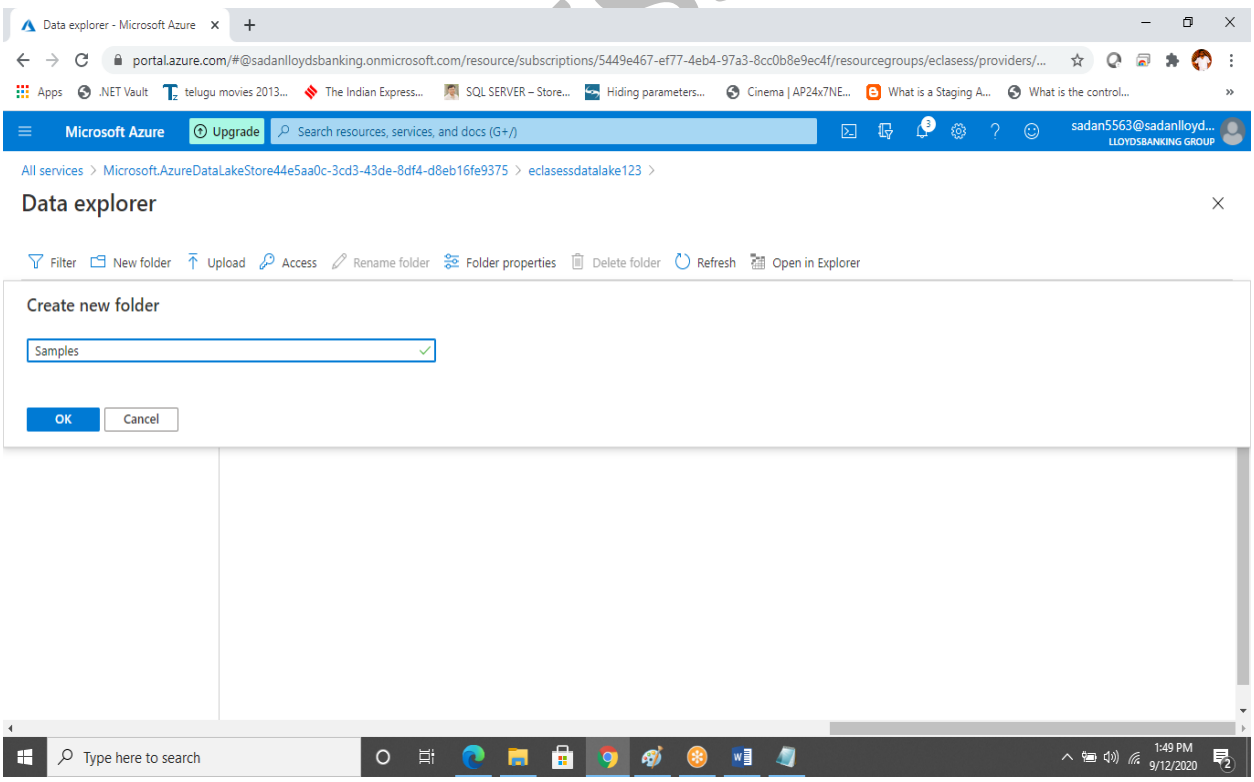
Step 10: click on Data explorer.



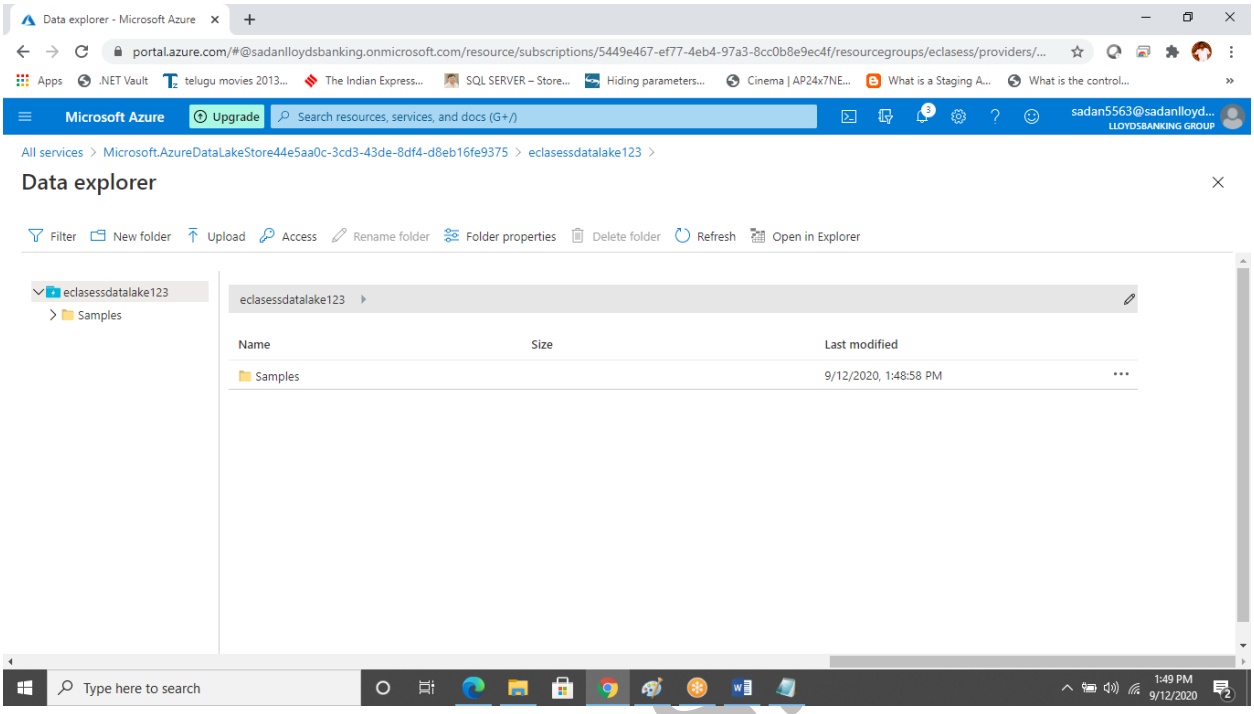
Step 11: You can see all options of data Explorer.



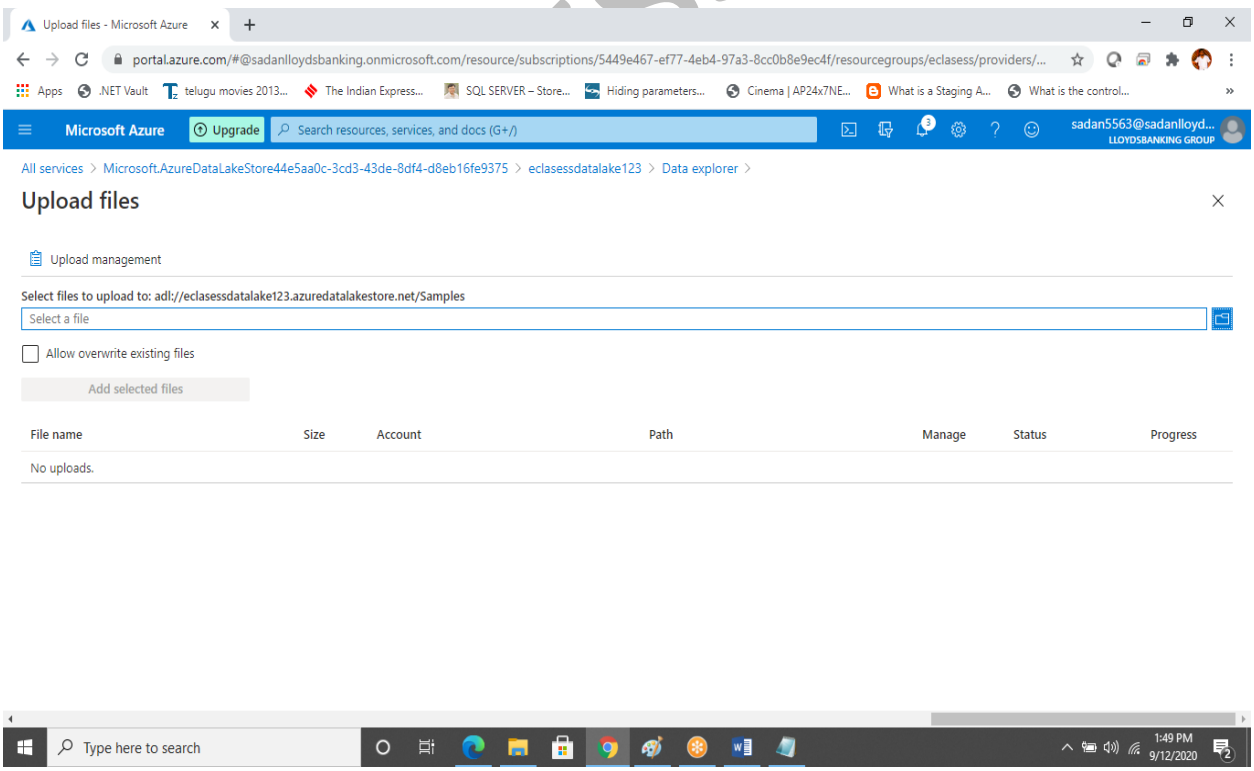
Step-12 : you can create new folder.



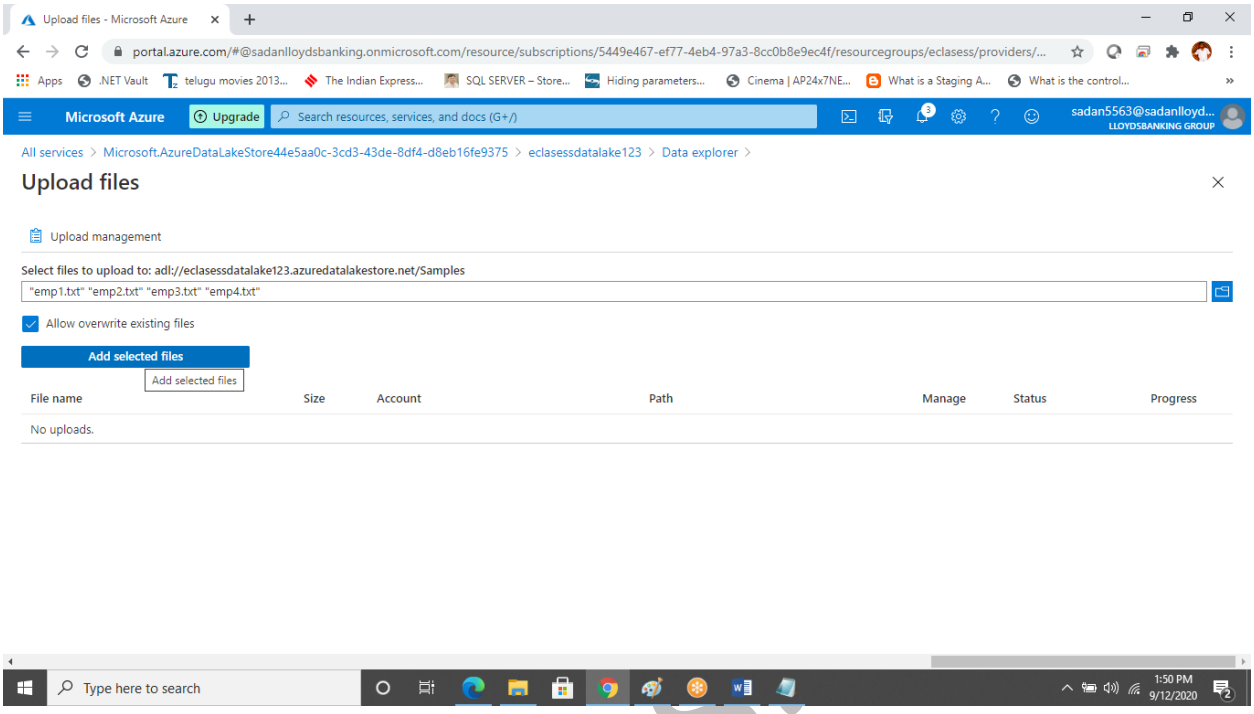
Step 13: Go to folder created as sample



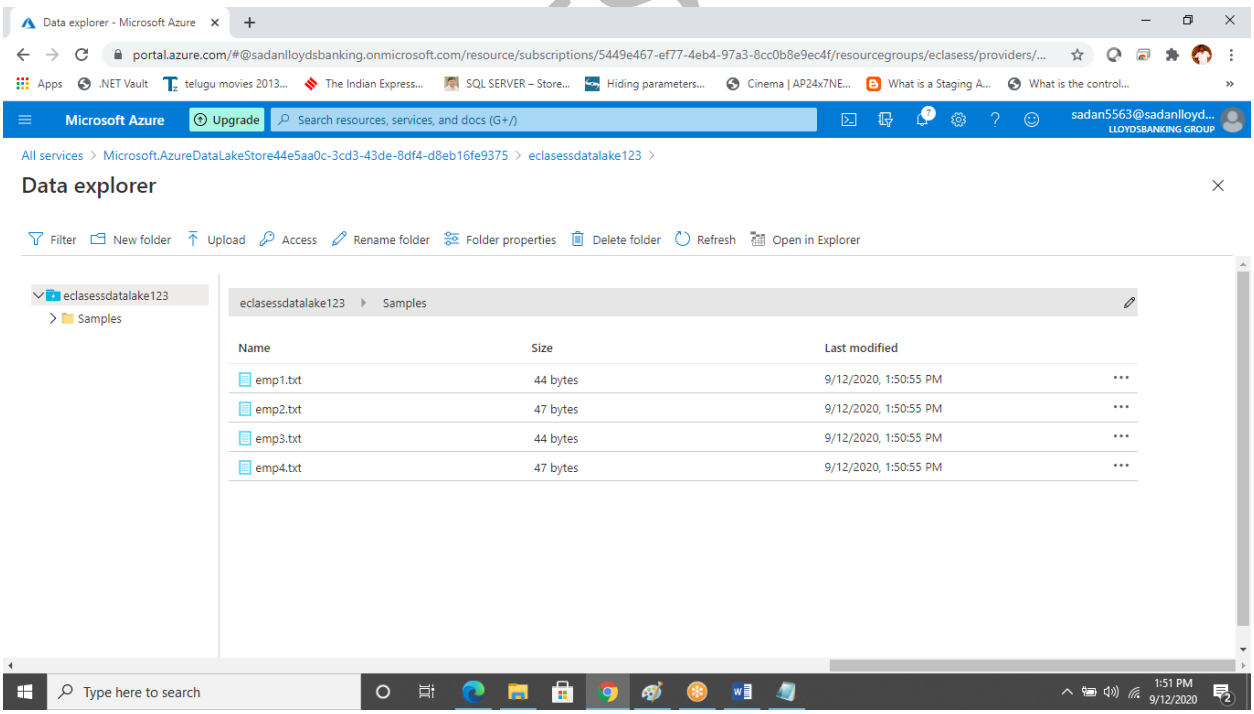
Step 14: Upload files



Step 15: add selected files.

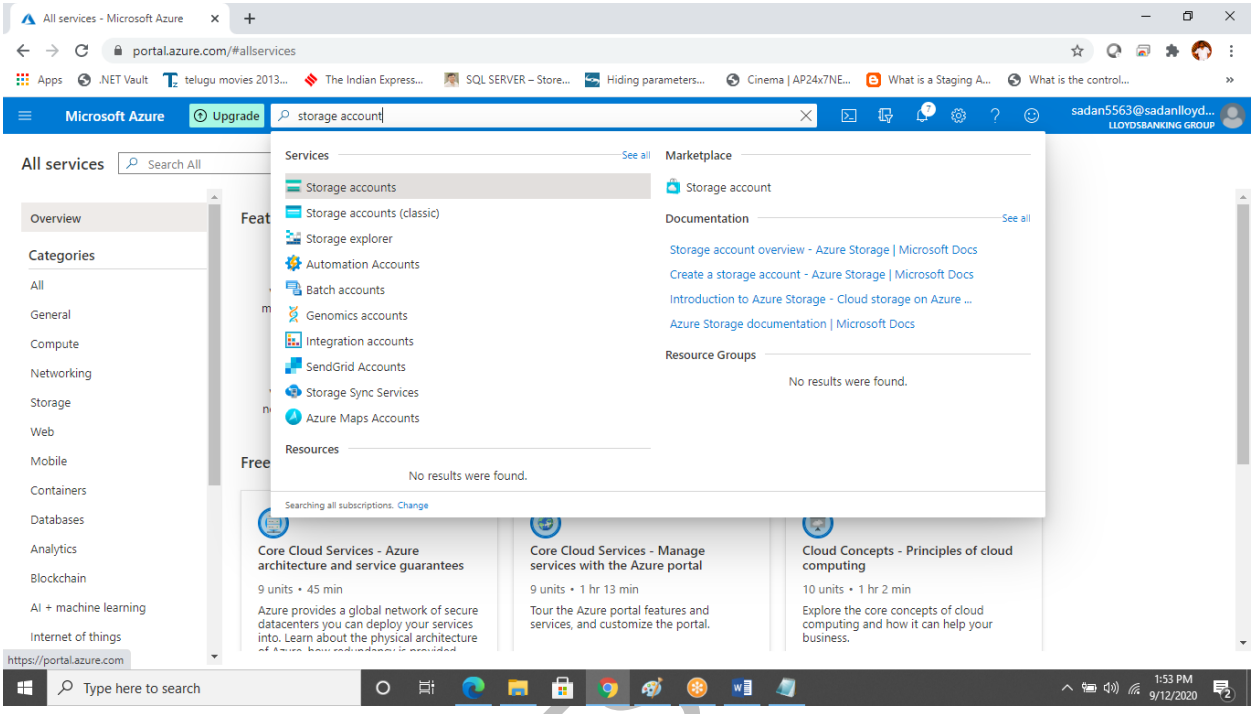


Step -16: See list of files added

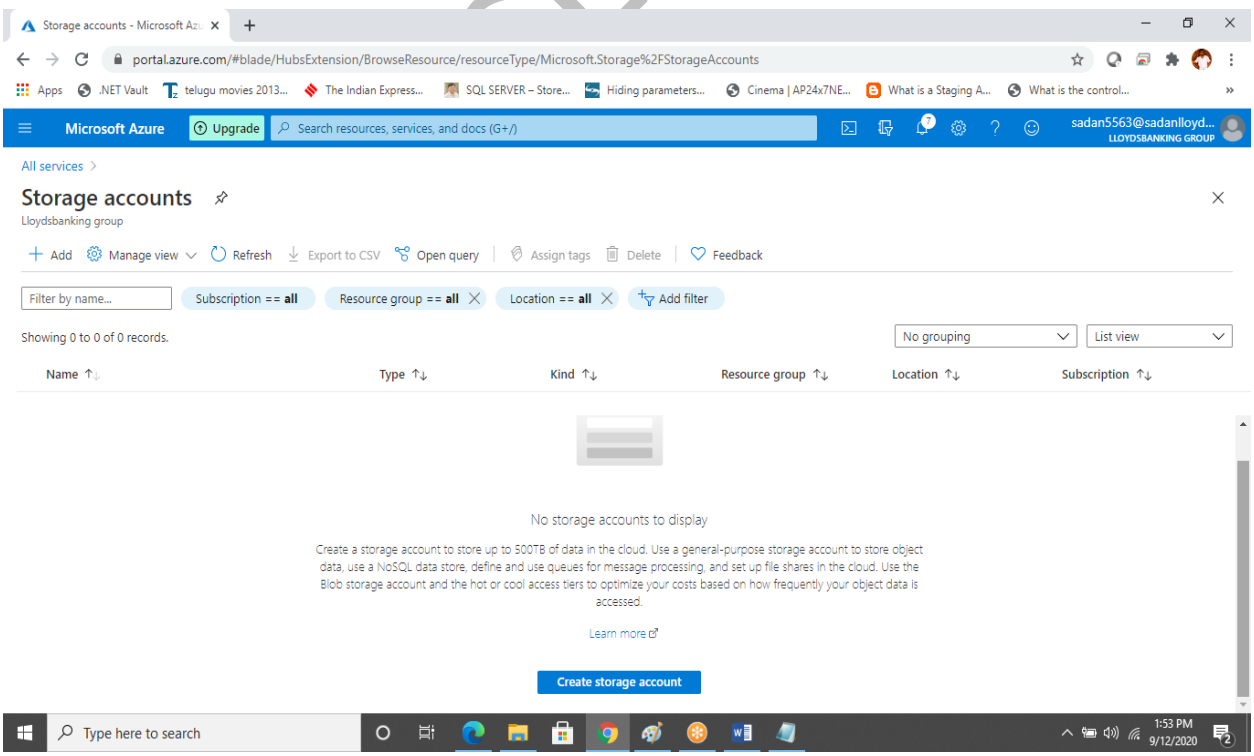


How to Create Data lake Storage Gen2 Account and upload Data

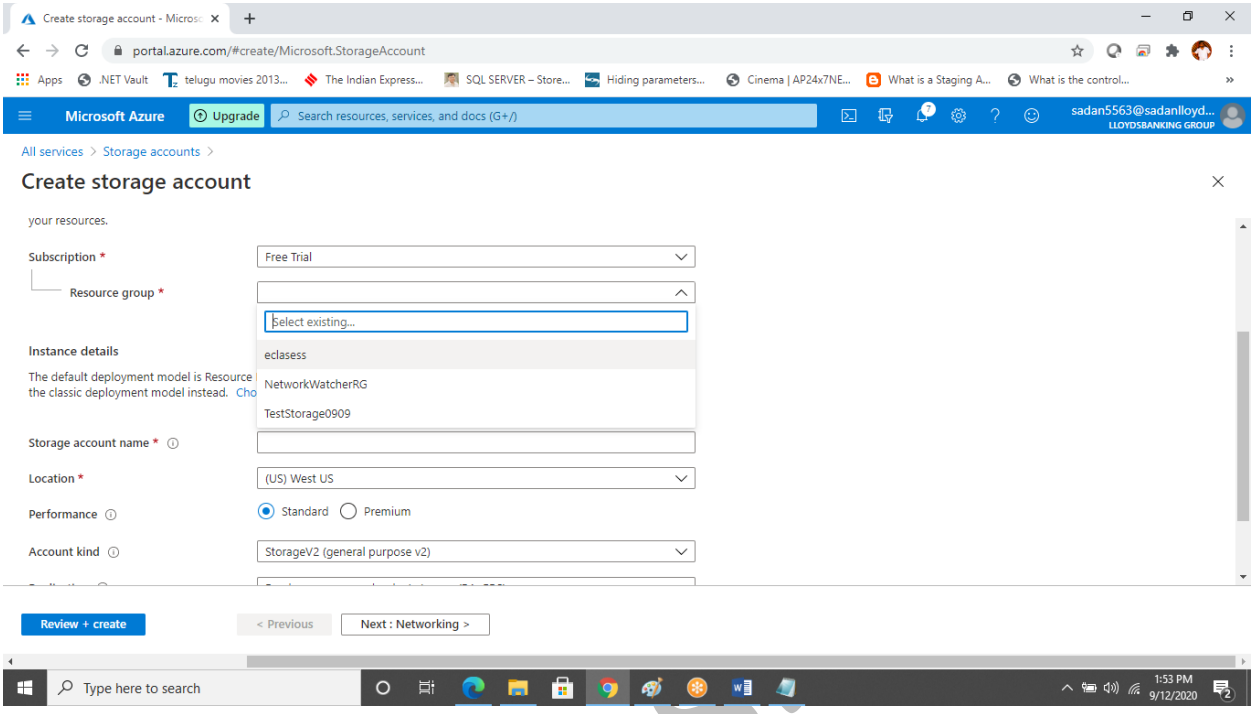
Step 1: Login to Azure Portal and search for Storage Account.



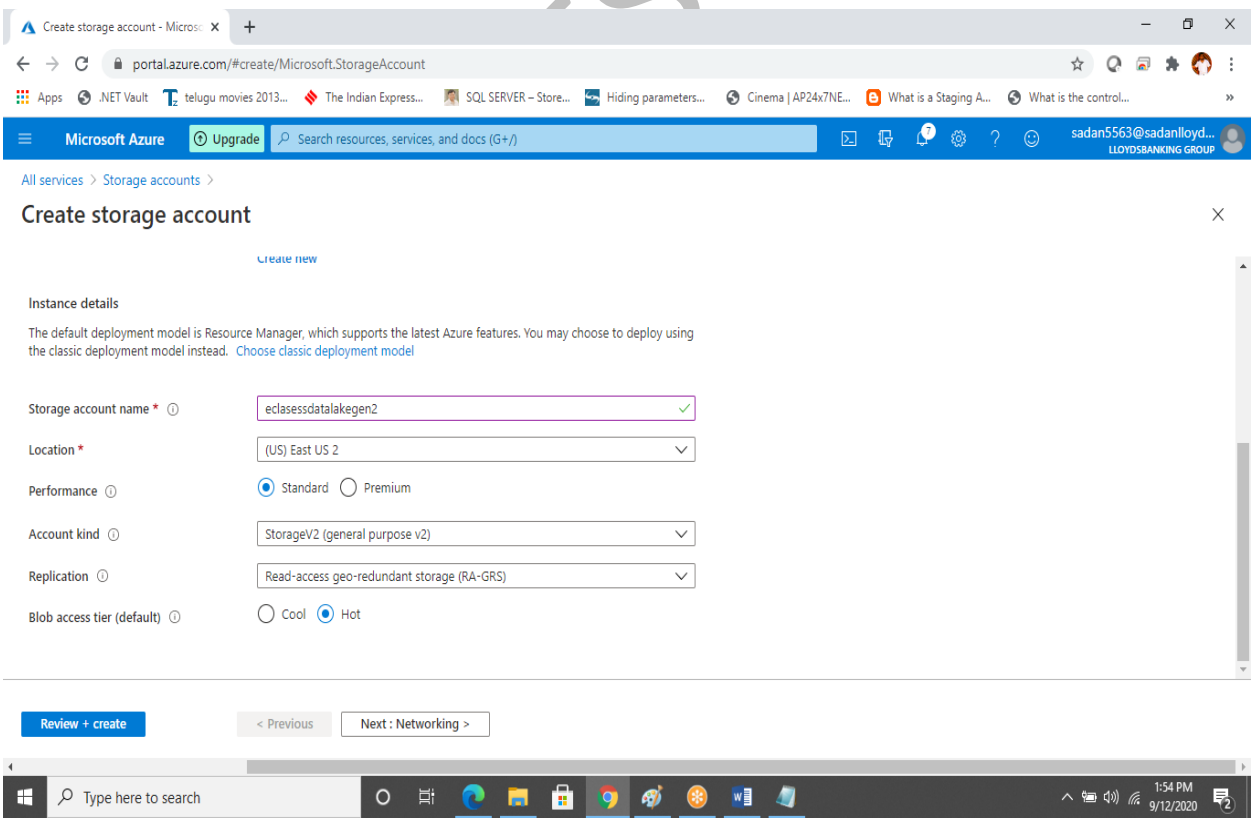
Step 2 : Click on Create on Storage Account



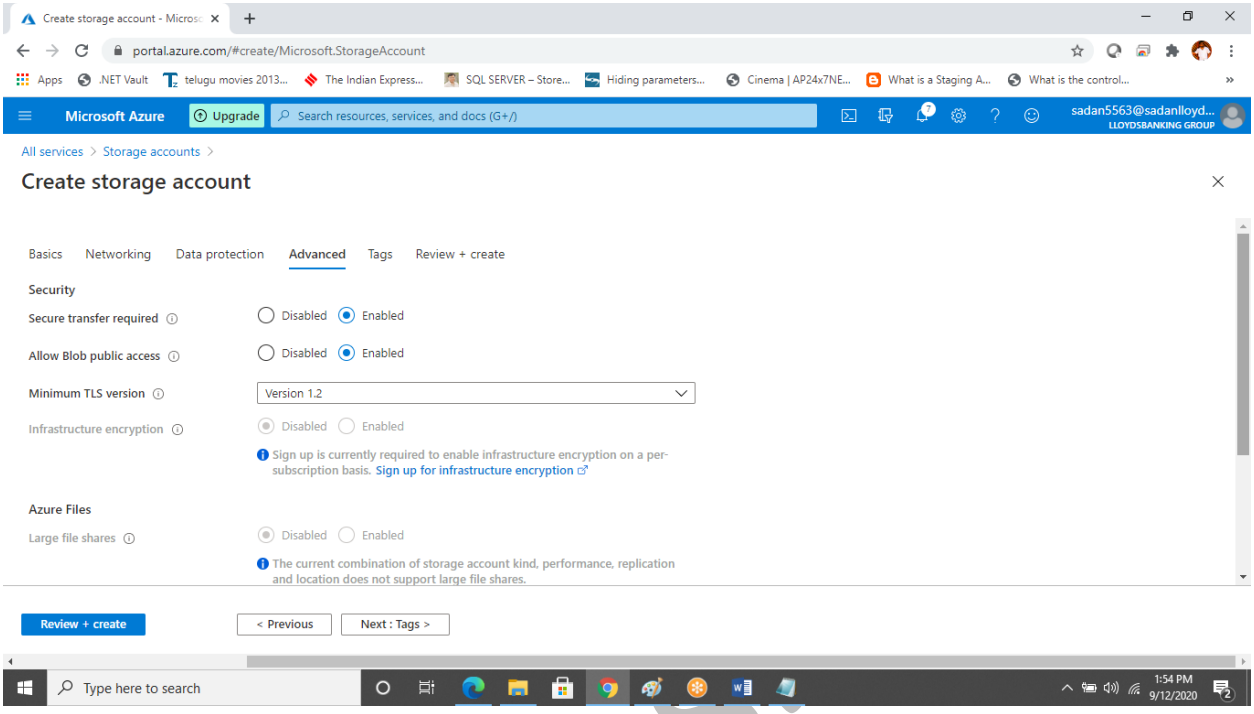
Step-3: Provide Subscription, Resource Group Details.



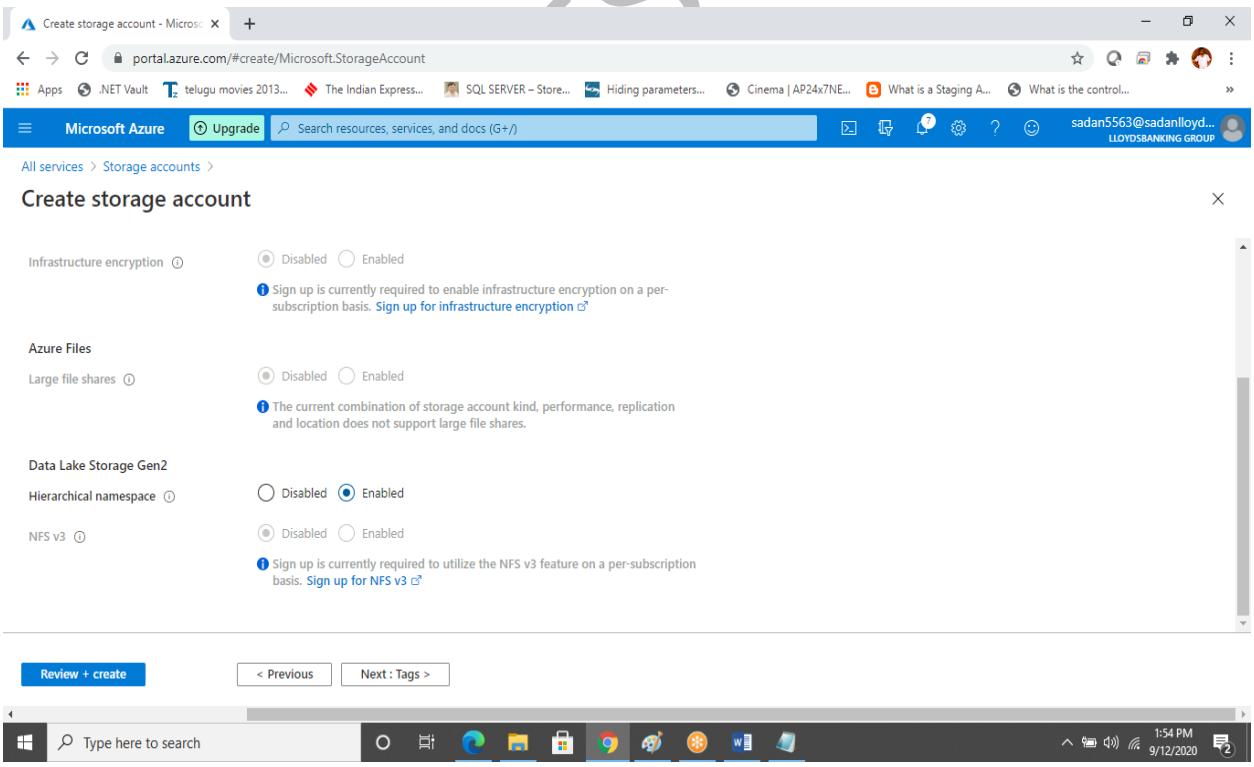
Step 4 : Give Storage Account Name (Any Name) and Location and keep all default options.



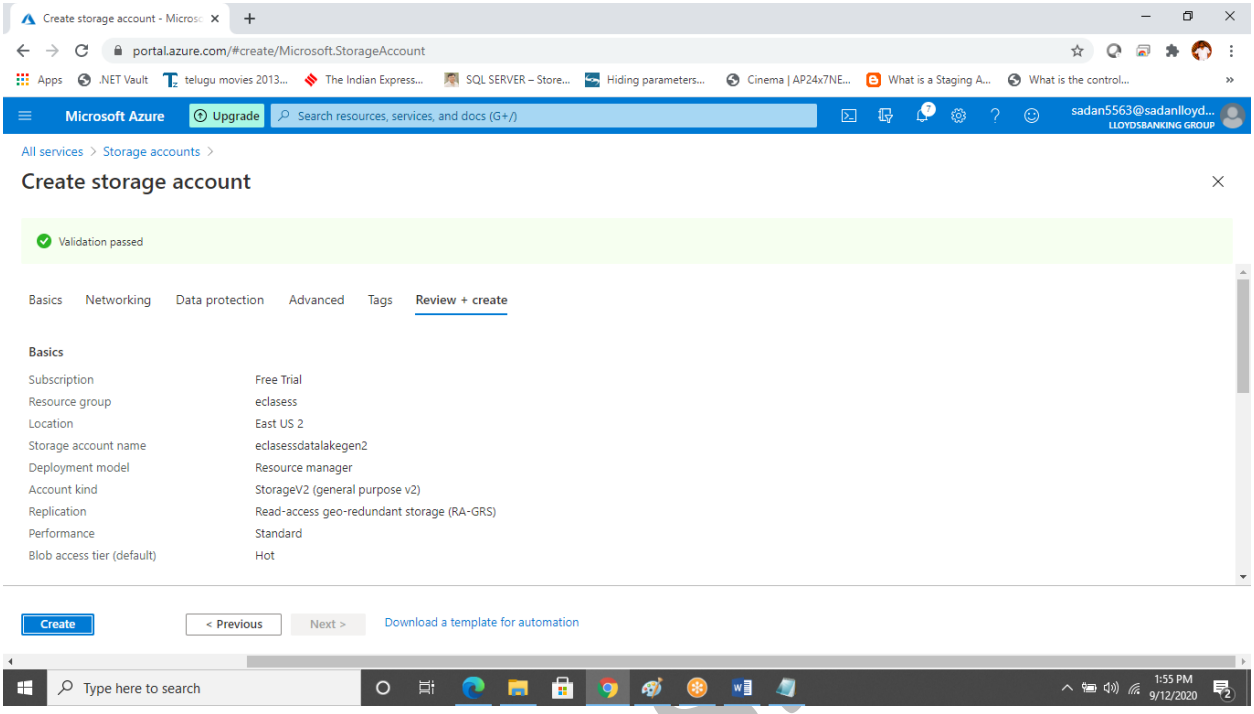
Step 5: go to advanced tab.



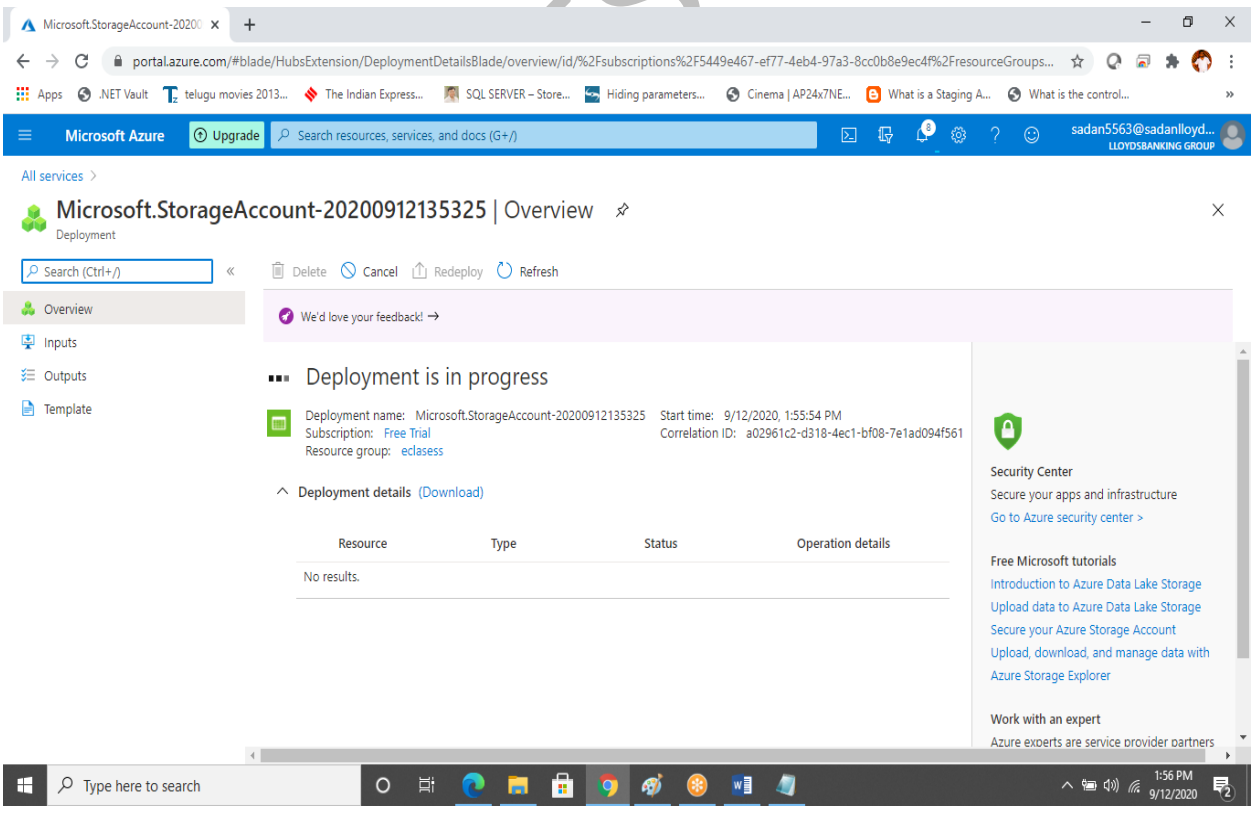
Step 6 : Enable Data Lake Storage Gen2 ? Hierarchal namespace?Enable.



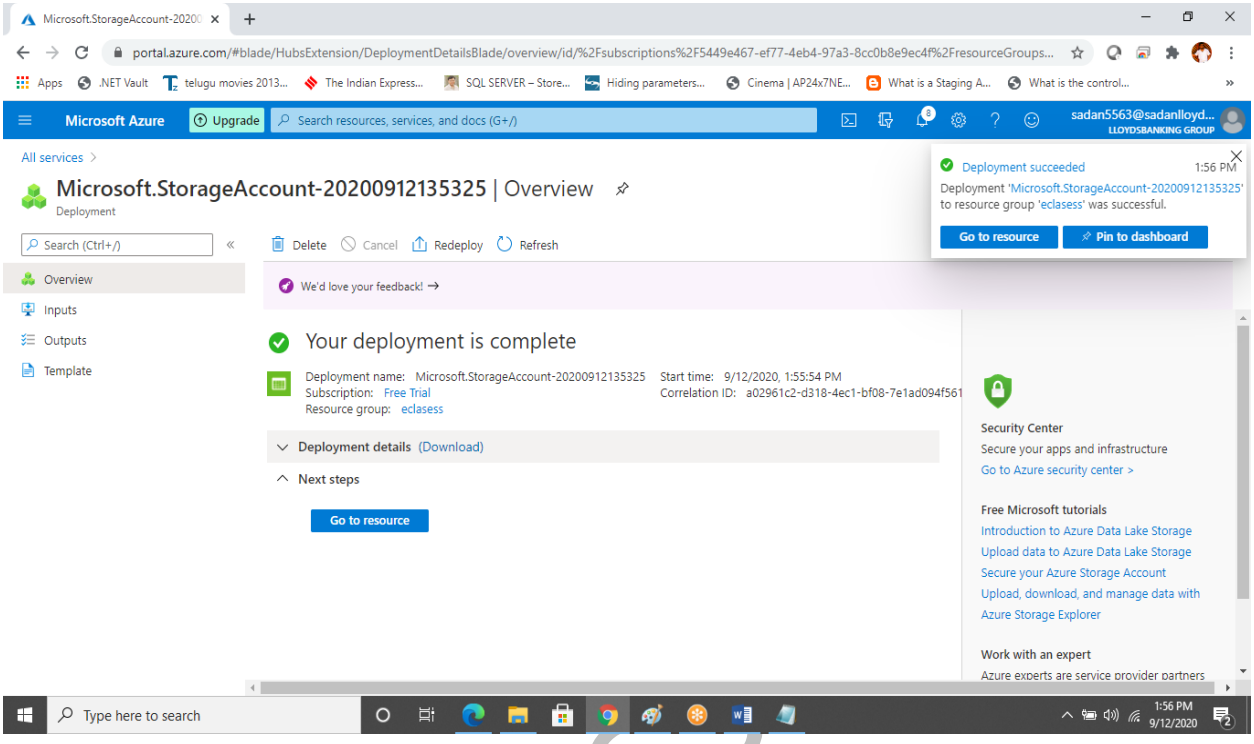
Step 7 : Click on review+ Create.



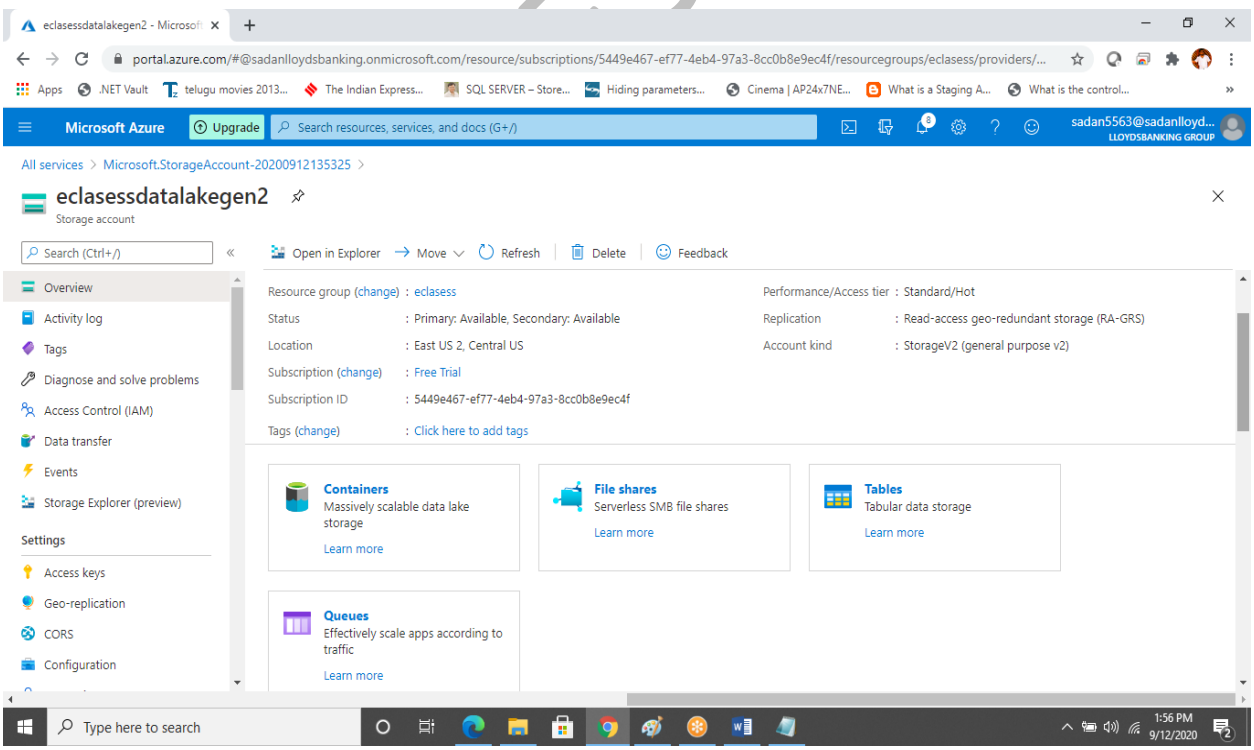
Step 8: Deployment is in Progress.



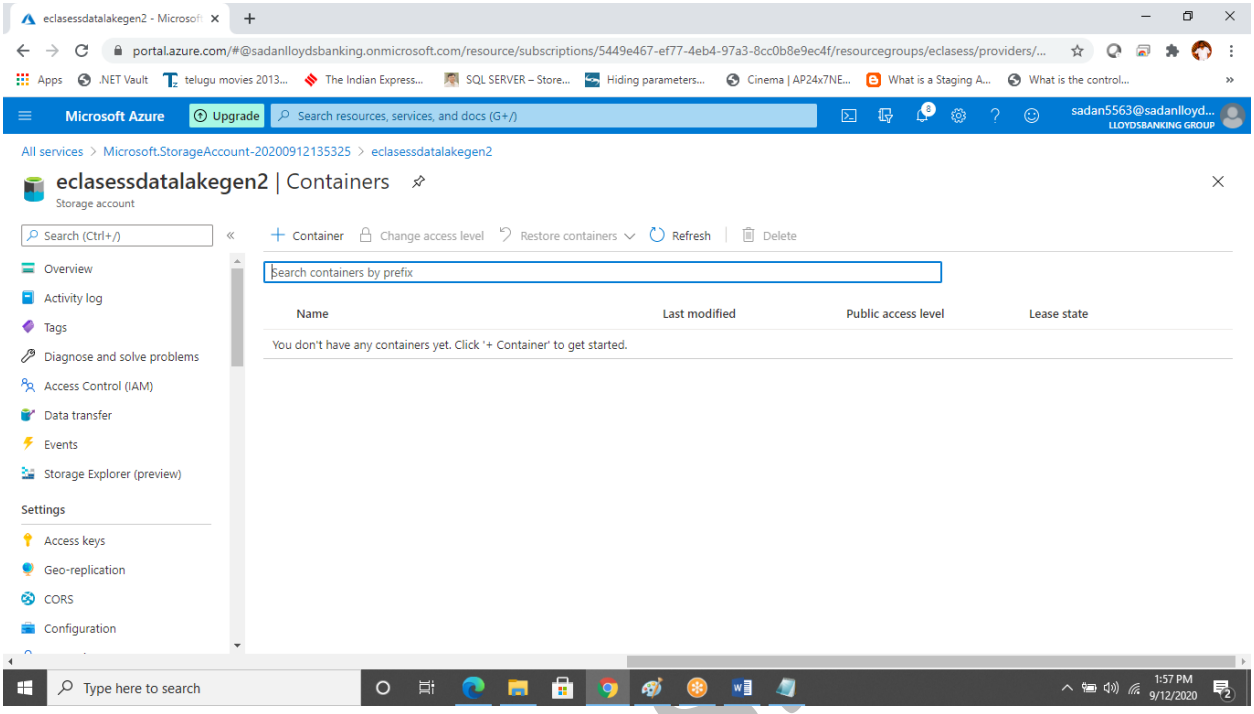
Step 9 : Click on Goto resource.



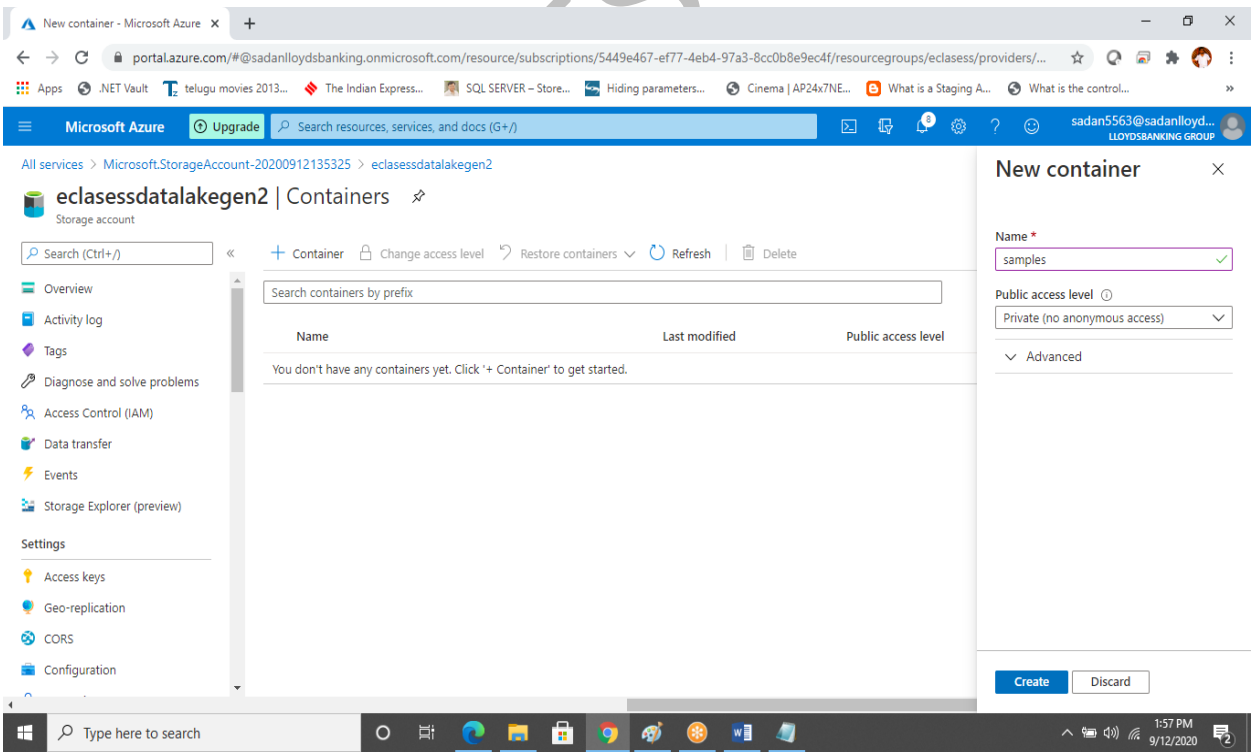
Step 10 : Click on Create Container.



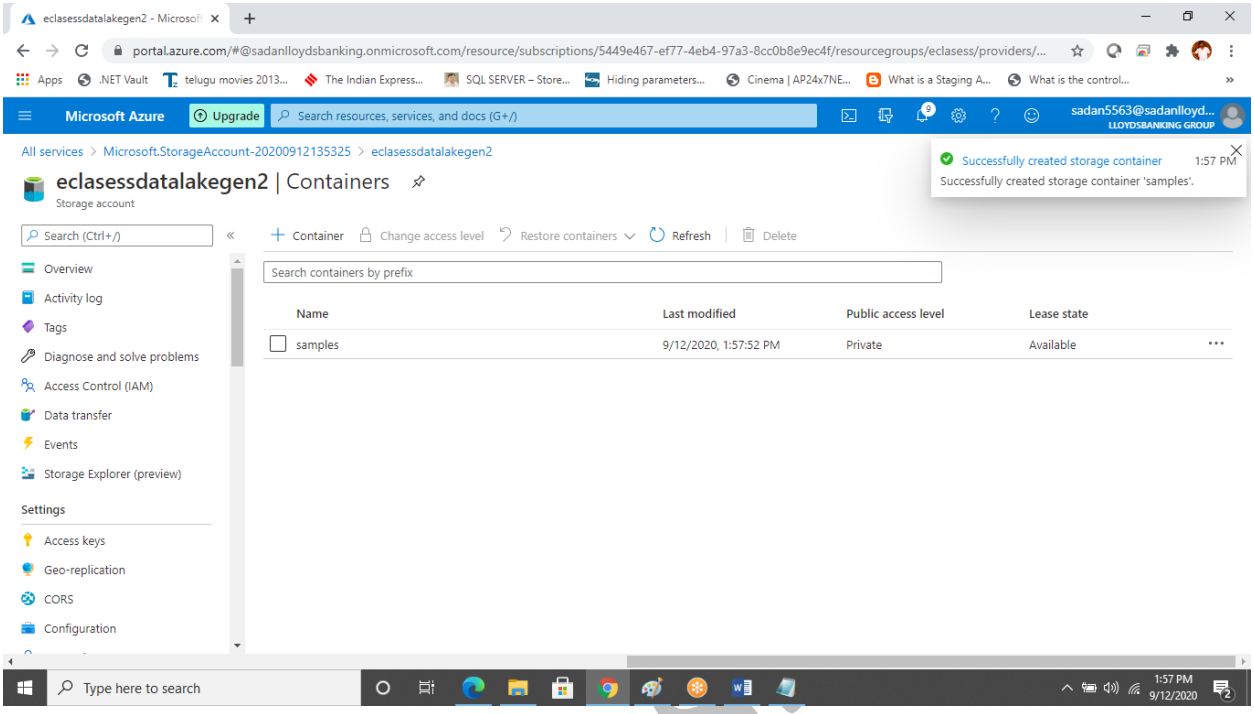
Step 11: Click on container.



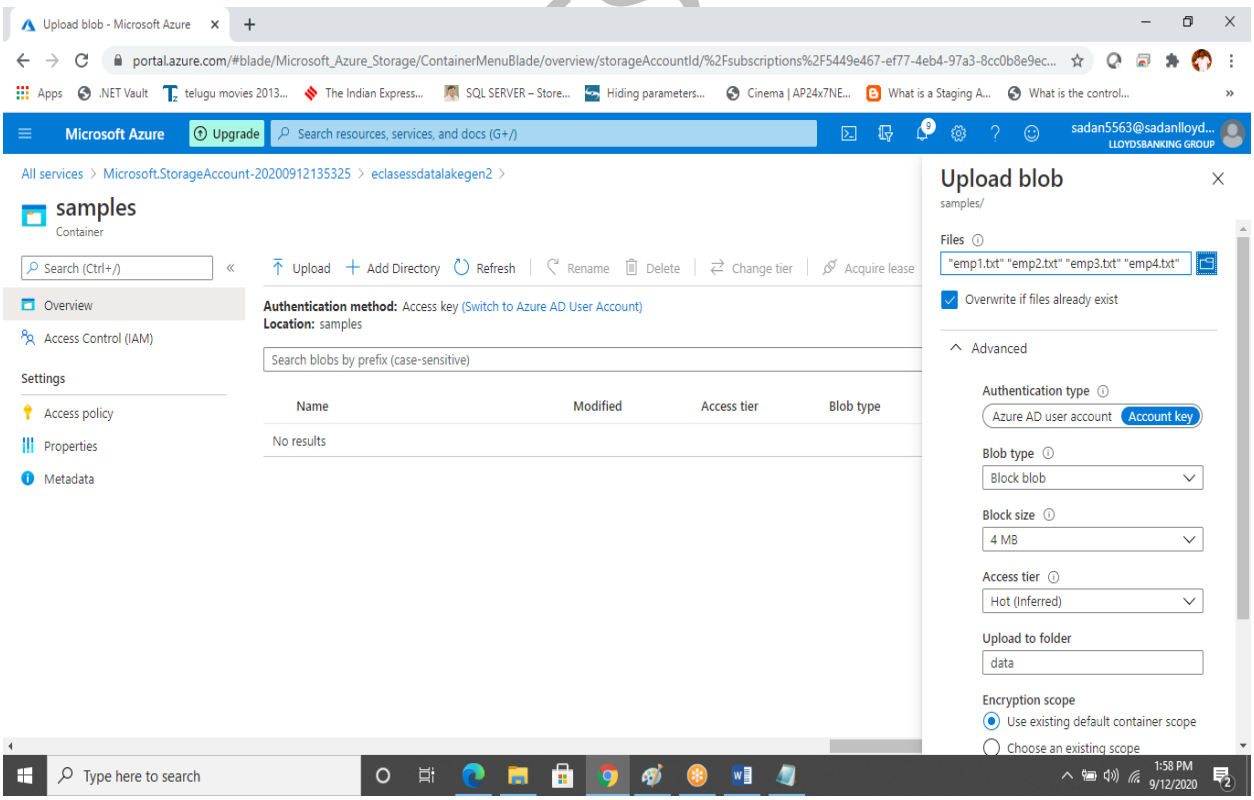
Step 12: Provide Container Name.



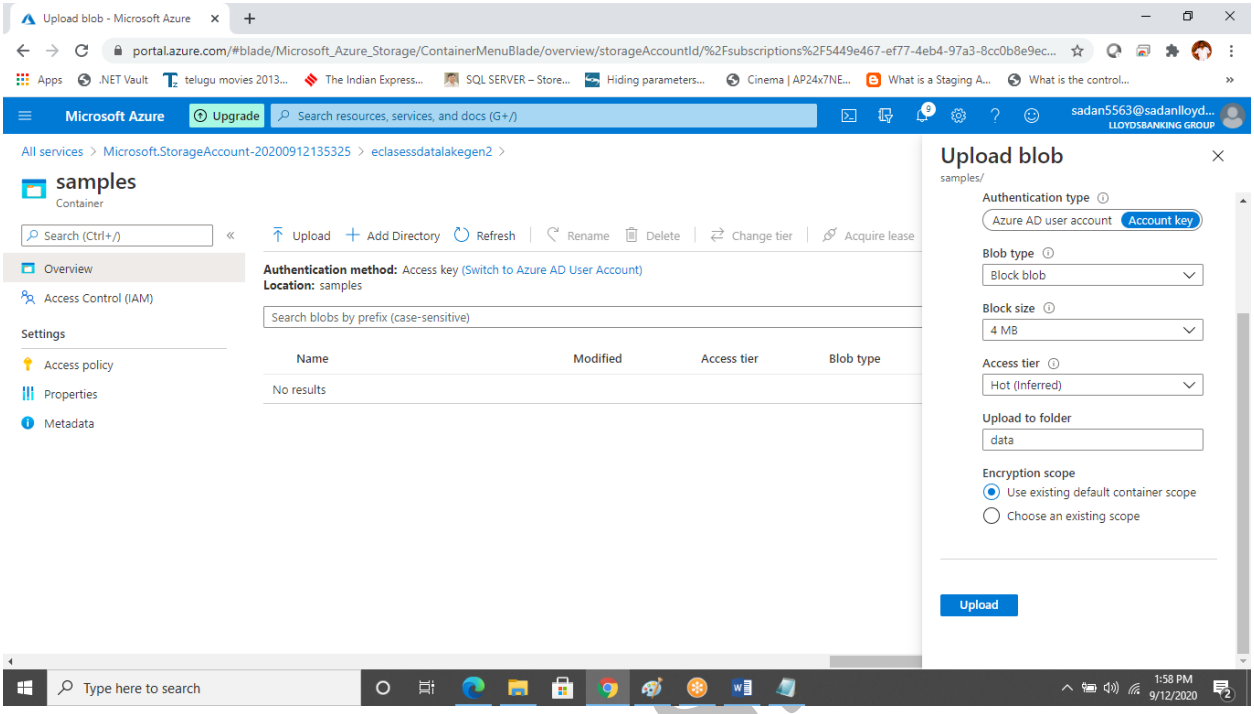
Step 13 : we can see created container.



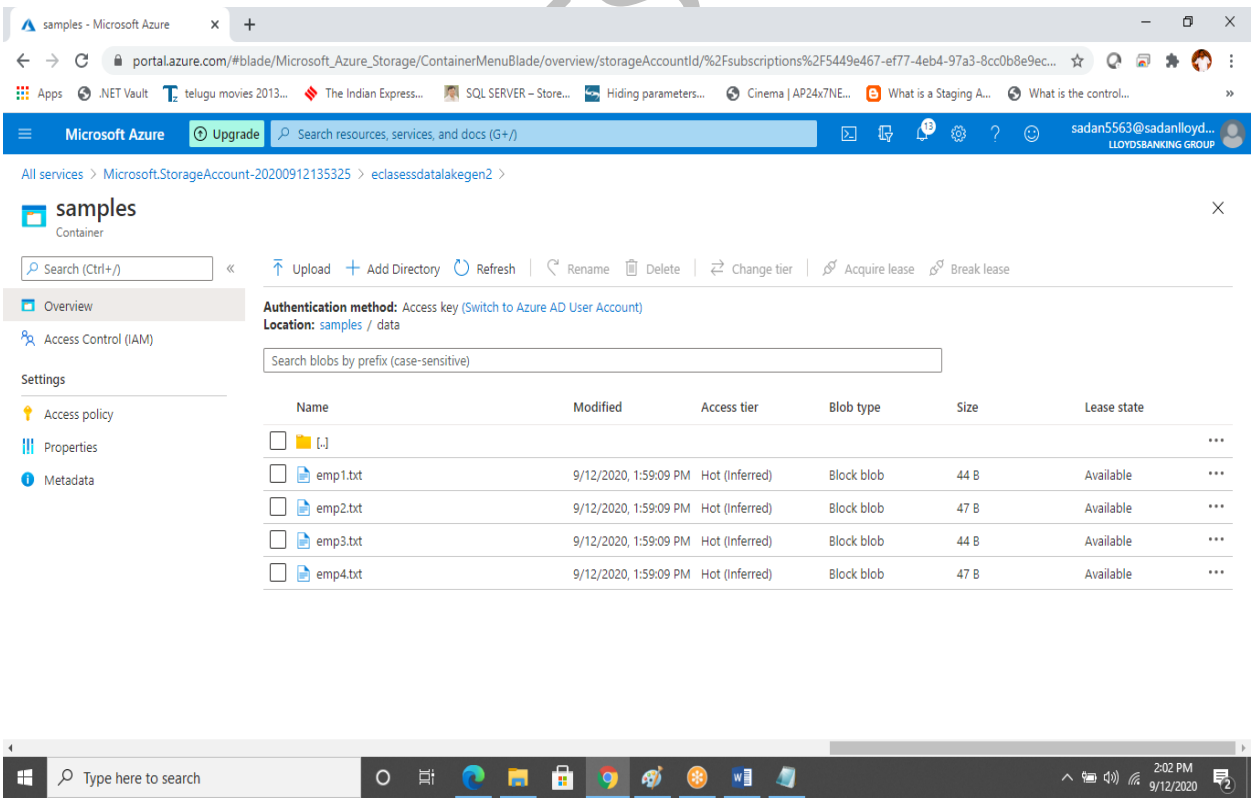
Step 14: You can upload files by click on upload.



Step 15: Click on Upload



Step 16: You can all list of files uploaded.

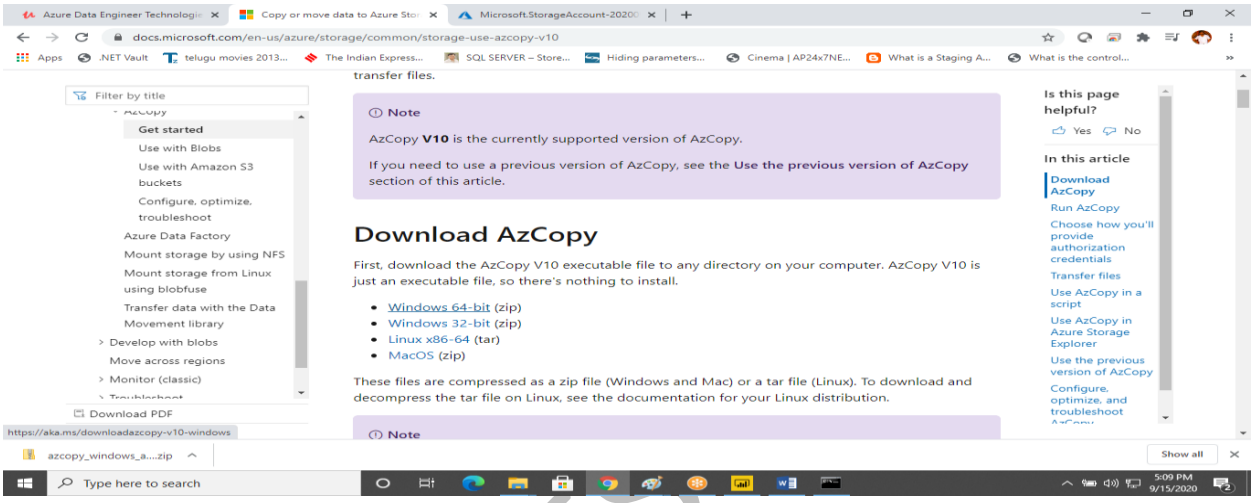


How to Copy data into Azure Data Lake using AZ Copy:

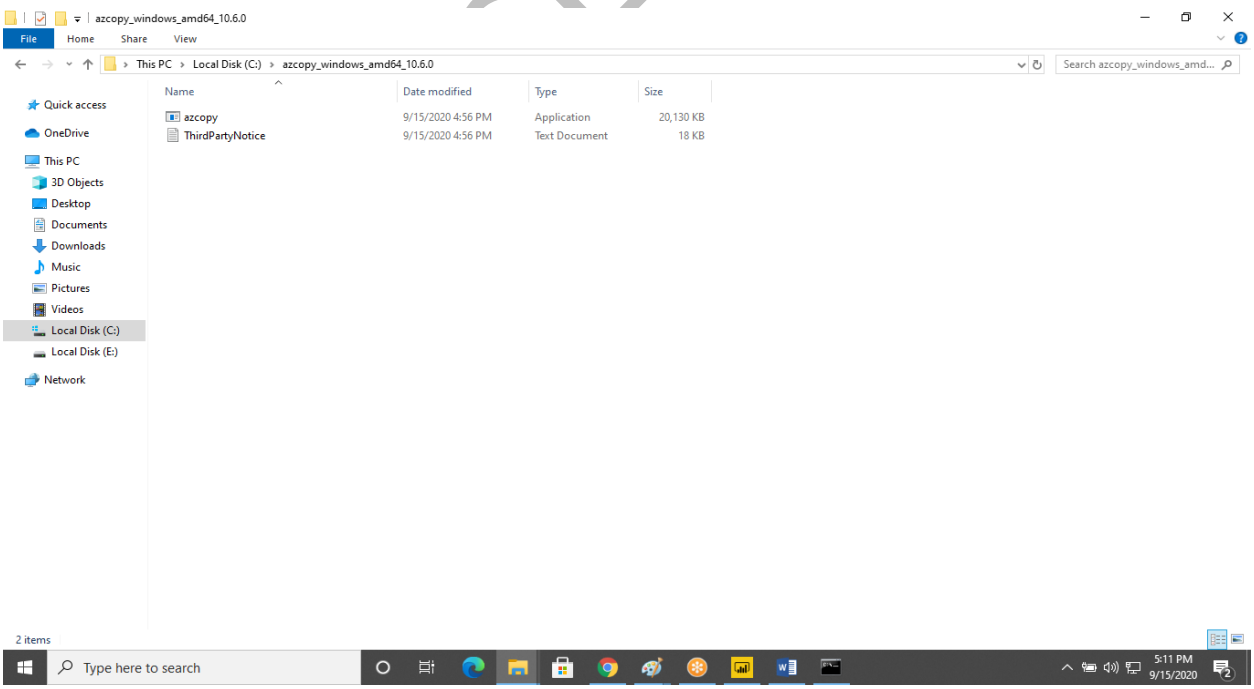
Copying data from on premises to Azure data Lake Storage Gen2.

Step-1 : Download Azure copy executable file from below path:

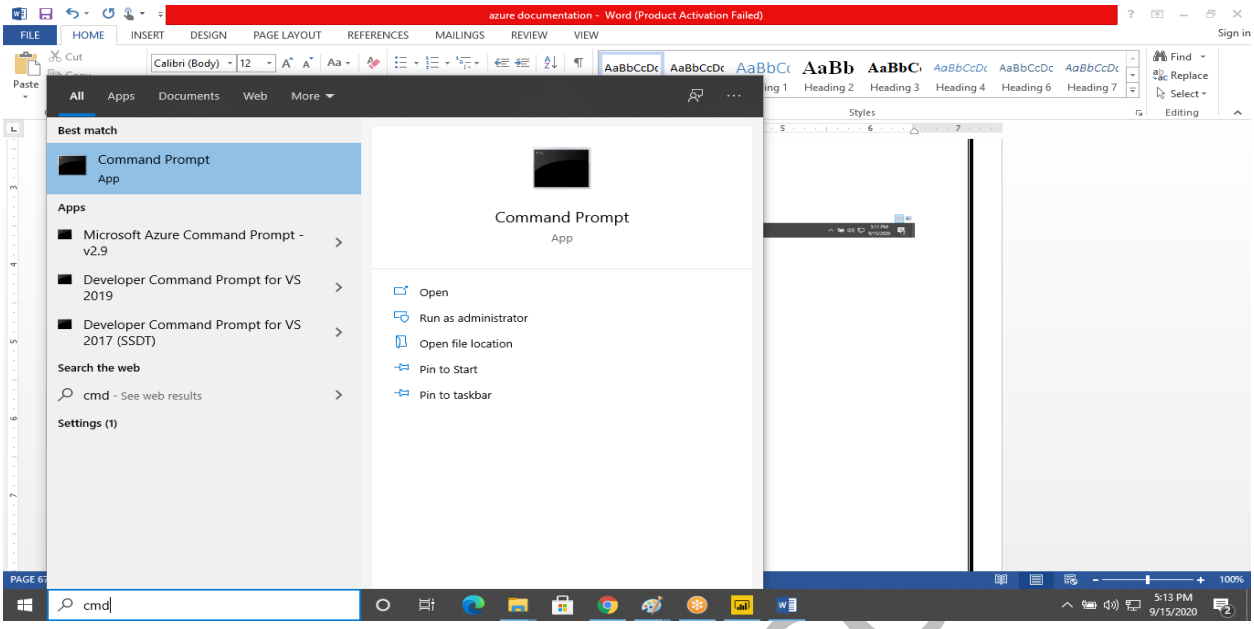
https://docs.microsoft.com/en-us/azure/storage/common/storage-use-azcopy-v10



Step 2 : Zip folder has downloaded and unzip folder and copy into C drive

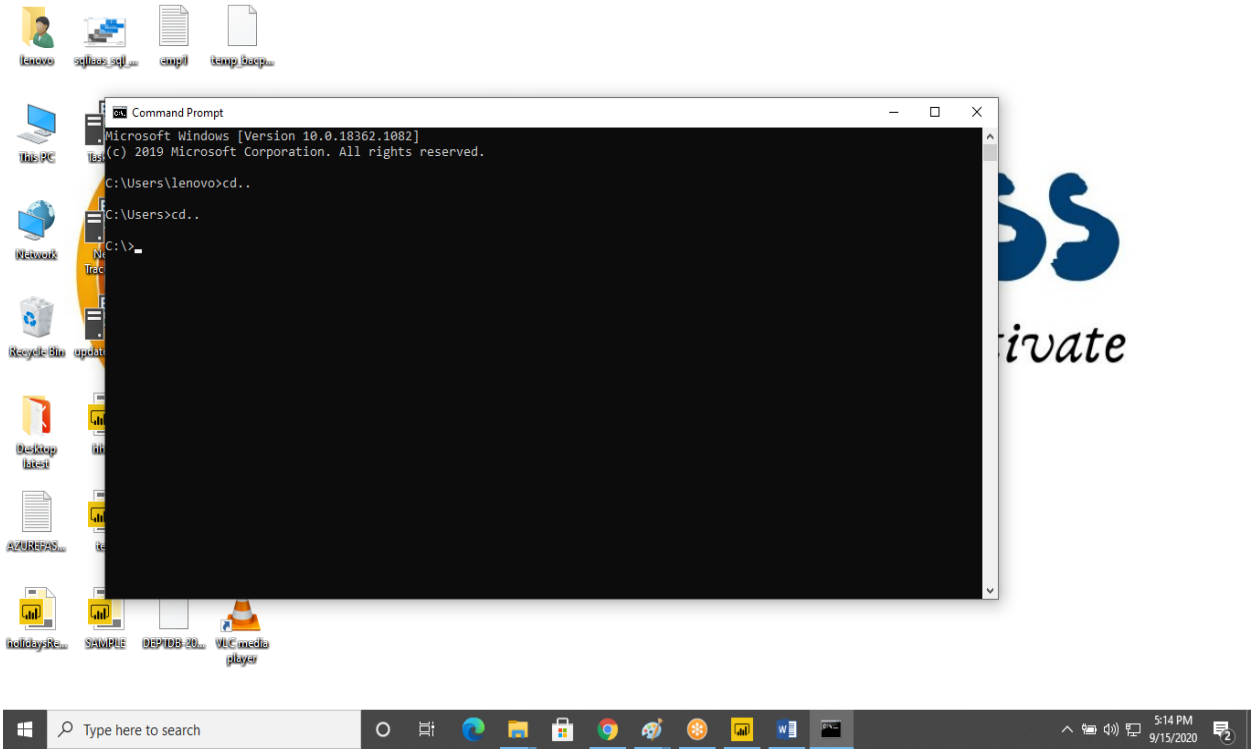


Step 3 : Open command prompt computer-Start-cmd



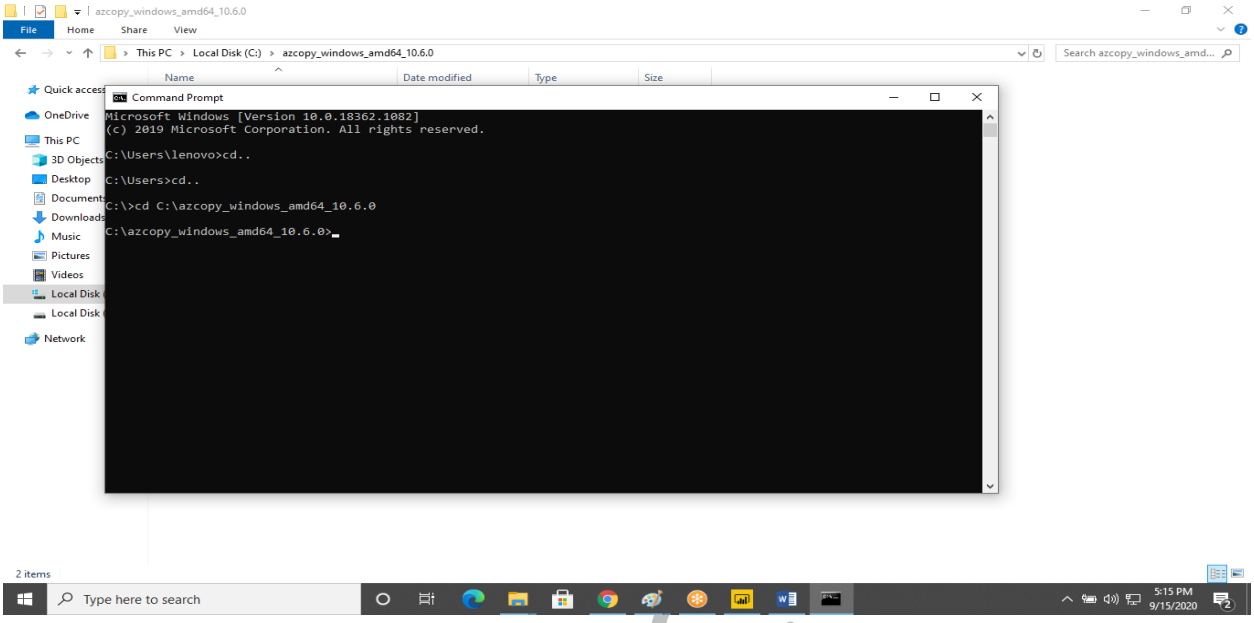
Step 4 : Change directory to C drive by typing below command in cmd prompt two times

cd..



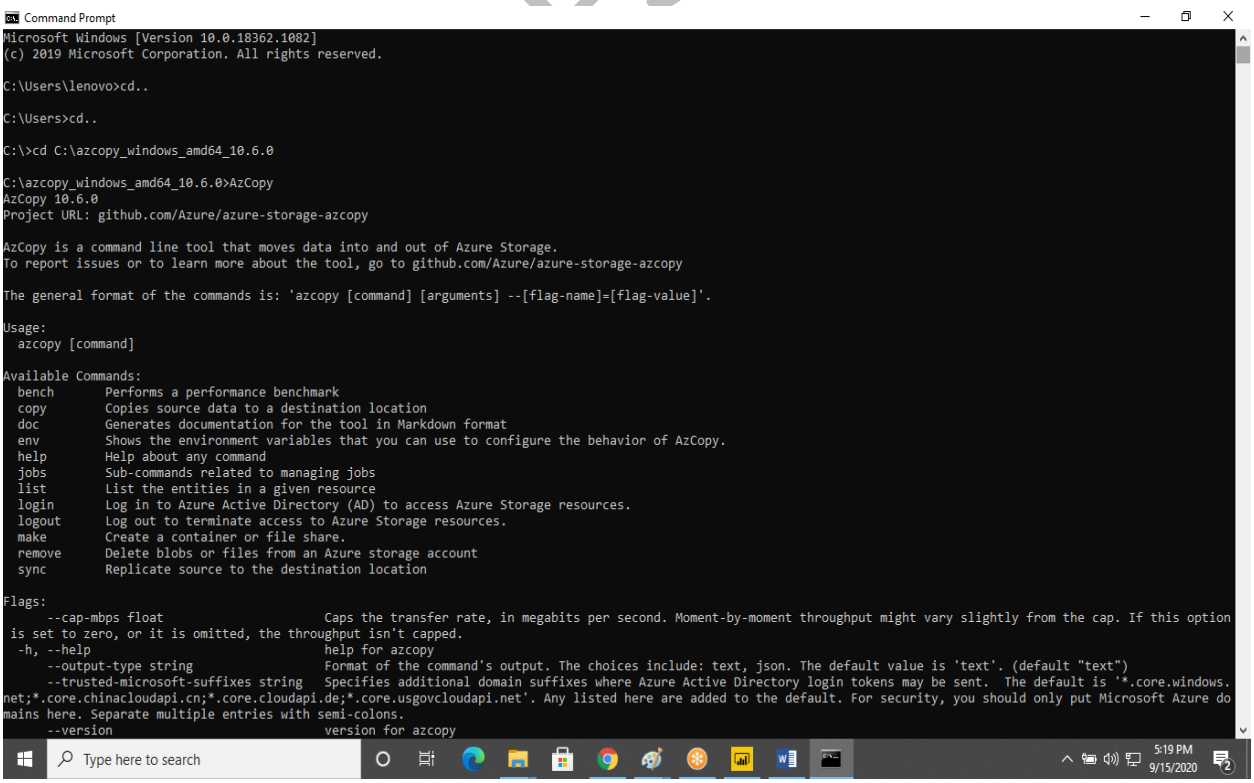
Step 5 : Change directory to AZ Copy file located in C Drive:

Cd C:\azcopy_windows_amd64_10.6.0



Step 6: Enter below command

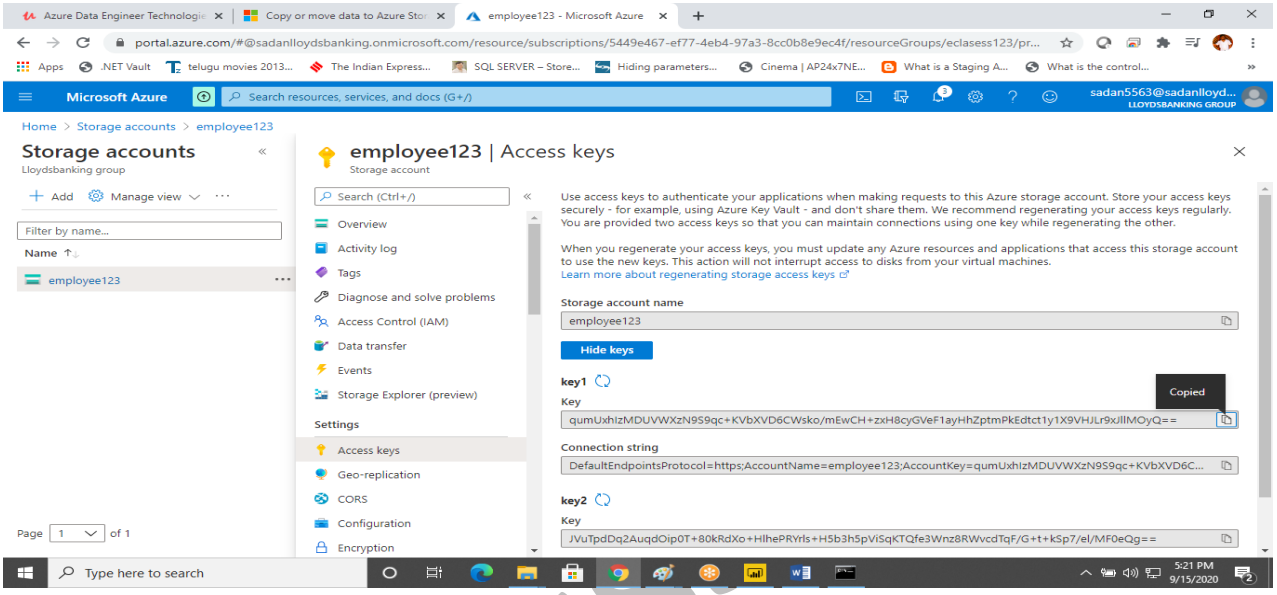
AZ Copy



Step 7: Take below information Access key and Azure Storage Account Name and path of Account.

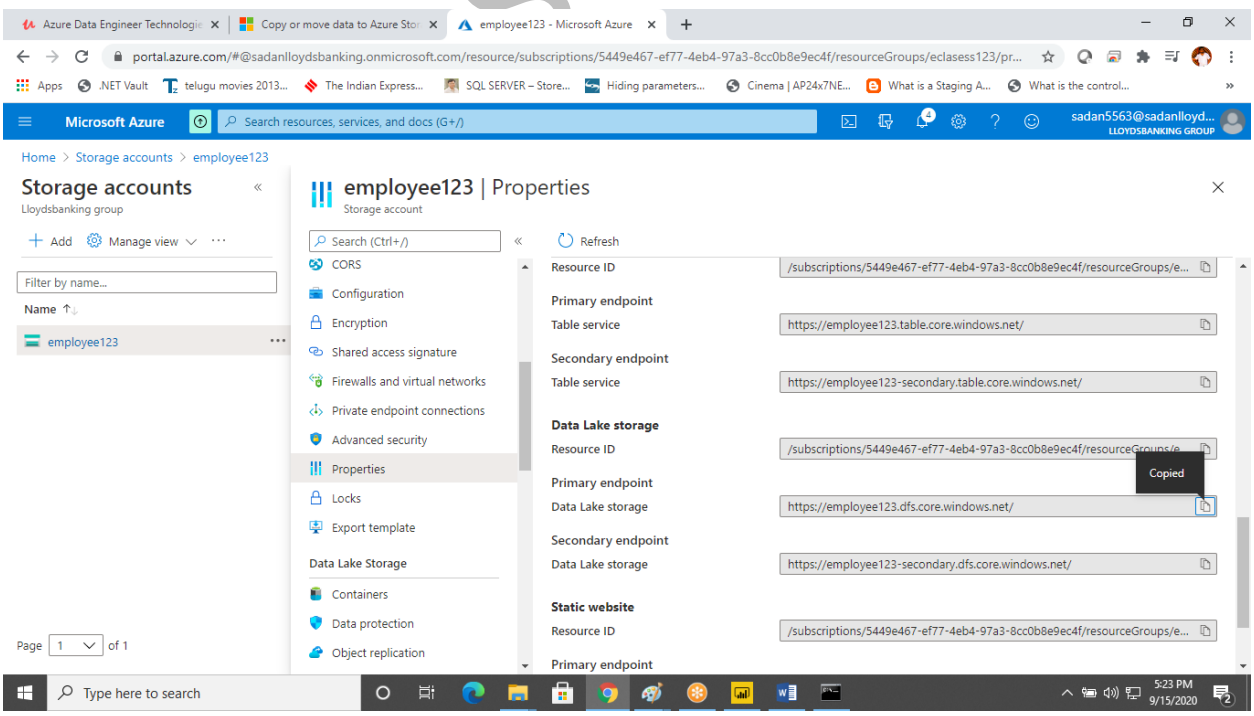
Access Key:

qumUxhIzMDUVWXzN9S9qc+KVbXVD6CWsko/mEwCH+zxH8cyGVeF1ayHhZptmPkEdtct1y1X9VHJLr9xJllMOyQ==



Step 8: Find endpoint path of Azure data Lake Gen2 Account under properties:

https://employee123.dfs.core.windows.net/

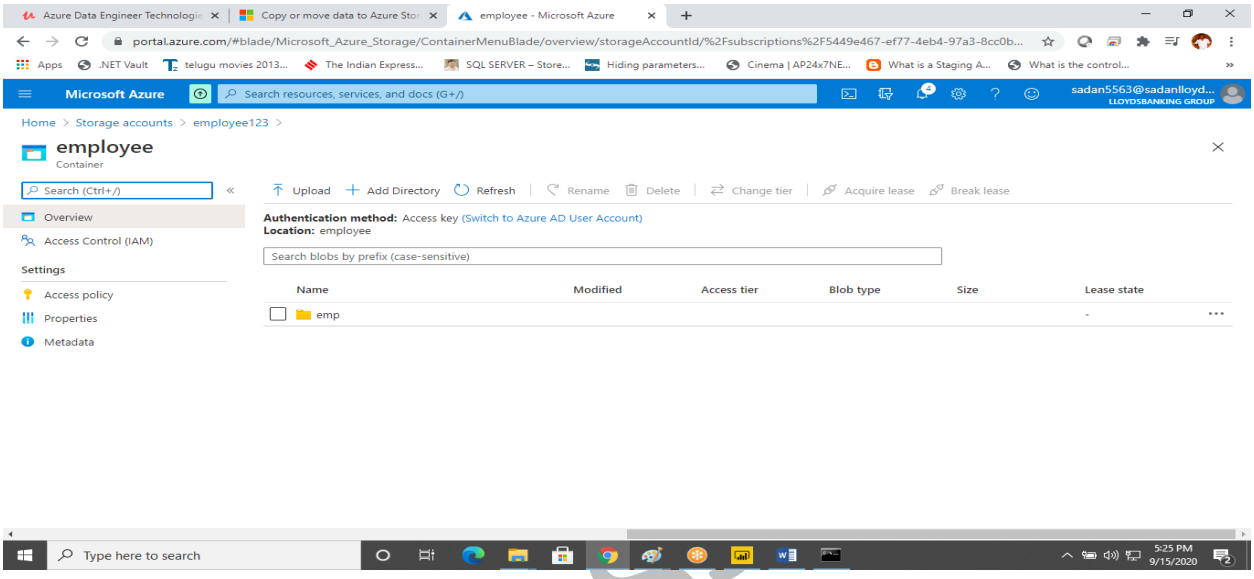


Step 9: Find Storage Account Name and Container Name and Folder Name to Load Data.

Account Name: employee123

Container Name: employee

Folder Name: EMP



Prepare below 3 commands to run in command prompt

1. //AzureStorageAccounName//

Set ACCOUNt_NAME=employee123

2. //Azure data Lake Storage Account Access key //

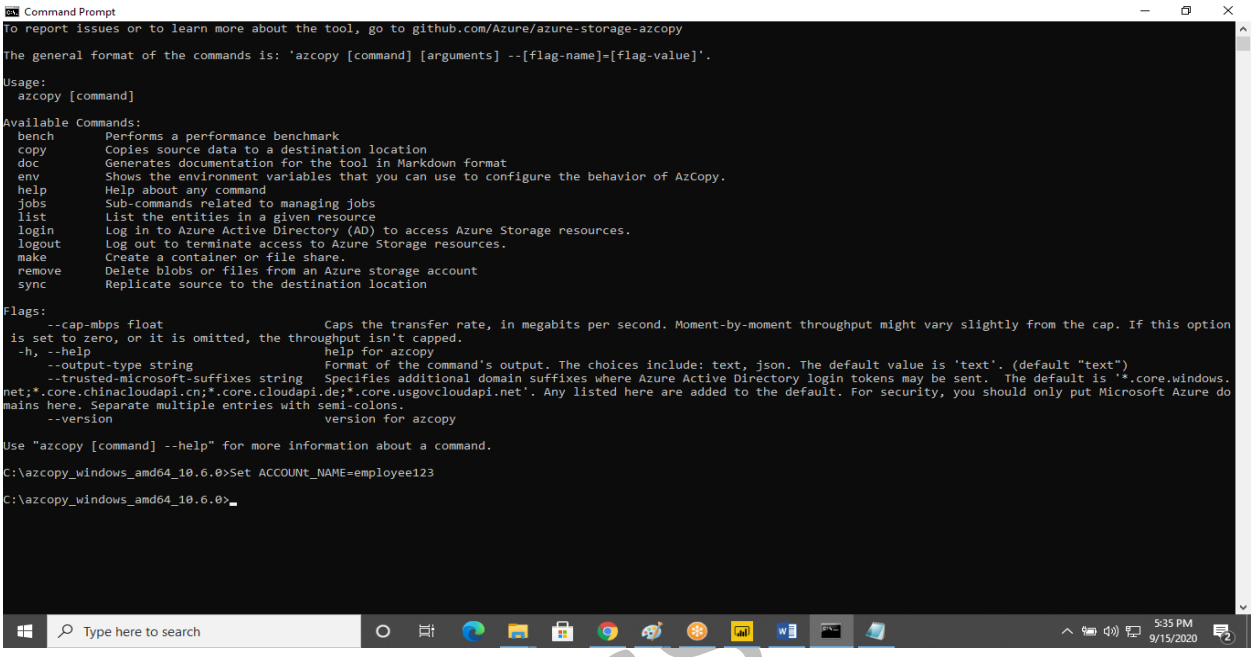
Set ACCOUNT_KEY=
qumUxhIzMDUVWXzN9S9qc+KVbXVD6CWsko/mEwCH+zxH8cyGVeF1ayHhZptmPkEdtct1y1X9V
HJLr9xJllMOyQ==

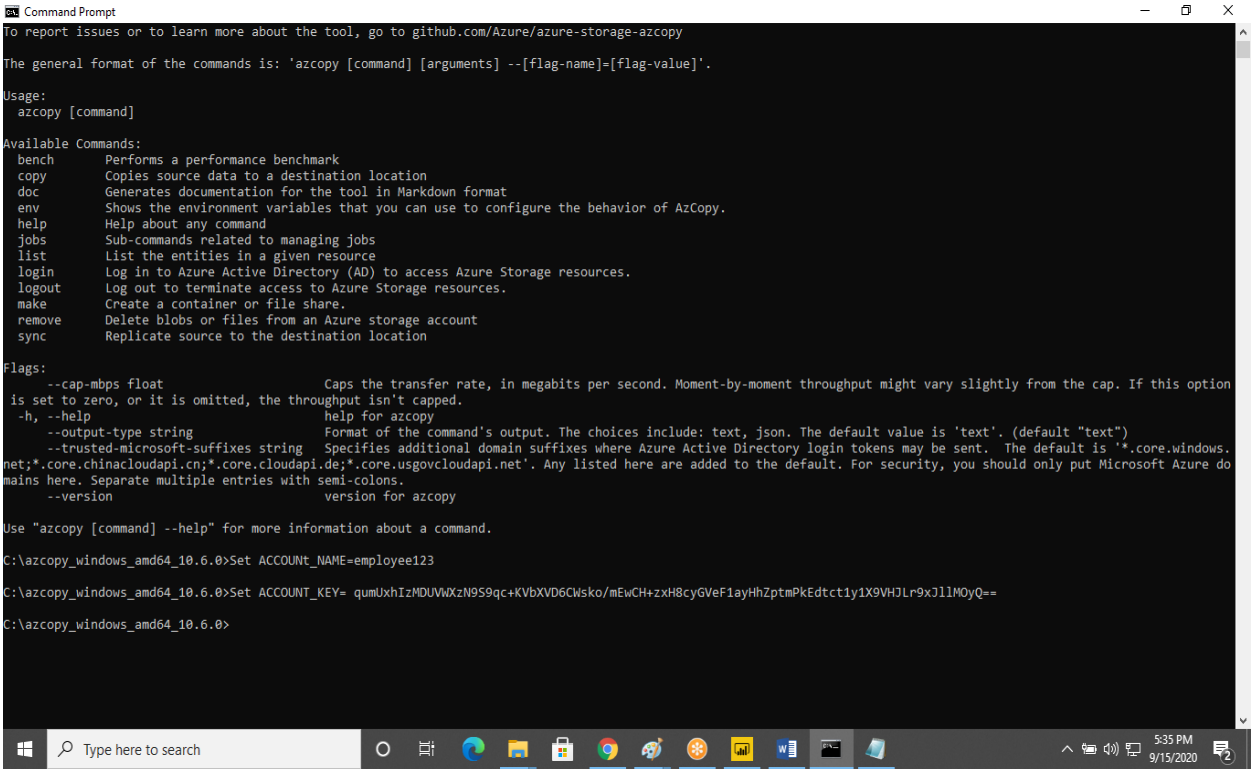
3. //Source files Folder path and destination data lake Account path includes containername and foldername//

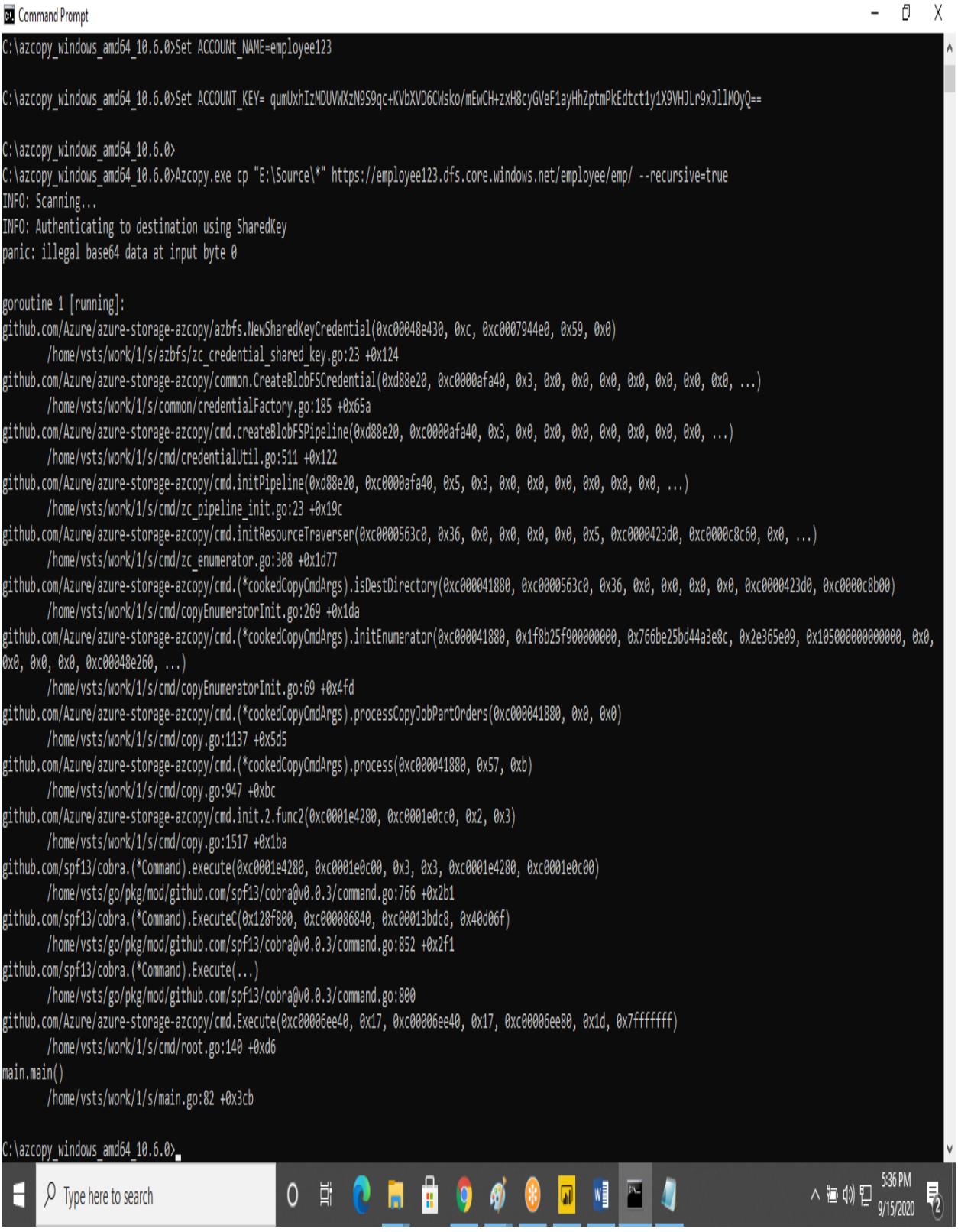
Azcopy.exe cp “E:\Source*” https://employee123.dfs.core.windows.net/employee/emp/ --recursive=true

ECLASESS

Step 10: Run above 3 commands in command prompt line.






Command Prompt
C:\azcopy_windows_amd64_10.6.0>Set ACCOUNt_NAME=employee123
C:\azcopy_windows_amd64_10.6.0>Set ACCOUNT_KEY= qumUxhIzMDUVWXzN9S9qc+KVbXVD6CWsko/mEwCH+zxH8cyGVeF1ayHhZptmPkEdtct1y1X9VHJLr9xJllMOyQ==
C:\azcopy_windows_amd64_10.6.0>
C:\azcopy_windows_amd64_10.6.0>Azcopy.exe cp "E:\Source*" https://employee123.dfs.core.windows.net/employee/emp/ --recursive=true
INFO: Scanning...
INFO: Authenticating to destination using SharedKey
panic: illegal base64 data at input byte 0
goroutine 1 [running]:
github.com/Azure/azure-storage-azcopy/azbfs.NewSharedKeyCredential(0xc00048e430, 0xc, 0xc0007944e0, 0x59, 0x0)
/home/vsts/work/1/s/azbfs/zc_credential_shared_key.go:23 +0x124
github.com/Azure/azure-storage-azcopy/common.CreateBlobFSCredential(0xd88e20, 0xc0000afa40, 0x3, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, ...)
/home/vsts/work/1/s/common/credentialFactory.go:185 +0x65a
github.com/Azure/azure-storage-azcopy/cmd.createBlobFSPipeline(0xd88e20, 0xc0000afa40, 0x3, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, ...)
/home/vsts/work/1/s/cmd/credentialUtil.go:511 +0x122
github.com/Azure/azure-storage-azcopy/cmd.initPipeline(0xd88e20, 0xc0000afa40, 0x5, 0x3, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, ...)
/home/vsts/work/1/s/cmd/zc_pipeline_init.go:23 +0x19c
github.com/Azure/azure-storage-azcopy/cmd.initResourceTraverser(0xc0000563c0, 0x36, 0x0, 0x0, 0x0, 0x0, 0x5, 0xc0000423d0, 0xc0000c8c60, 0x0, ...)
/home/vsts/work/1/s/cmd/zc_enumerator.go:308 +0x1d77
github.com/Azure/azure-storage-azcopy/cmd.(*cookedCopyCmdArgs).isDestDirectory(0xc000041880, 0xc0000563c0, 0x36, 0x0, 0x0, 0x0, 0x0, 0xc0000423d0, 0xc0000c8b00)
/home/vsts/work/1/s/cmd/copyEnumeratorInit.go:269 +0x1da
github.com/Azure/azure-storage-azcopy/cmd.(*cookedCopyCmdArgs).initEnumerator(0xc000041880, 0x1f8b25f900000000, 0x766be25bd44a3e8c, 0x2e365e09, 0x105000000000000, 0x0,
0x0, 0x0, 0x0, 0xc00048e260, ...)
/home/vsts/work/1/s/cmd/copyEnumeratorInit.go:69 +0x4fd
github.com/Azure/azure-storage-azcopy/cmd.(*cookedCopyCmdArgs).processCopyJobPartOrders(0xc000041880, 0x0, 0x0)
/home/vsts/work/1/s/cmd/copy.go:1137 +0x5d5
github.com/Azure/azure-storage-azcopy/cmd.(*cookedCopyCmdArgs).process(0xc000041880, 0x57, 0xb)
/home/vsts/work/1/s/cmd/copy.go:947 +0xbc
github.com/Azure/azure-storage-azcopy/cmd.init.2.func2(0xc0001e4280, 0xc0001e0cc0, 0x2, 0x3)
/home/vsts/work/1/s/cmd/copy.go:1517 +0x1ba
github.com/spf13/cobra.(*Command).execute(0xc0001e4280, 0xc0001e0c00, 0x3, 0x3, 0xc0001e4280, 0xc0001e0c00)
/home/vsts/go/pkg/mod/github.com/spf13/cobra@v0.0.3/command.go:766 +0x2b1
github.com/spf13/cobra.(*Command).ExecuteC(0x128f800, 0xc000086840, 0xc00013bdc8, 0x40d06f)
/home/vsts/go/pkg/mod/github.com/spf13/cobra@v0.0.3/command.go:852 +0x2f1
github.com/spf13/cobra.(*Command).Execute(...)
/home/vsts/go/pkg/mod/github.com/spf13/cobra@v0.0.3/command.go:800
github.com/Azure/azure-storage-azcopy/cmd.Execute(0xc00006ee40, 0x17, 0xc00006ee40, 0x17, 0xc00006ee80, 0x1d, 0x7fffffff)
/home/vsts/work/1/s/cmd/root.go:140 +0xd6
main.main()
/home/vsts/work/1/s/main.go:82 +0x3cb
C:\azcopy_windows_amd64_10.6.0>
Type here to search
5:36 PM
9/15/2020

Azure Data Factory

Azure Data factory is ETL Tool available in Azure cloud.

ETL stands for Extracting Data from different source systems and Transforming Data based on business requirement and loading into destination.

What are Major components of Azure Data Factory?

- Pipeline
- Activity
- Datasets
- Linked Services
- Integration Runtimes
- Triggers

1. **Pipeline.**

   A data factory can have one or more pipelines. A pipeline is a logical grouping of activities that together perform a task like copy data from source and apply transformations, and Load data into destination.

2. **Activities**

   Activities are defined to perform actions inside pipeline. There are several activities available, based on requirement we have to choose which activity needs to use to perform action.

   Append Variable: Append Variable activity could be used to add a value to an existing array variable defined in a Data Factory pipeline.

   Set Variable: Set Variable activity can be used to set the value of an existing variable of type String, Bool, or Array defined in a Data Factory pipeline.

   Execute Pipeline: The Execute Pipeline activity allows a Data Factory pipeline to invoke another pipeline.

   If Condition: If Condition activity allows directing pipeline execution, based on evaluation of certain expressions.

   Get Metadata: Get Metadata activity can be used to retrieve metadata of any data in Azure Data Factory.

   For Each: The For each activity defines a repeating control flow in your pipeline.

ECLASESS.COM

Lookup: Lookup activity can retrieve a dataset from any of the Azure Data Factory supported data sources.

Filter: Filter activity can be used in a pipeline to apply a filter expression to an input array.

Until: Until activity executes a set of activities in a loop until the condition associated with the activity evaluates to true.

Wait: Wait activity allows pausing pipeline execution for specified time period.

Web: Web activity can be used to call a custom REST endpoint from a Data Factory pipeline.

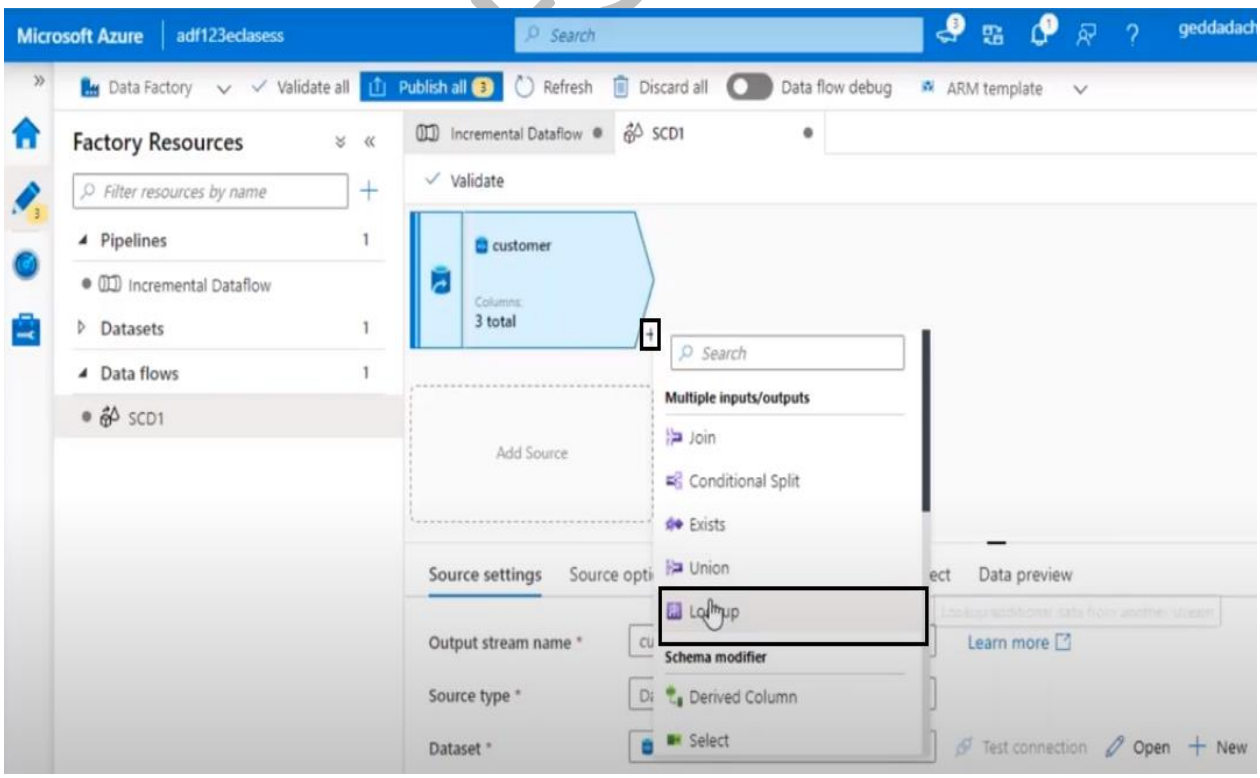
Azure Function: The Azure Function activity allows you to run Azure Functions in a Data Factory pipeline.

Data Flows:

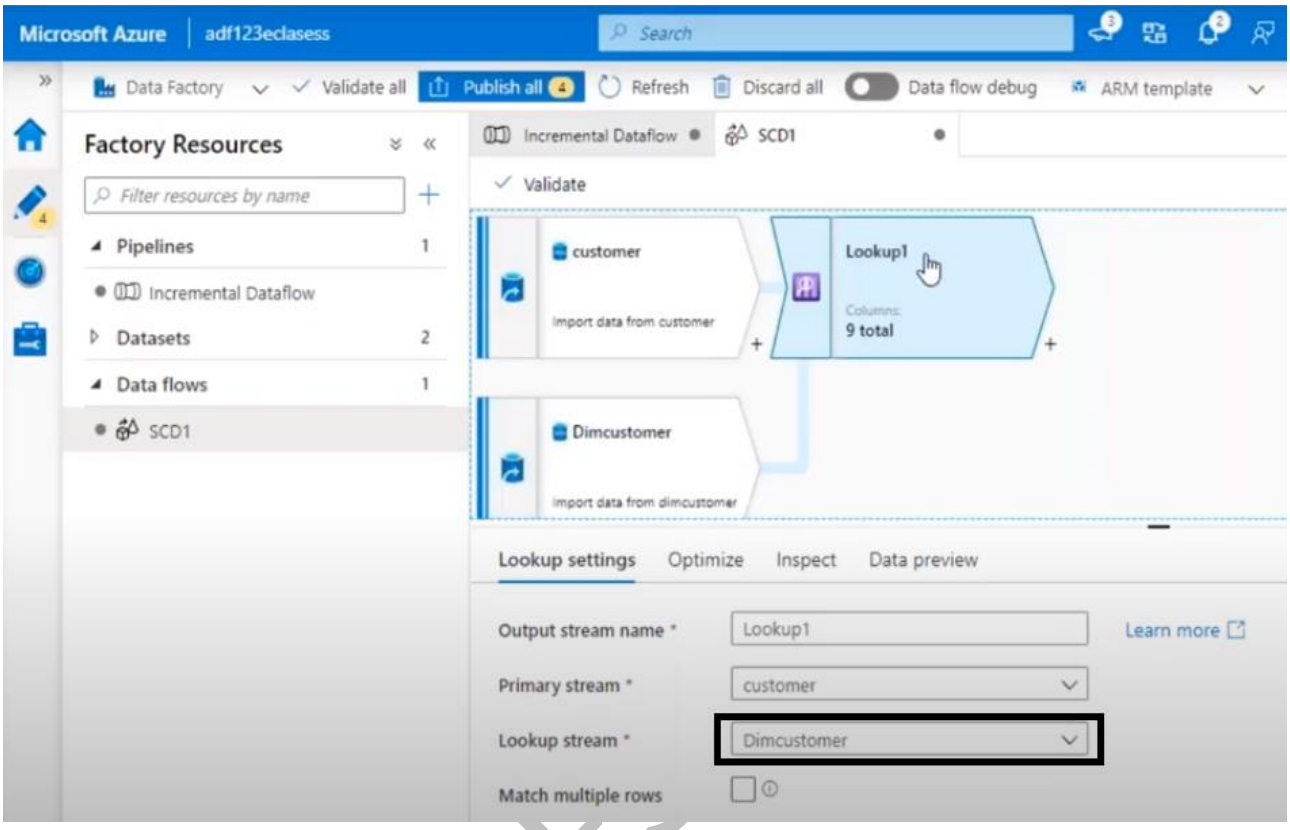
Data Flow is a new feature of Azure Data Factory (ADF) that allows you to develop graphical data transformation. It contains several transformations to perform ETL process.

- **Lookup:** Comparing two Datasets based on condition by matching common key column and returning matched and unmatched records.

Step A: Click on '+' symbol and Click on lookup transformation.

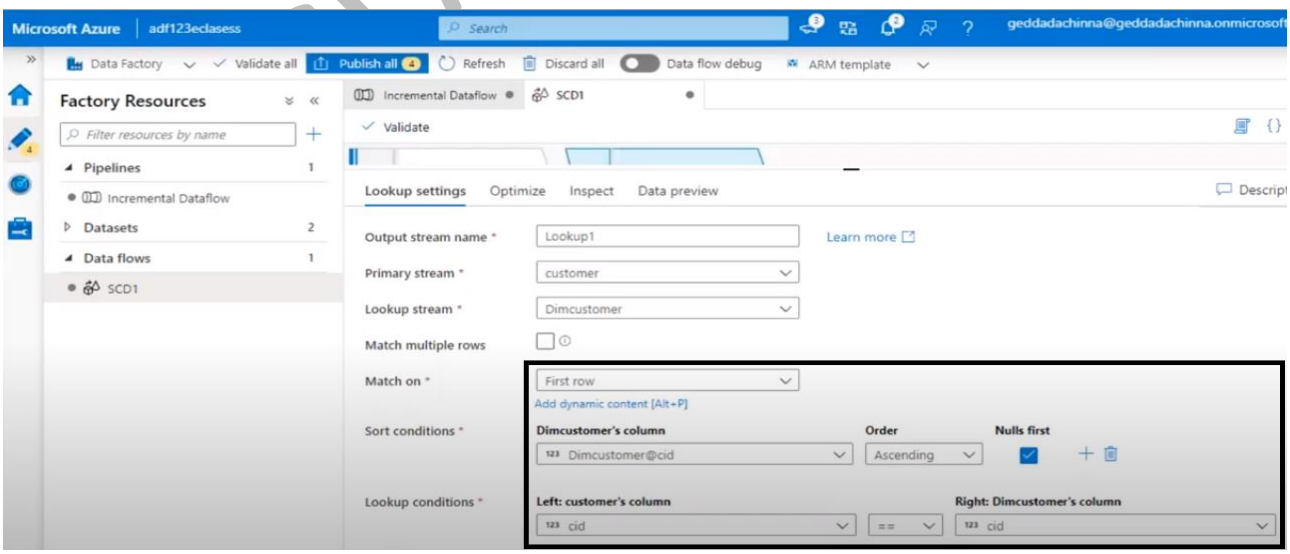


Step B: Click on Lookup Stream----Select Lookup stream as dim customer



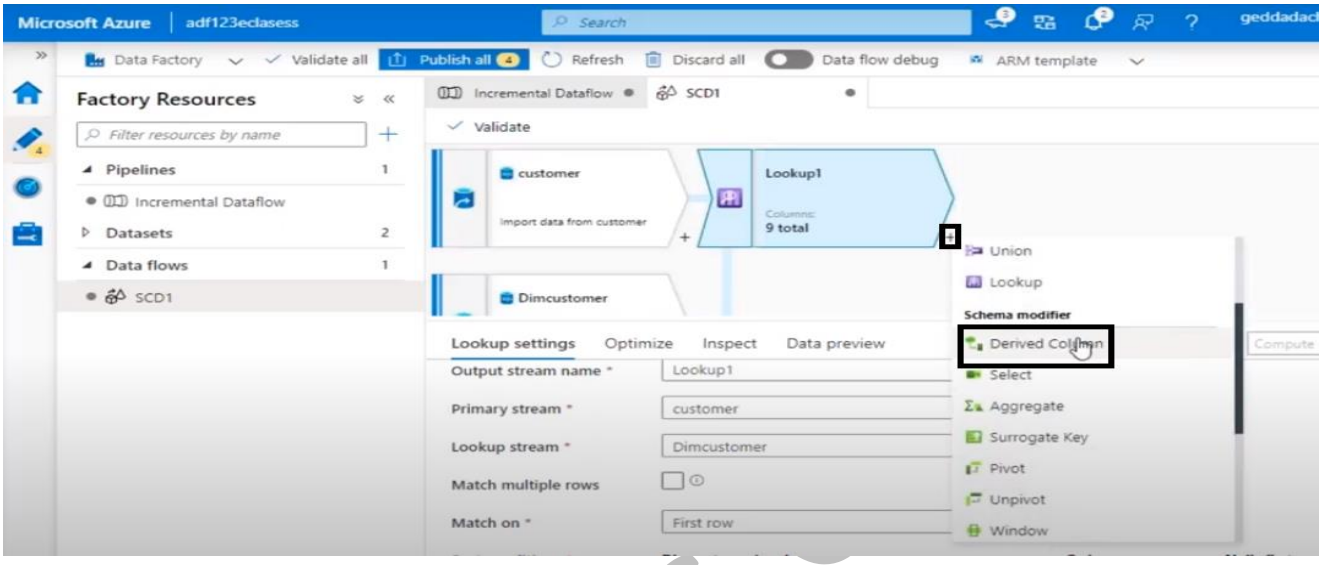
Step C: Go to Lookup conditions---- Select cid column from the customer table on the left side----Select cid column from the dimcustomer column on the right side.

Go to Match on---Select First row-----Go to Sort conditions----Select Dimcustomer@cid----Select Order as Ascending

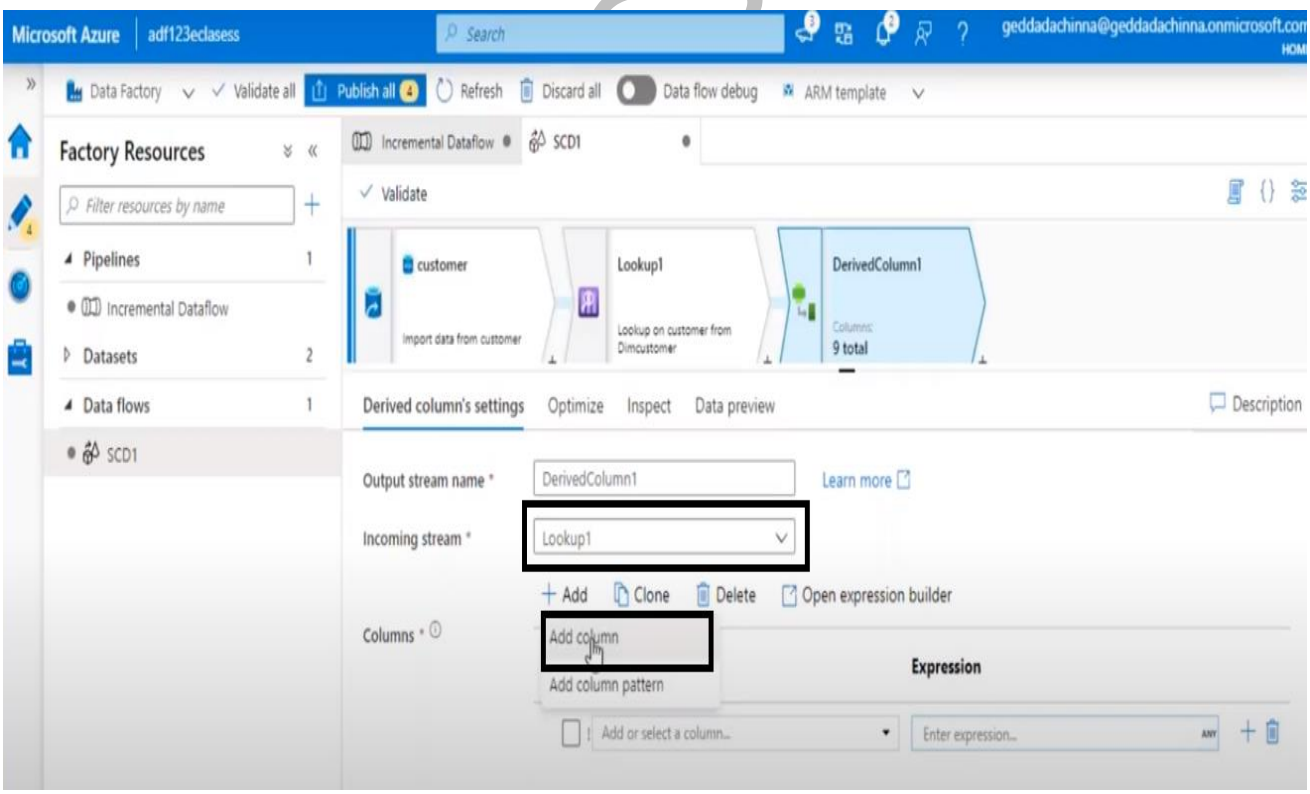


Derived Column: If you want to add new column or replace existing column by deriving new expression inside derived column Transformation.

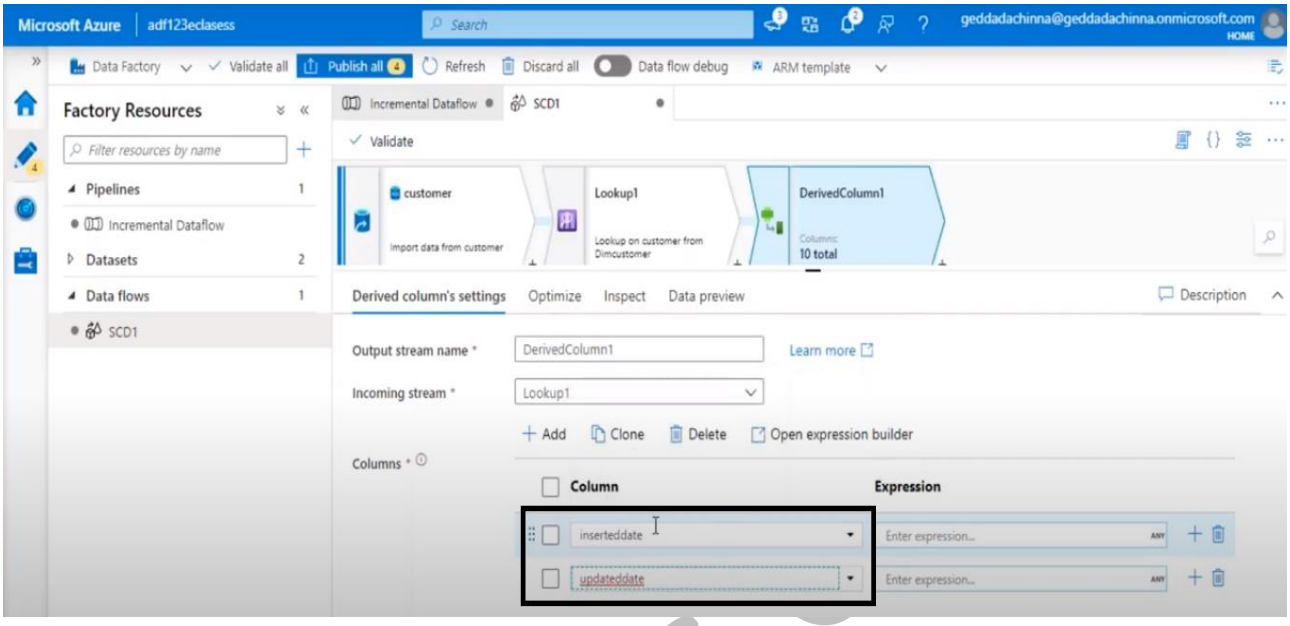
Step A: Click on '+' symbol---Click on Derived Column



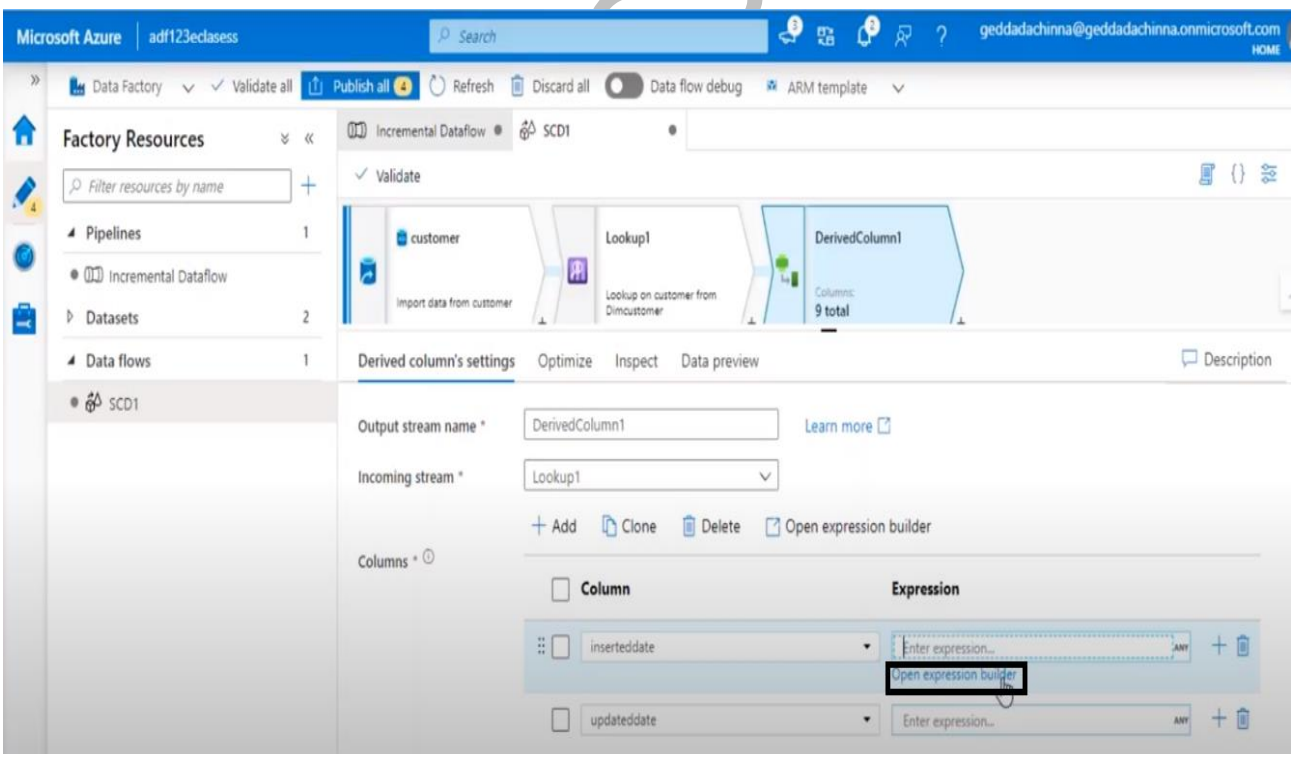
Step B: Select Incoming stream---Go to Columns----Click on Add Column



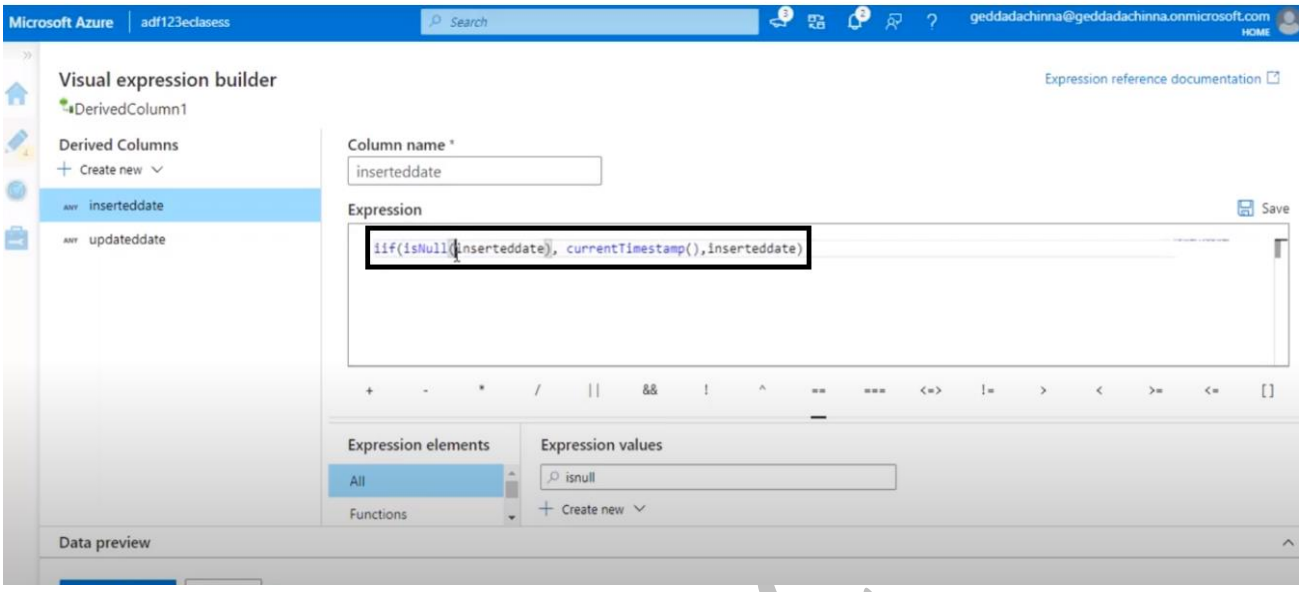
Step C: Select inserteddate in the first row and updateddate in the second row (inserteddate and updateddate are present in the dimcustomer table)



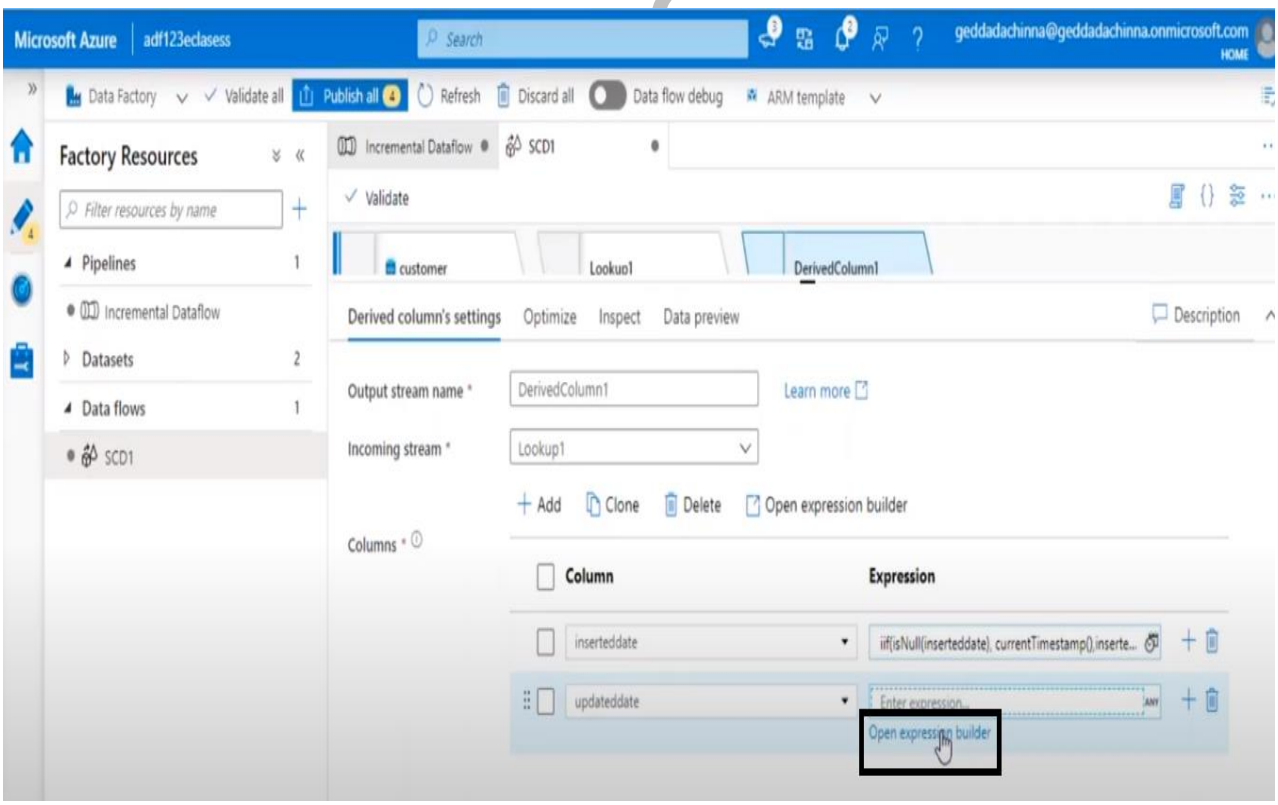
Step D: Click on Open Expression Builder of the inserteddate



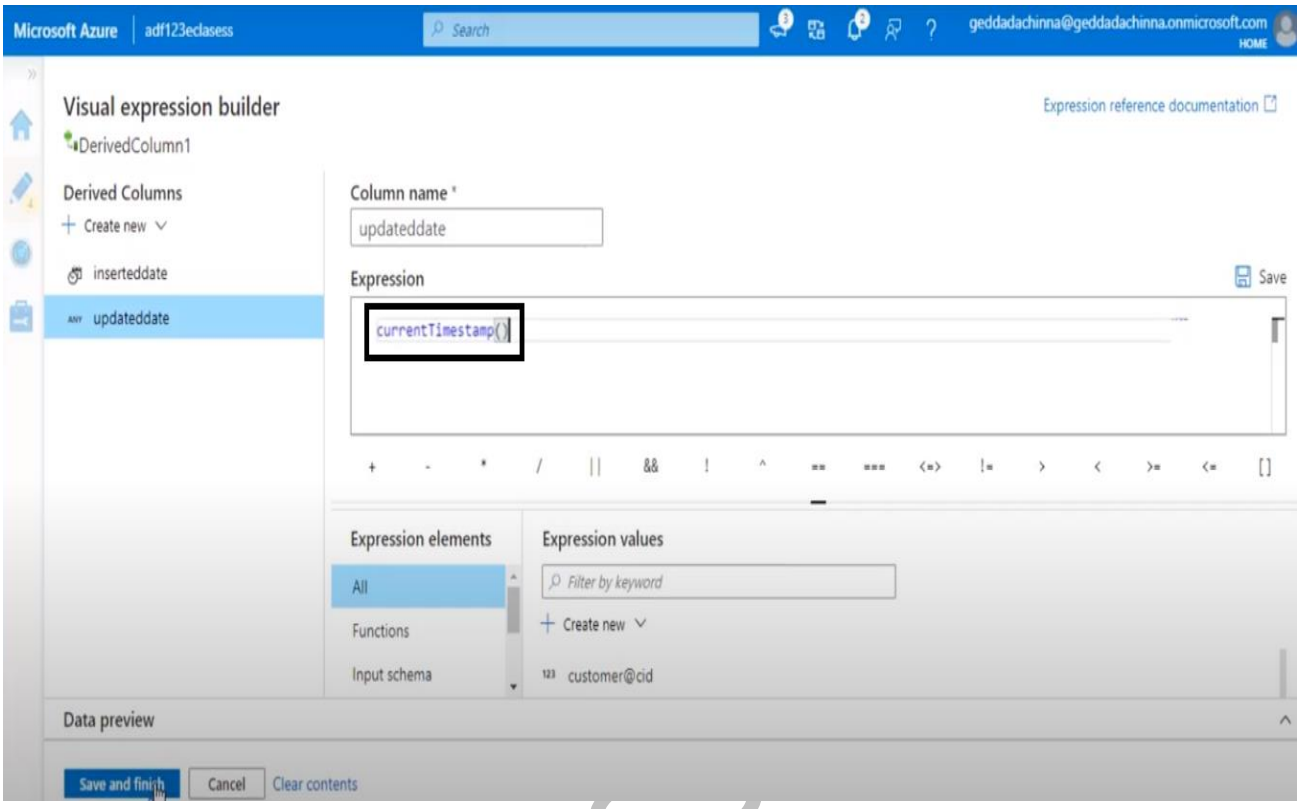
Step E: Write an Expression----Click on Save and Finish



Step F: Click on open expression builder of the updateddate



Step G: Write an expression---Click on Save and Finish



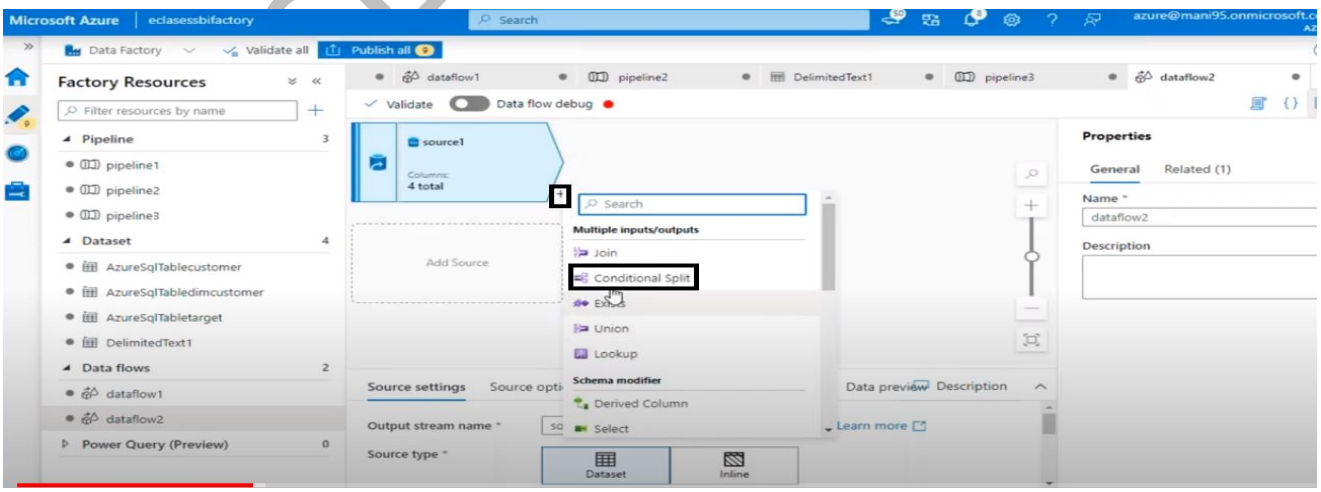
Conditional Split: Splitting input Data Based on Conditions

Ex: All Hyd Data Load into One destination.

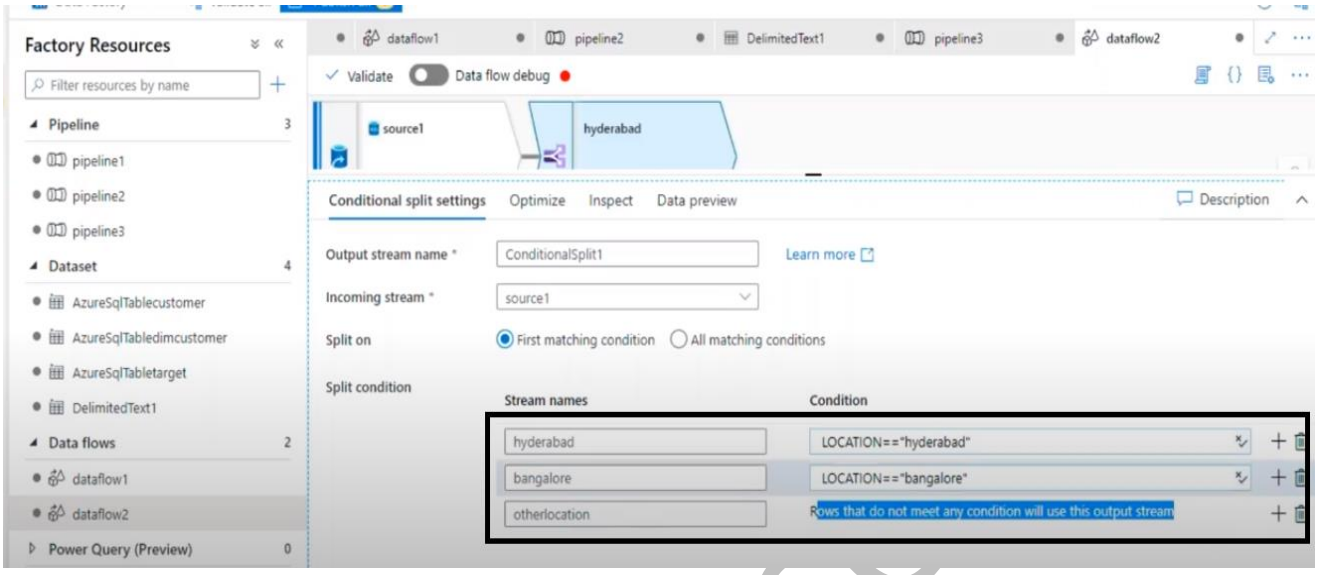
All Bang Data Load into another Destination.

Other Locations data needs to load into different destination.

Step A: Click on '+' symbol----Select Conditional Split

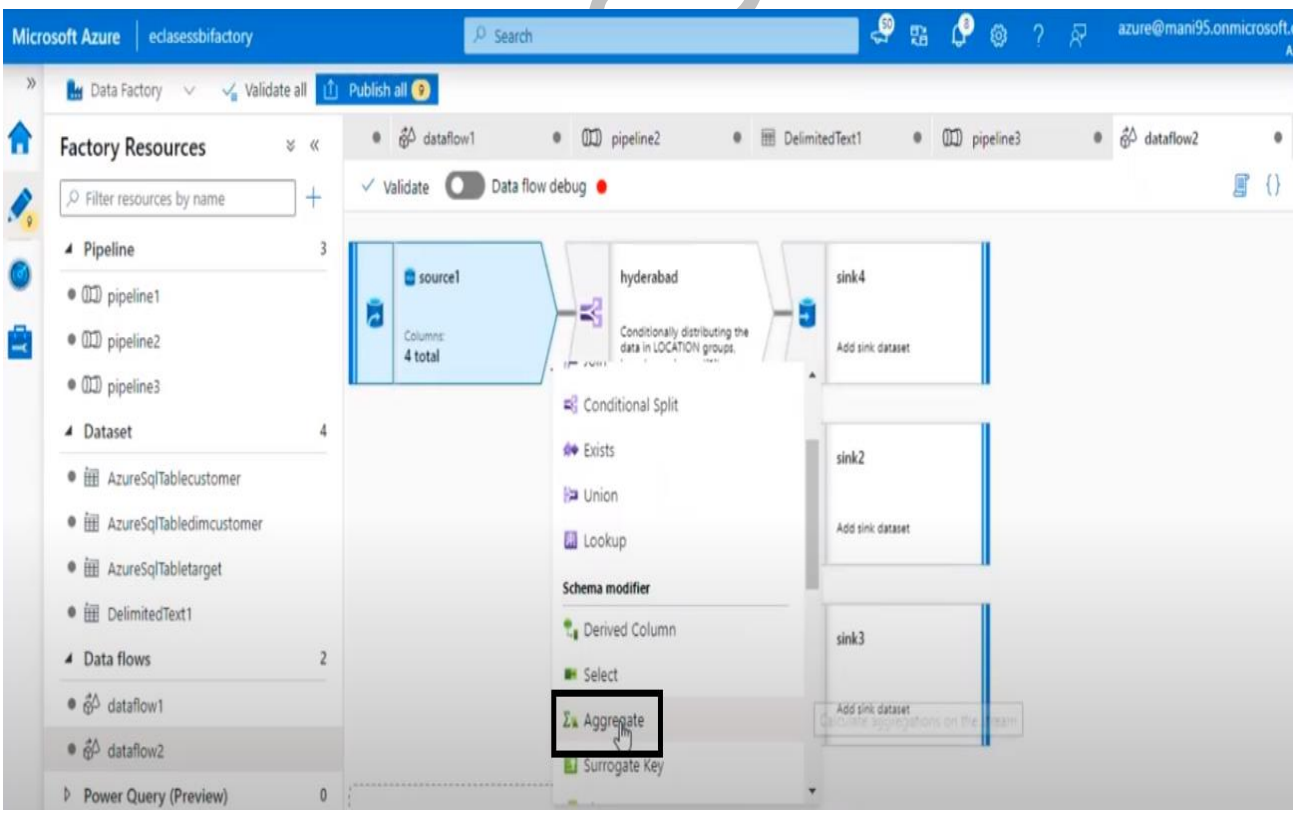


Step B: Write conditions for splitting the data as per locations

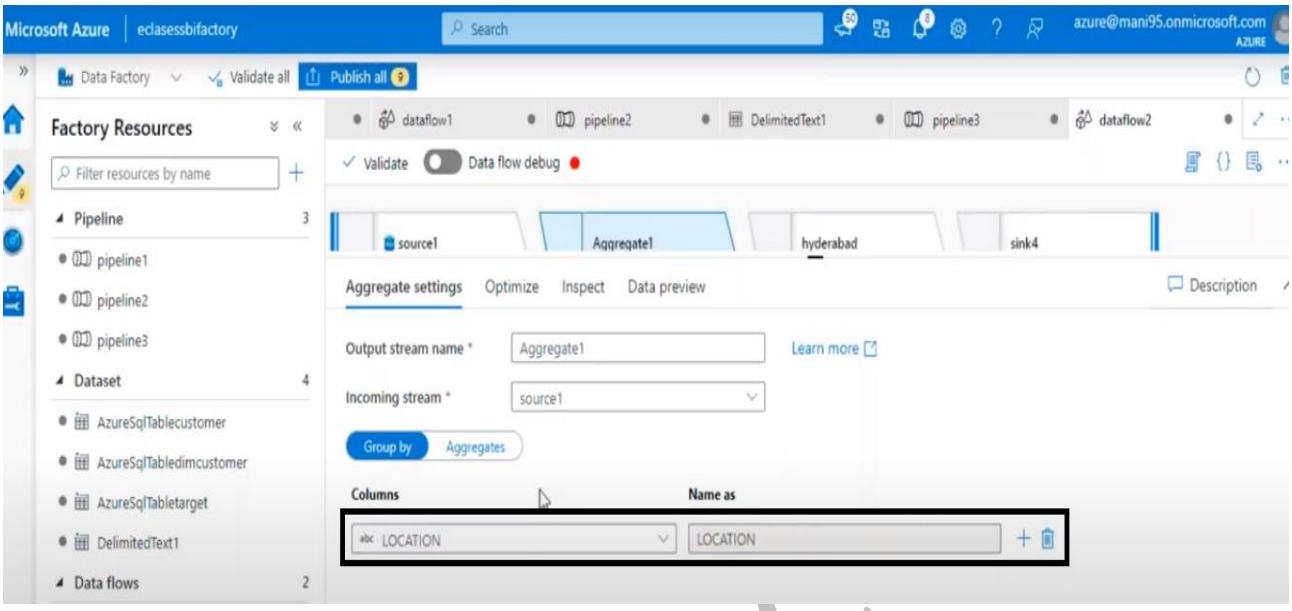


Aggregate: If you want to perform any aggregate operations like SUM, MIN, MAX, COUNT, AVG by grouping on input data.

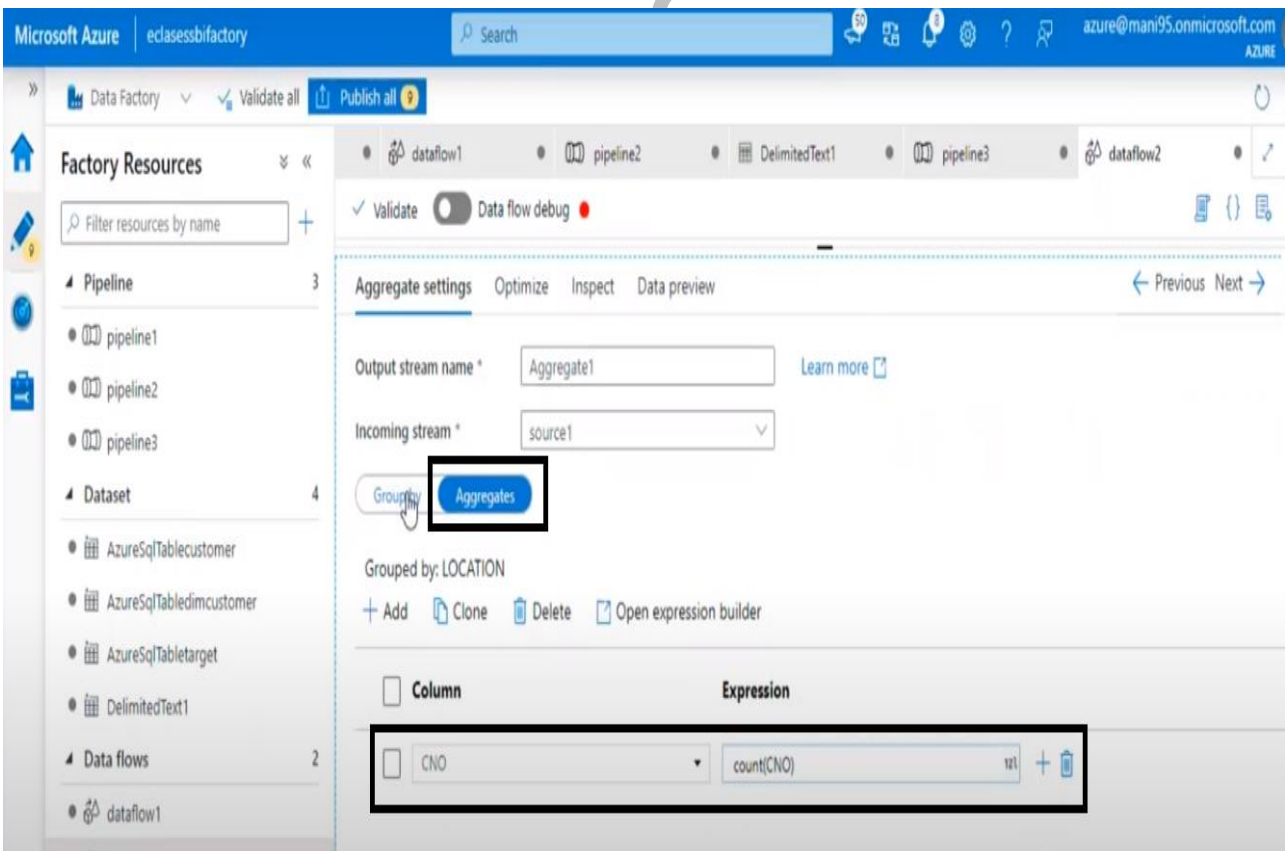
Step A: Click on '+' symbol-----Select Aggregate



Step B: Select the grouping column

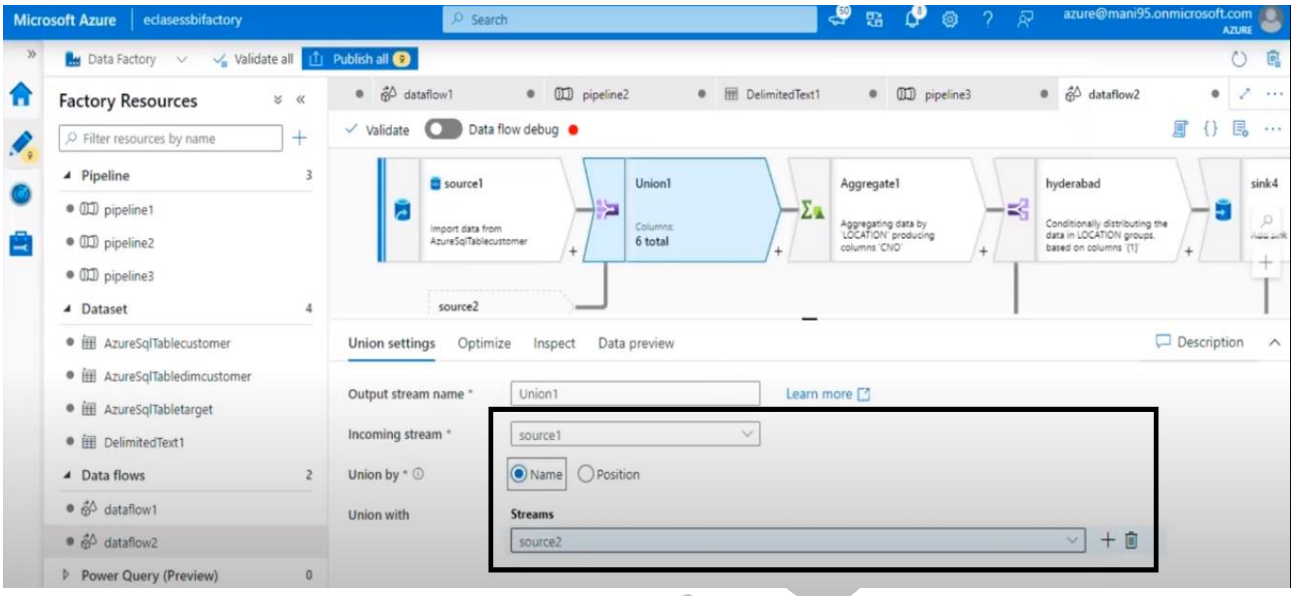


Step C: Click on Aggregates----write expression



Union: If want to append or combining more than two datasets data into one dataset.

For example: Combining Source1 and Source2 datasets



Join : Merging or joining two tables by using common key columns by performing different kind of joins

Inner join

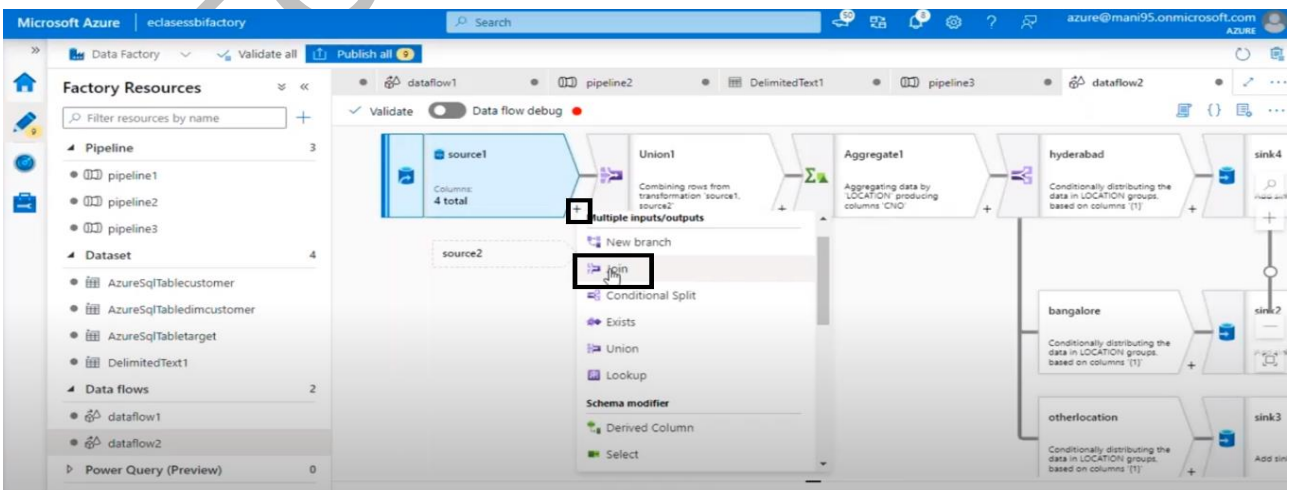
Left outer Join

Right Outer Join

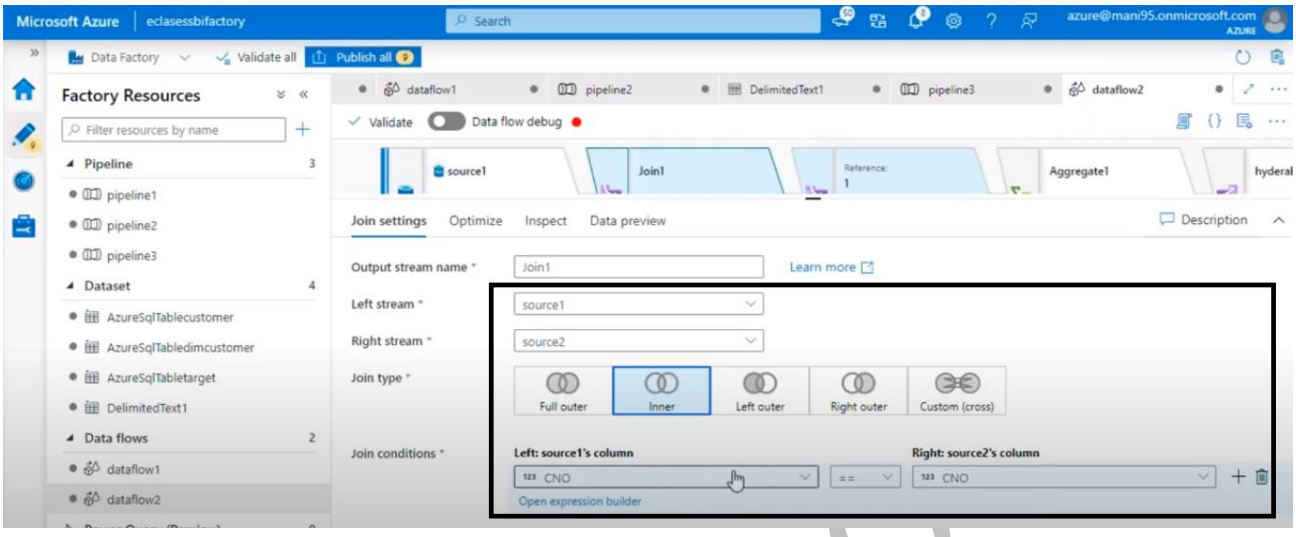
Full Outer join

Custom Cross Join

Step A: Click on '+' symbol-----Select Join



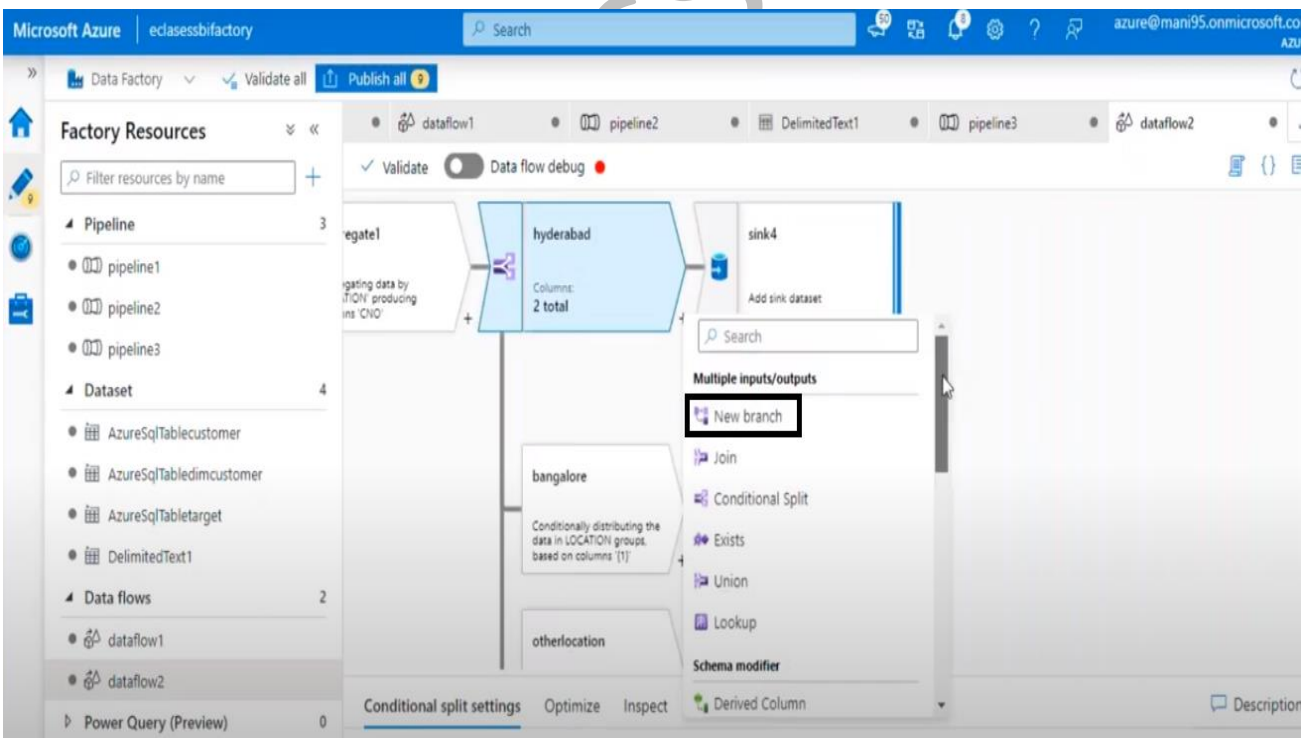
Step B: Select Left stream & Right stream-----Select join type-----Select Common key columns



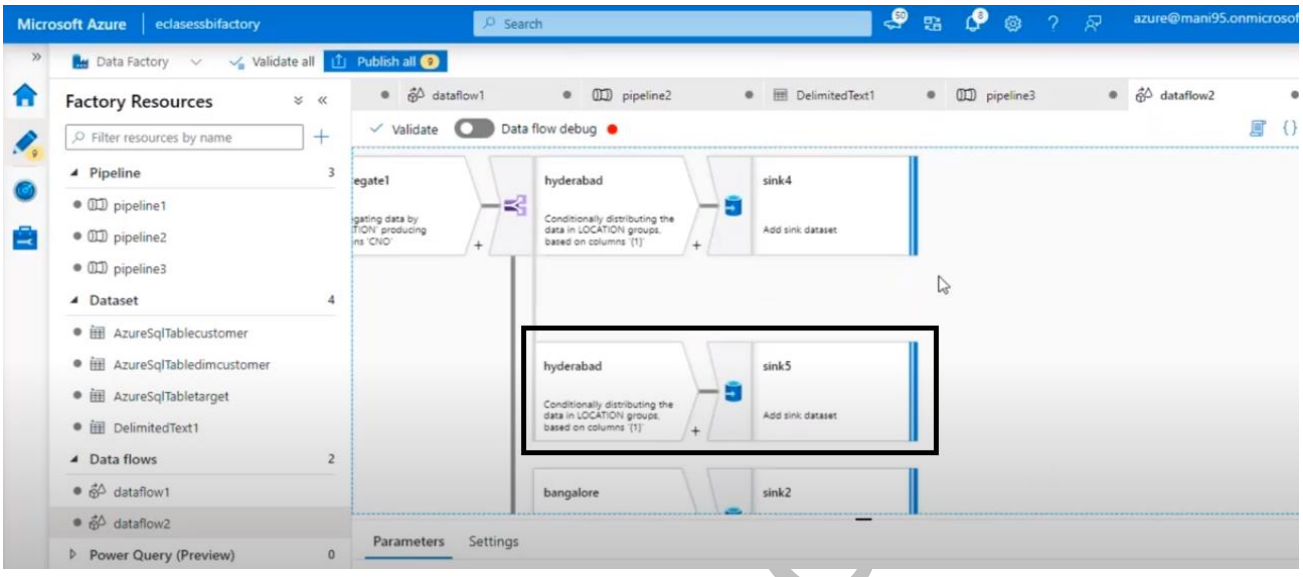
New branch: It will take one input and giving multiple output with same set of data.

Ex: When same data wants to Load into Multiple destinations.

Step A: Click on '+' symbol----Select New Branch



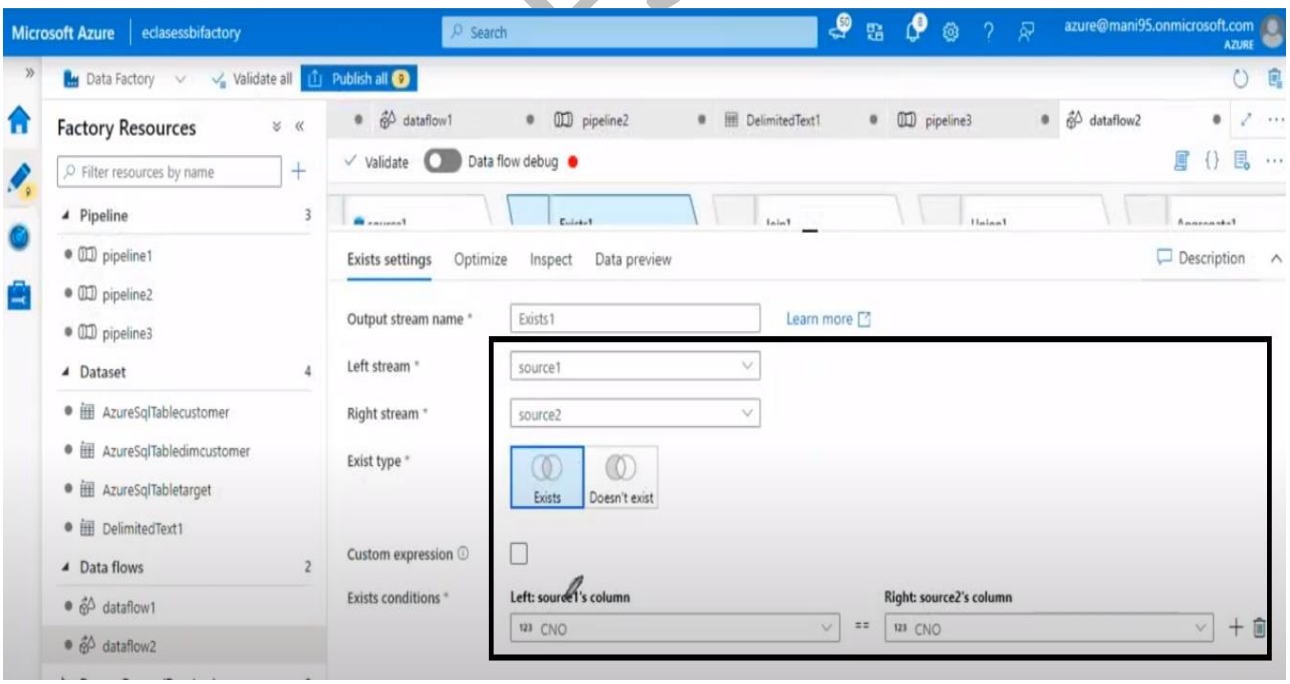
Step B: Add destination for hyderabad data



Exist :

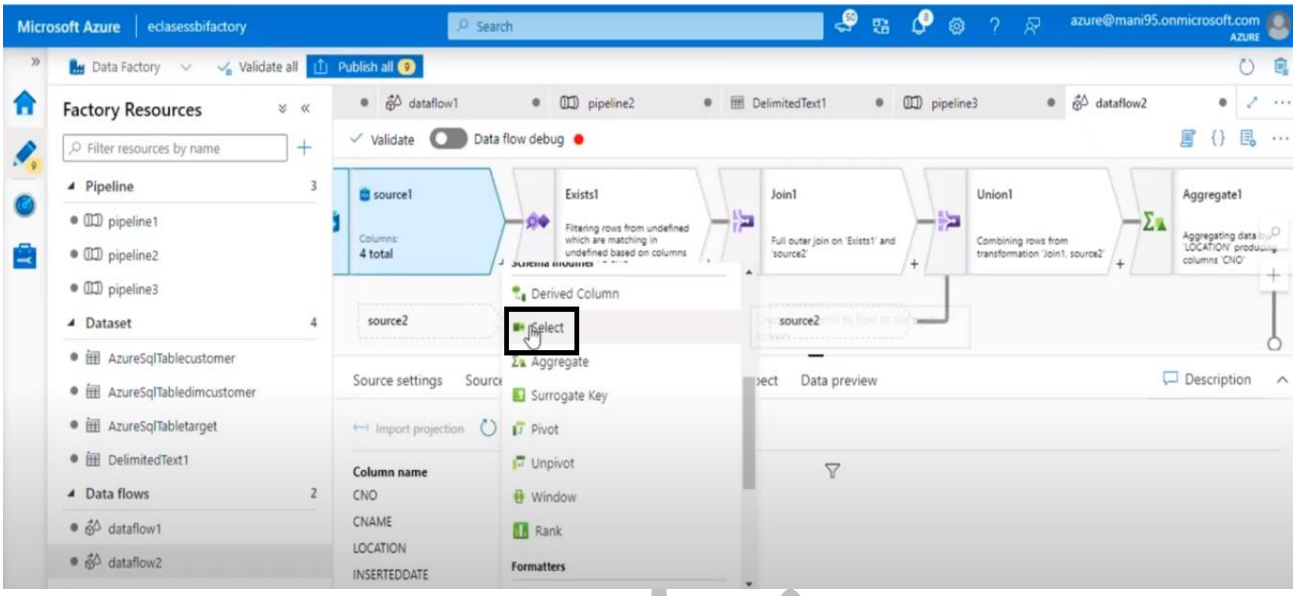
It is identifying what are all records from source1 available in source2

Doesn't exist: It is identifying what are all records from source1 doesn't exist in source2

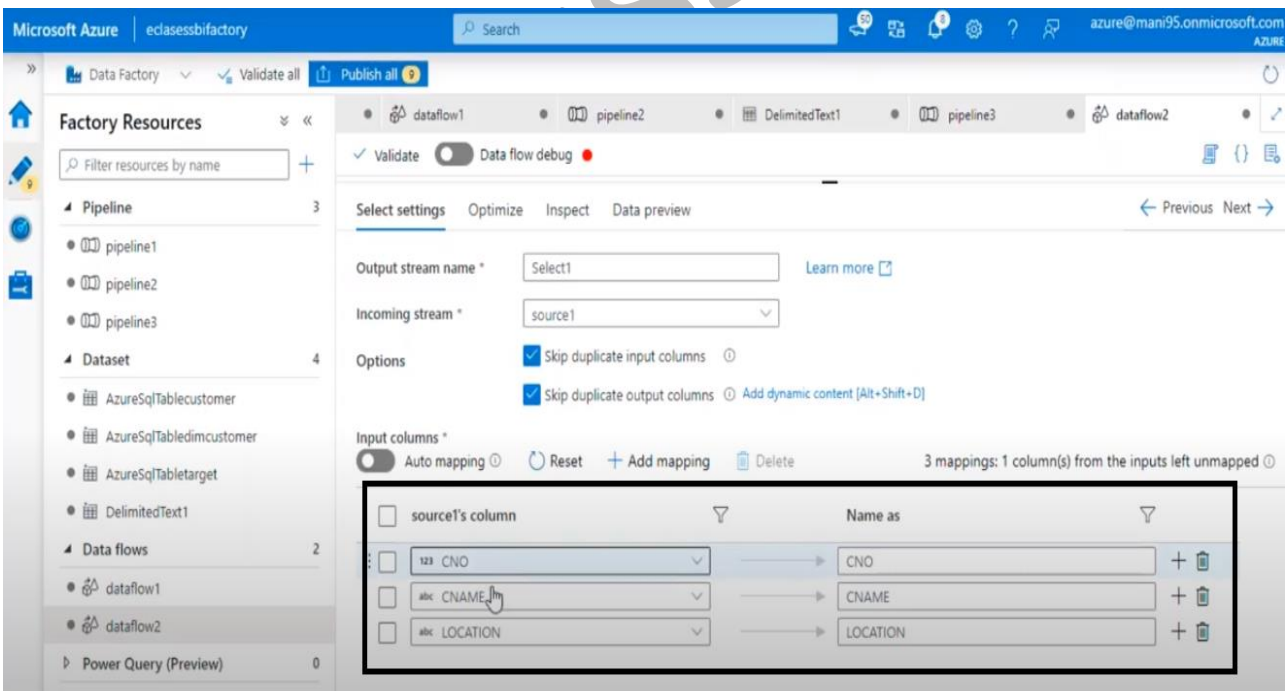


Select: This Transformation can use to skip columns and keeping whatever column you want to pass to next transformation.

Step A: Click on '+' symbol-----Click on Select



Step B: Select Columns which want to pass to next transformation



Alter Row: It is marking each row based on operation which records needs to be

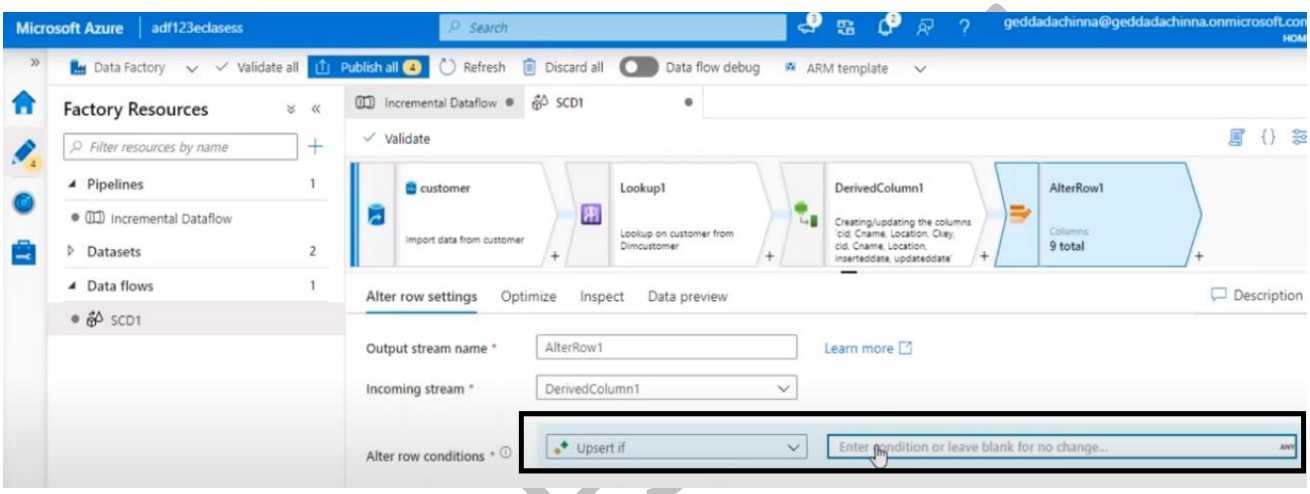
Inserted

Updated

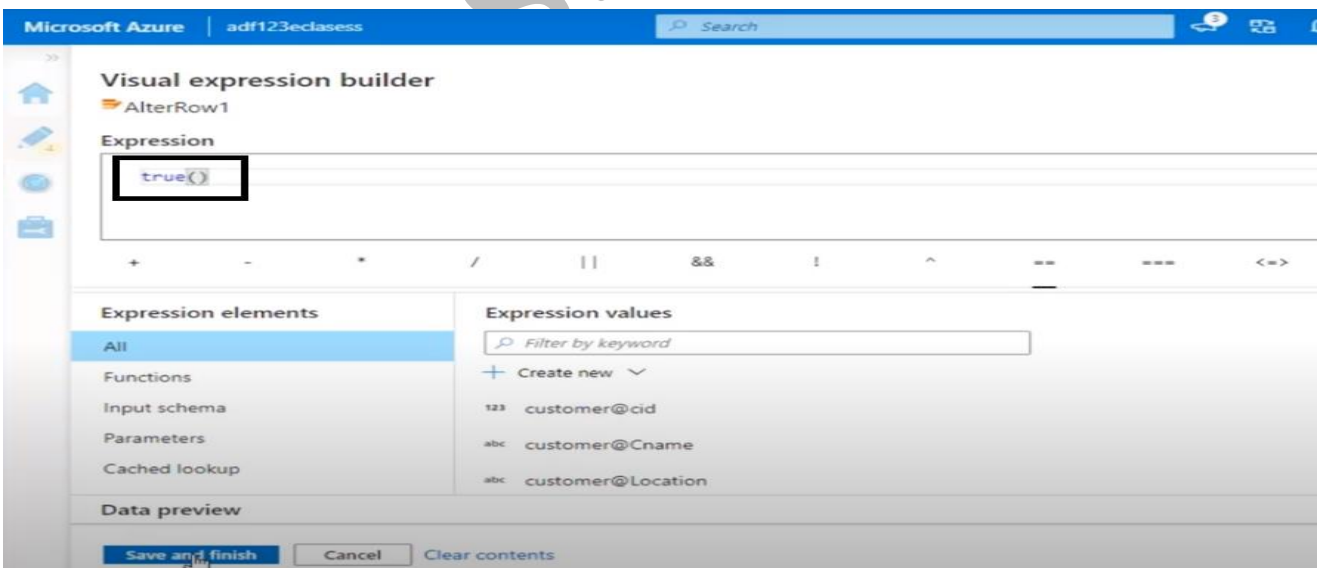
Deleted

Upsert is combination of Insert + Update

Step A: Go to Alter row conditions in Alter row transformation----Select upsertif----Click on Enter Condition

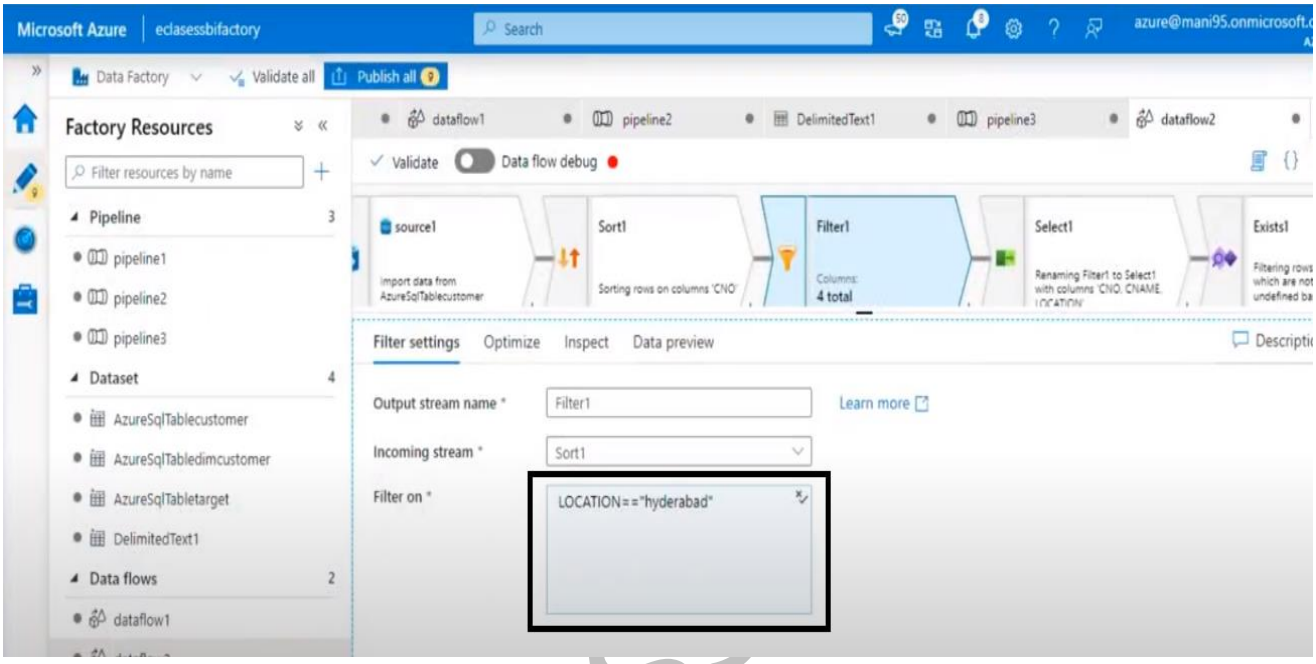


Step B: Write an expression----Click on Save and Finish



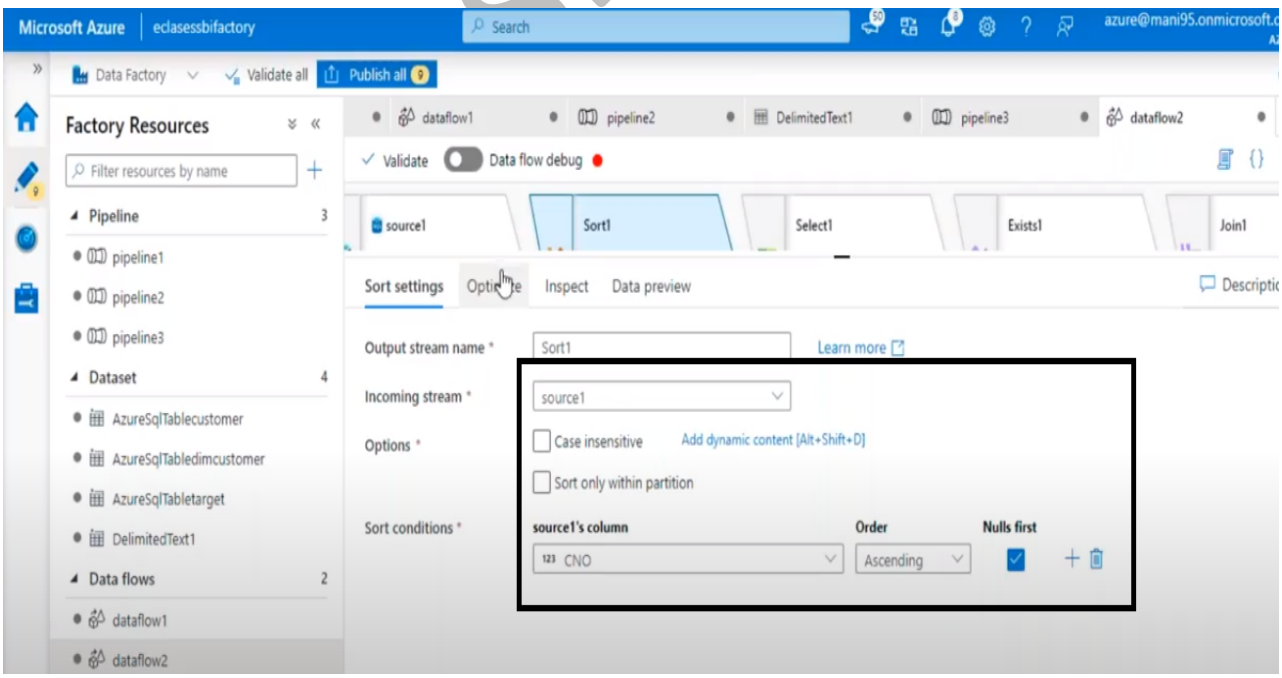
Filter: If you want to apply any filter conditions on input stream data before passing to next transformation.

For example: Filtering data based on hyderabad location

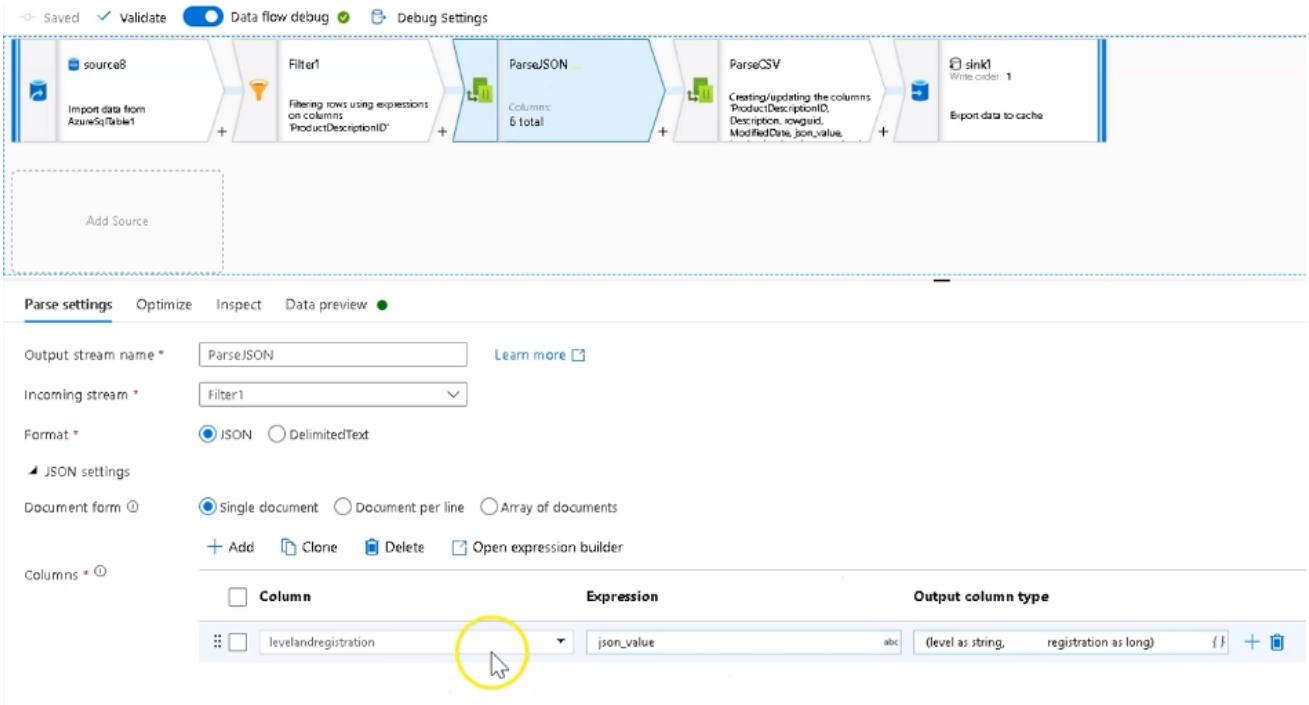


Sort: If you want to sort input source data based on column in ascending or descending order.

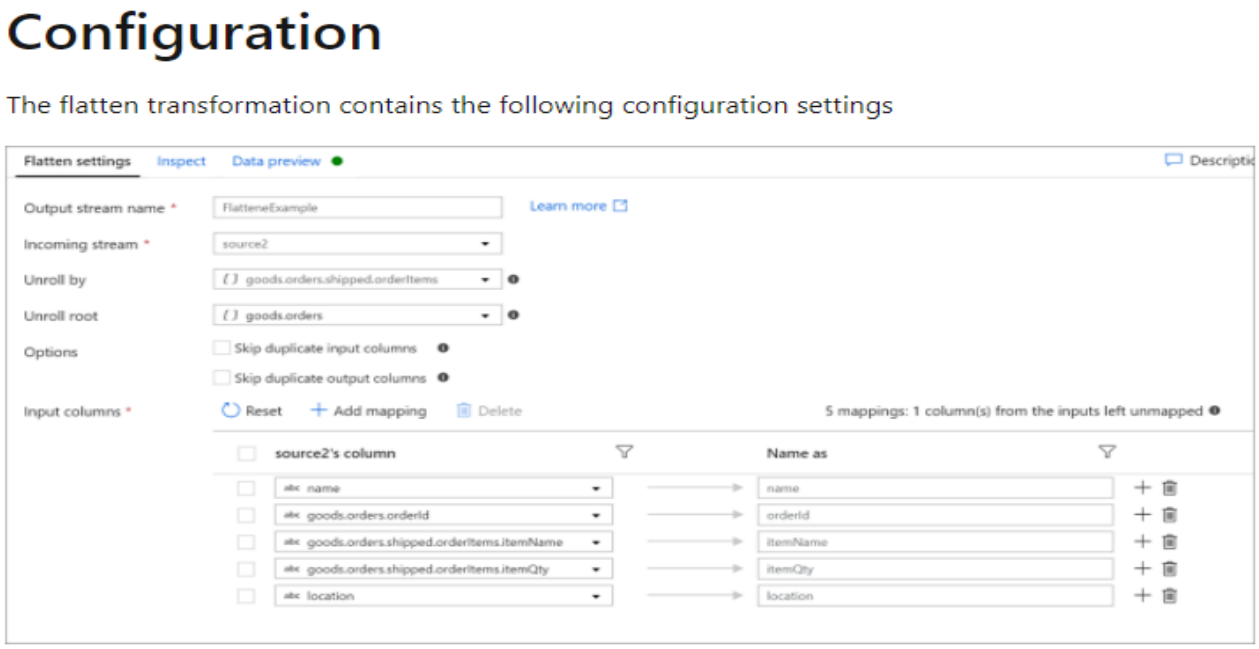
For example: Sorting data based on CNO



Parse: Use the Parse transformation to parse columns in your data that are in document form. The current supported types of embedded documents that can be parsed are JSON, XML, and delimited text.

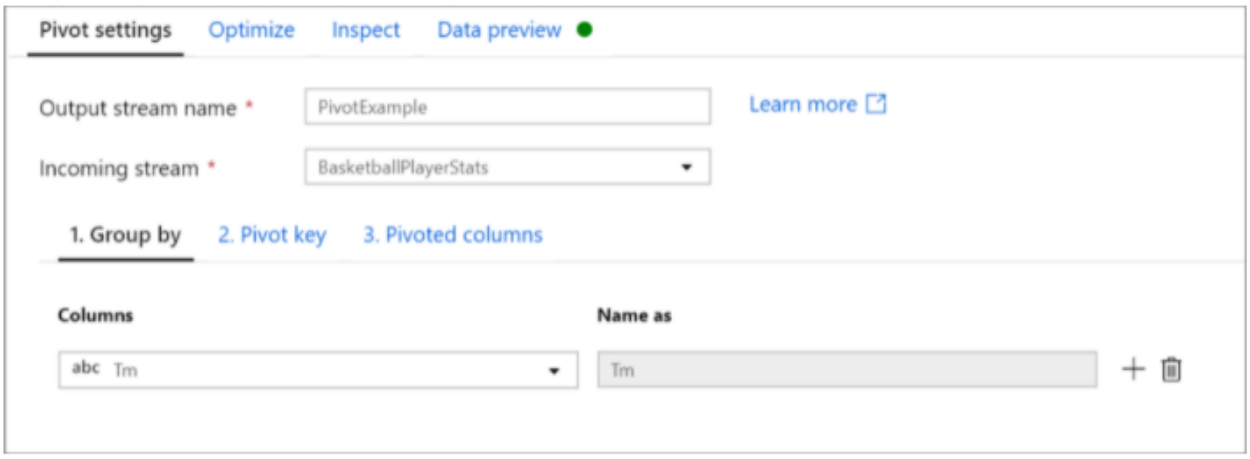


Flatten: Use the flatten transformation to take array values inside hierarchical structures such as JSON and unroll them into individual rows. This process is known as denoralization.



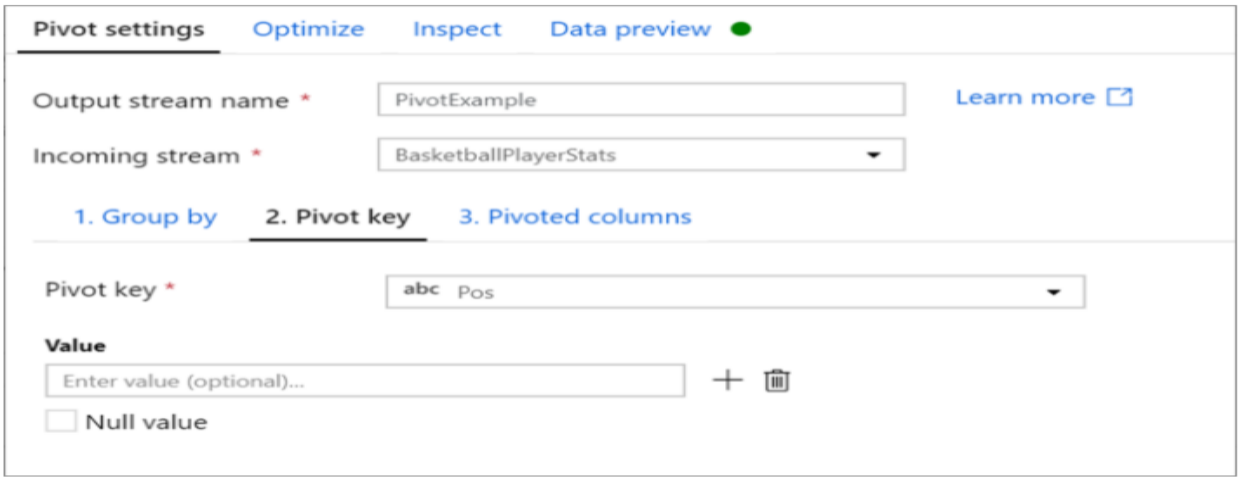
Pivot: Pivot is an aggregation transformation where you select group by columns and generate pivot columns using aggregate functions.

Group by



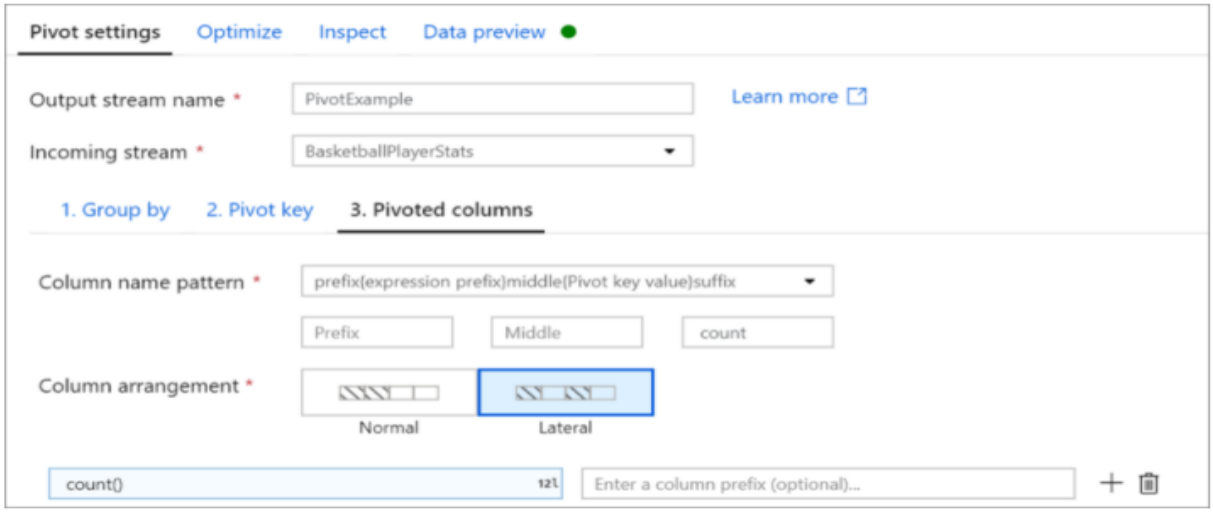
Select which columns to aggregate the pivoted columns over. The output data will group all rows with the same group by values into one row. The aggregation done in the pivoted column will occur over each group.

Pivot key



The pivot key is the column whose row values get pivoted into new columns. By default, the pivot transformation will create a new column for each unique row value.

Pivoted columns



For each unique pivot key value that becomes a column, generate an aggregated row value for each group. You can create multiple columns per pivot key. Each pivot column must contain at least one aggregate function.

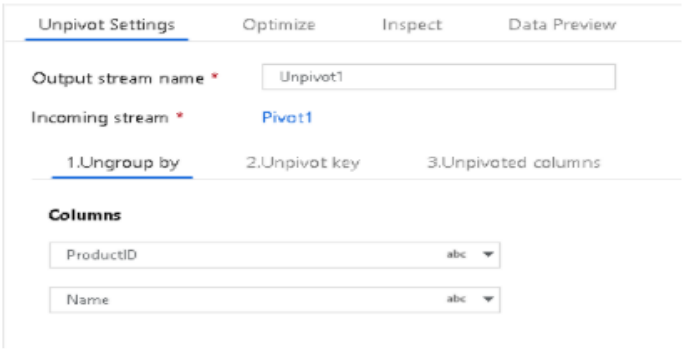
Column name pattern: Select how to format the column name of each pivot column. The outputted column name will be a combination of the pivot key value, column prefix and optional prefix, suffice, middle characters.

Column arrangement: If you generate more than one pivot column per pivot key, choose how you want the columns to be ordered.

Column prefix: If you generate more than one pivot column per pivot key, enter a column prefix for each column. This setting is optional if you only have one pivoted column.

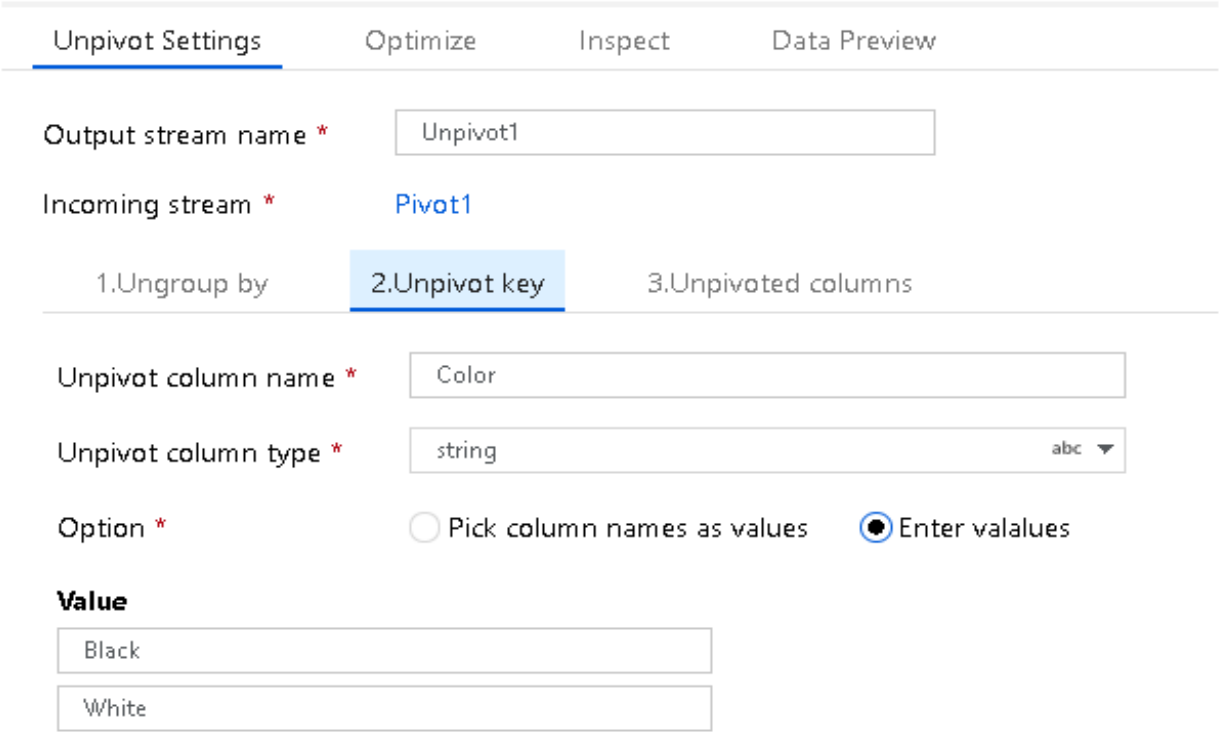
Unpivot: Use to Unpivot the data

Ungroup By

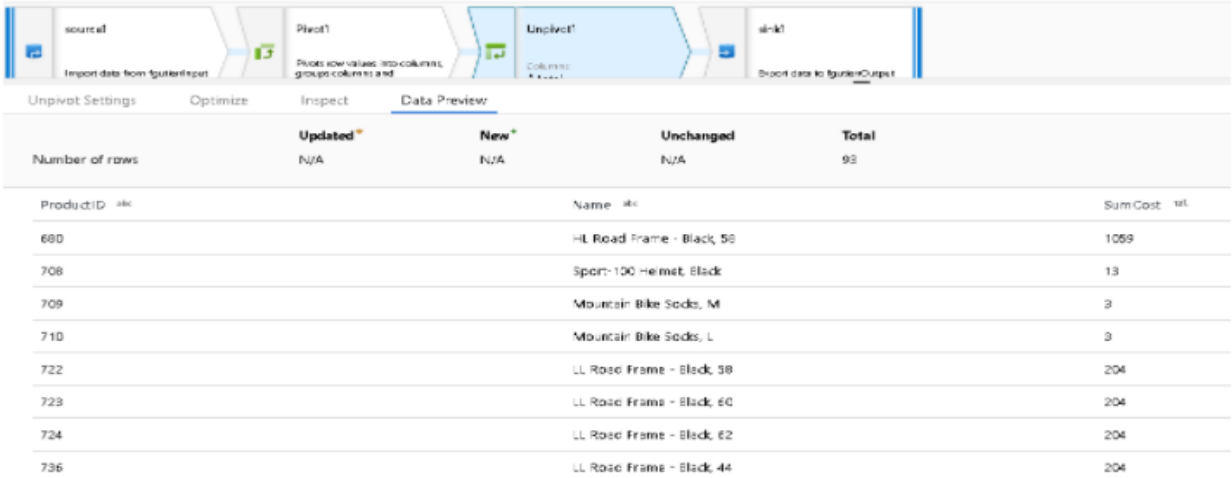


First, set the columns that you wish to ungroup by for your unpivot aggregation. Set one or more columns for ungrouping with the + sign next to the column list.

Unpivot Key



Unpivoted Columns

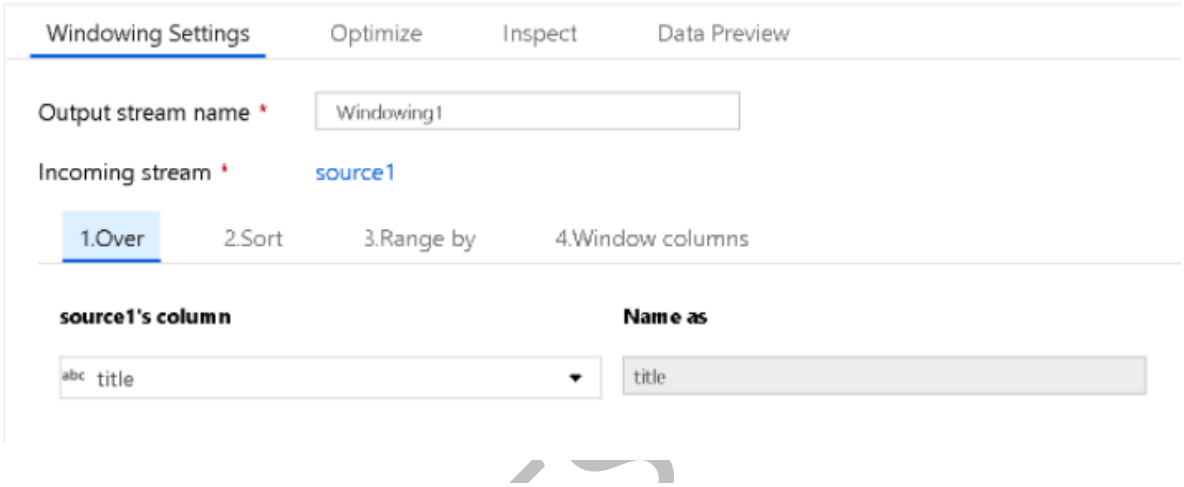


Lastly, choose the column name for storing the values for unpivoted columns that are transformed into rows.

Window: The Window transformation is where you will define window-based aggregations of columns in your data streams. In the Expression Builder, you can define different types of aggregations that are based on data or time windows (SQL OVER clause) such as LEAD, LAG, NTILE, CUMEDIST, RANK, etc.).

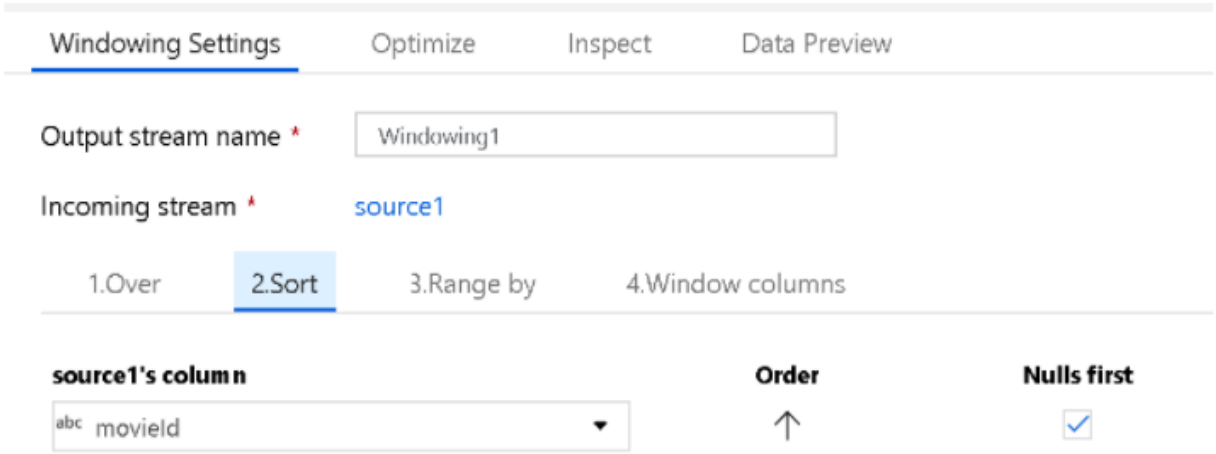
Over

Set the partitioning of column data for your window transformation. The SQL equivalent is the `Partition By` in the Over clause in SQL. If you wish to create a calculation or create an expression to use for the partitioning, you can do that by hovering over the column name and select "computed column".



Sort

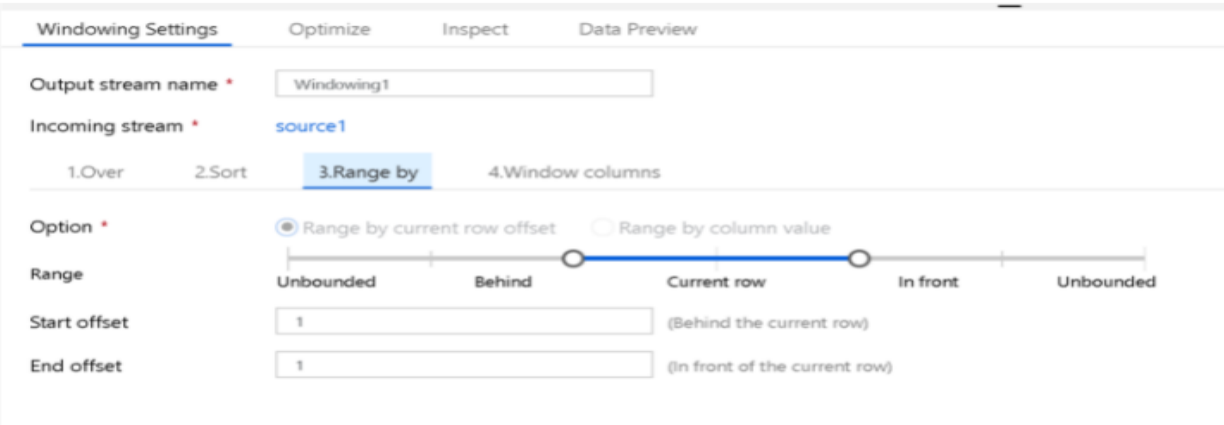
Another part of the Over clause is setting the `Order By`. This will set the data sort ordering. You can also create an expression for a calculate value in this column field for sorting.



Range By

Next, set the window frame as Unbounded or Bounded. To set an unbounded window frame, set the slider to Unbounded on both ends. If you choose a setting between Unbounded and Current Row, then you must set the Offset start and end values. Both values will be positive integers. You can use either relative numbers or values from your data.

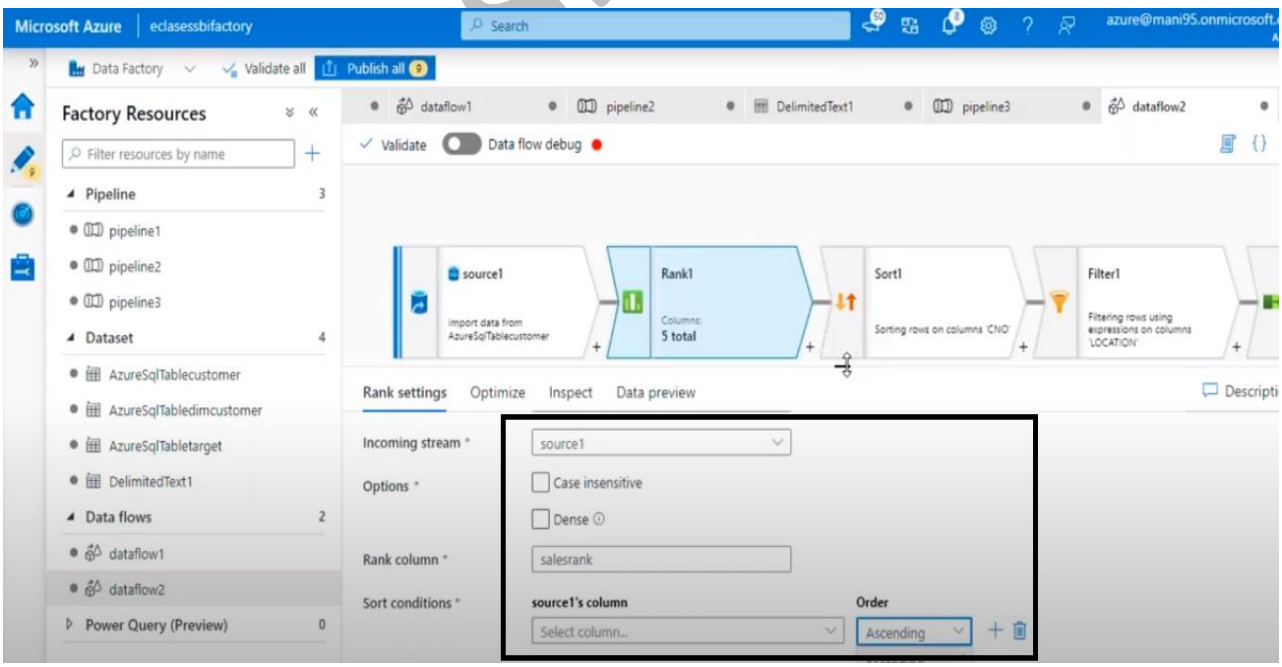
The window slider has two values to set: the values before the current row and the values after the current row. The Start and End offset matches the two selectors on the slider.



Window columns

Lastly, use the Expression Builder to define the aggregations you wish to use with the data windows such as RANK, COUNT, MIN, MAX, DENSE RANK, LEAD, LAG, etc.

Rank: Providing Rank or Dense Rank based on column specified in such order ascending or descending



3. **Data Sets :**

   Dataset is a named view of data that simply points or references, the data you want to use in your activities as inputs and outputs. Datasets identify data within different data stores

4. **Linked Services :**

   Linked services contain connection strings that the service uses at runtime to connect to data sources.

5. **Integration Runtimes :**

   The Integration Runtime (IR) is the compute infrastructure used by Azure Data Factory to connect to different networks and performing data movements and executing activities.

   There are 3 different types of integration Runtimes.

   - Azure or Auto resolve Integration Runtimes

     This is created automatically by azure data factory to connect to cloud networks and services available inside azure.

   - Self -Hosted Integration Runtime

     This Integration runtime used to connect to on premises and virtual private networks. We need to setup this manually.

   - SSIS Integration Runtime

     This integration runtime is used to lift and shift of on premises SSIS packages into Azure Data Factory to schedule and run inside cloud.

6. **Triggers**

   Triggers are used to schedule Data factory pipelines on schedule based.

   There are three different kind of triggers

   - Normal Schedule Triggers.
   - Event Based Triggers
   - Tumbling window Triggers

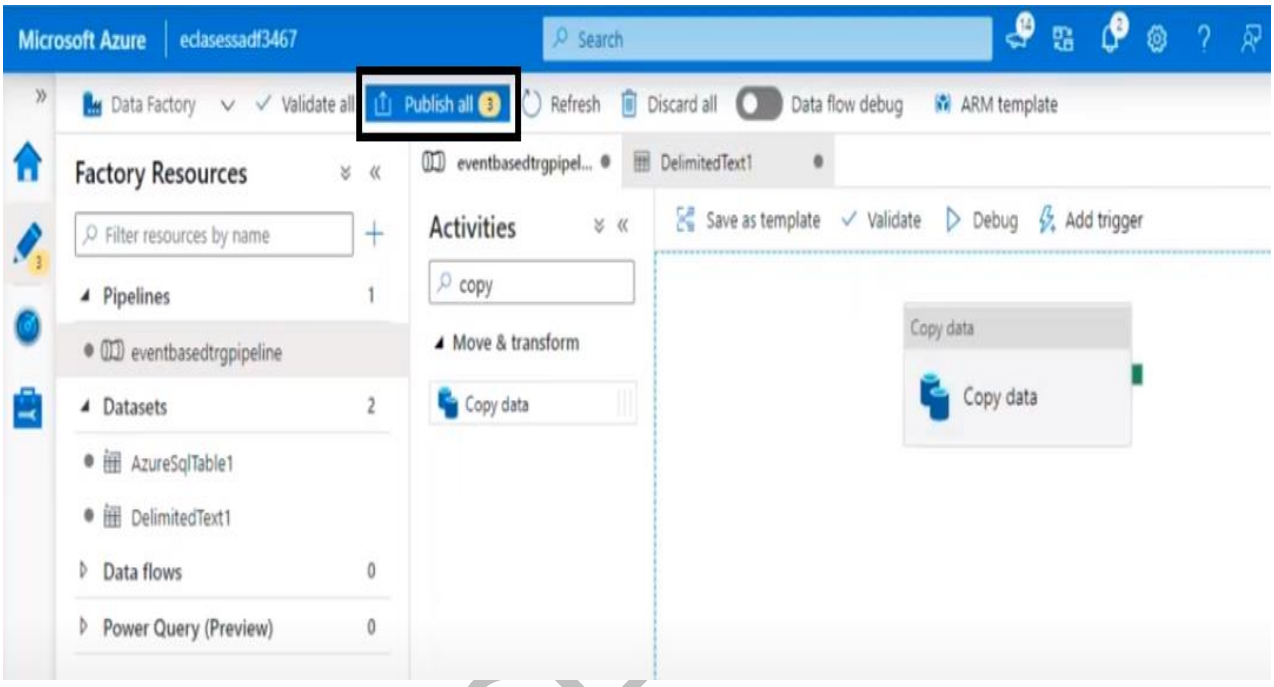
ECLASESS.COM

Normal Schedule Triggers:

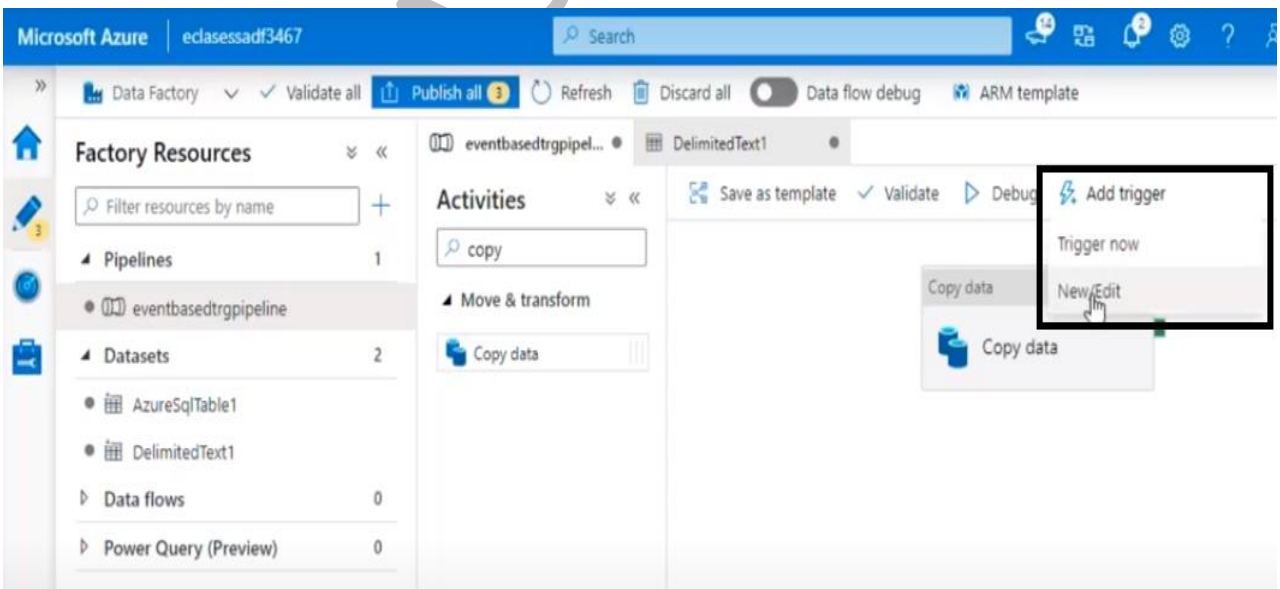
This is normal schedule trigger based on wall clock

Daily or weekly or monthly on specific time zone

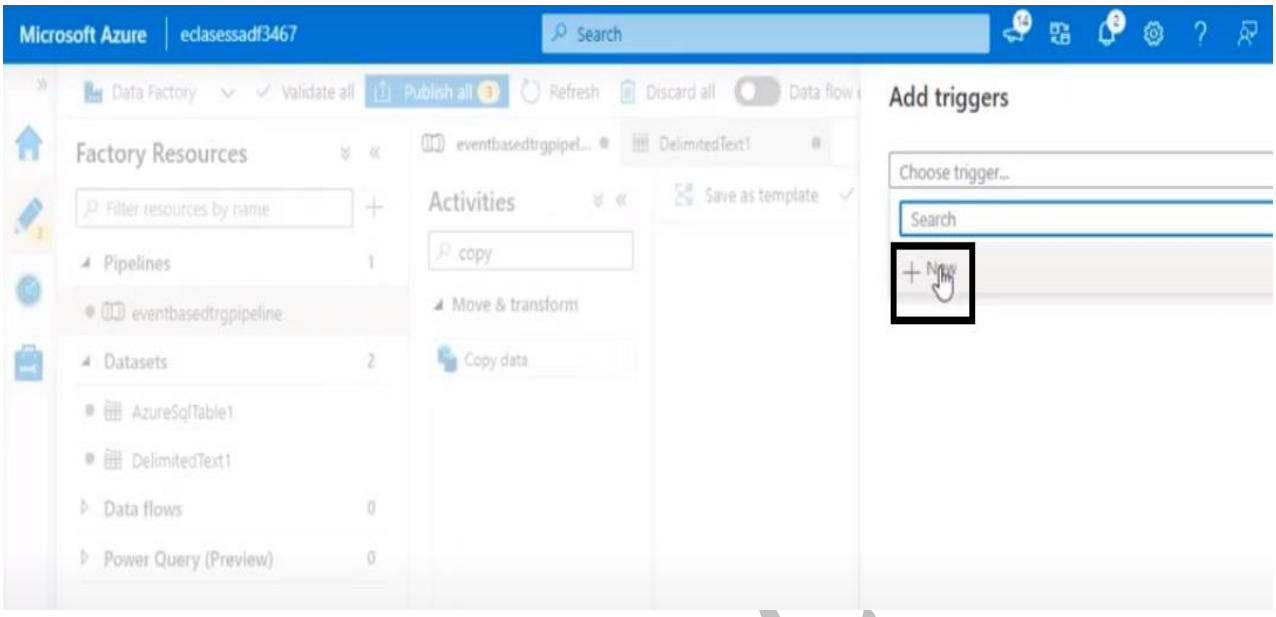
Step 1: Click on publish all



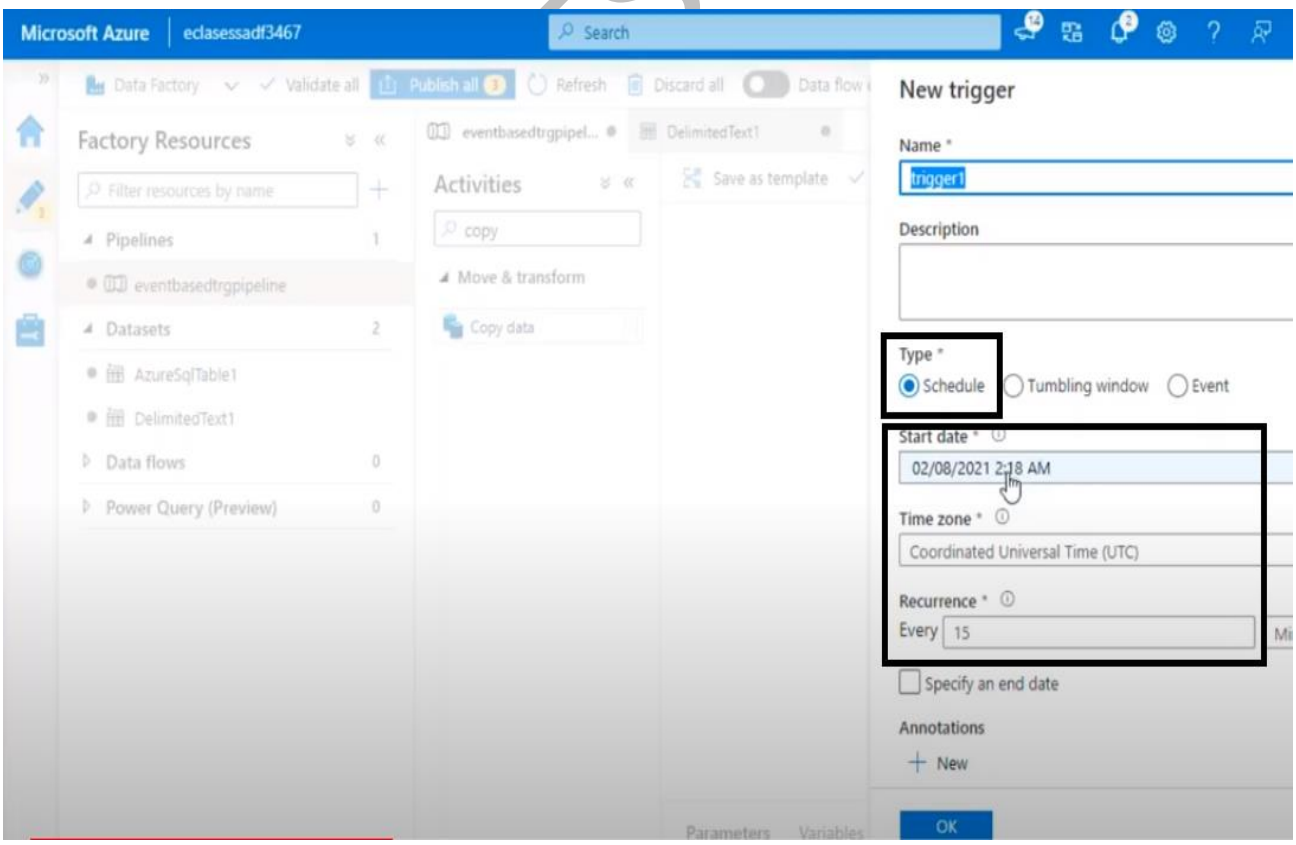
Step 2: Click on Add trigger----New/Edit



Step 3: Click on Choose trigger-----New



Step 4: Select Type as Schedule-----Give Start date----Select Time zone-----Give Recurrence as per requirement-----Click on OK

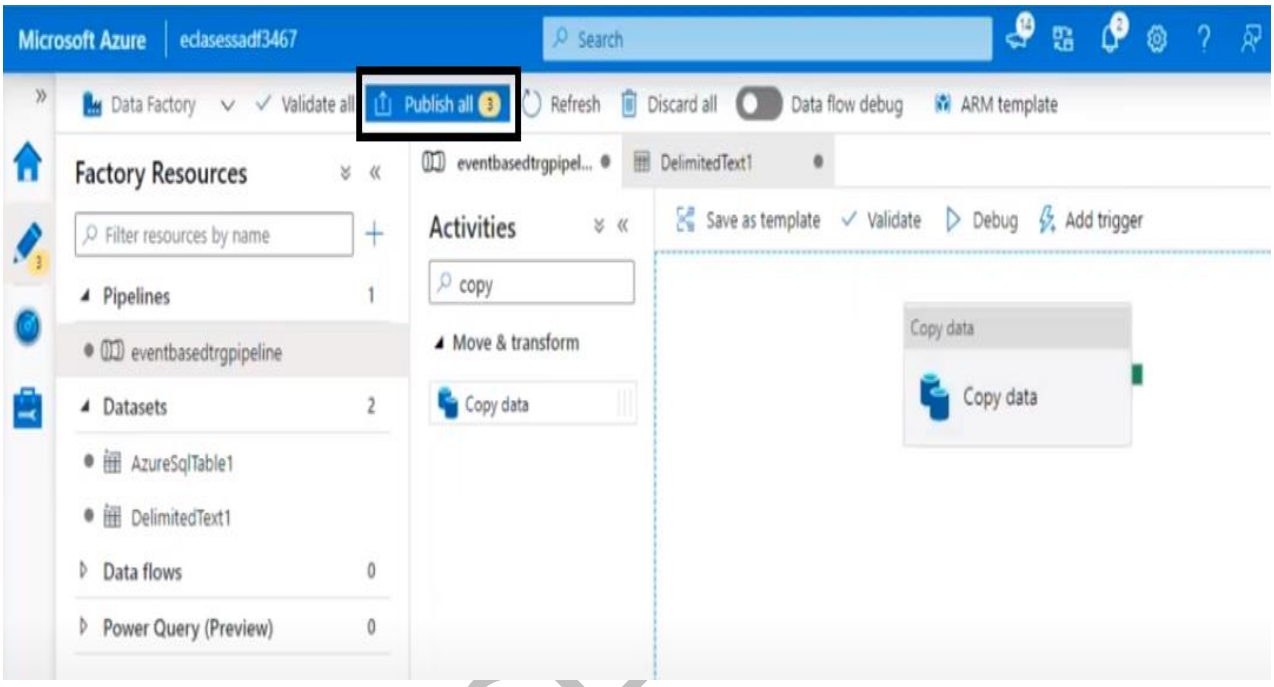


Event based Triggers:

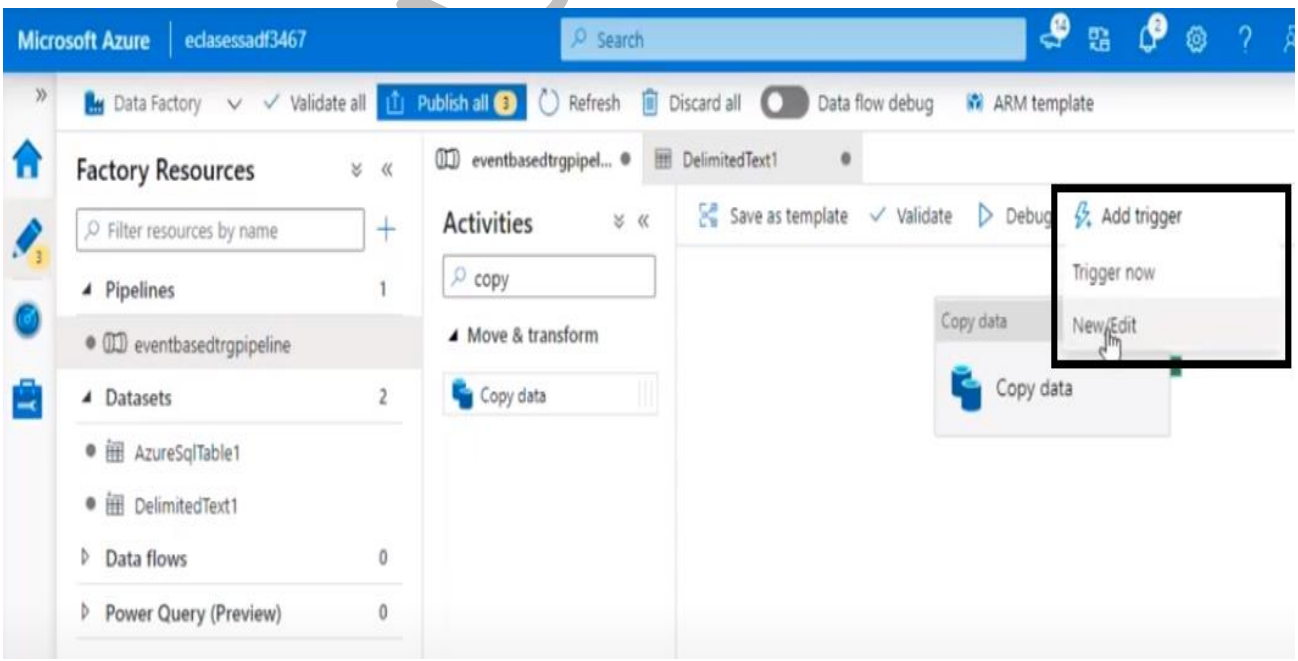
When event occurred like uploading or deleting file from storage account,

Automatically trigger will execute. No need to specify any time.

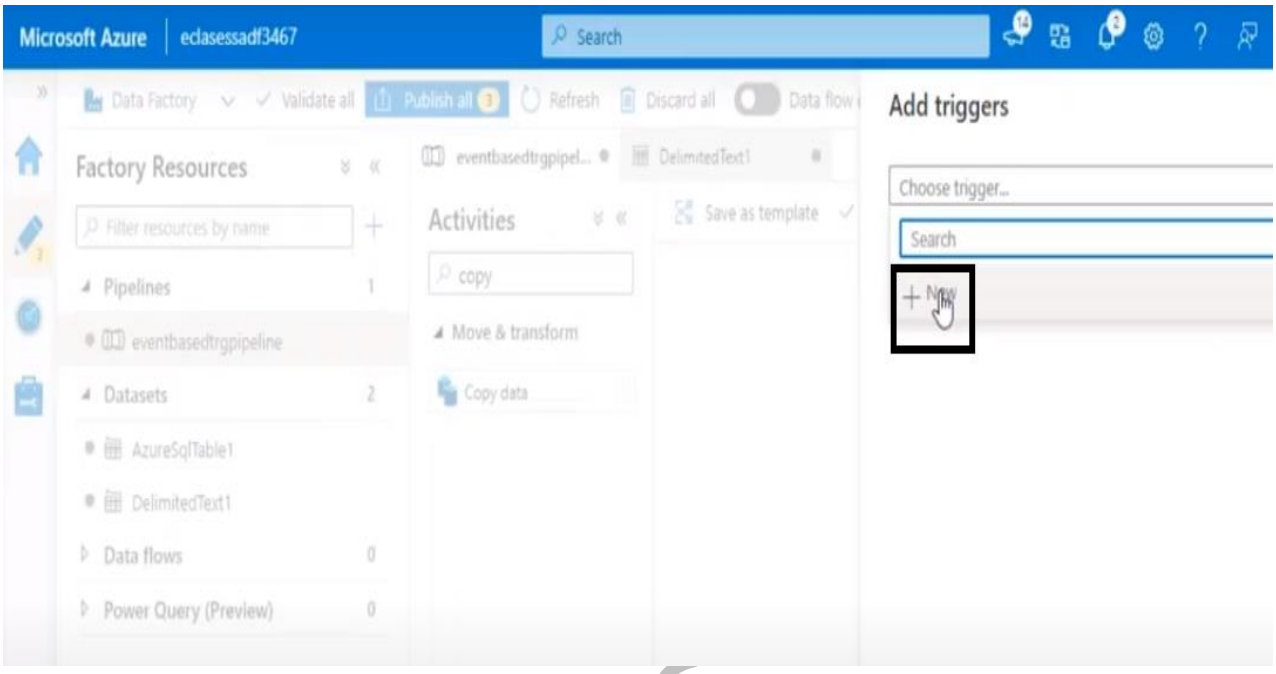
Step 1: Click on publish all



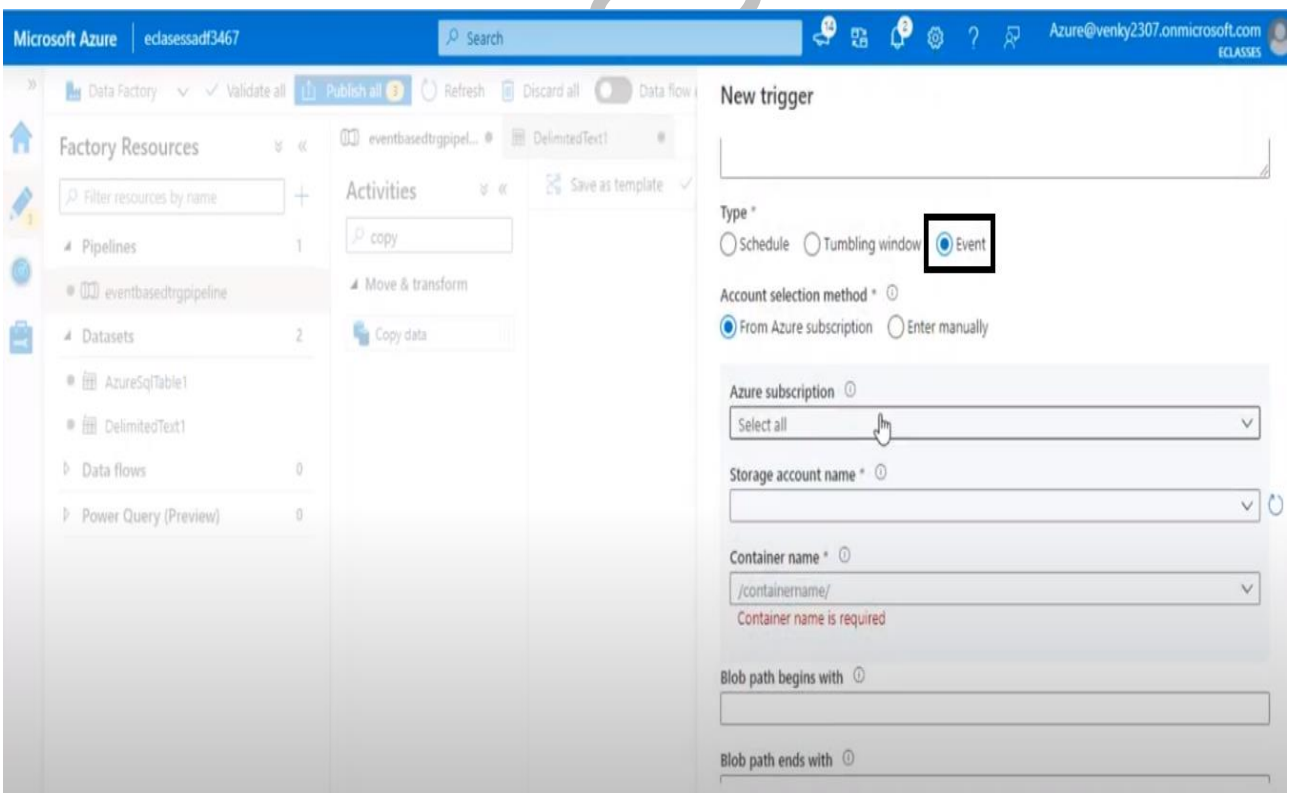
Step 2: Click on Add trigger----New/Edit



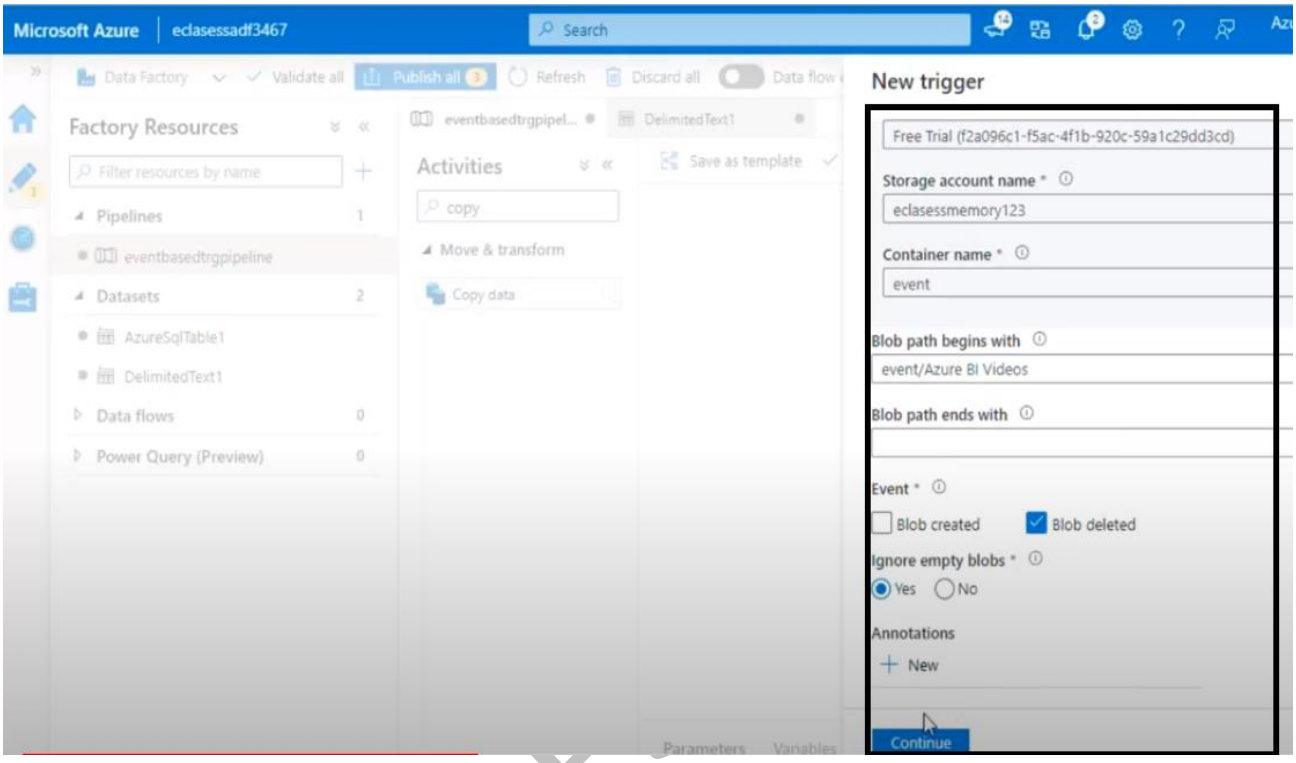
Step 3: Click on Choose trigger-----New



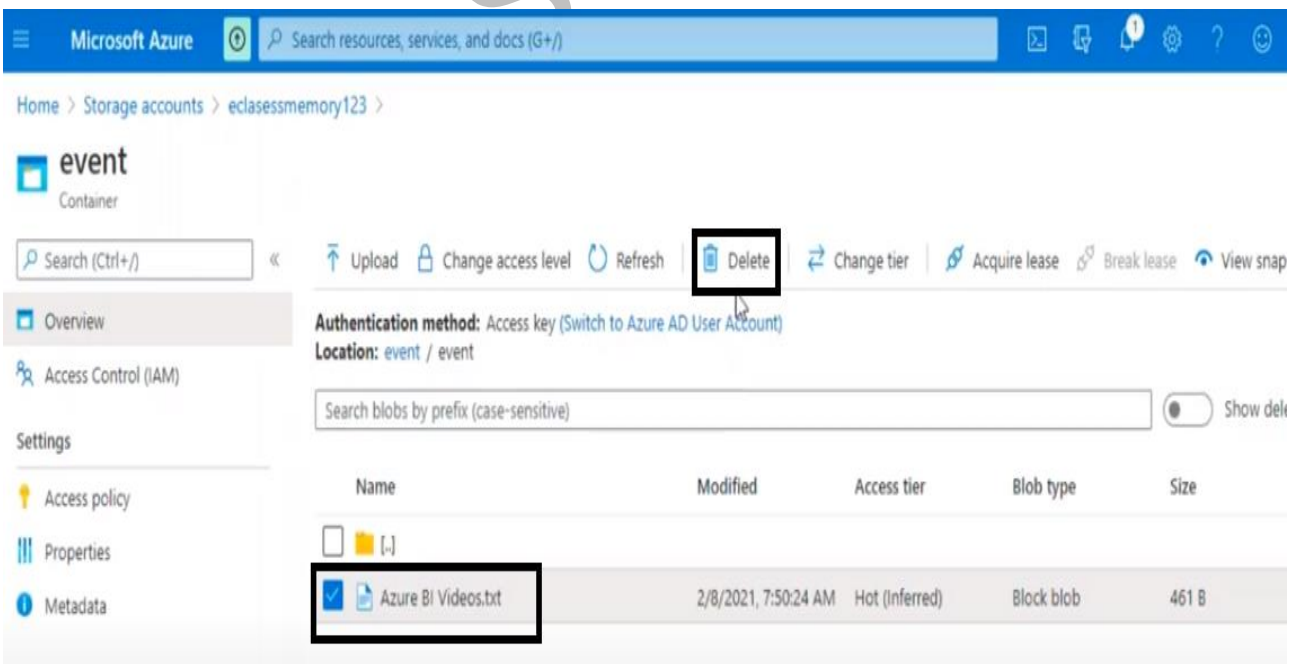
Step 4: Select Type as Event



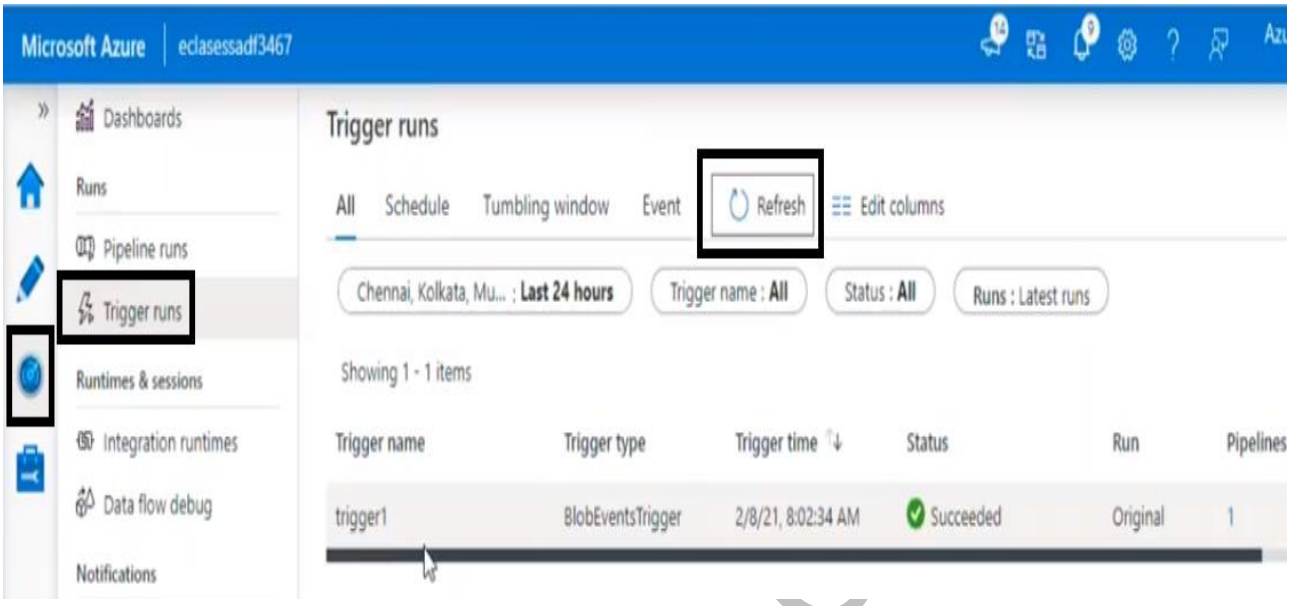
Step 5: Select Subscription----Give storage account name-----Select container name-----Give Blob path begins with or Blob path ends with-----Select Blob Created or Blob deleted as per requirement----Click on Continue



Step 6: Delete the file in the path which is mentioned in Step 5



Step 7: Go to Monitor window----Click on Trigger runs----Click on Refresh-----Trigger got executed successfully



Tumbling window Triggers:

Tumbling window triggers are running pipelines from a specific start date in periodic manner from past date, these are used to run pipelines in historical manner.

How to Copy Data from Blob to Azure SQL Database using COPY ACTIVITY:

Copy Data Activity used to copy data from any source to any destination without using any transformations.

Dataset: Preparing the data from source and destination

Linked Service: A connection to source and destination

For Example: We have to follow below steps for Copy data from Blob Storage to Azure SQL

Prerequisites:

Create Storage account

Create blob Container inside storage account

Upload files with sample employee data

Create Azure SQL database

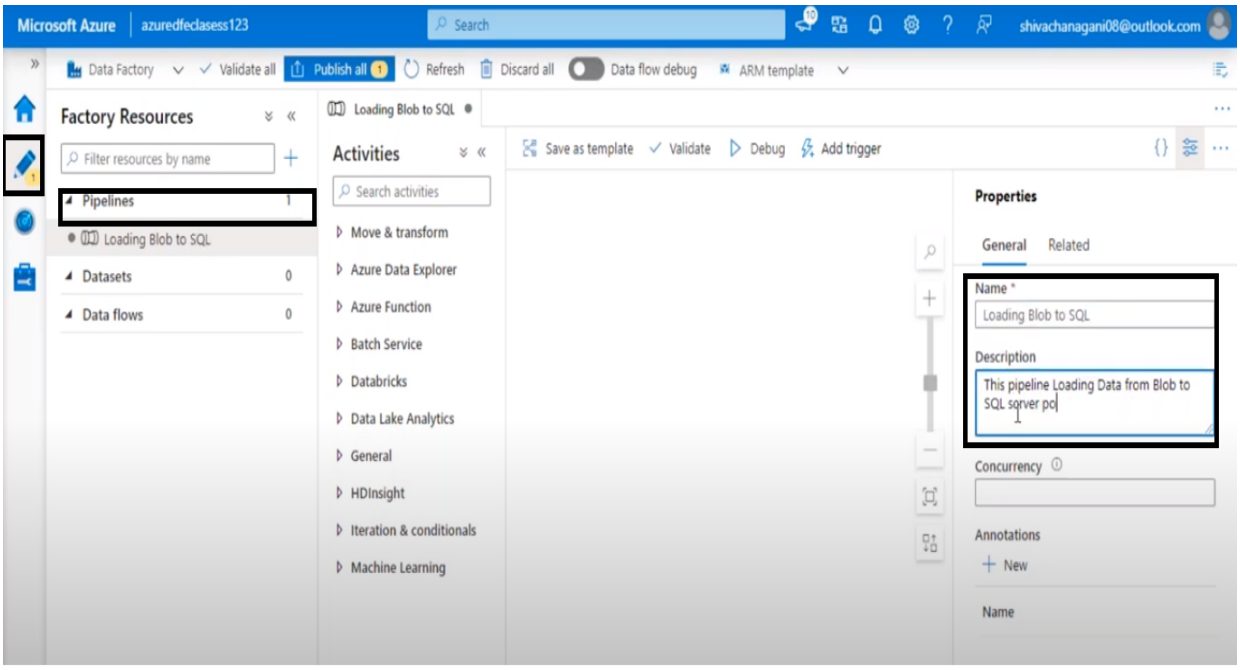
Create employee Table by using below script in Azure SQL Database.

Create Table EMP (Eno int, Ename Varchar (100), Sal int)

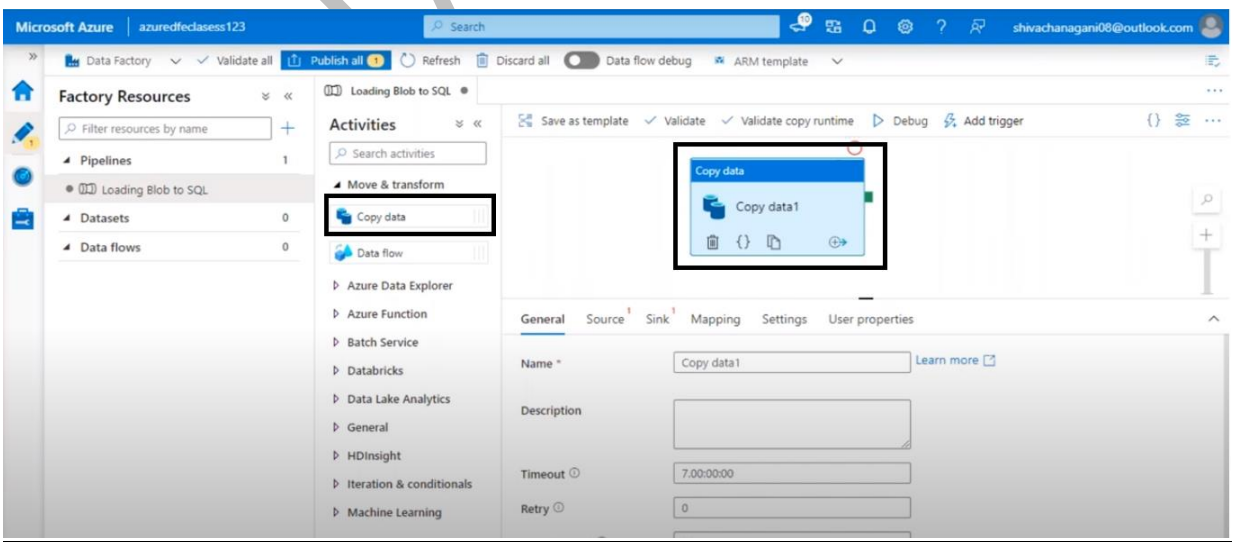
Creating Pipeline:

Step 1: Click on Author tab—Pipelines---new pipeline

Give a proper name to the pipeline and write description



Step 2: Drag the Copy data activity to the development area



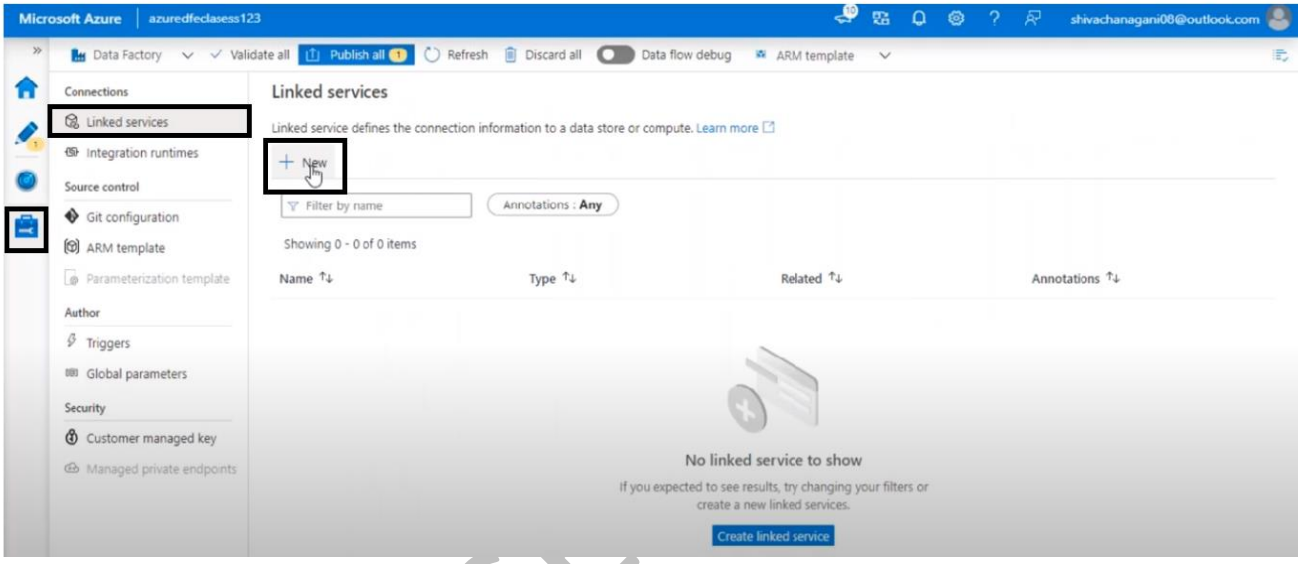
Creating Linked Service:

We need to create two linked services (one linked service for Blob Storage and another for Azure SQL Database)

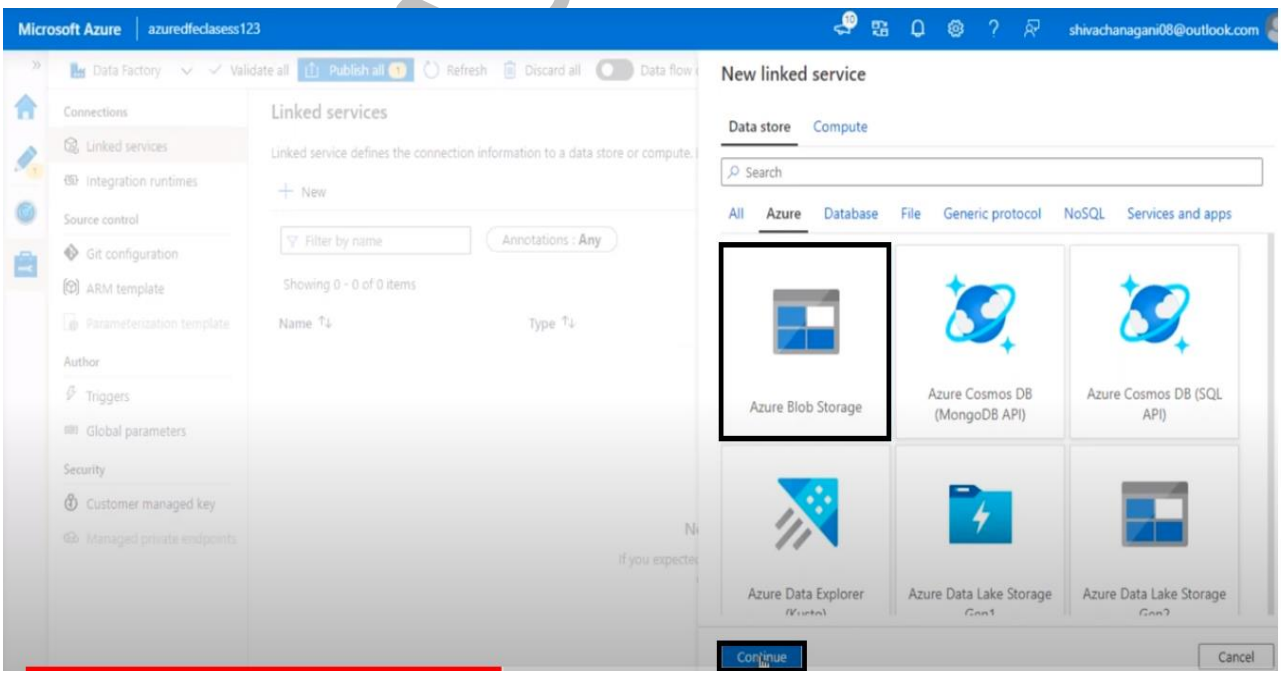
Blob Storage: This is the Storage which contains files inside it.

Azure SQL: We need to create Azure SQL Database and also create one table inside it.

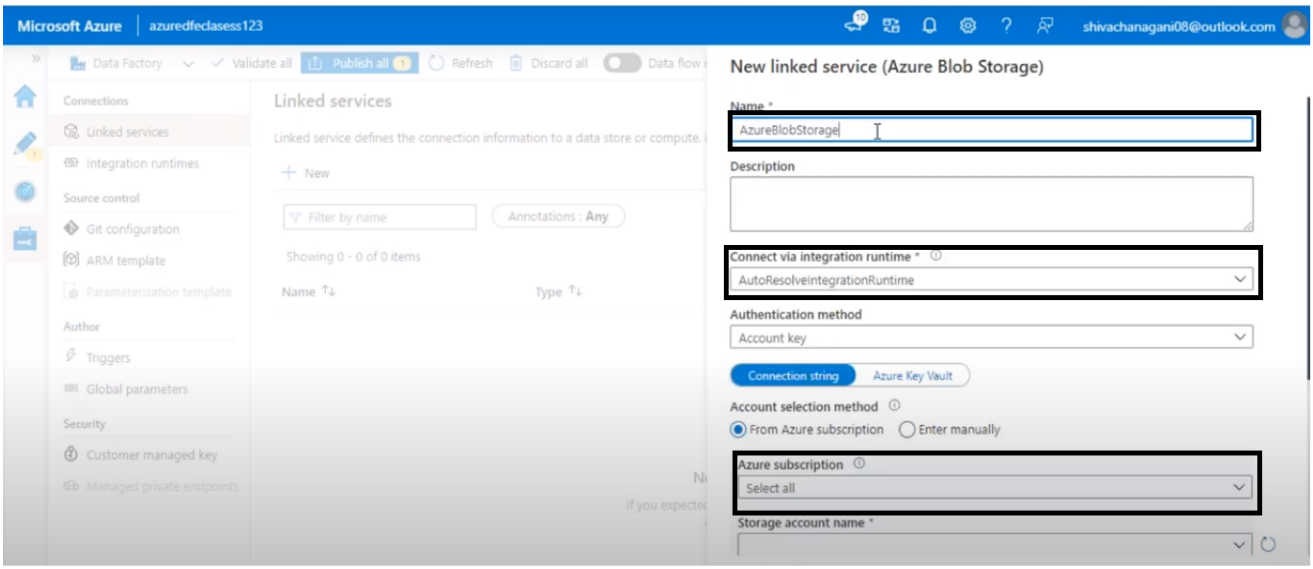
Step 3: Go to Manage Window----Click on Linked Services----Click on New



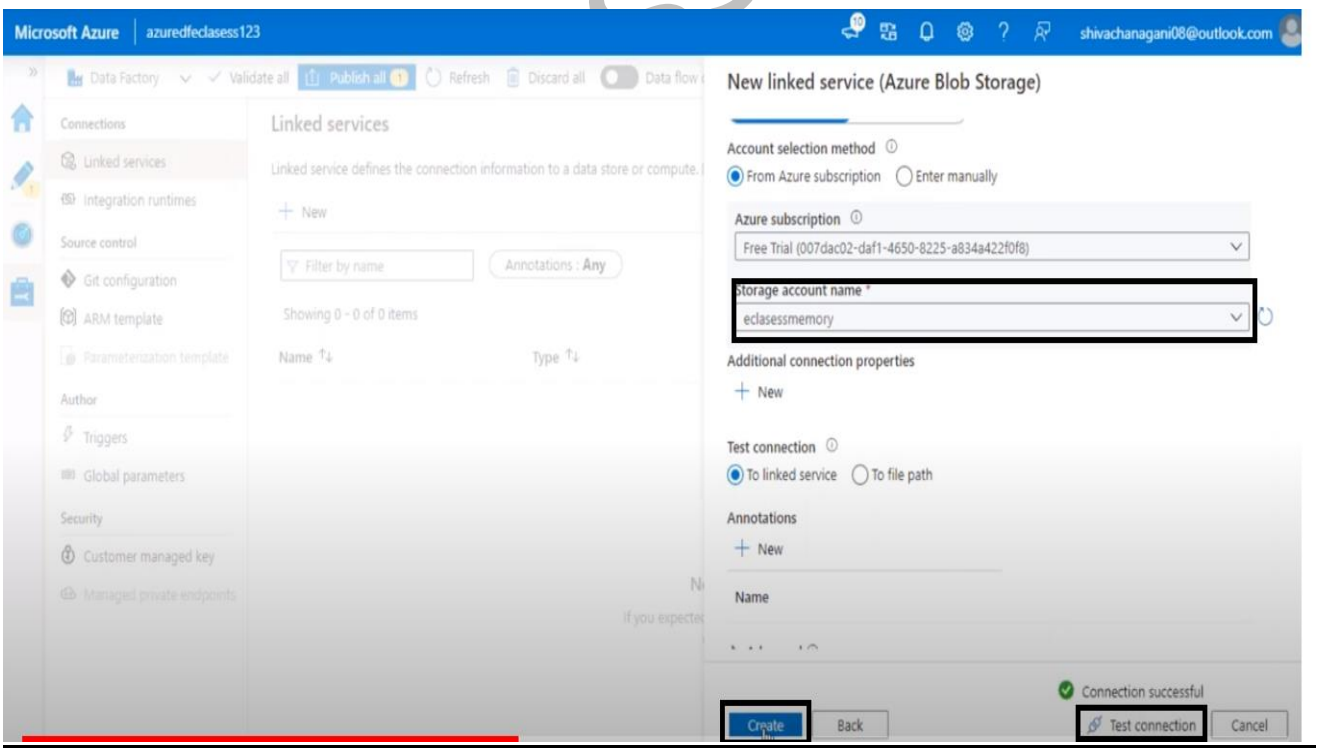
Step 4: Click on Blob Storage and Continue



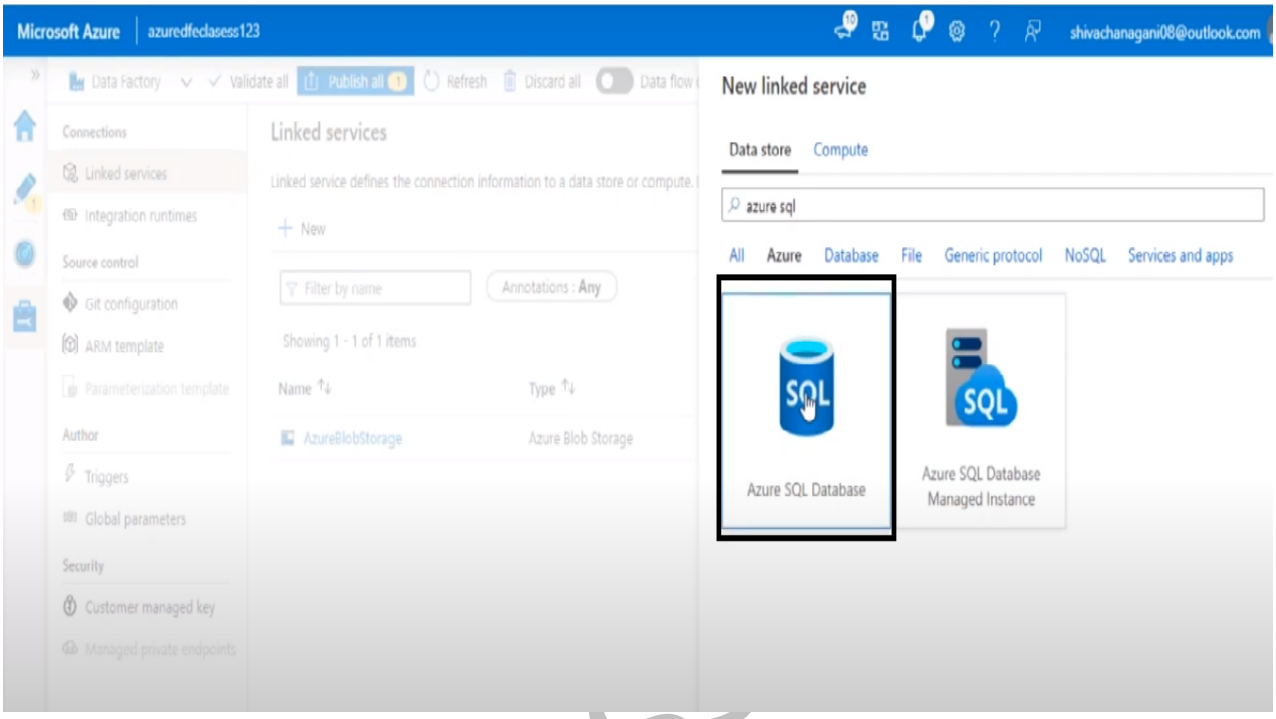
Step 5: Give a proper name on the Name tab---Select Auto resolve integration runtime--Select Subscription



Step 6: Select Blob Storage Account Name from which we are going to load the data to Azure SQL---Test the connection---if it successful then only and click on create

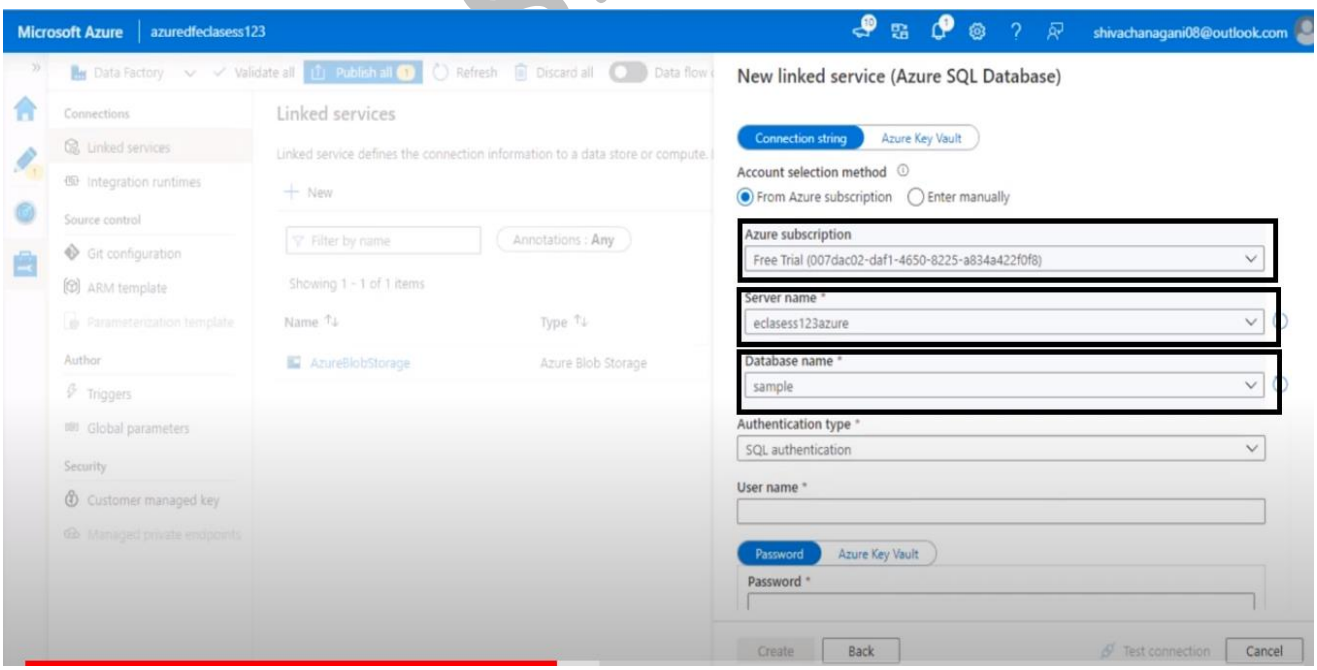


Step 7: Go to Manage Window----Click on Linked Services----Click on New----Select Azure SQL Database

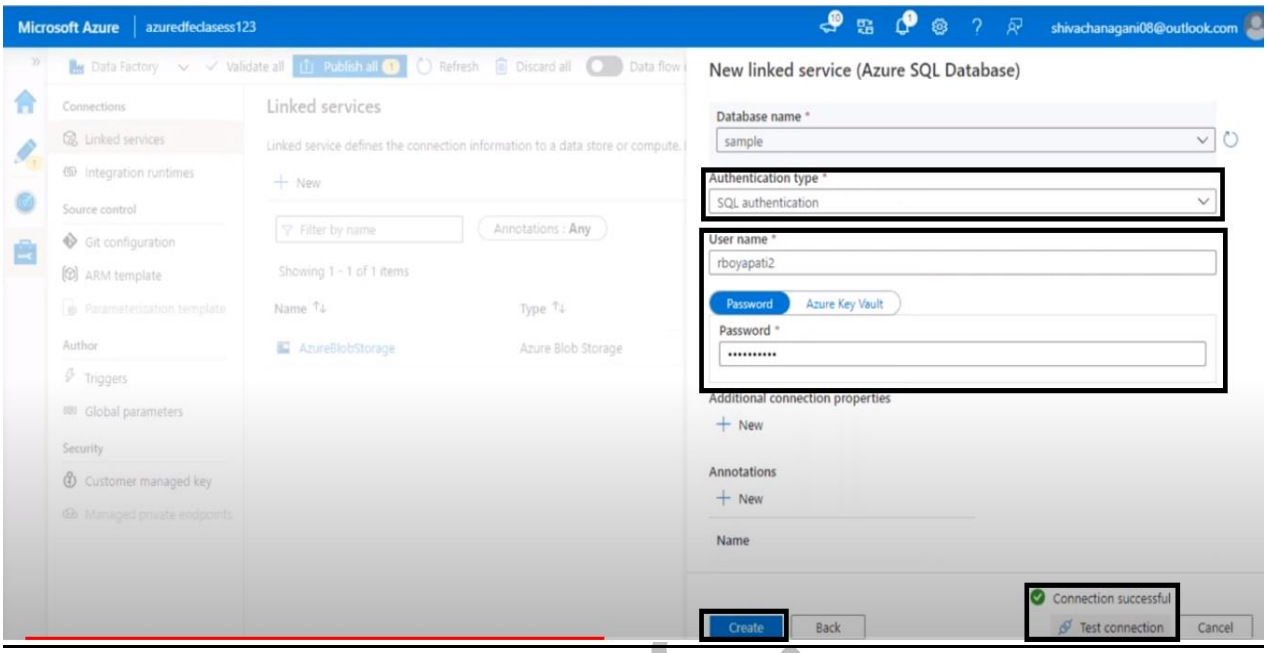


Step 8: Give a Proper name on the name tab---select auto resolve integration runtime---

Select subscription---select sever name---select database name

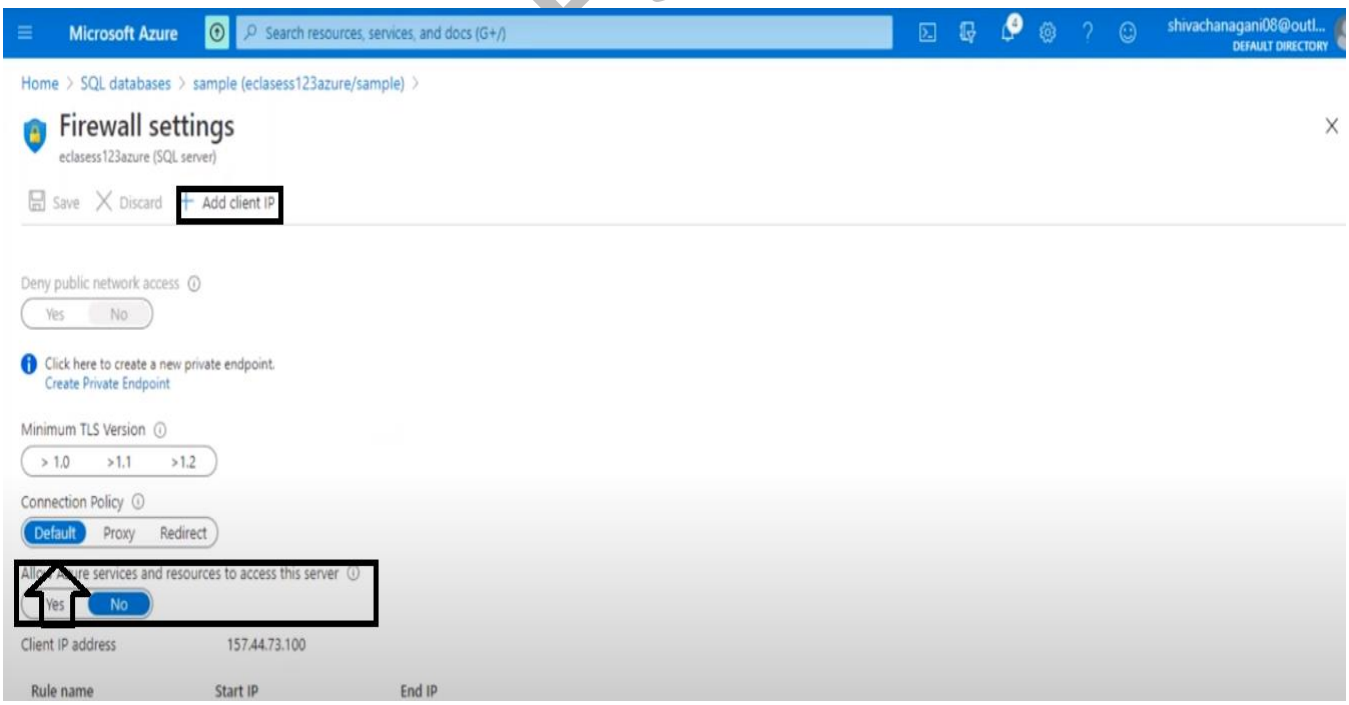


Step 9: Select authentication type---enter user name and password of Azure SQL server---test the connection---if it is success then click on create.



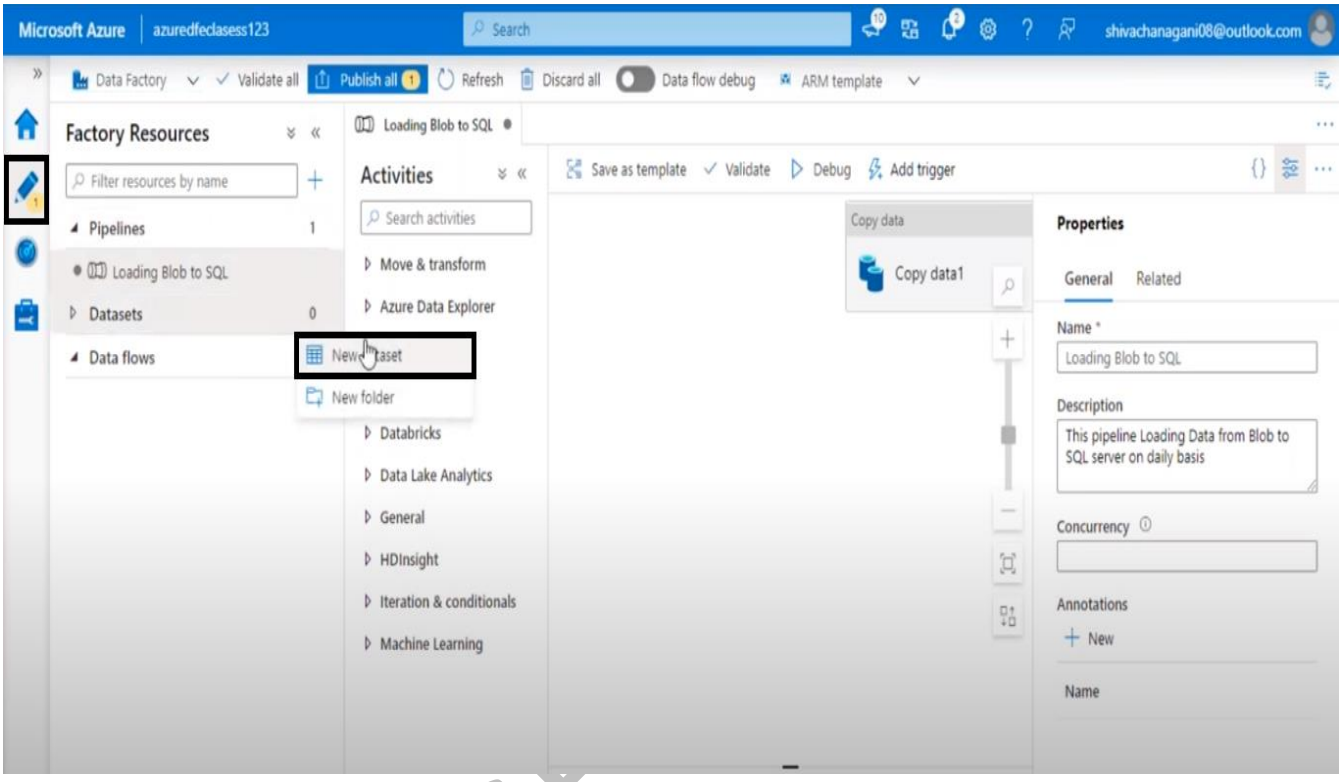
Step 10: If connection gets fail while testing the connection, we need to follow below steps

SQL Database---Firewall Settings---Add Client IP and also we need to allow the azure services

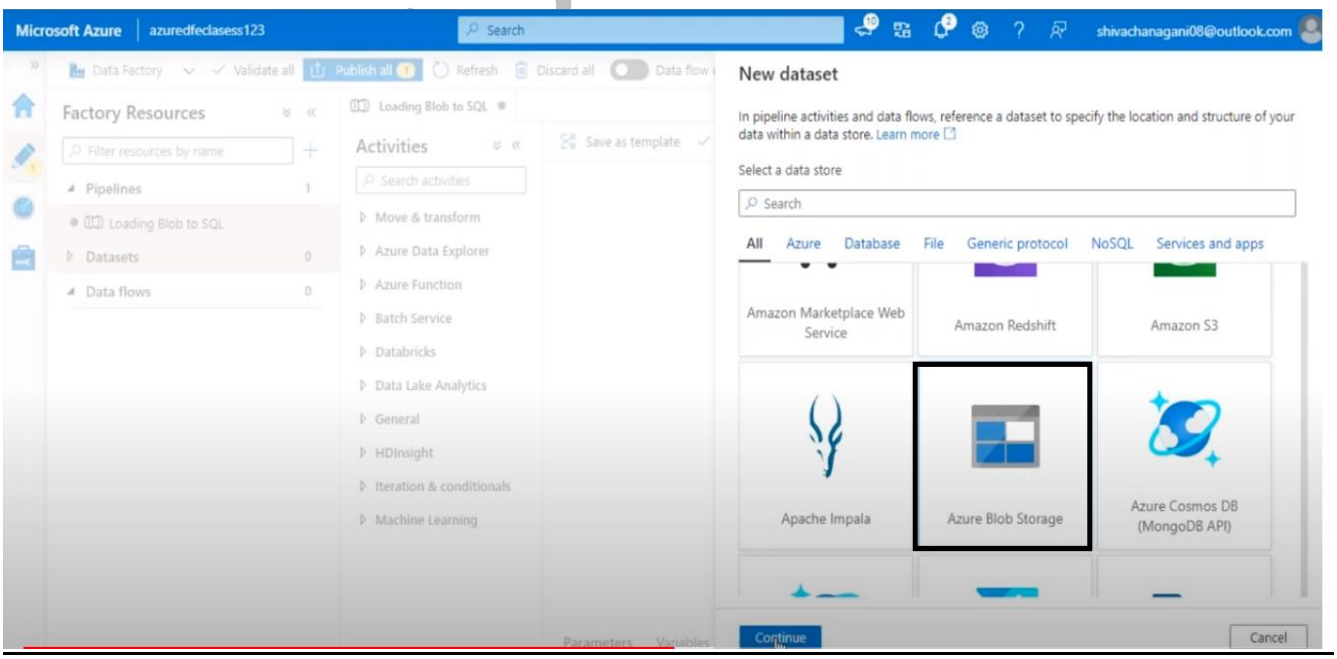


Creating Datasets:

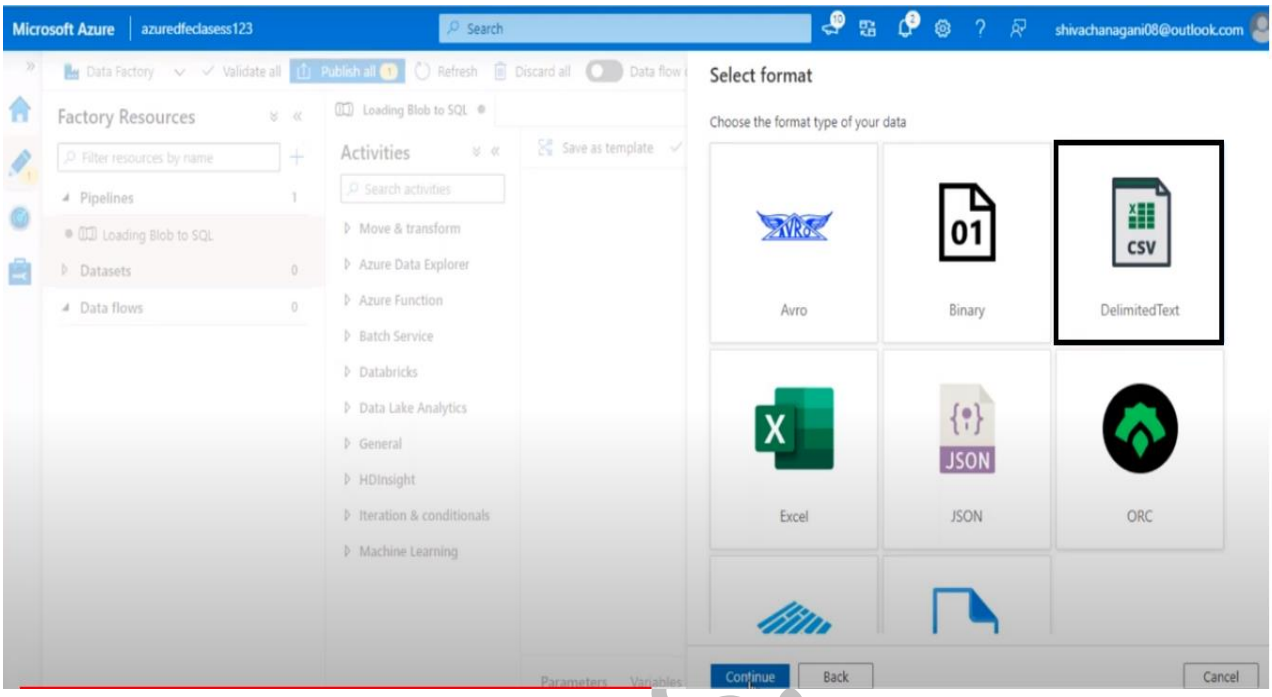
Step 11: Click on Author tab----click on Dataset----click on New dataset



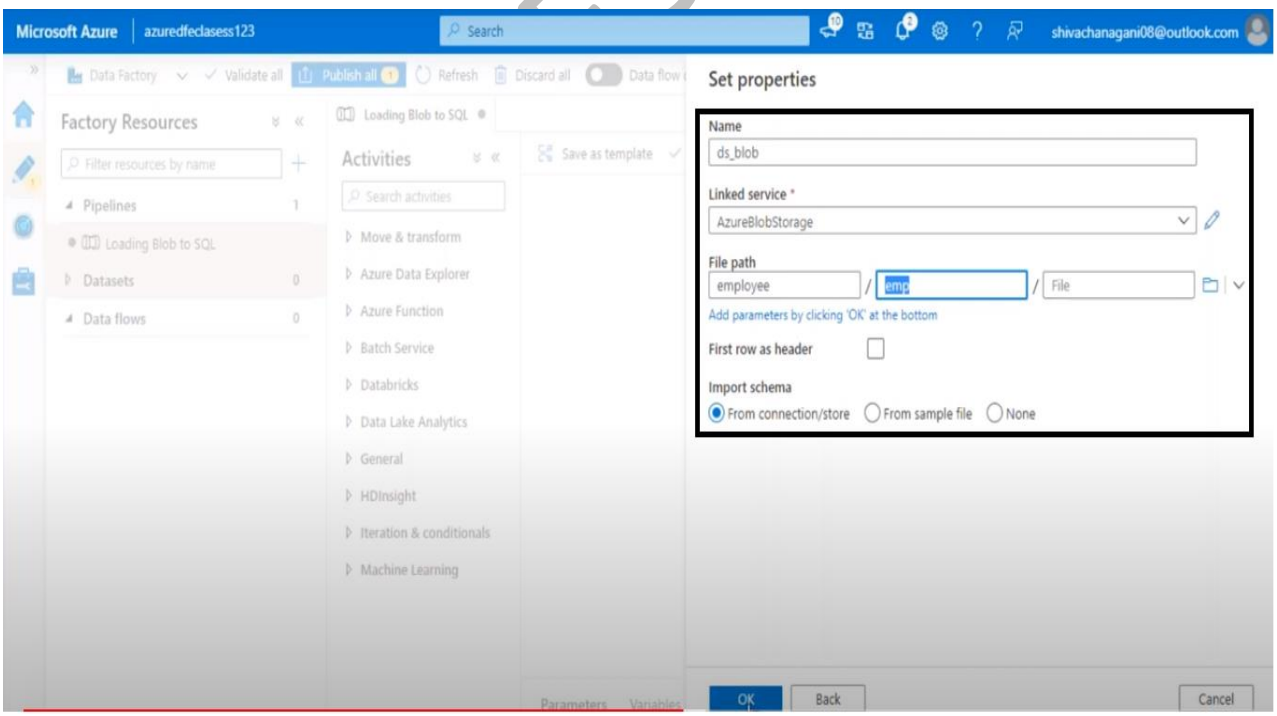
Step 12: Click on Azure Blob Storage and continue



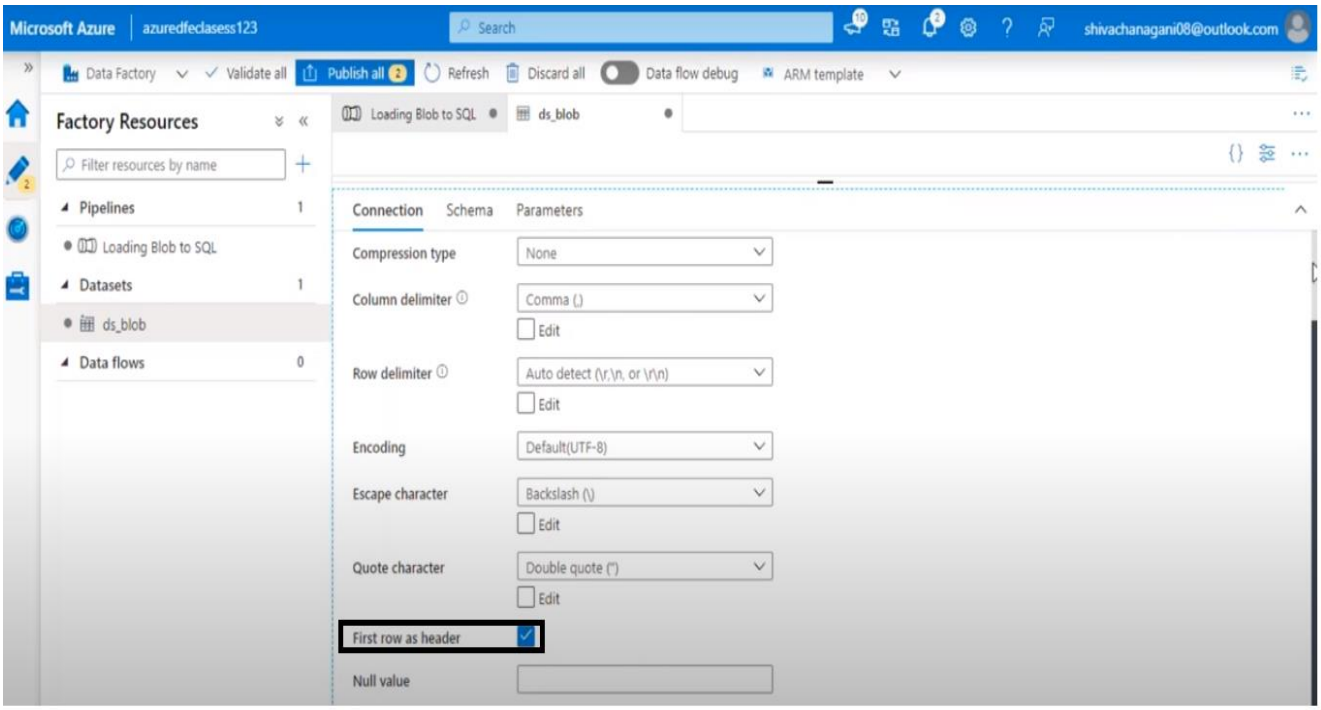
Step 13: Click on Delimited Text and continue



Step 14: Give a proper name on the name tab----Select Linked Service---Select the File Path where the files are stored----Click OK

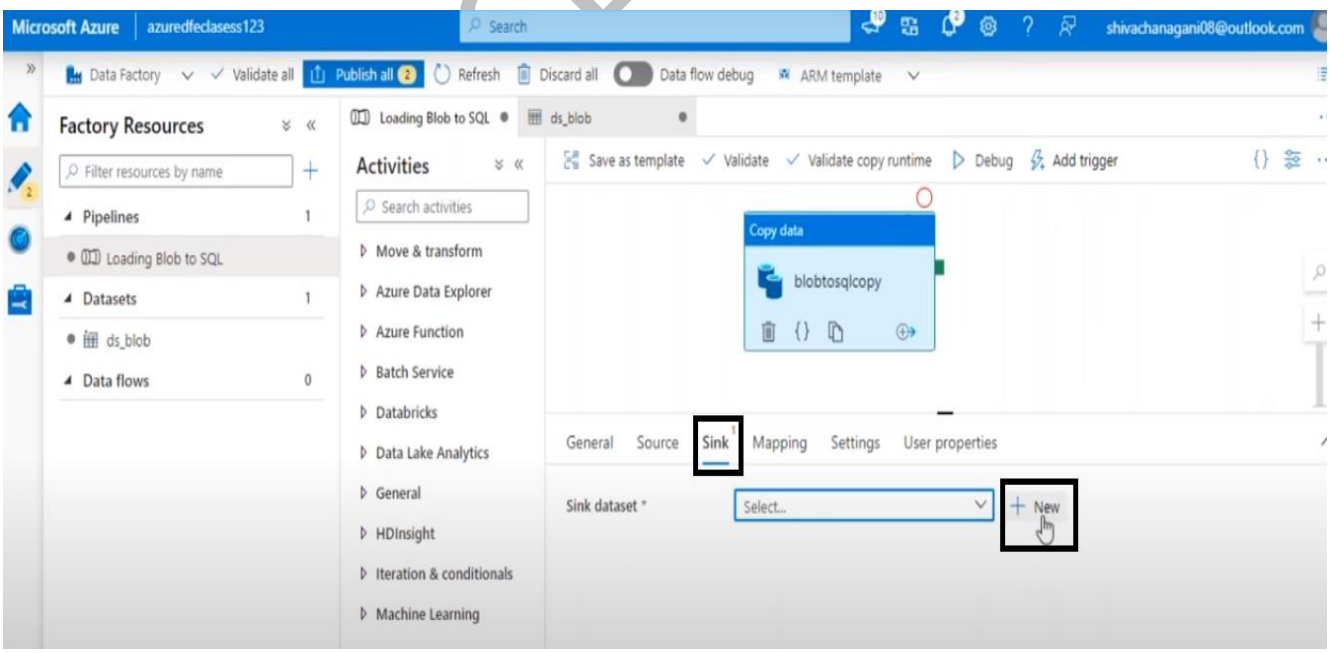


Step 15: Enable First row as header

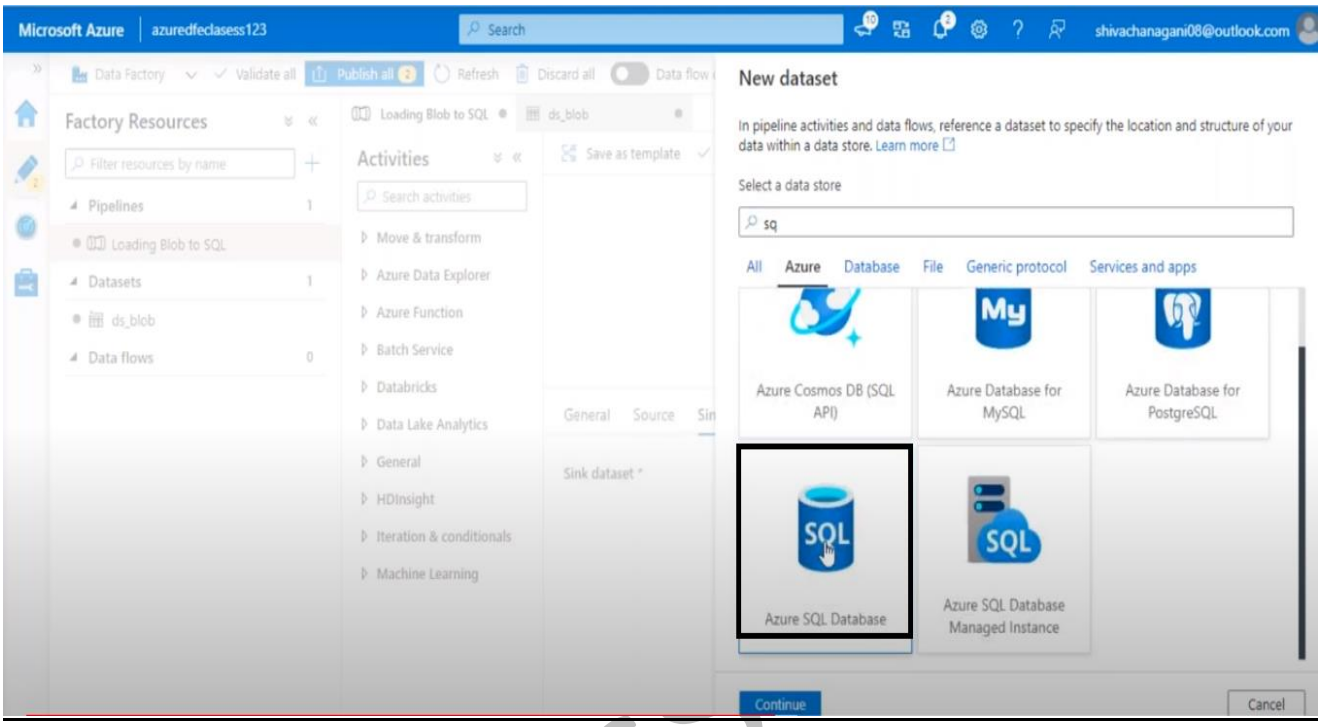


Step 16: Create dataset for the Sink

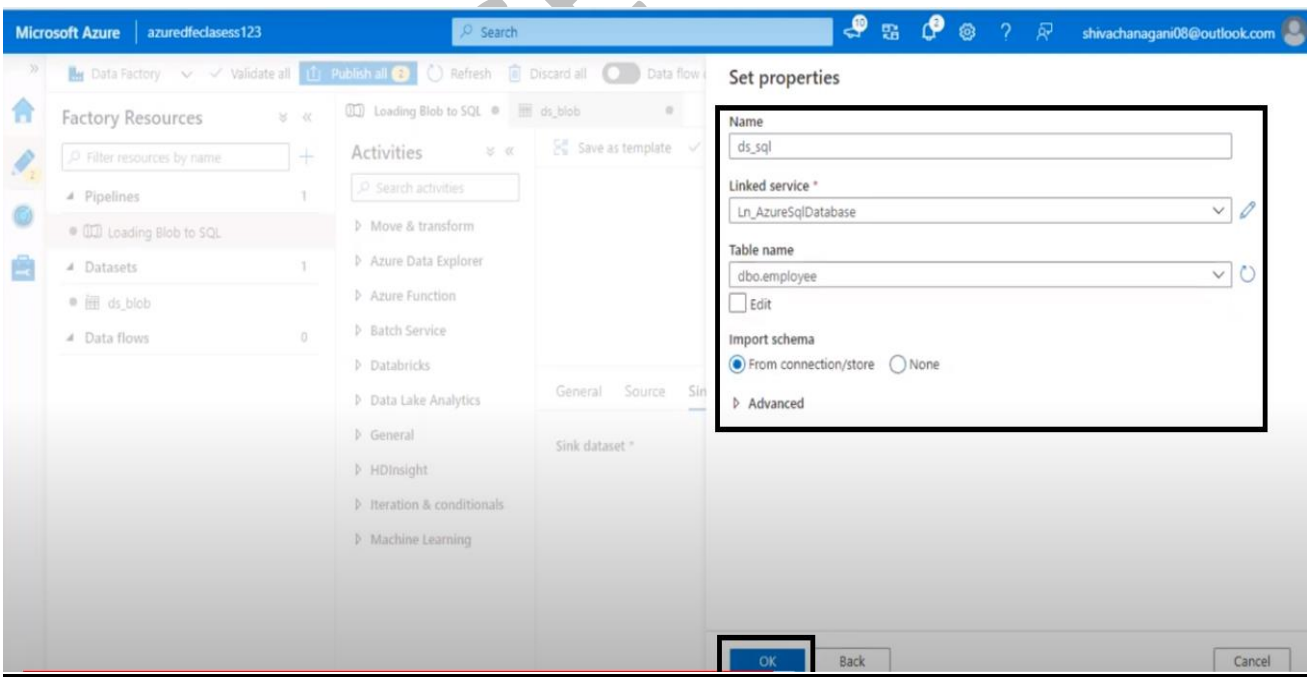
Click on Sink----New



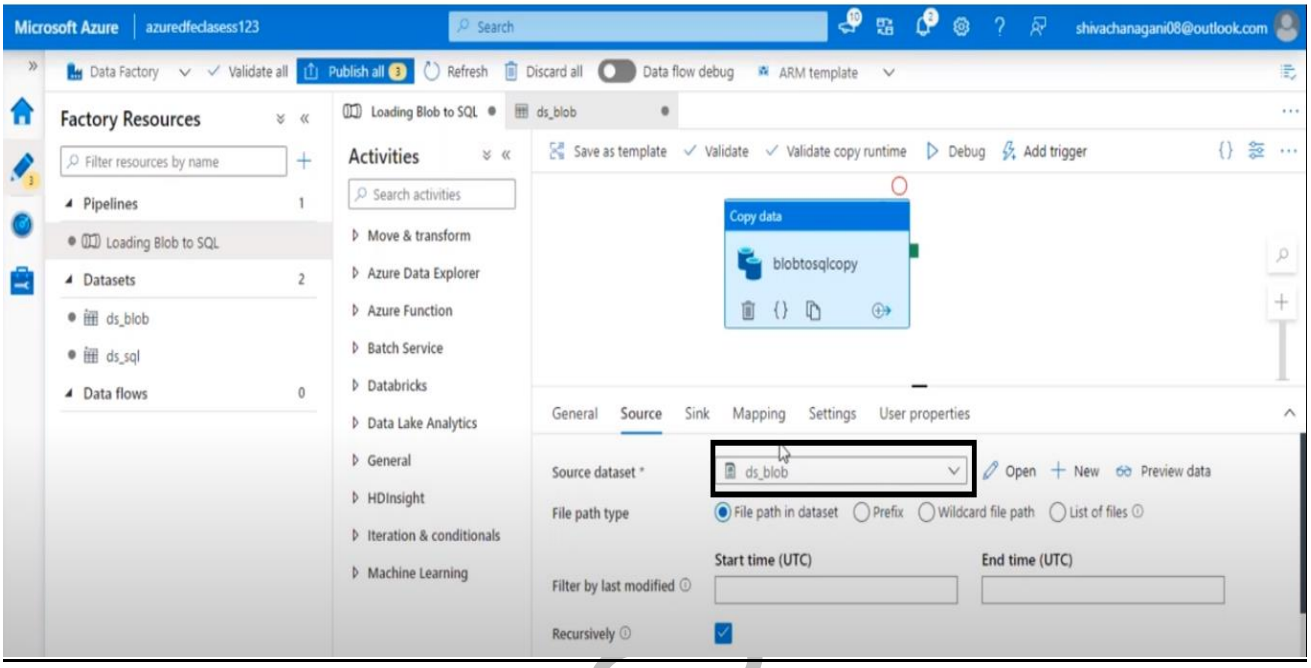
Step 17: Click on Azure SQL Database and Continue



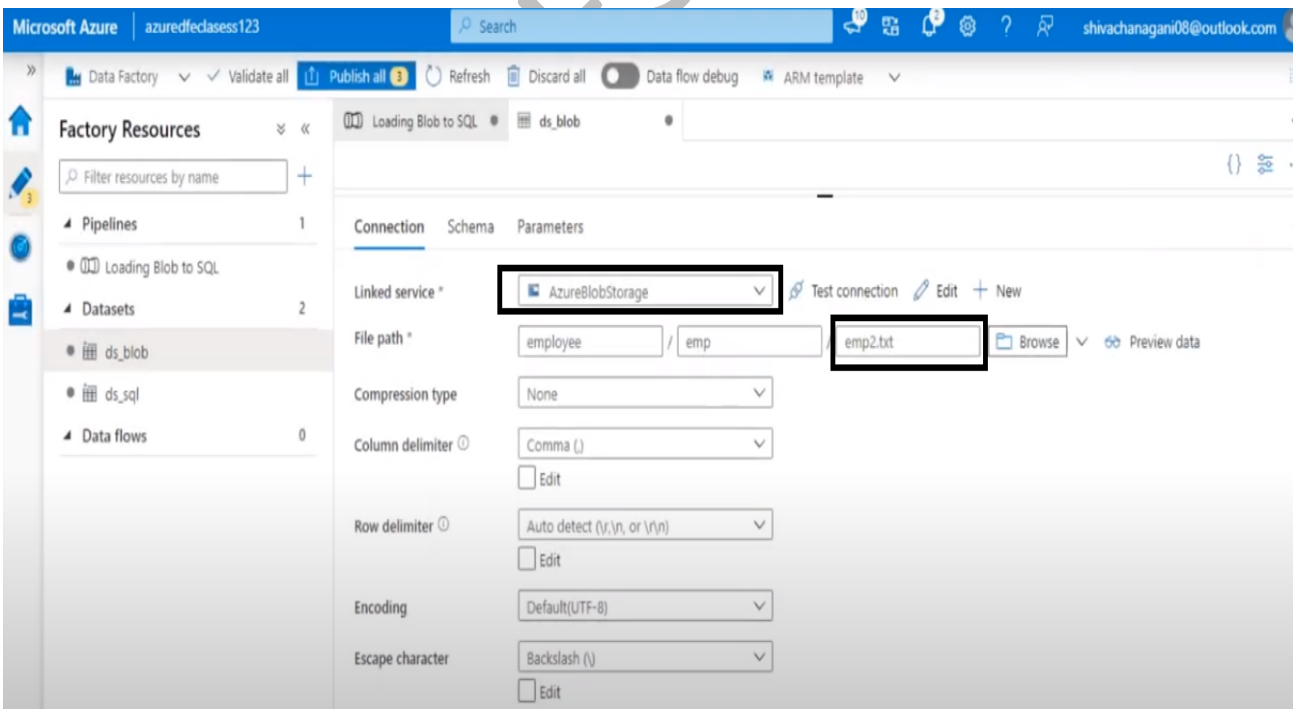
Step 18: Give proper name on the name tab----Select the linked service—Select Table Name and click on ok



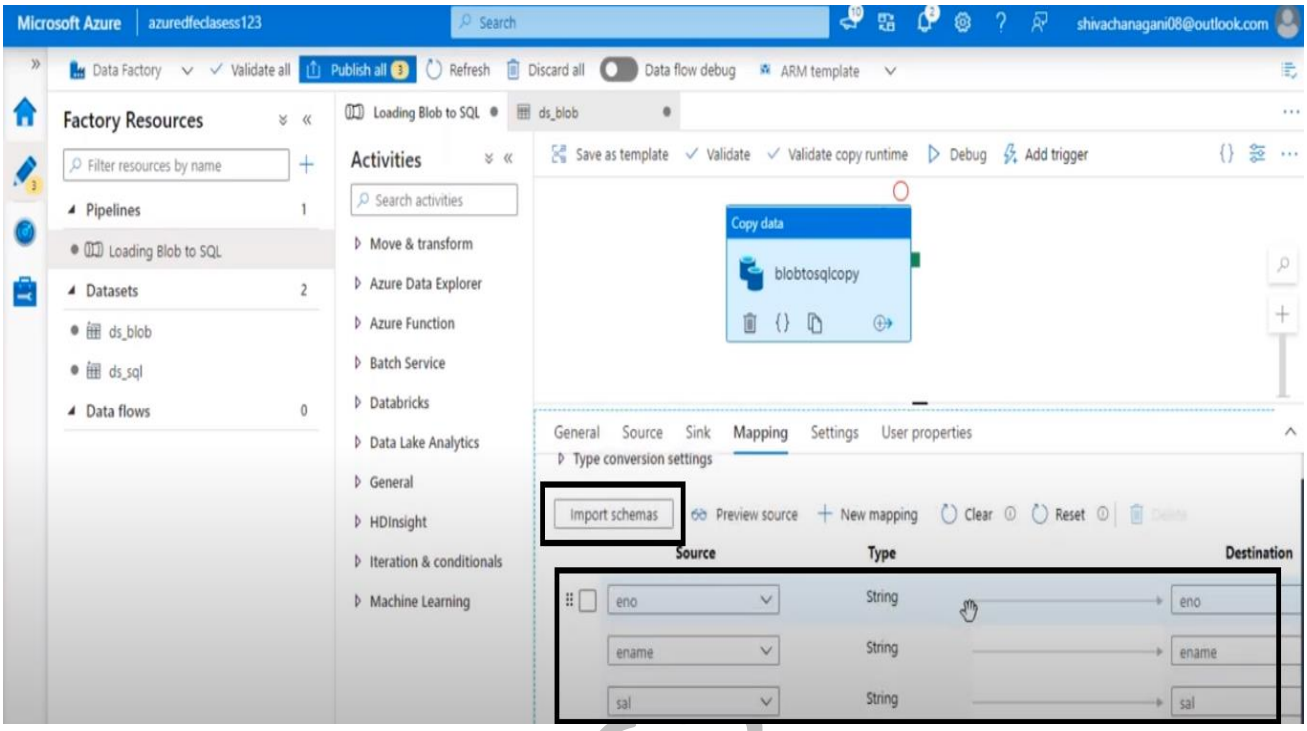
Step 19: Click on Source tab and select linked service for the blob storage which we already created earlier



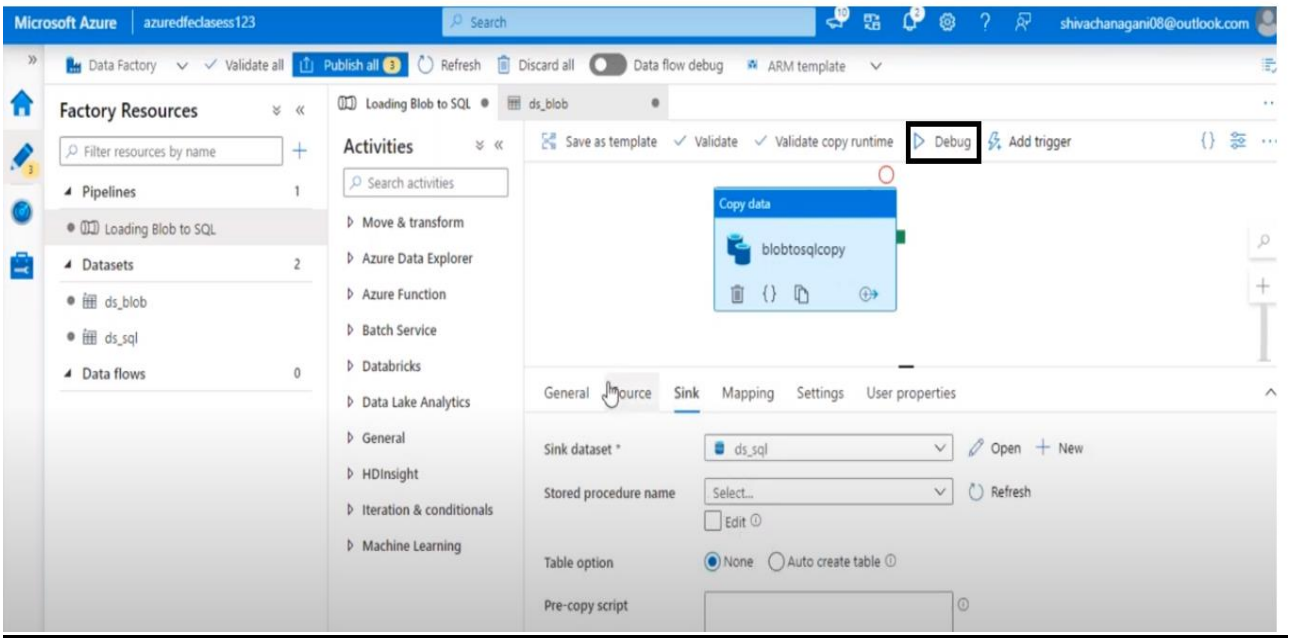
Step 20: Browse one file for mapping source to sink



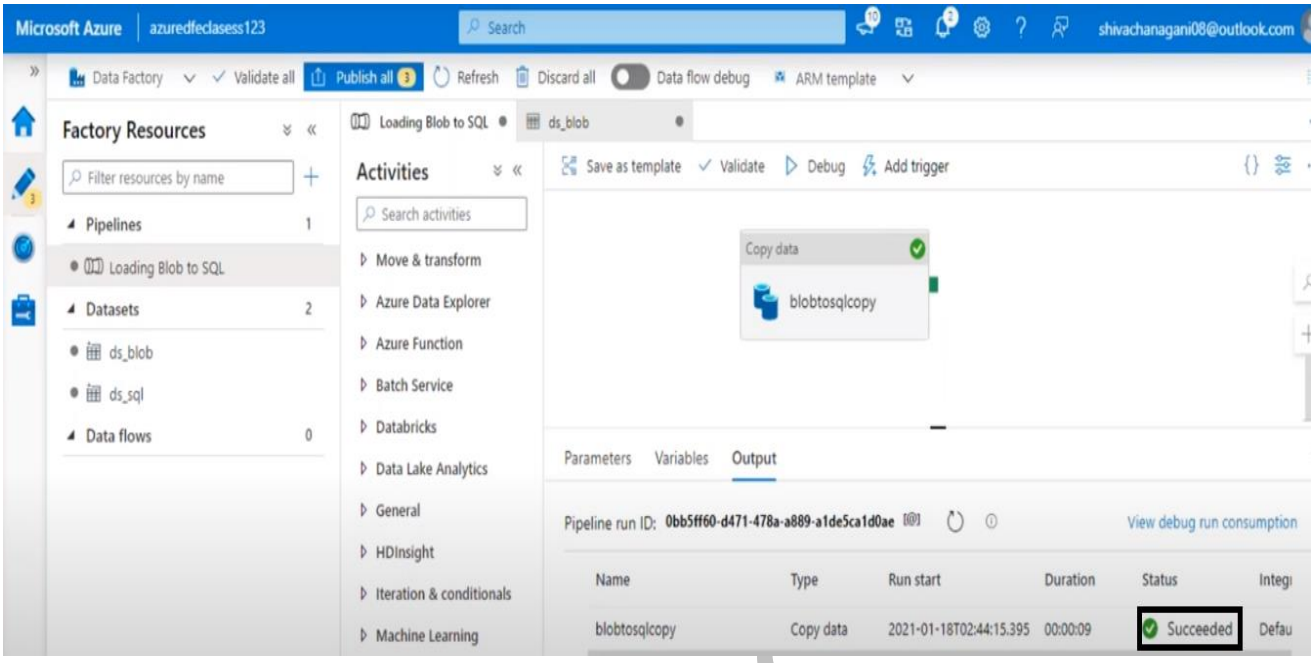
Step 21: Click on Mapping---Import Schemas---Do Mapping



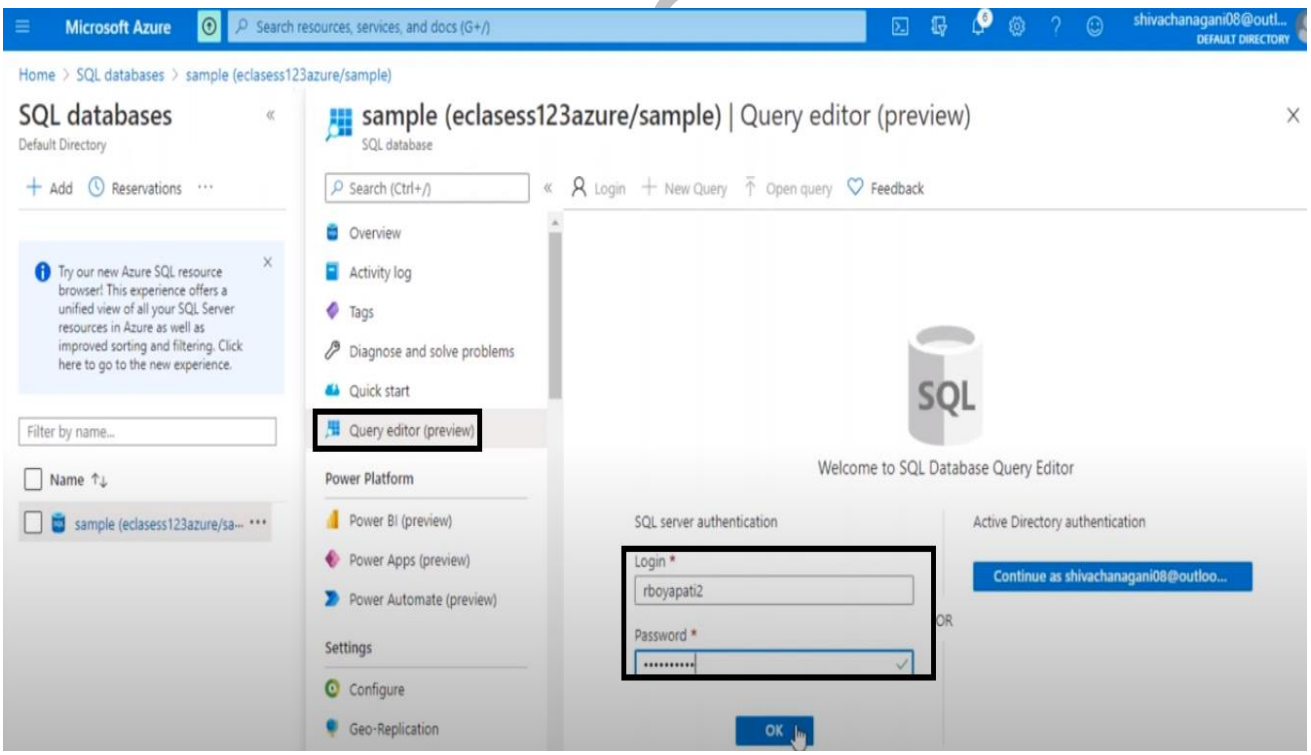
Step 22: Click on Debug



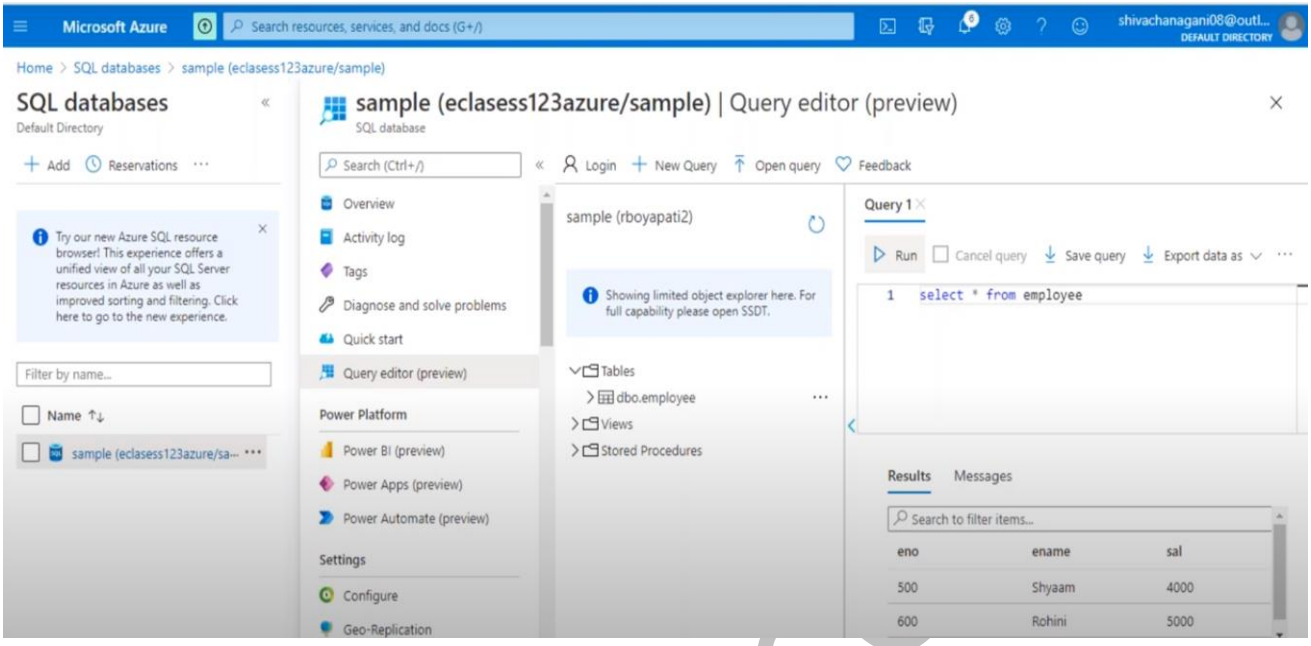
Step 23: Check the status of the copy activity



Step 24: Go to Query Editor----enter login credentials

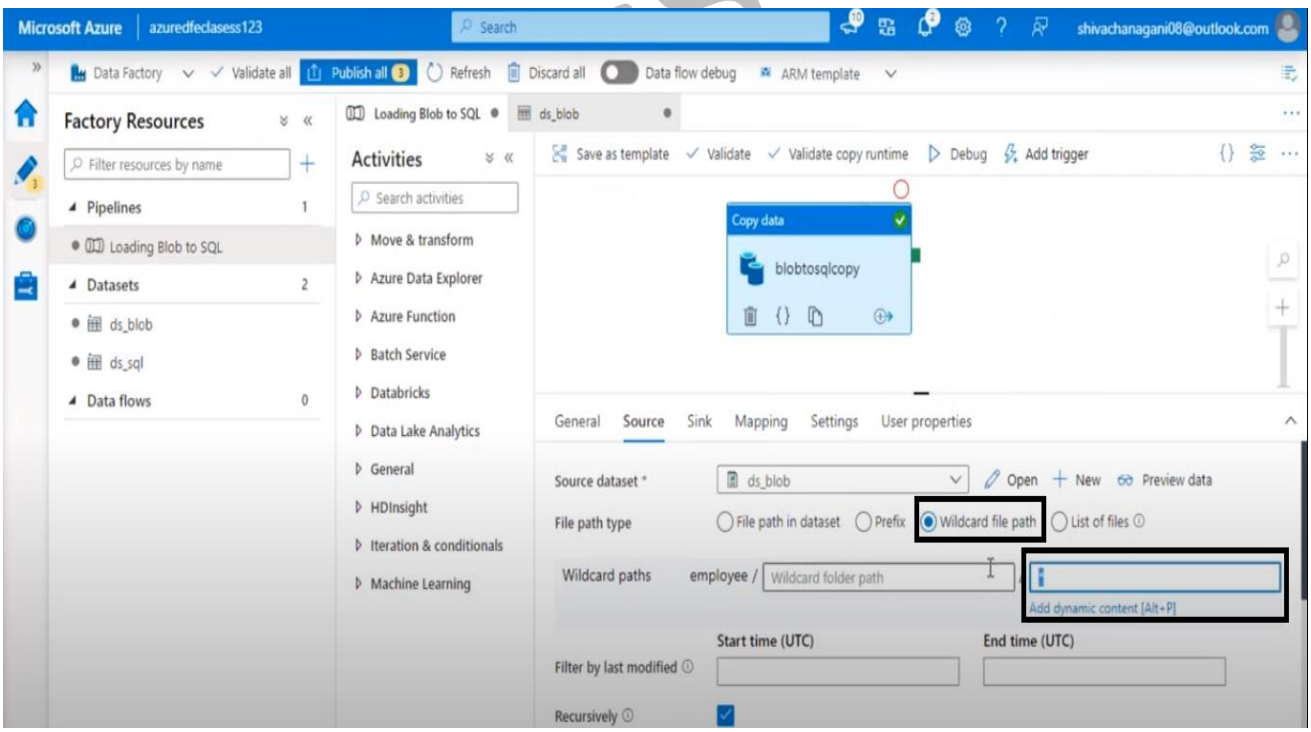


Step 25: Query the SQL to retrieve the data to check whether blob storage data loaded successfully or not

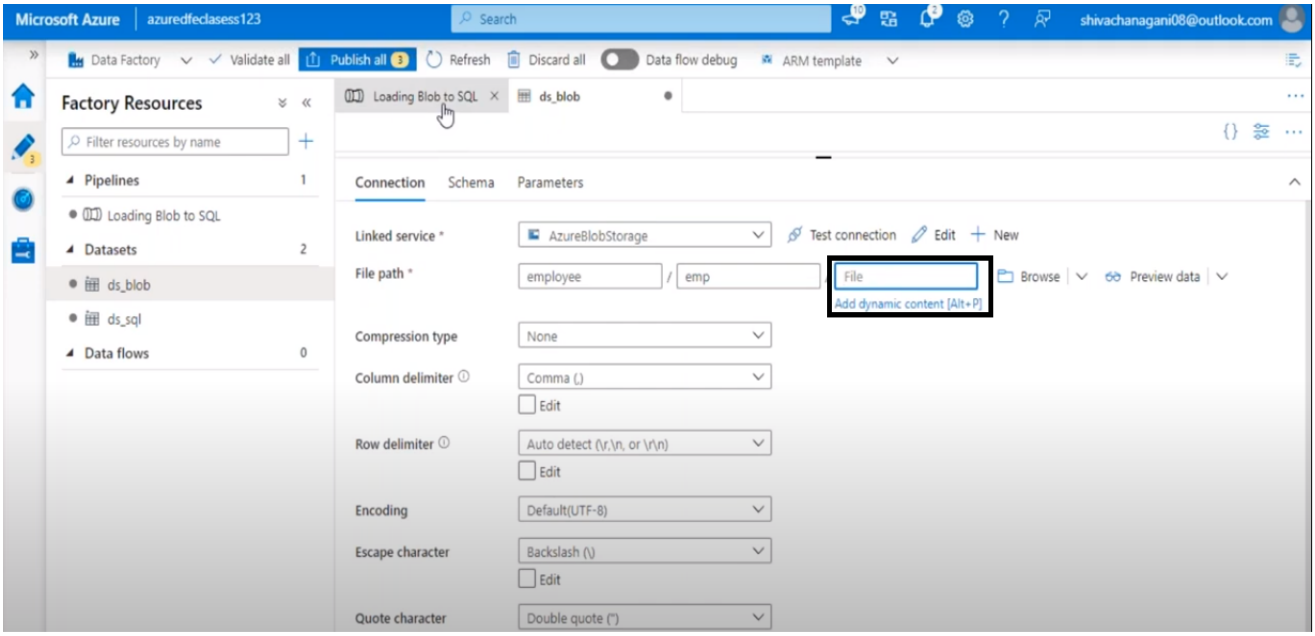


Step 26:

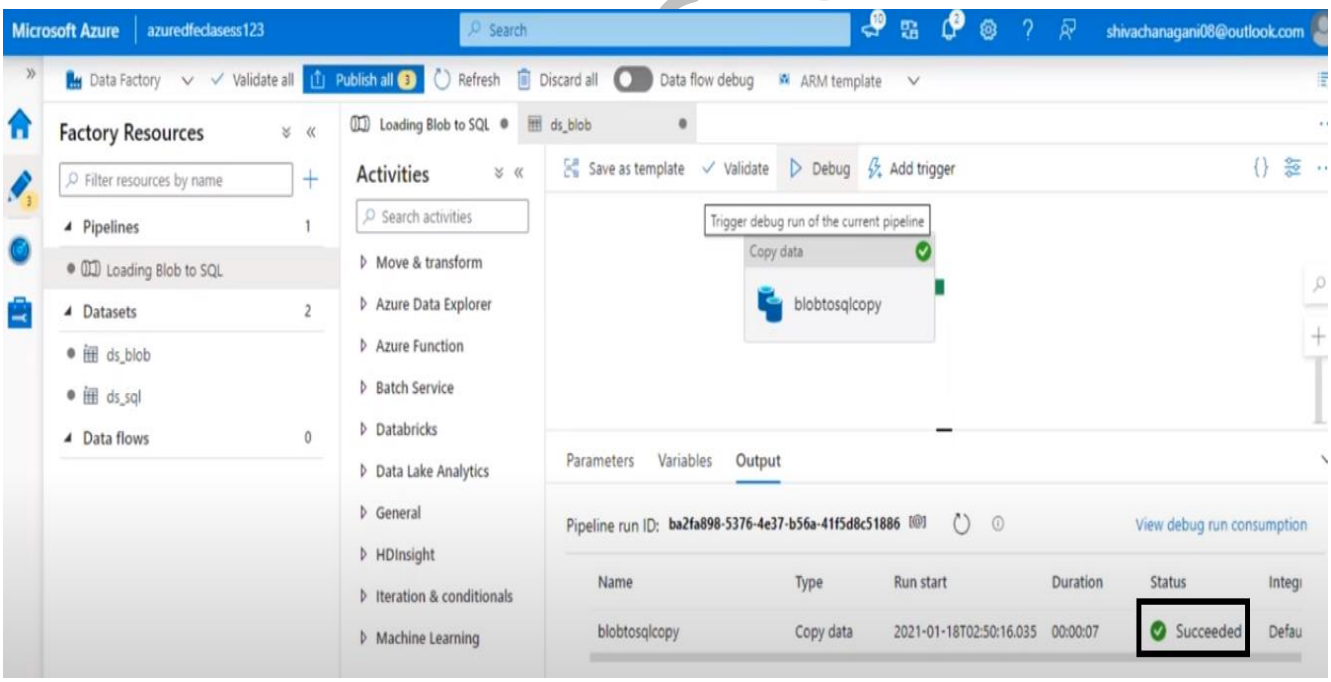
To load the all the files in the blob storage we need to go pipeline---select wild card path----Keep *



Step 27: Go to source data set---keep file tab as empty



Step 28: Click on debug and check the status



Loading Data from On Premises SQL Server to Azure SQL Database using Self Hosted Integration runtime:

Prerequisites:

SQL Server needs to be installed in on premises machine.

Create Database in On Premises SQL Server

Create Sample Table and insert some records in Source

Create Table EMP (Eno int, Ename Varchar (100), Sal int)

Insert into EMP Values (100,'Ram', 1000)

Insert into EMP Values (200,'Vihaan', 2000)

Create Azure SQL Database.

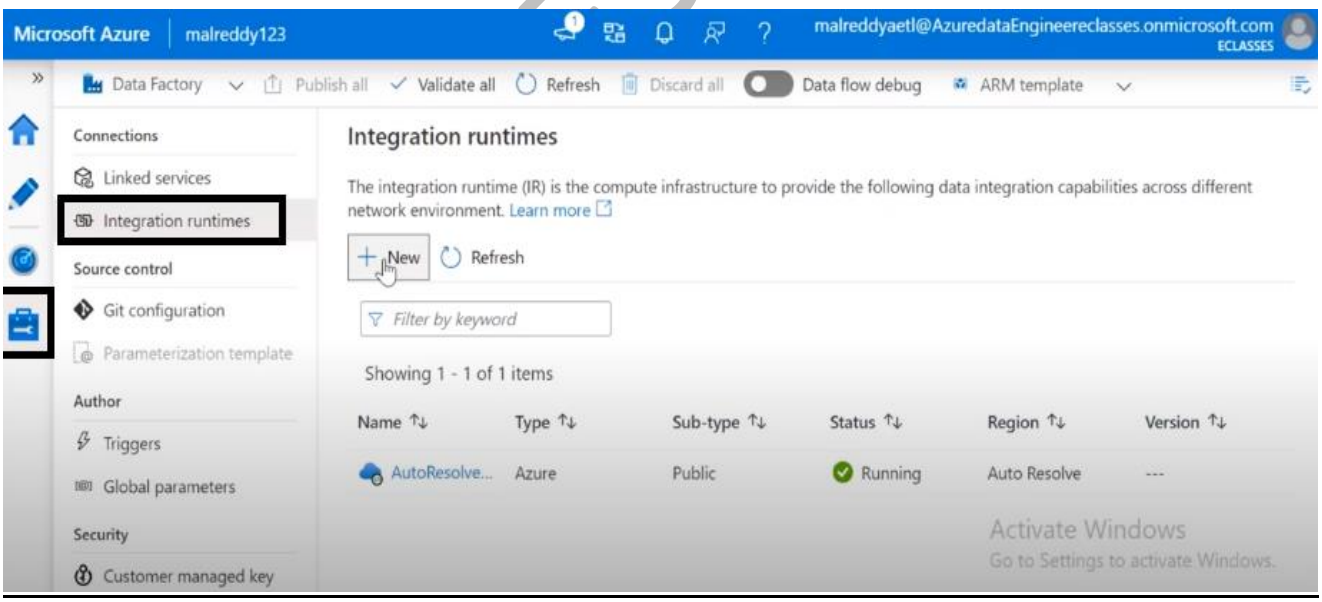
Create employee Table by using below script in Azure SQL Database.

Create Table EMP (Eno int, Ename Varchar (100), Sal int)

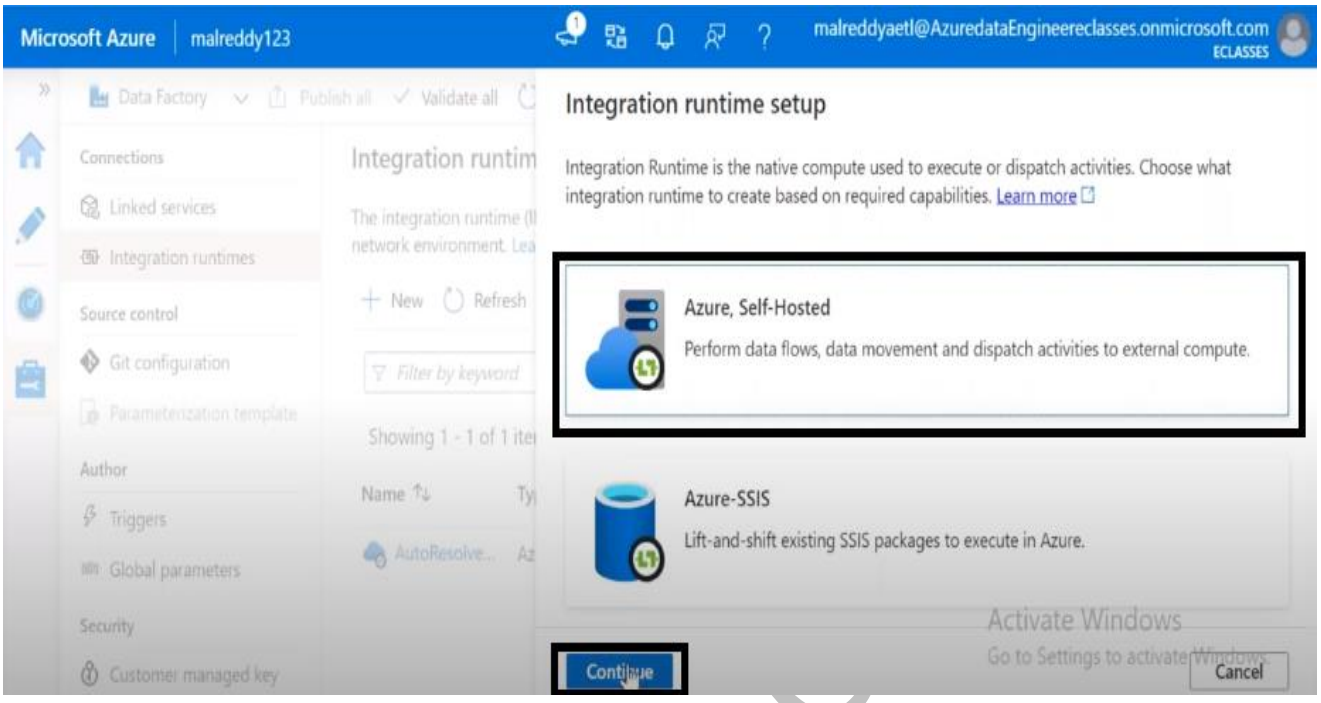
We need to install Self Hosted Integration Runtime for loading the data from on premises or any private virtual networks to Azure cloud.

Configuration of Self Hosted Integration Runtime:

Step 1: Go to manage window----Click on Integration runtimes----Click on New

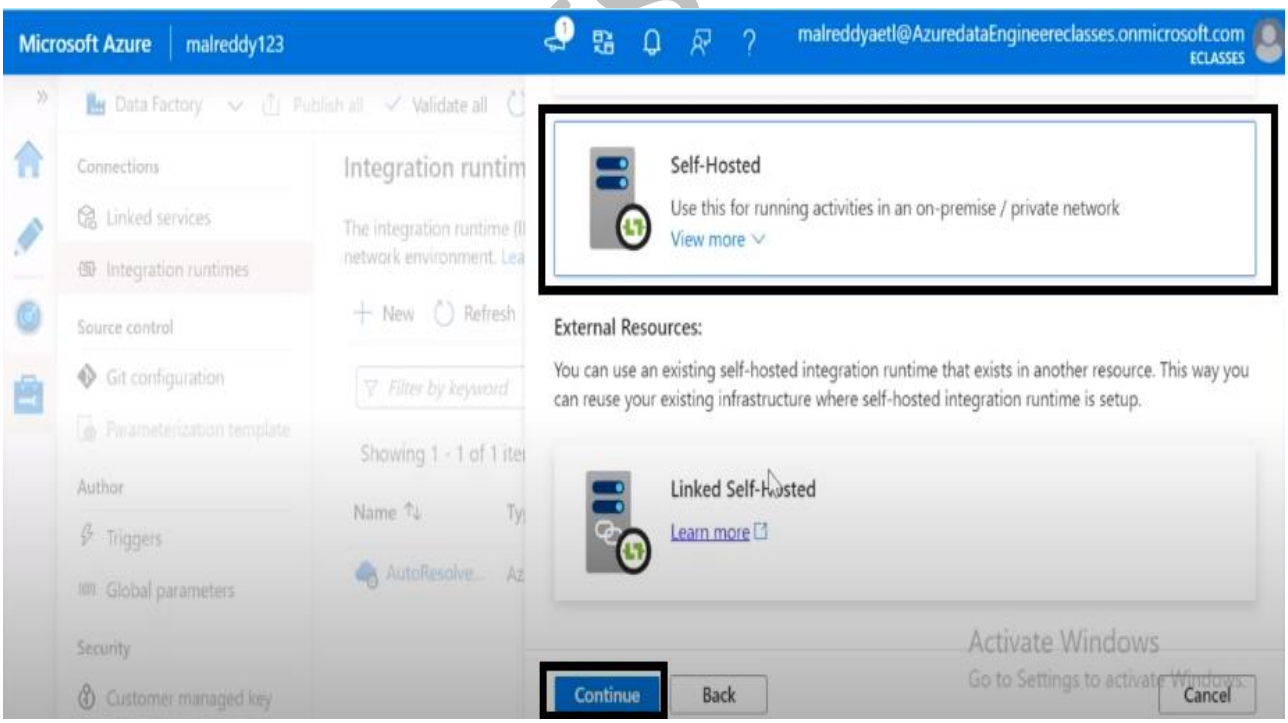


Step 2: Click on Azure, Self-Hosted and Continue

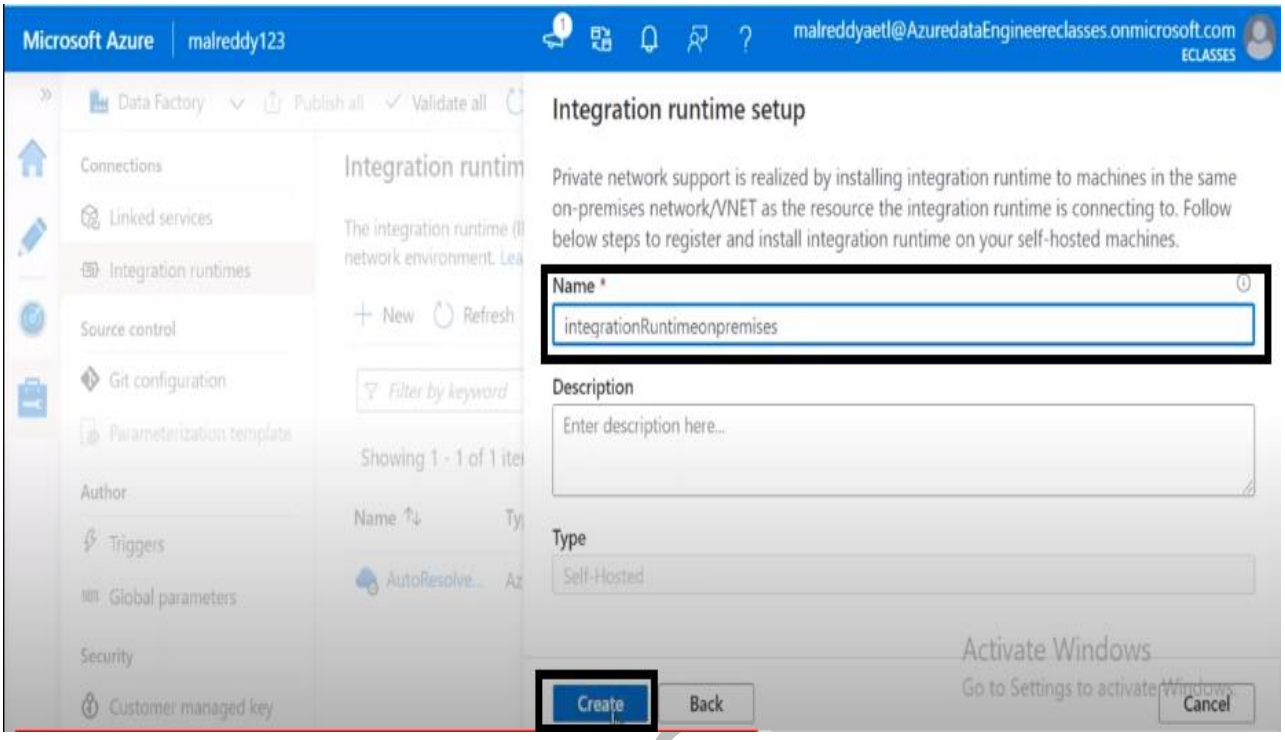


Step 3: Click on Self-Hosted and Continue

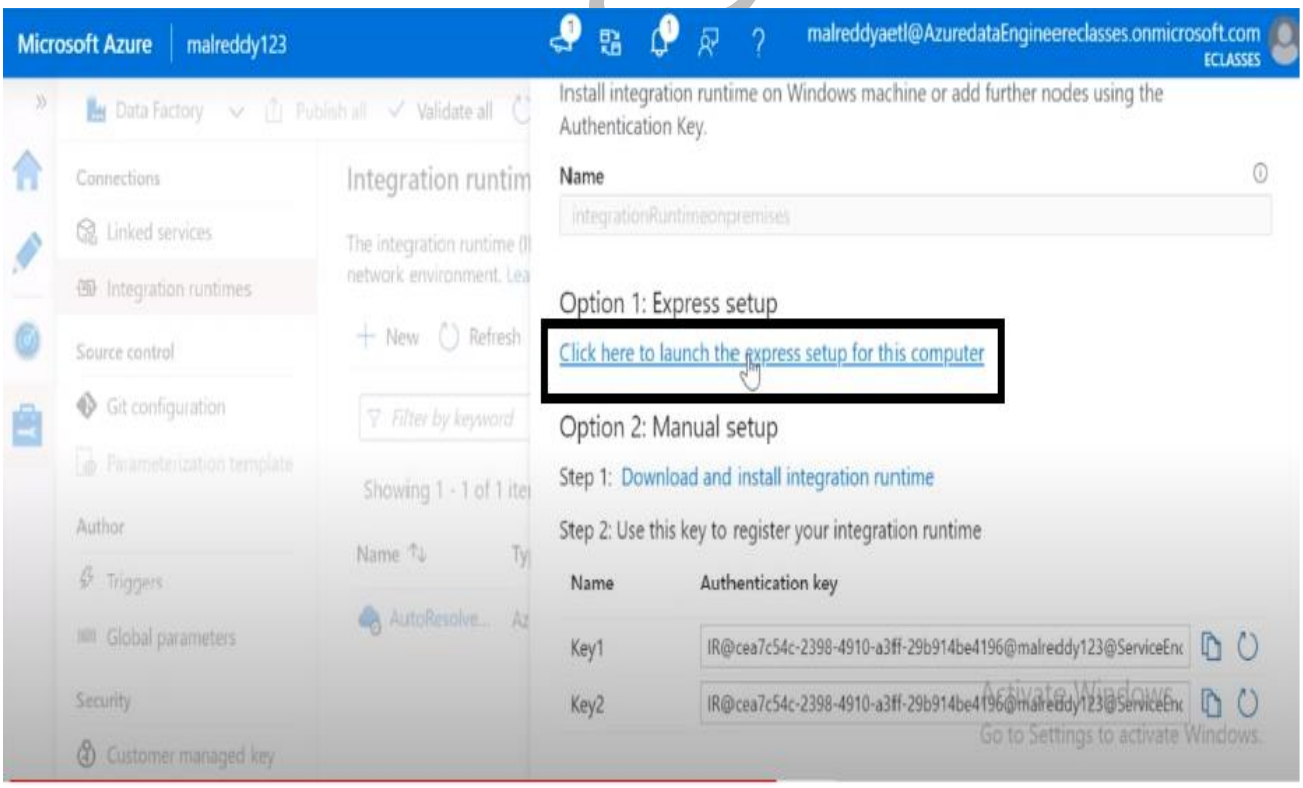
Note: We need select Linked Self-Hosted if it had already created by someone



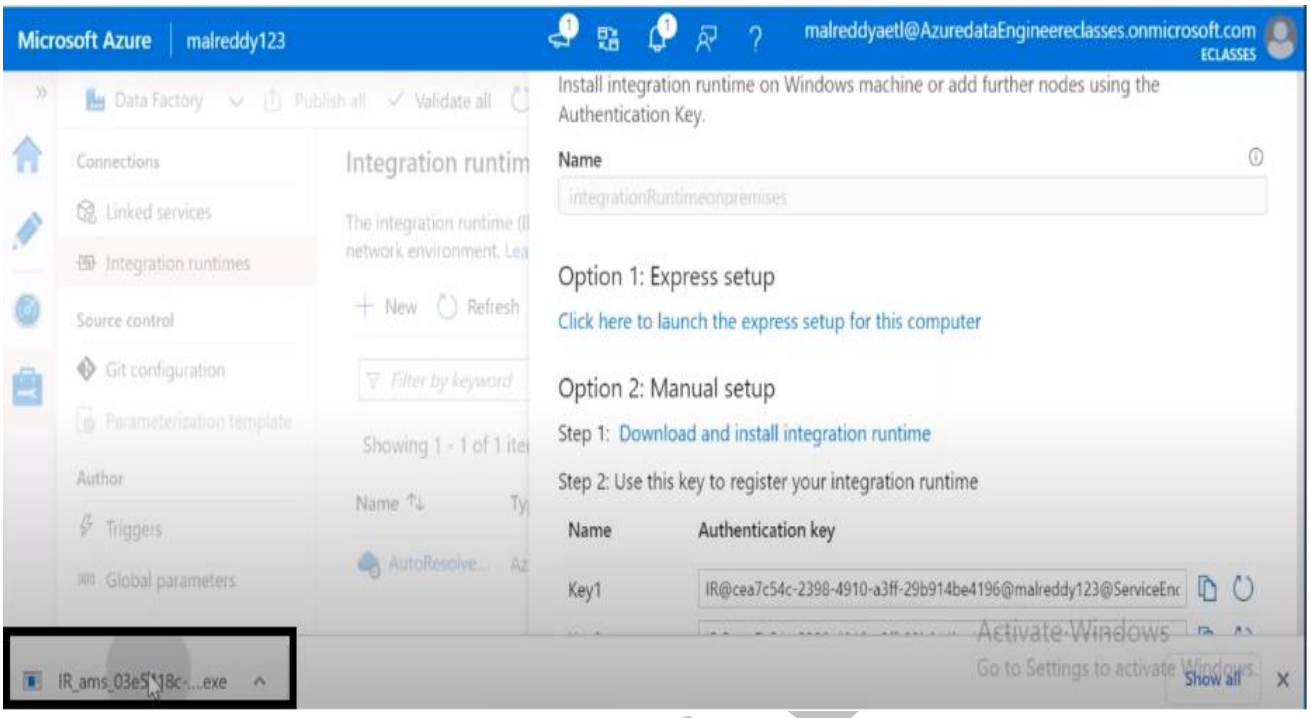
Step 4: Give a proper name and Click on Create



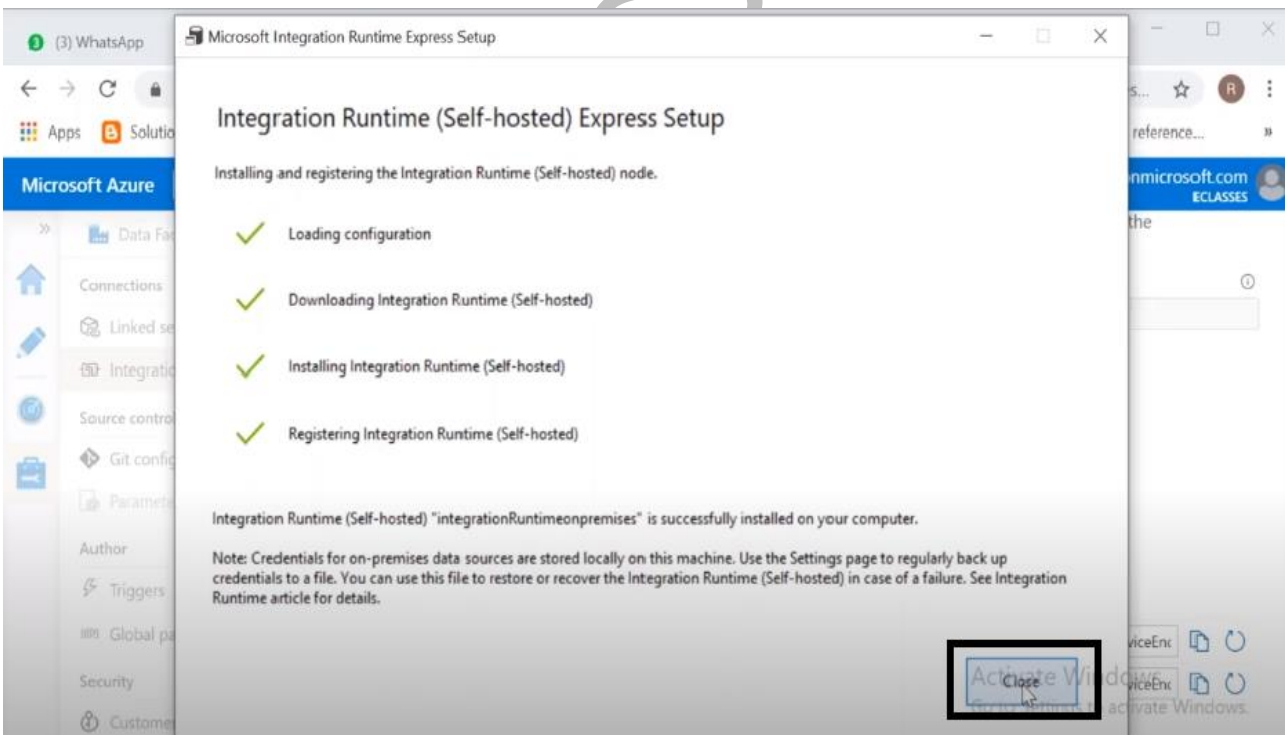
Step 5: Click on Express setup (Also we can manually install by clicking on Manual setup)



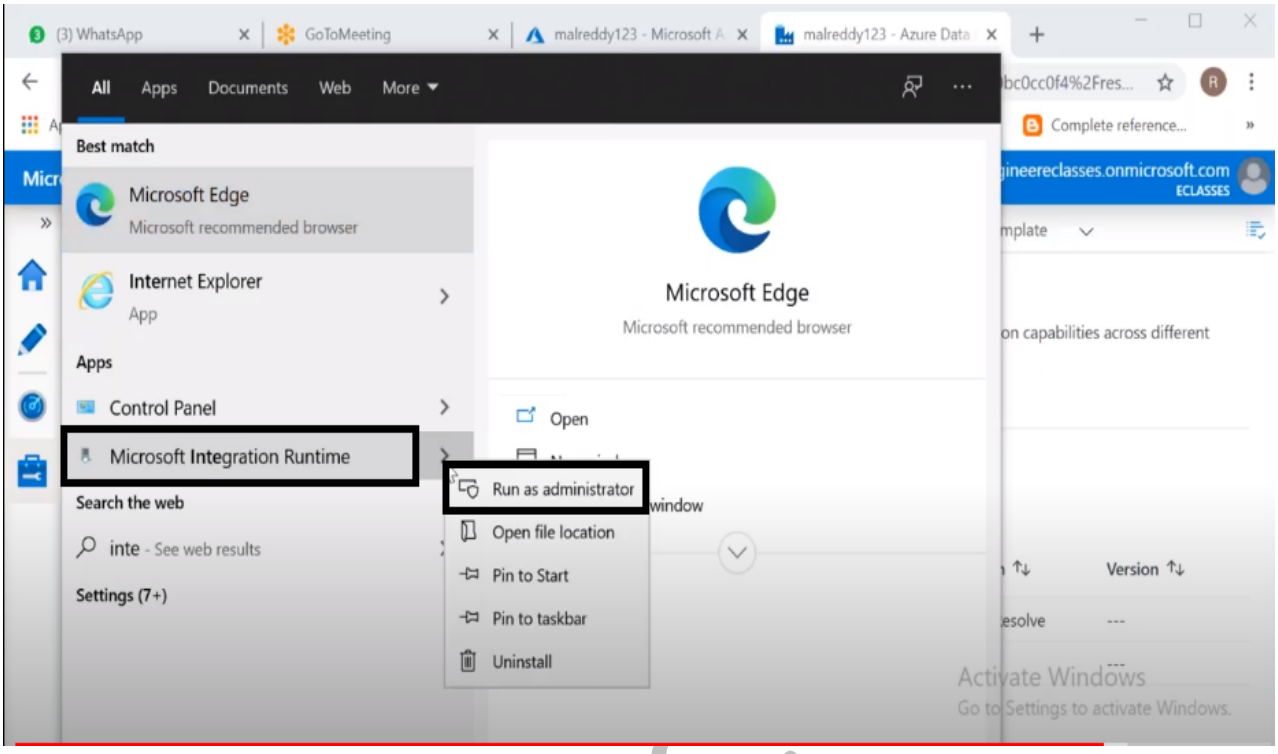
Step 6: Click on downloaded file



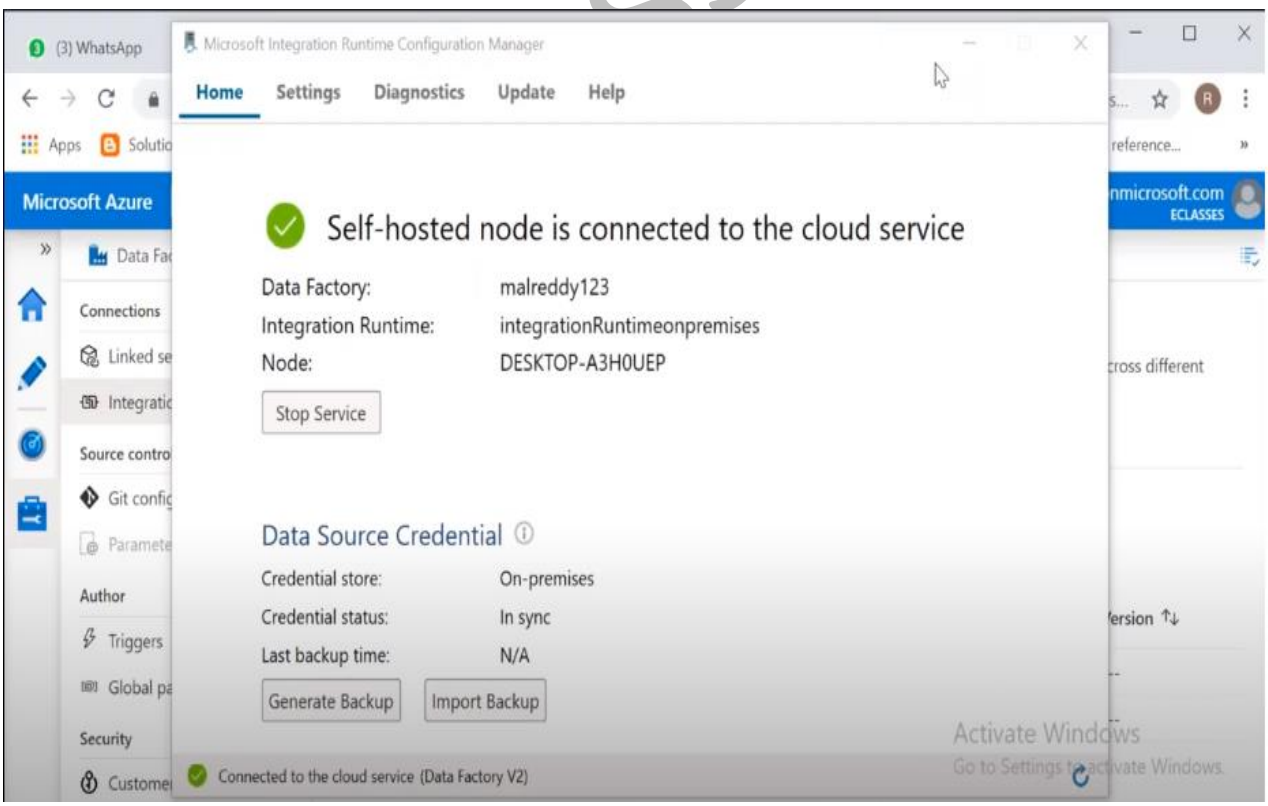
Step 7: Click on Close after installation



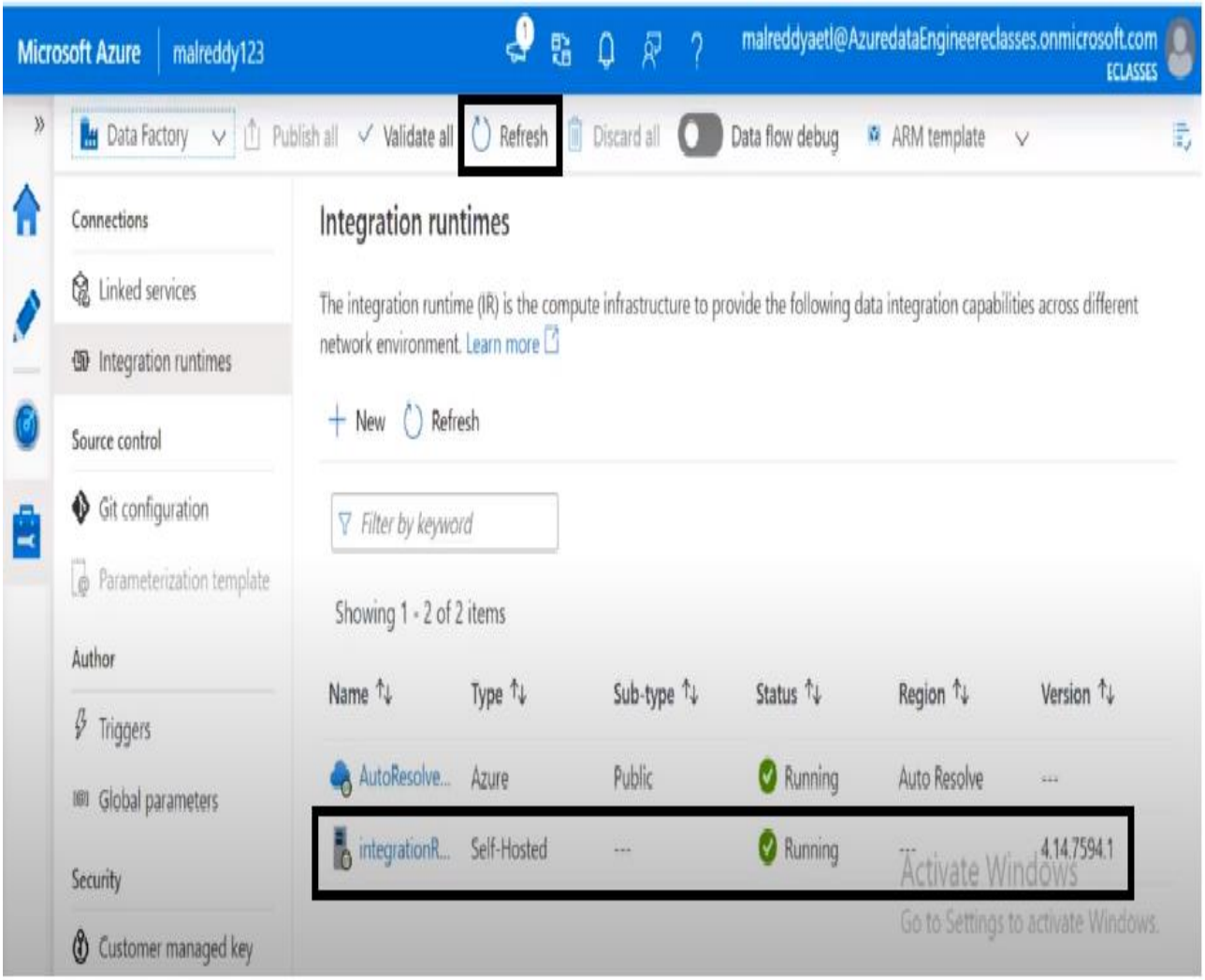
Step 8: Select Microsoft integration runtime----Run as administrator



Step 9: We will see below screen, minimize it



Step 10: Click on refresh or reload the page to check whether the self-hosted integration runtime successfully installed on not

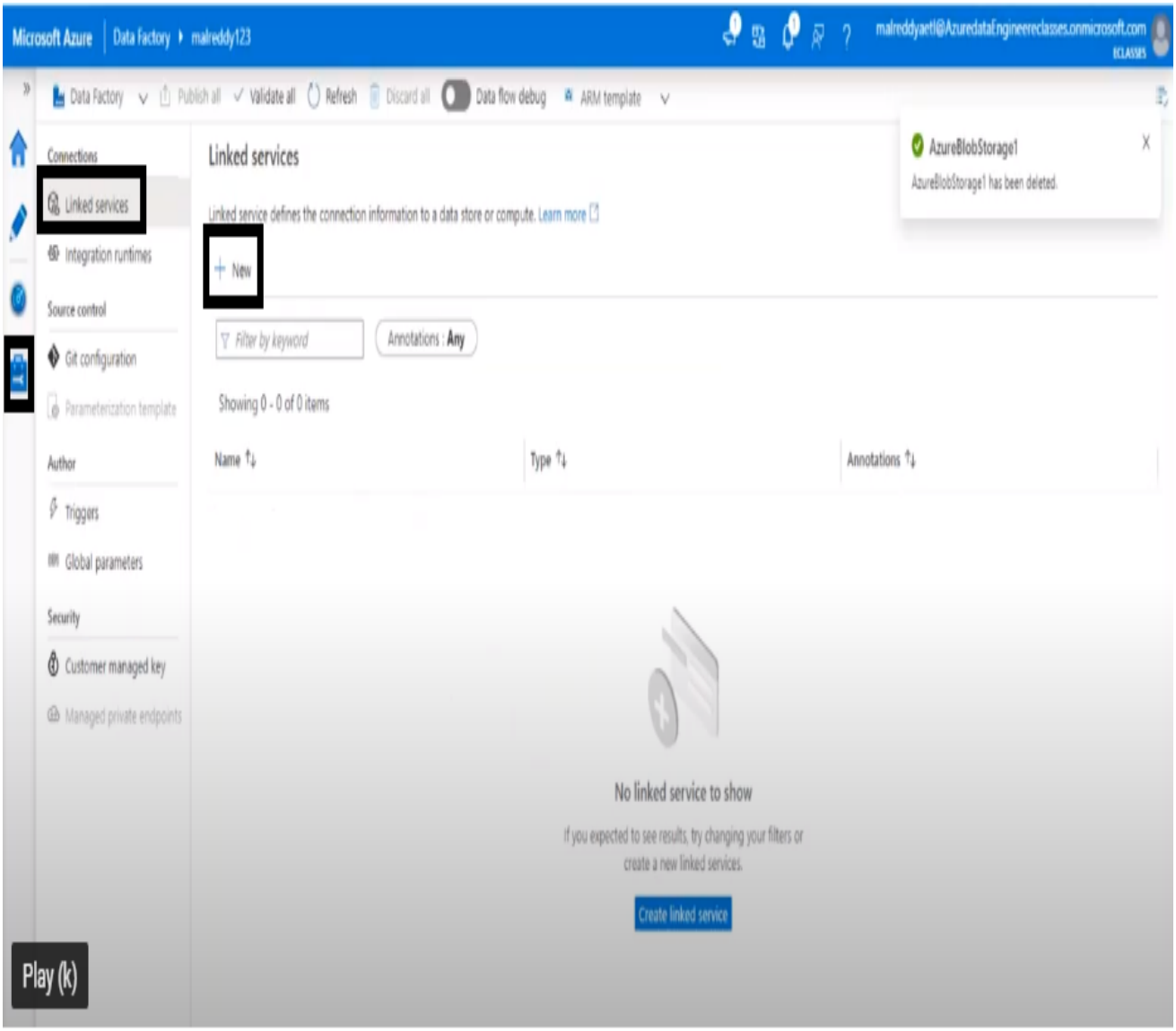


Loading data from on premises SQL server to azure SQL Database:

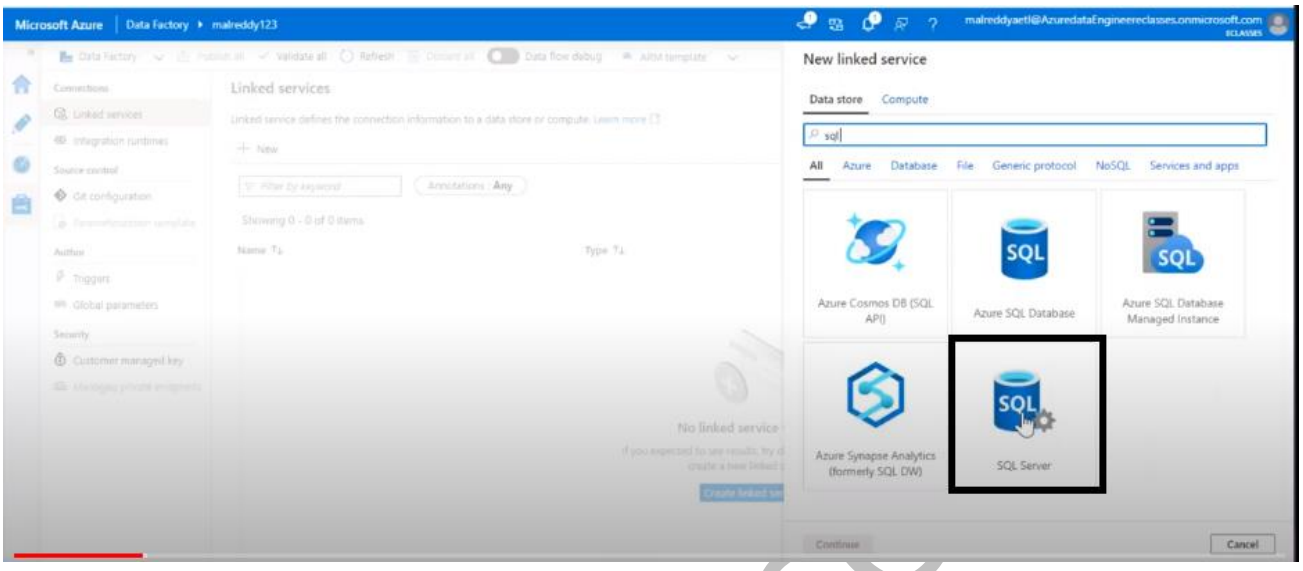
We can use Copy Activity for loading the data from on premises to Azure data factory.

Create a table inside azure SQL database to get the data from on premises SQL server.

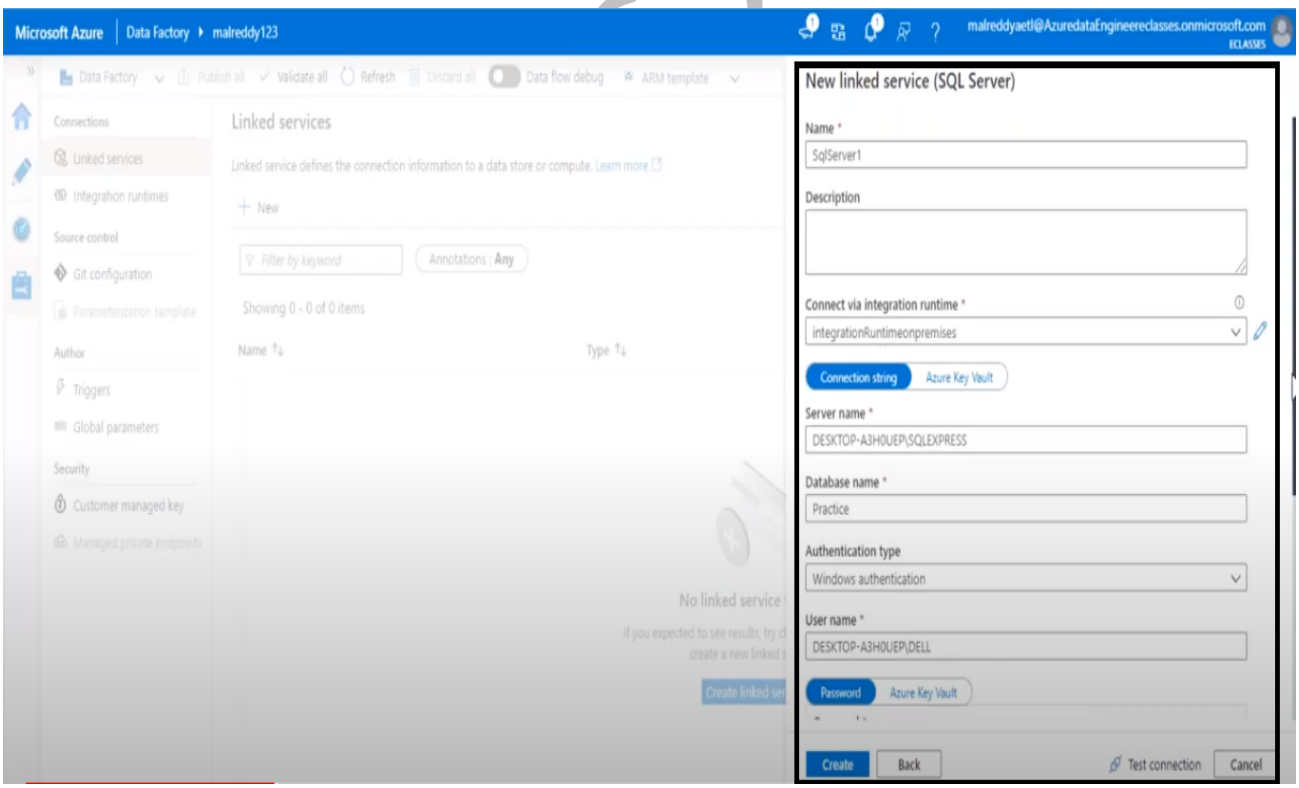
Step 1: Click on Manage window----Linked Services----New



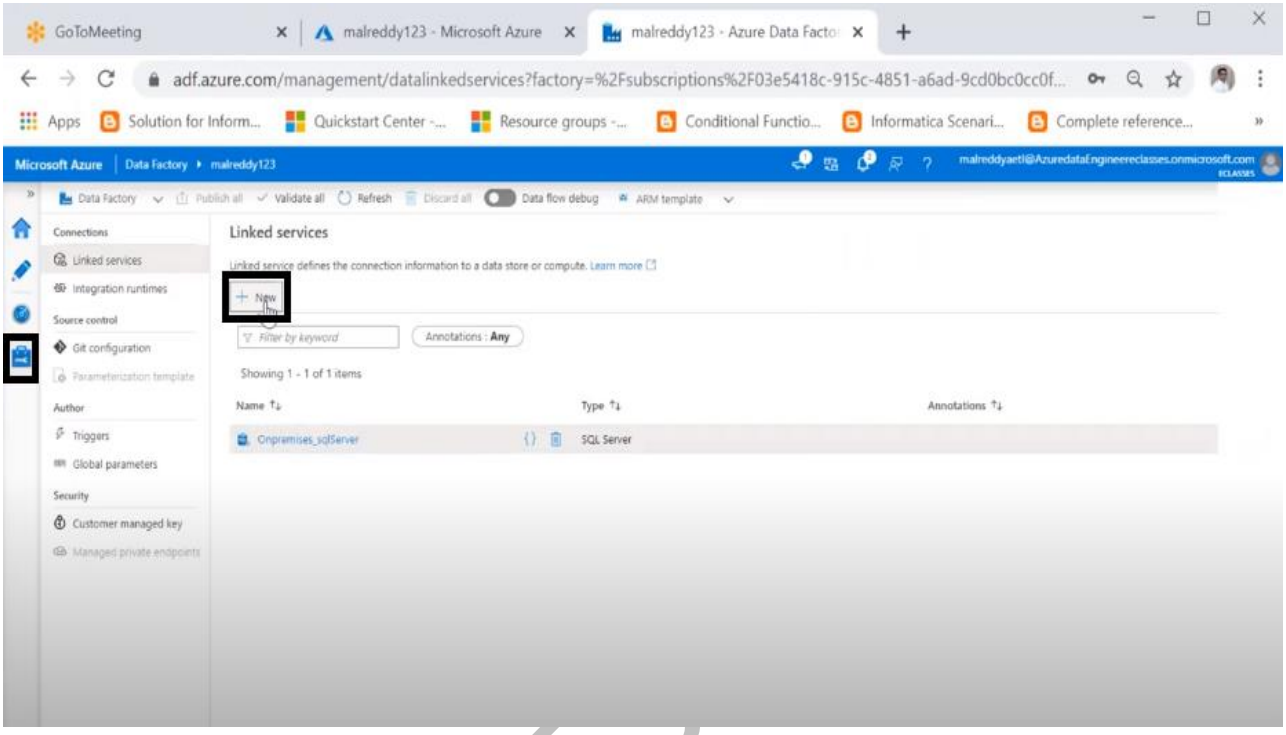
Step 2: Select SQL server



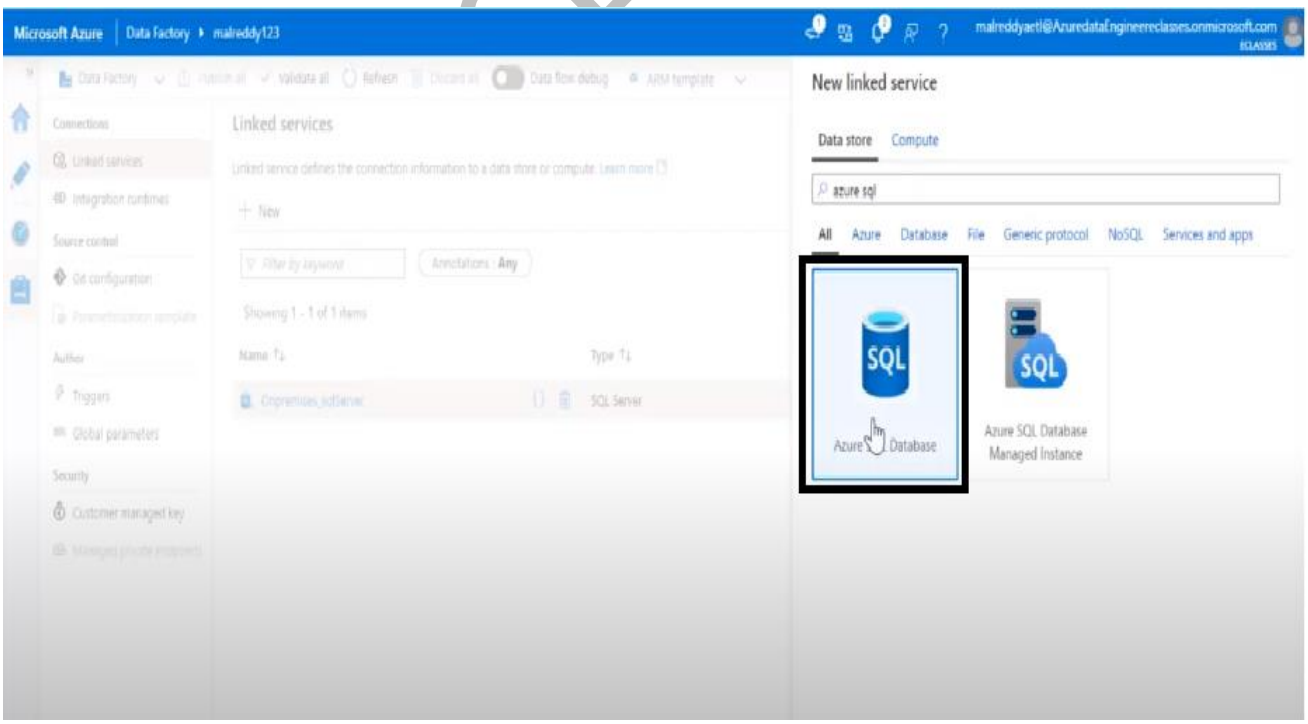
Step 3: Give a proper name----select self-hosted integration runtime----select server name and database name----select authentication type------give user name and password----Test the connection---If connection gets success then click on create.



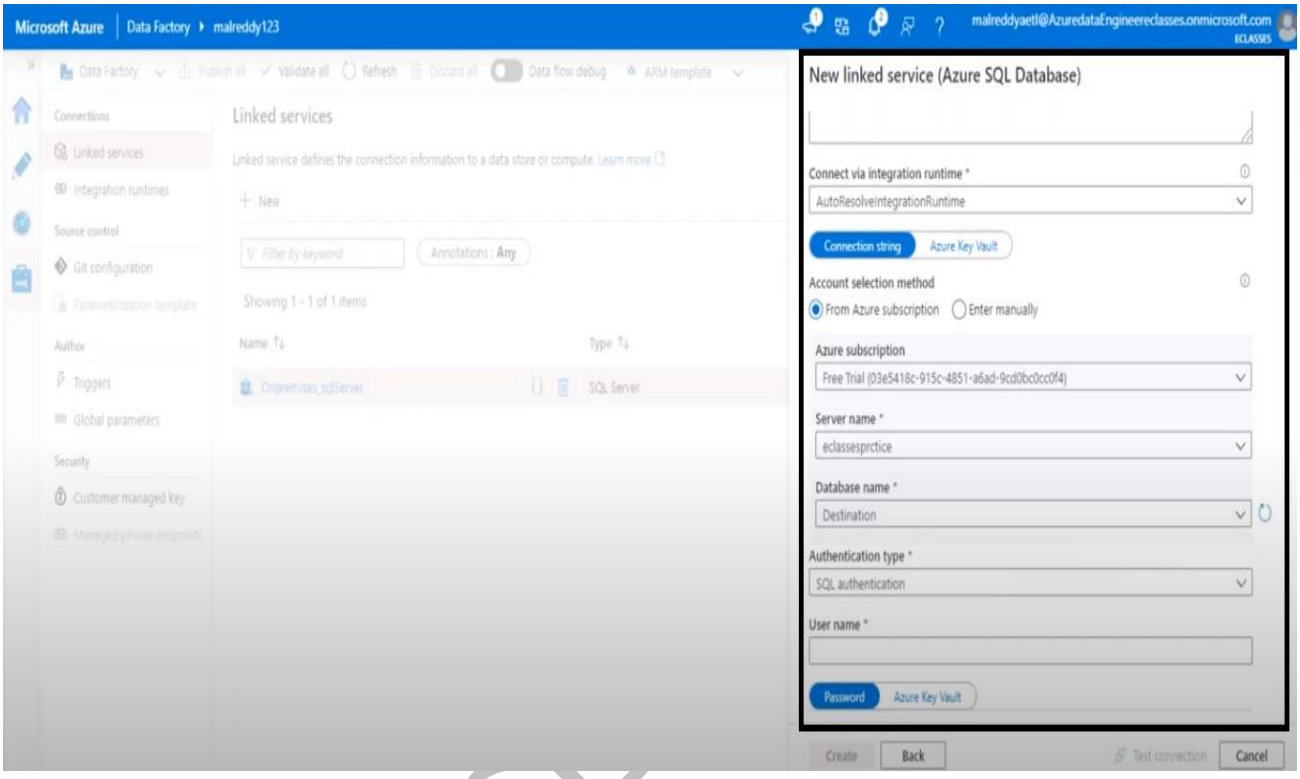
Step 4: Click on new for creating linked service for azure SQL database



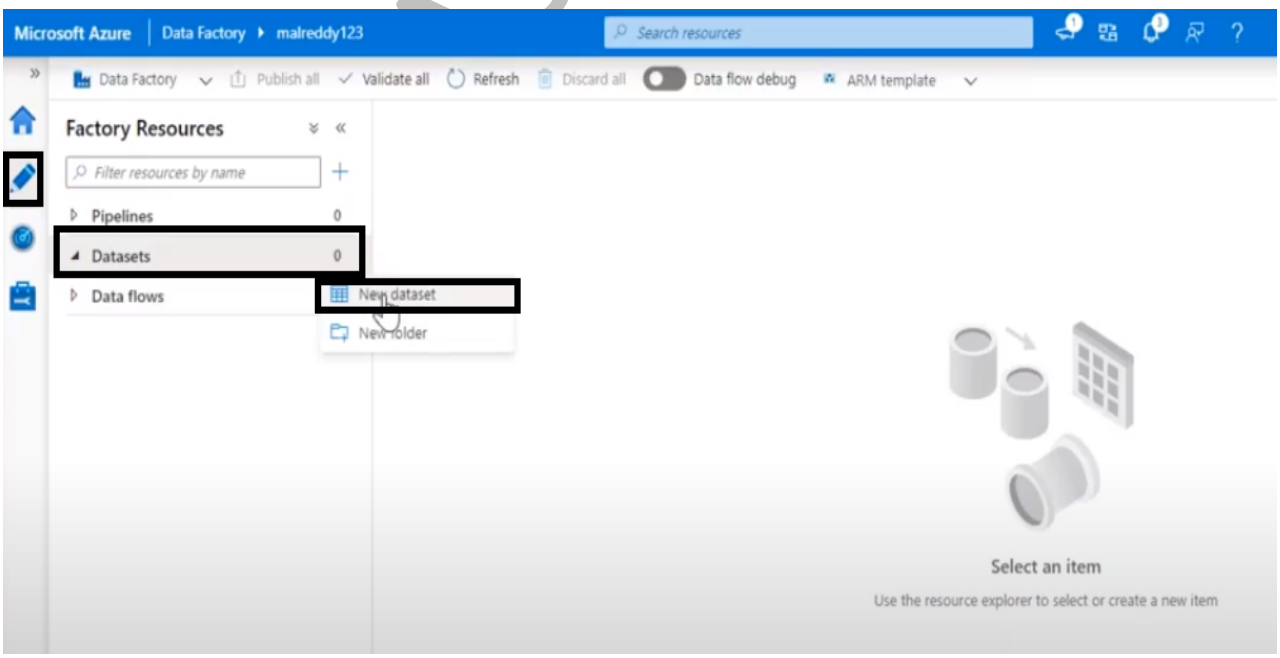
Step 5: Select Azure SQL Database



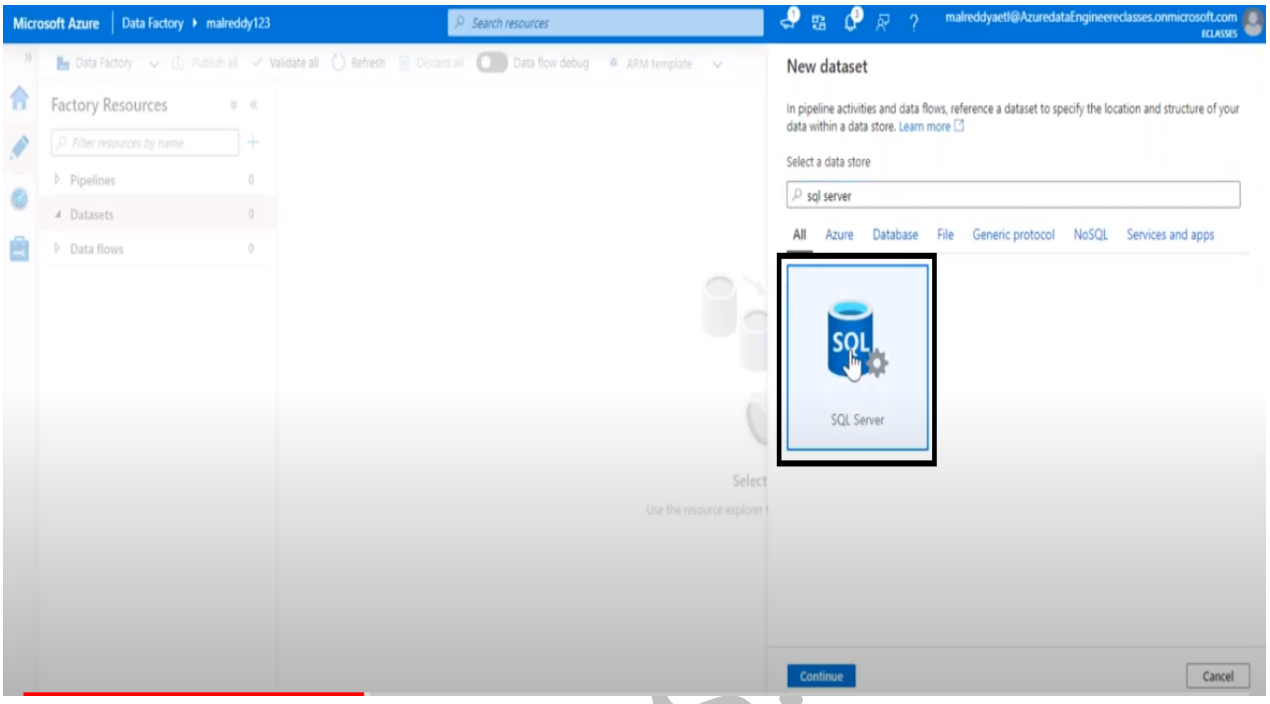
Step 6: Give a proper name----select auto resolve integration runtime----select subscription--select server name and database name----select authentication type------give user name and password----Test the connection---If connection gets success then click on create.



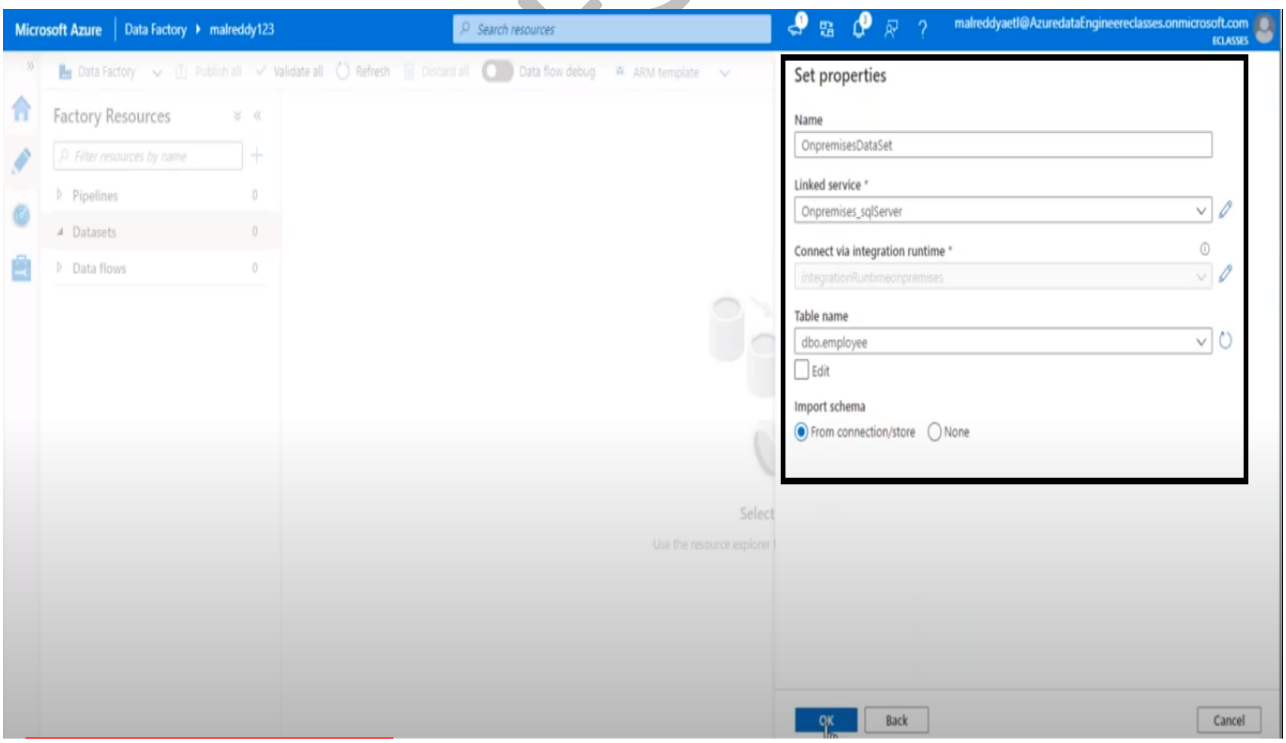
Step 7: Click on Author window-----Datasets---New dataset



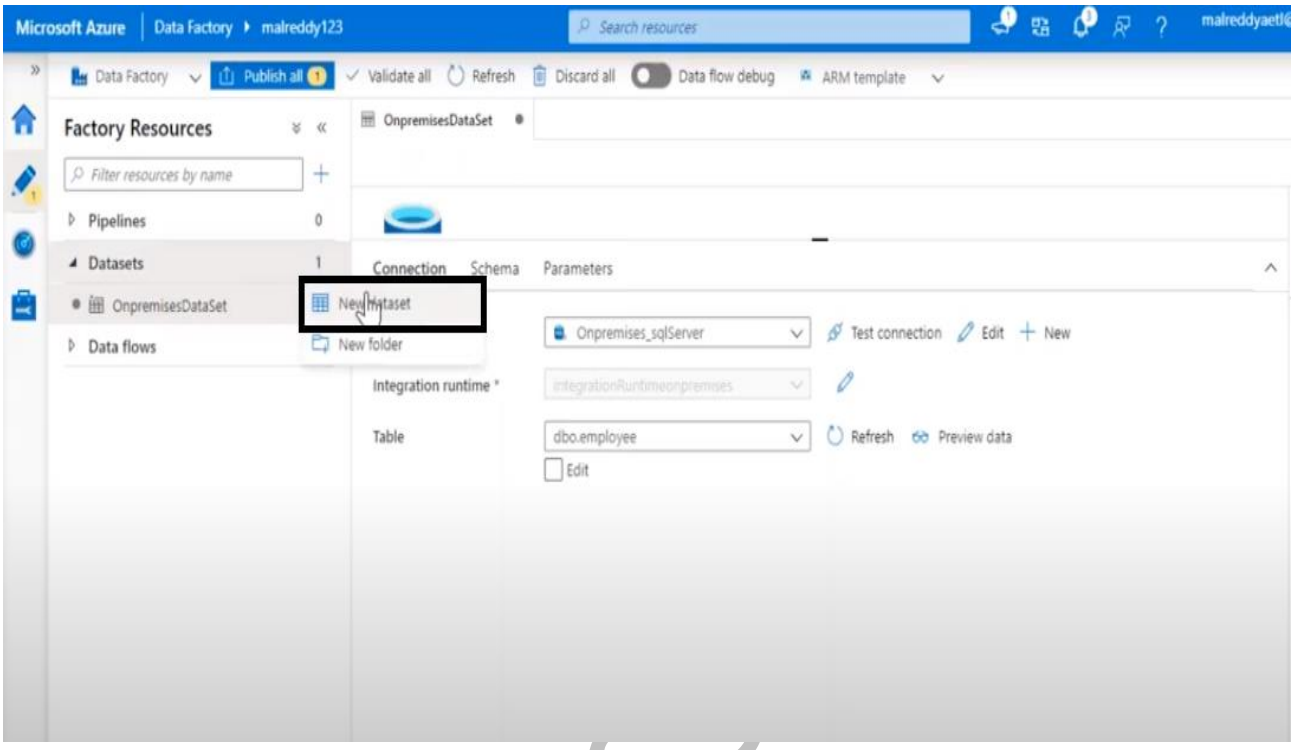
Step 8: Click on SQL Server and Continue



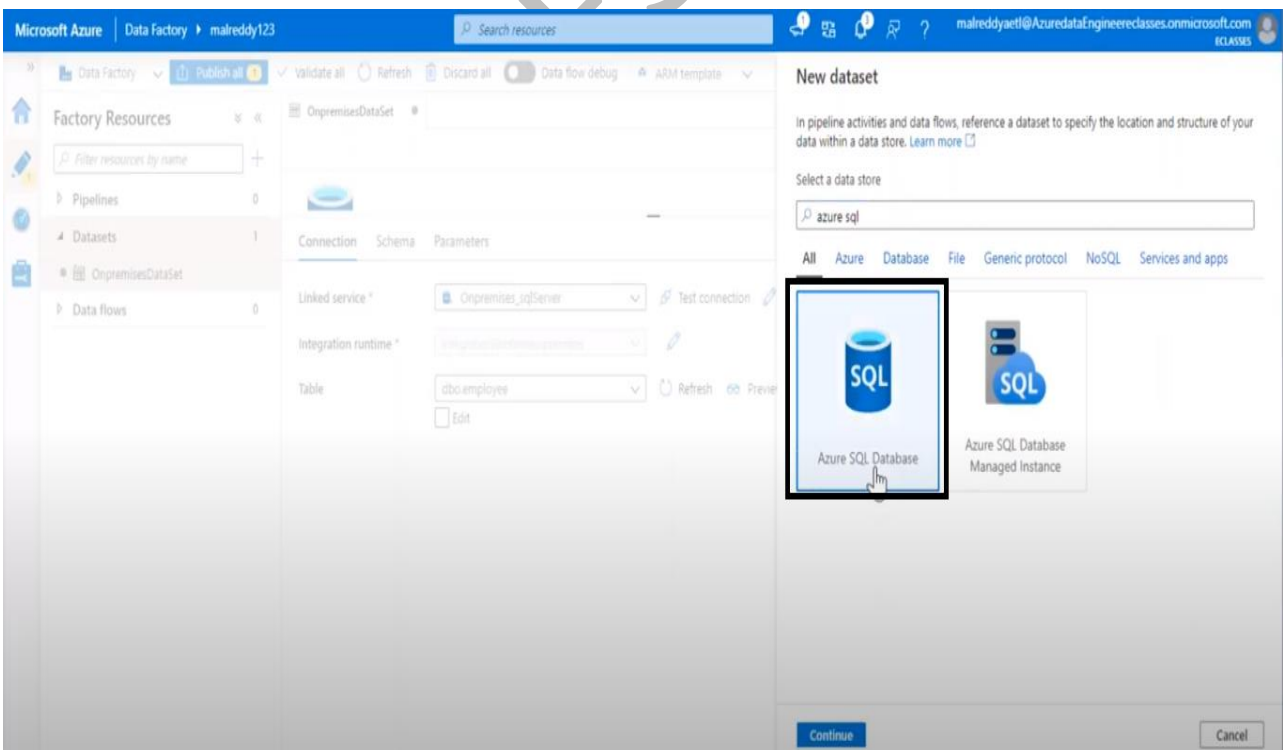
Step 9: Give a proper to the source dataset----Select Linked Service----Select Table Name---Click OK



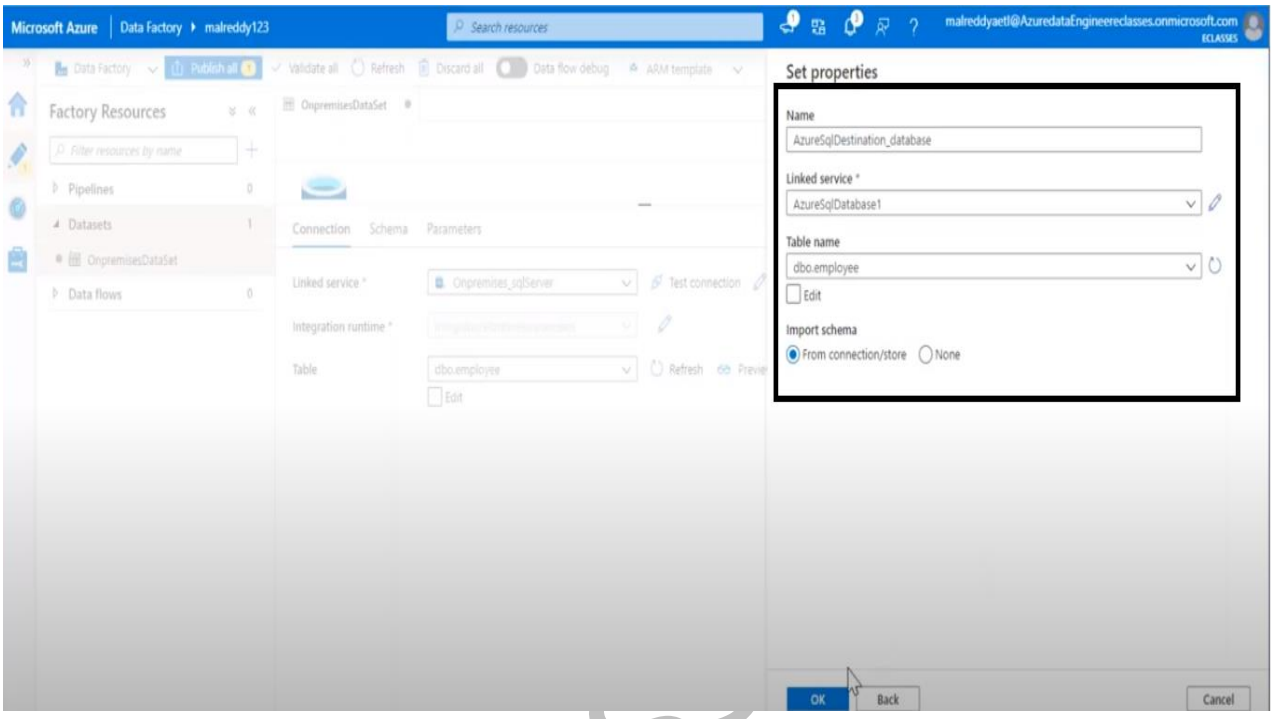
Step 10: Click on New dataset for creating the azure SQL database



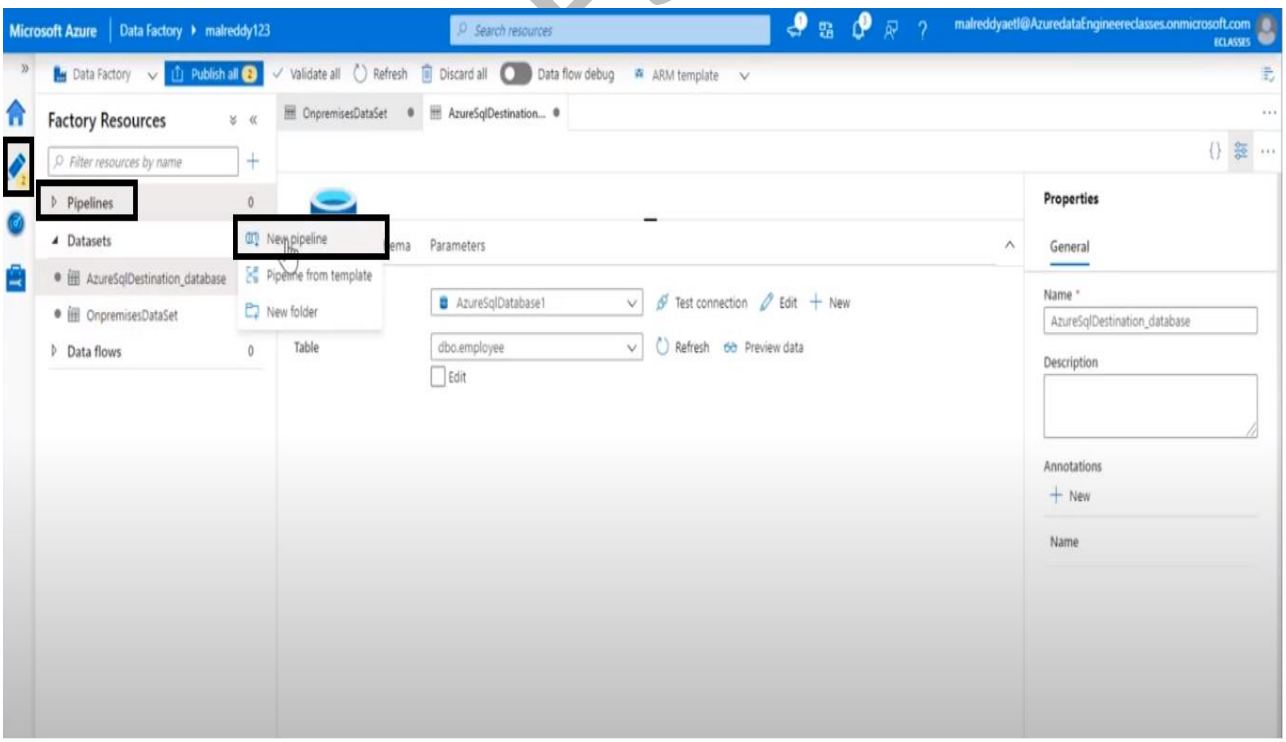
Step 11: Select Azure SQL database and Continue



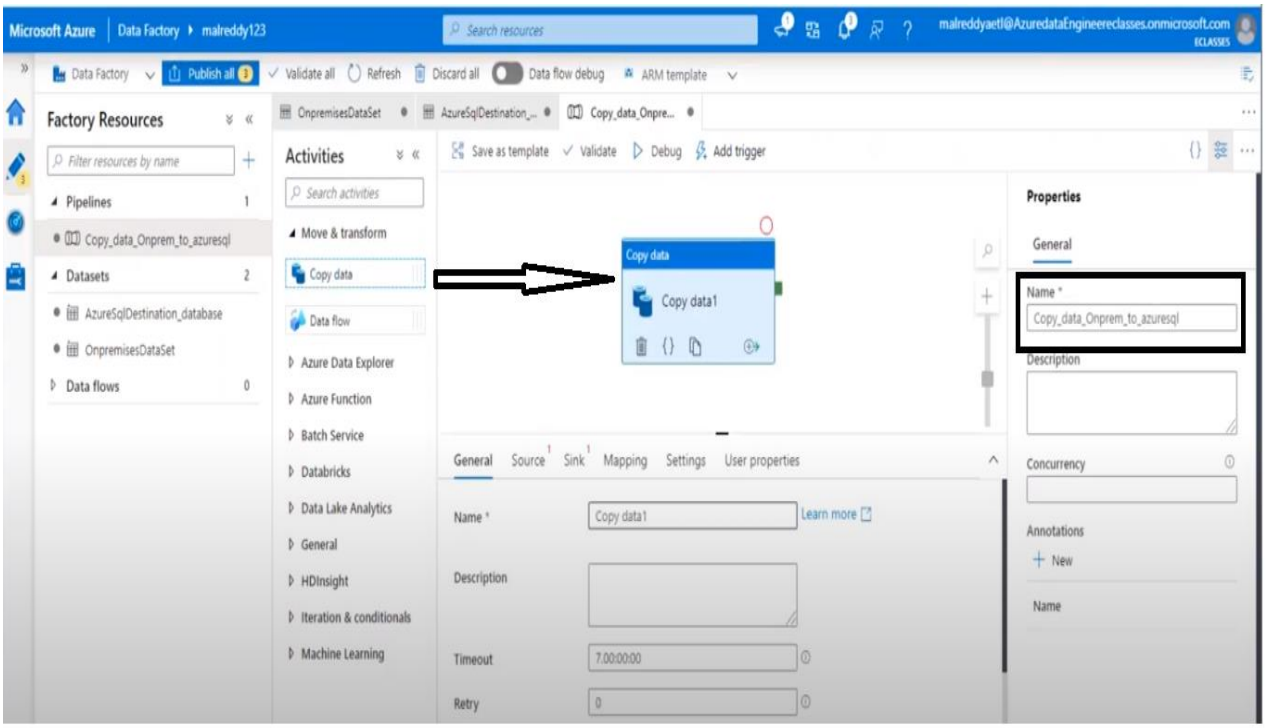
Step 12: Give proper name for the target dataset----Select Linked Service-----Select Table Name and Click on OK



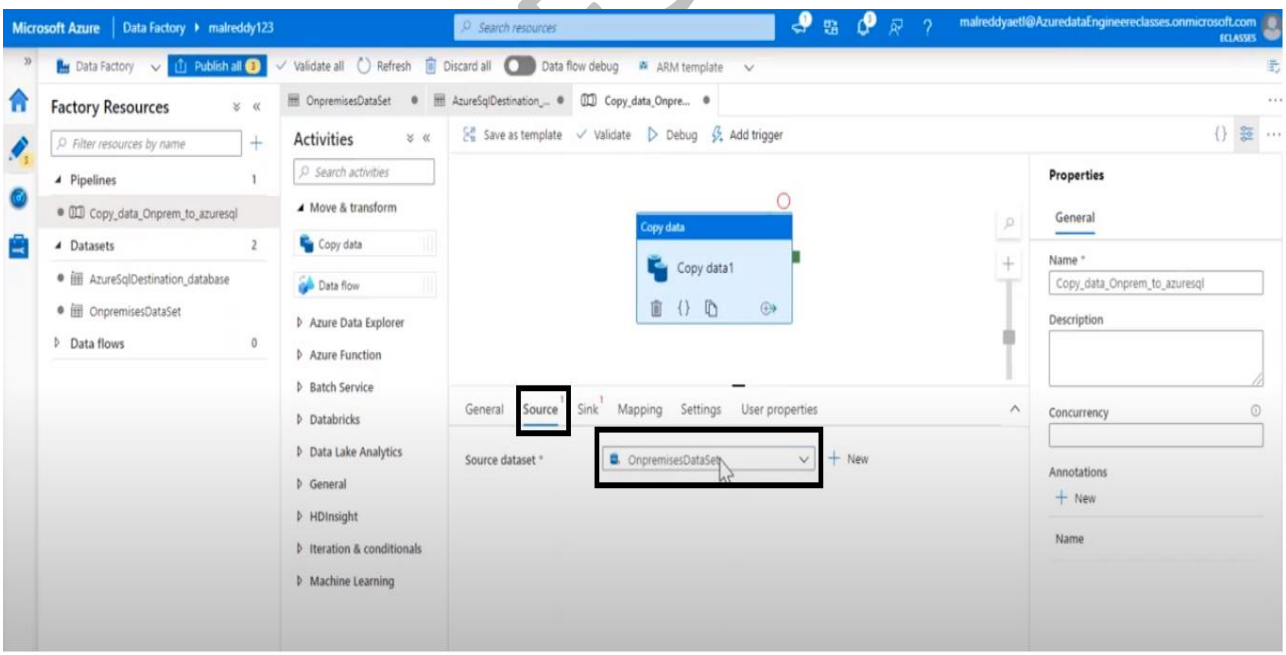
Step 13: Go to Author window---Click on Pipelines---New Pipeline



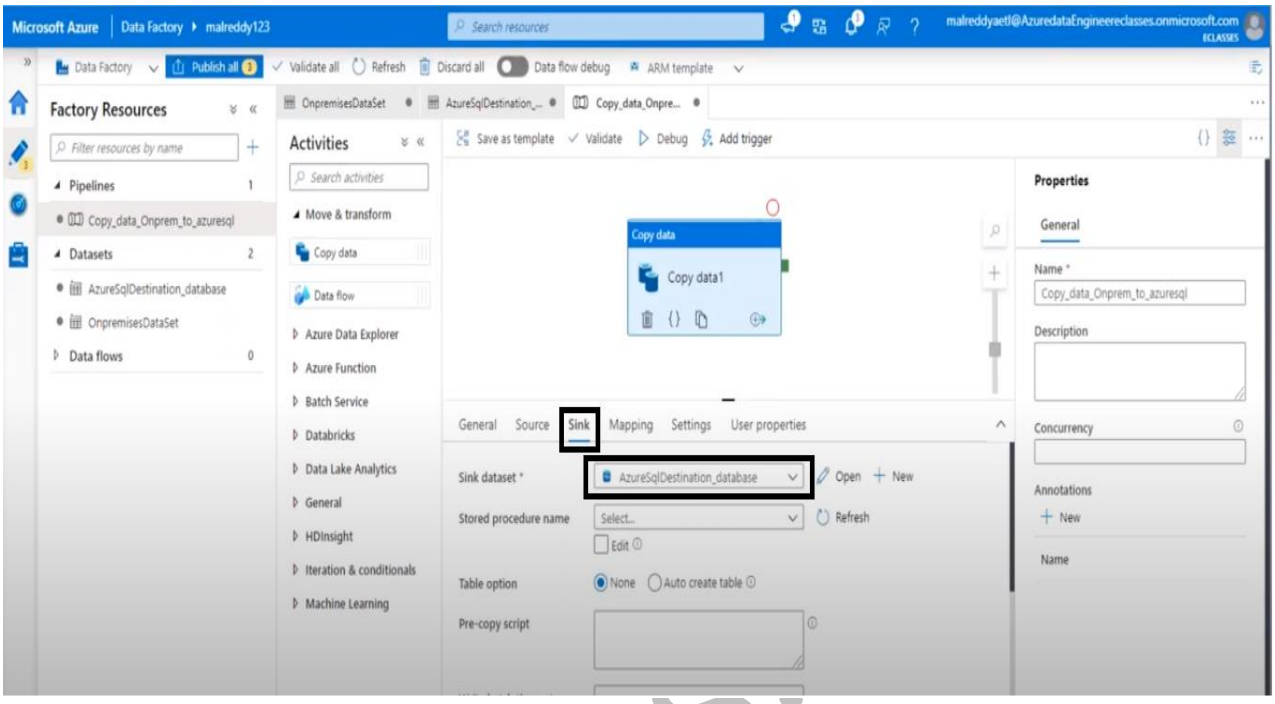
Step 14: Give a proper name to the pipeline----drag the copy activity to development area



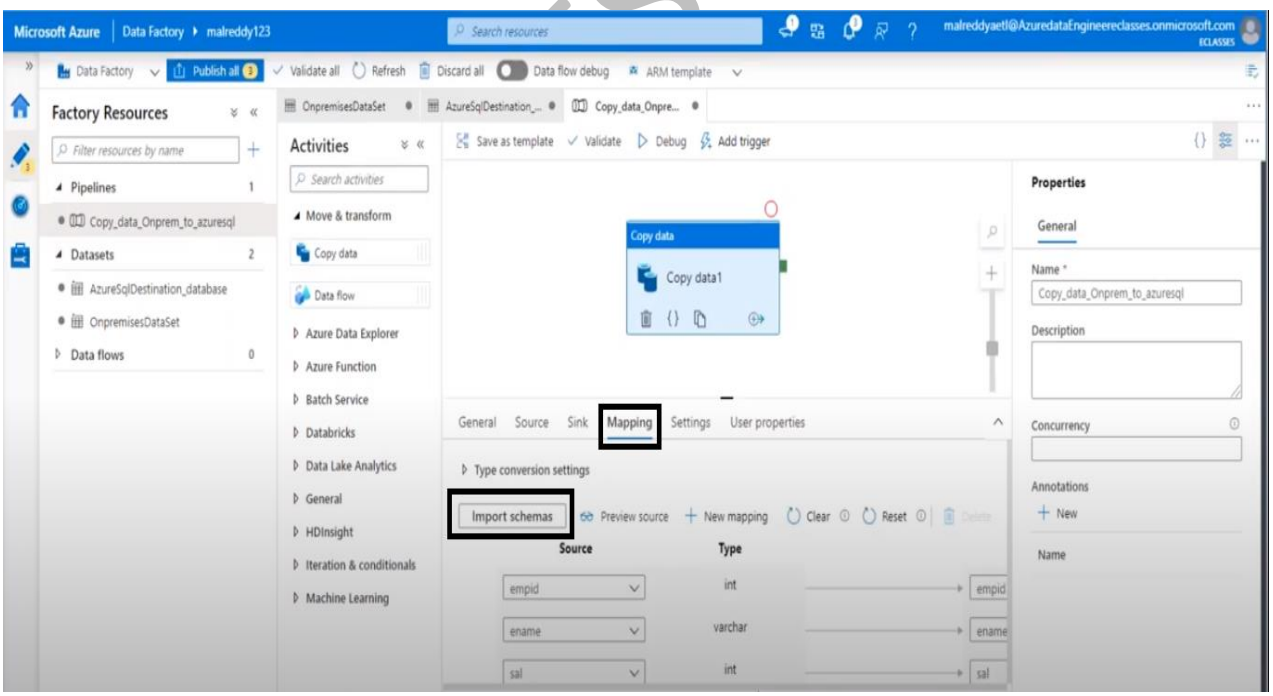
Step 15: Click on Source----Select source dataset



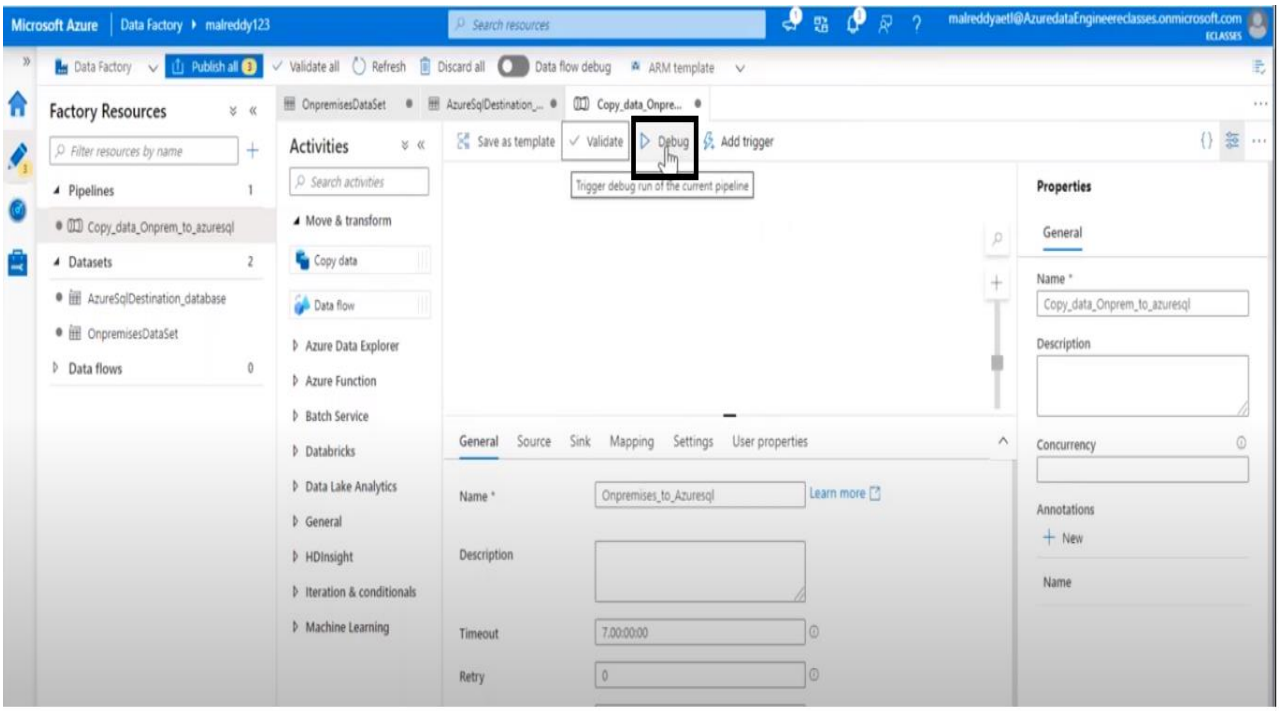
Step 16: Click on Sink----Select Sink Dataset



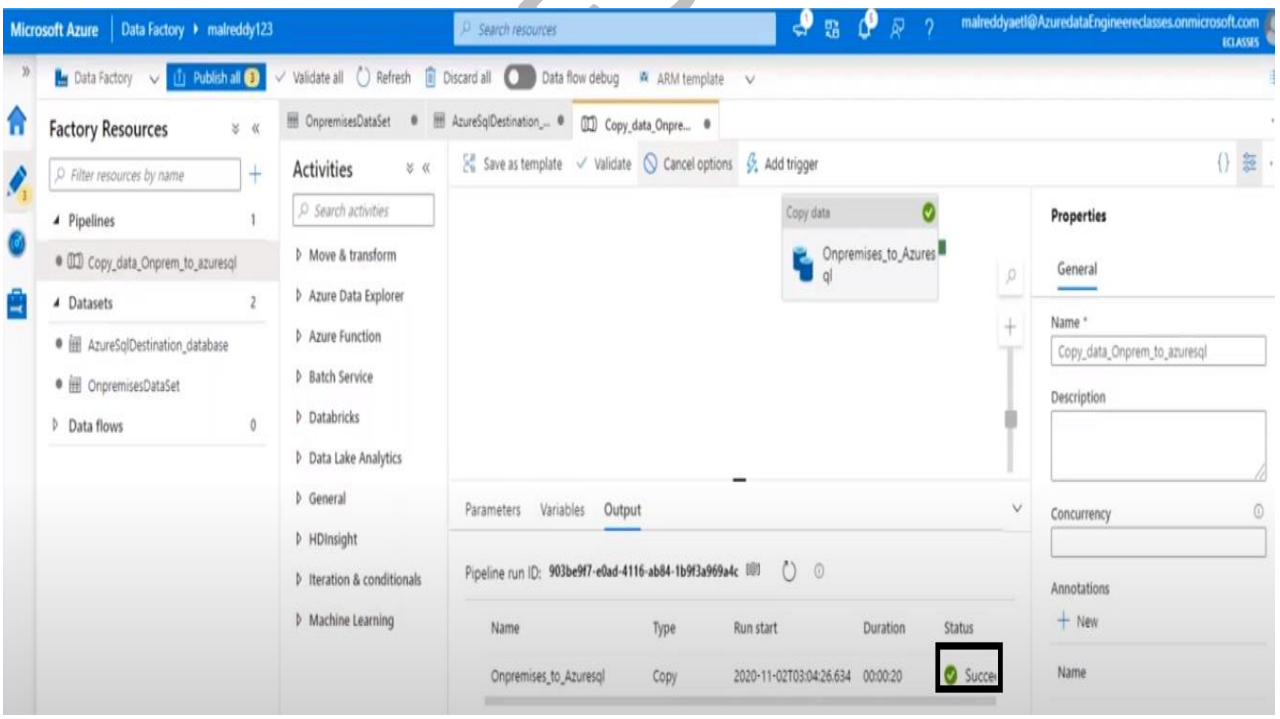
Step 17: Click on Mapping----Import schemas---Do Mapping



Step 18: Click on Debug



Step 19: Check the pipeline status



How to perform Incremental Loading in Azure Data Factory

Incremental loading means inserting newly added records in source table into target table

Based on watermark value.

Prerequisites:

Create Azure SQL Database name it as Source.

Create Student table.

Create Another Azure SQL database Name it as Target.

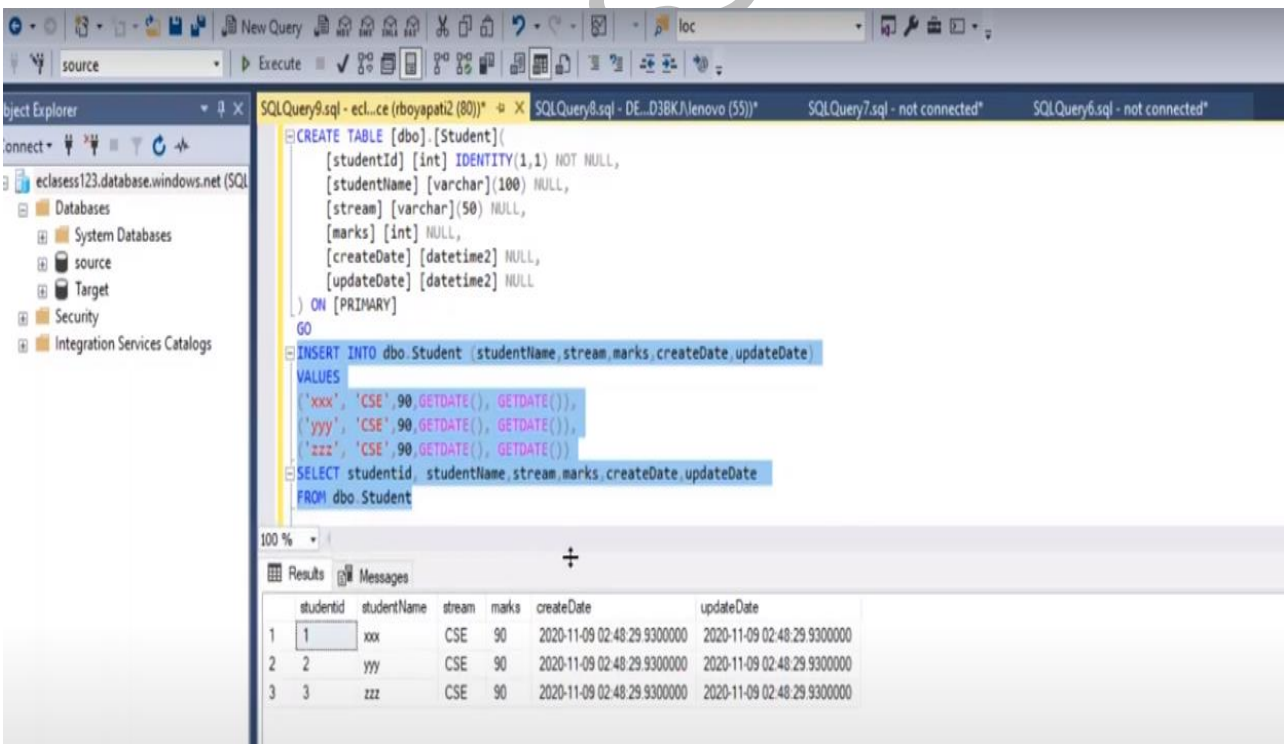
Create table STG student Table.

For example:

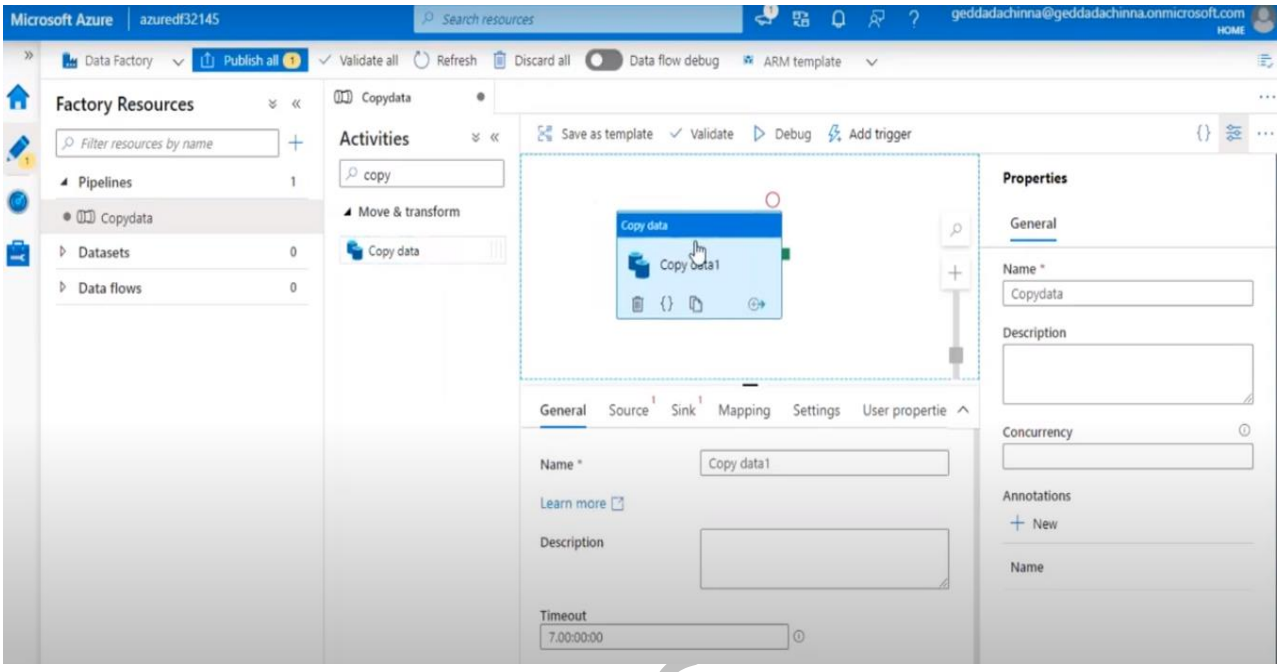
Loading data from a dbo.Student table to dbo.Stgstudent table

Step 1:

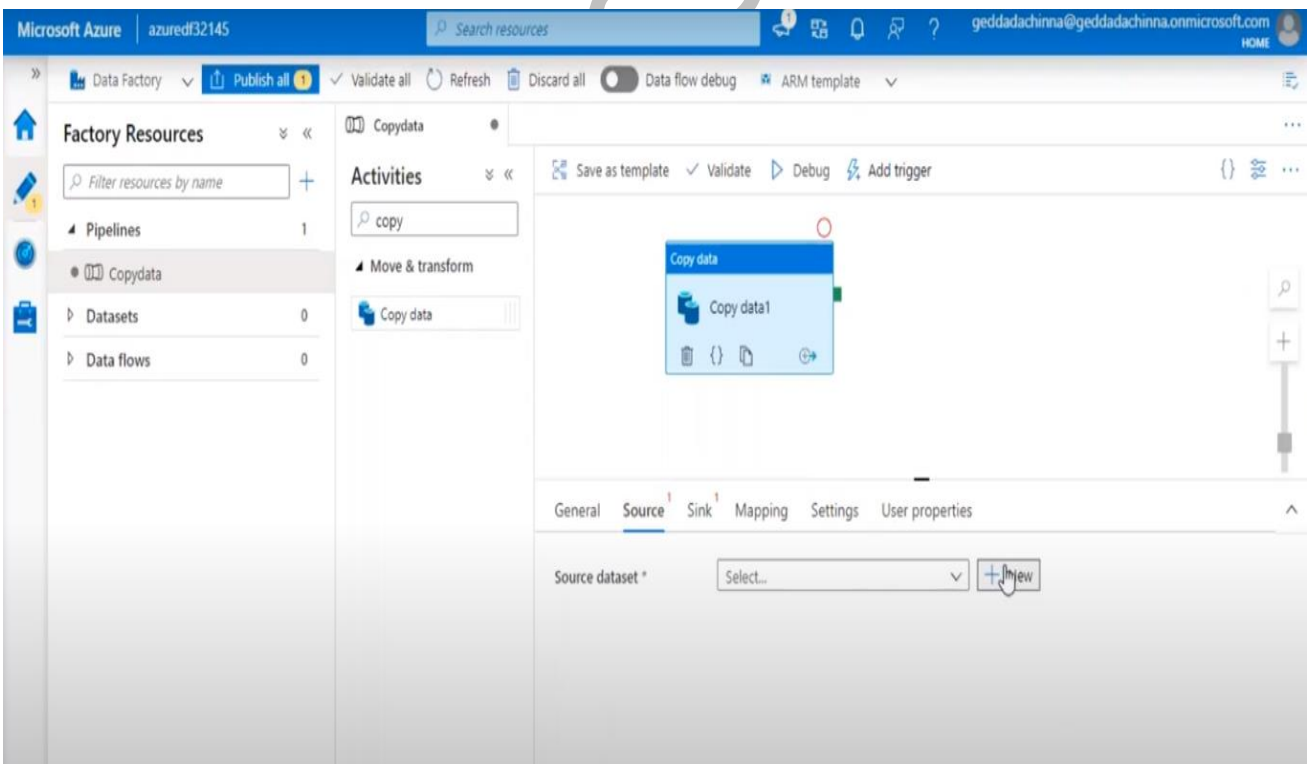
Create a dbo.Student table (Source) and insert some sample data. Create a dbo.Stgstudent as a target table



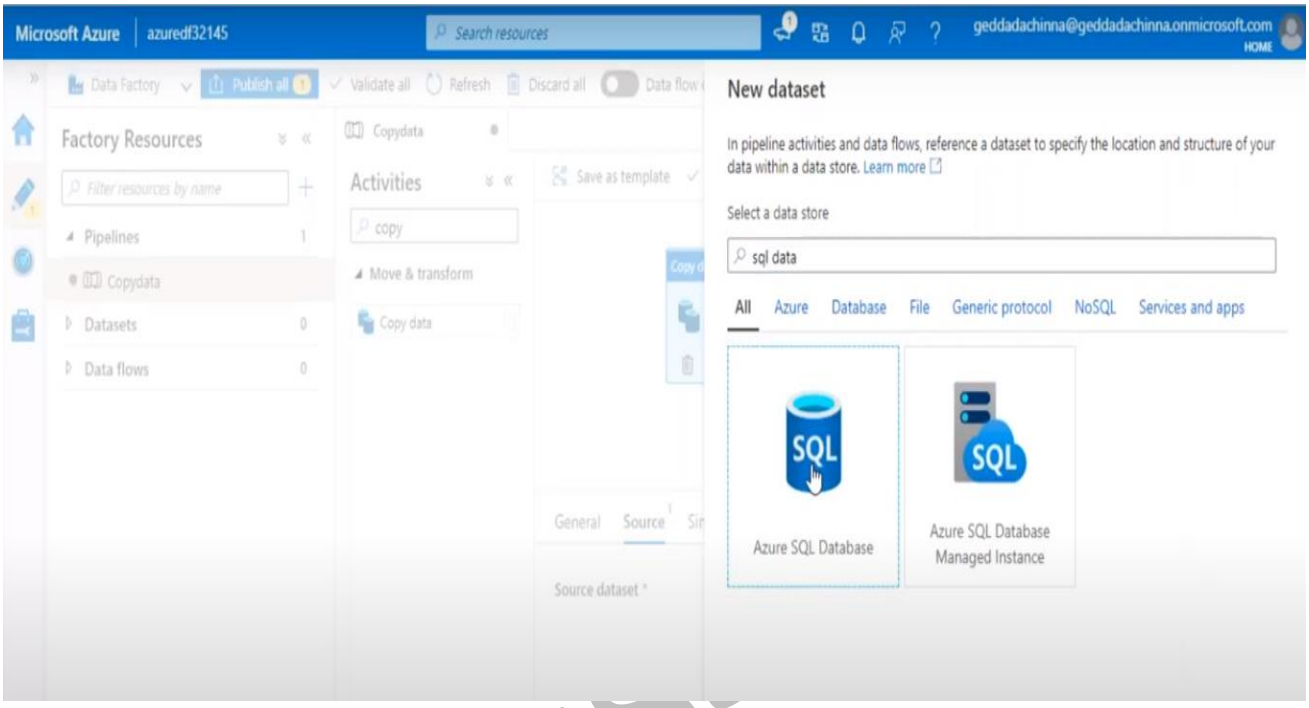
Step 2: Drag Copy Activity into development area inside pipeline



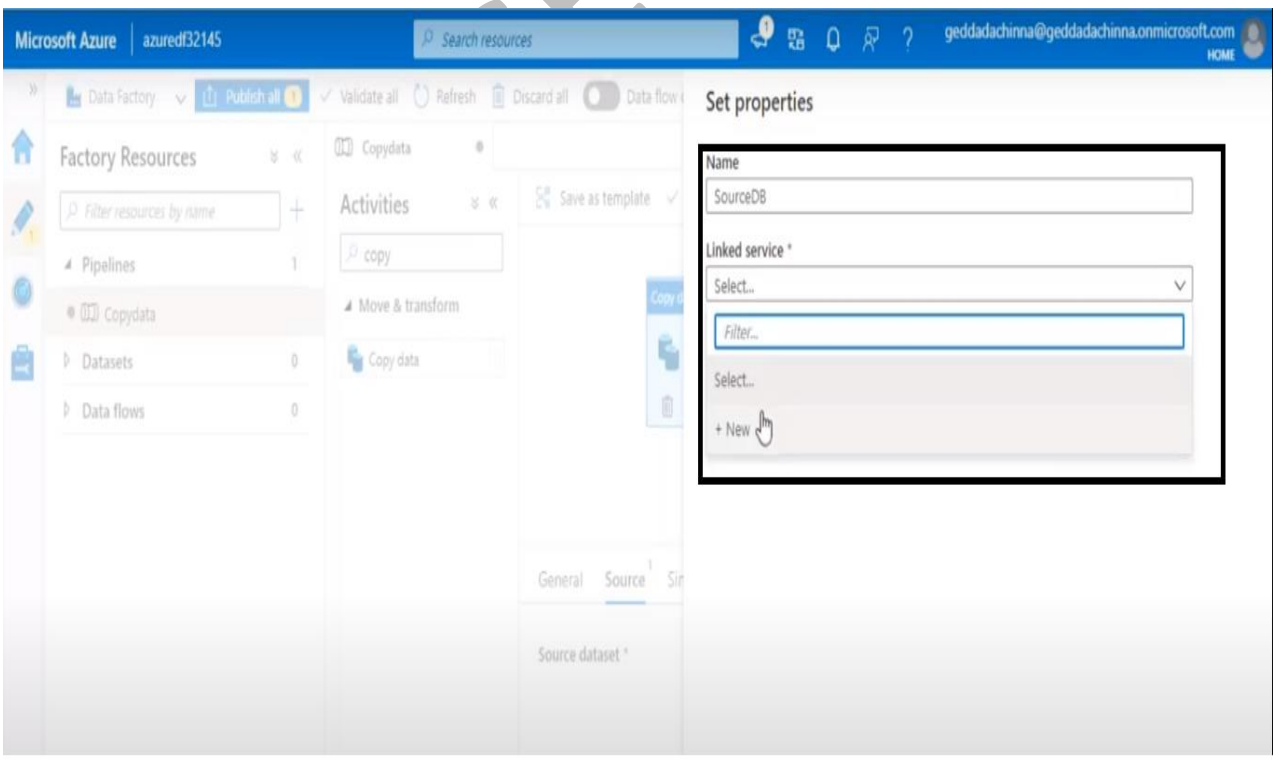
Step 3: Click on Source and New



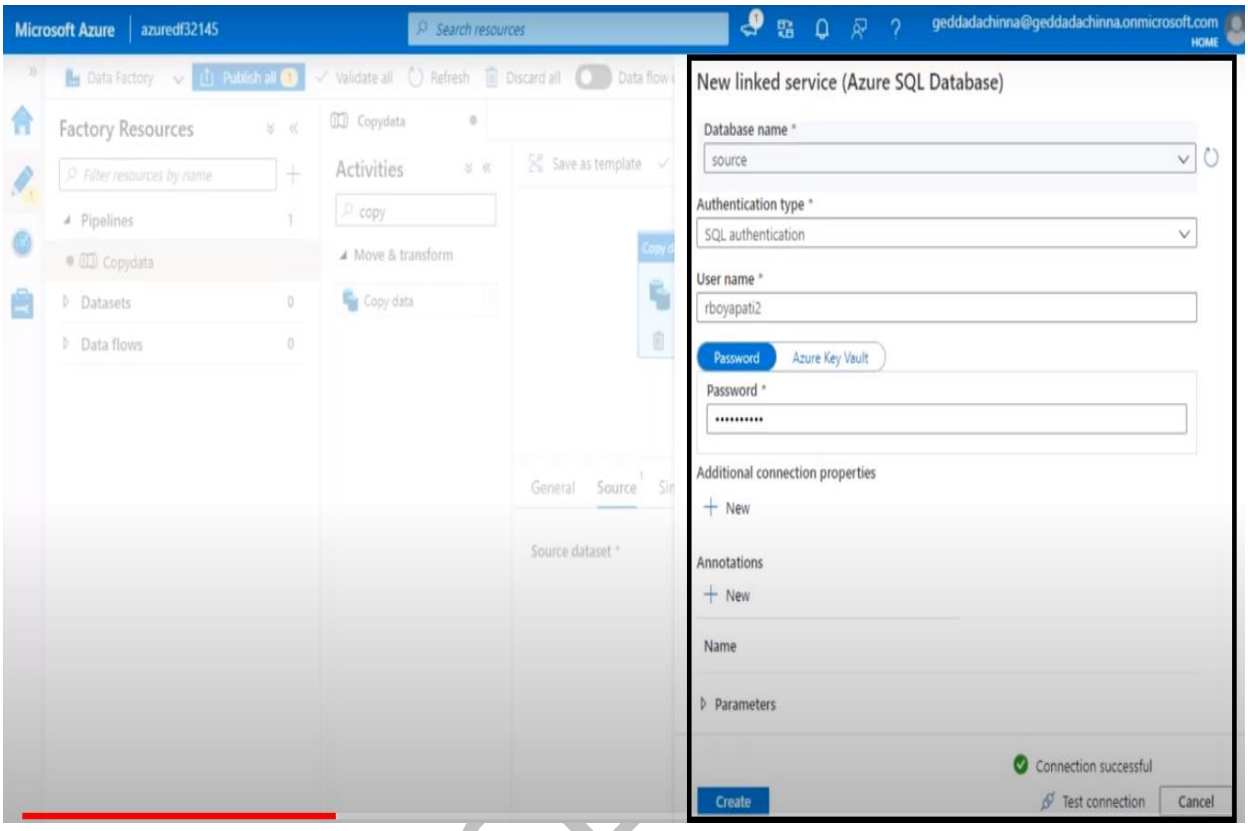
Step 4: Click on Azure SQL Database



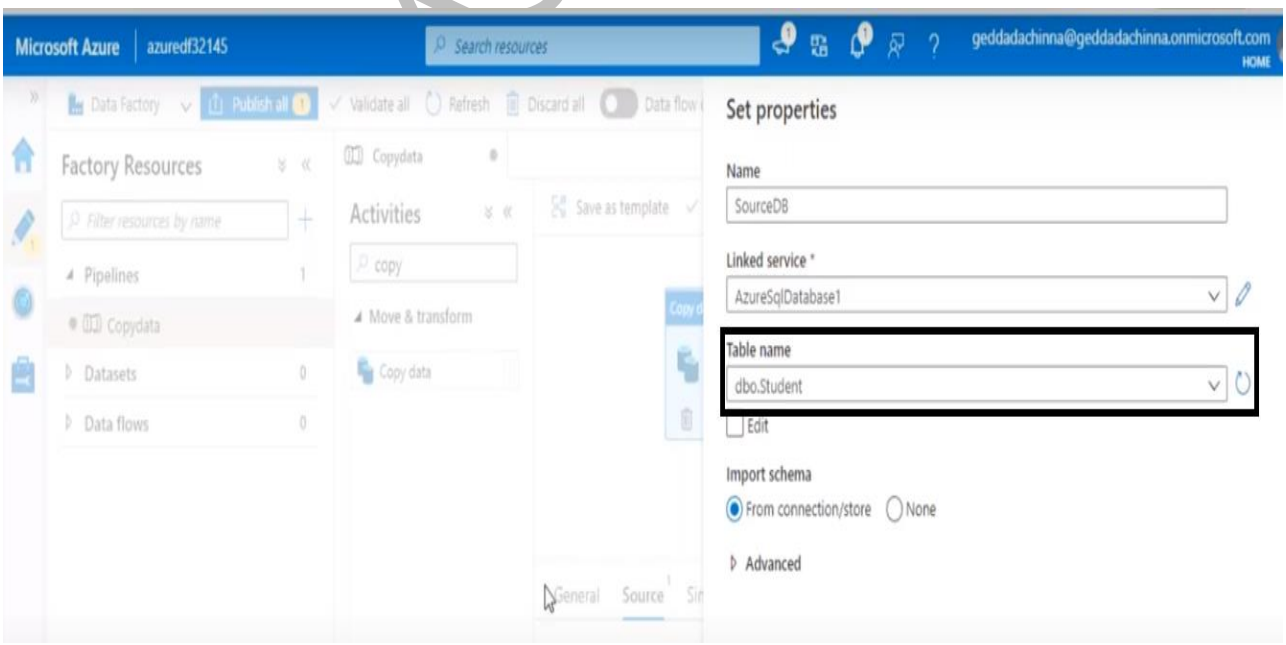
Step 5: Give a proper name for the Source dataset----Click on Linked service and Select New



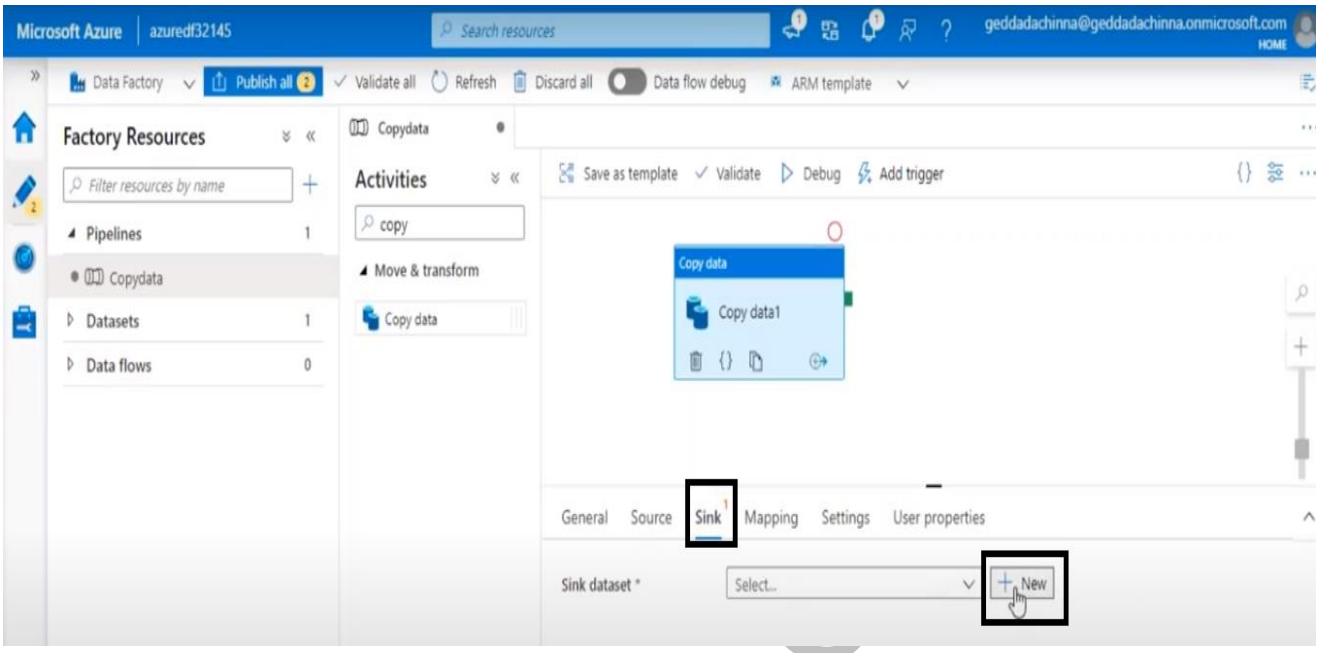
Step 6: Give a proper name for the linked service---Select Subscription-----Select Database name---- Select Authentication type---Give password----Test Connection---Click on Create



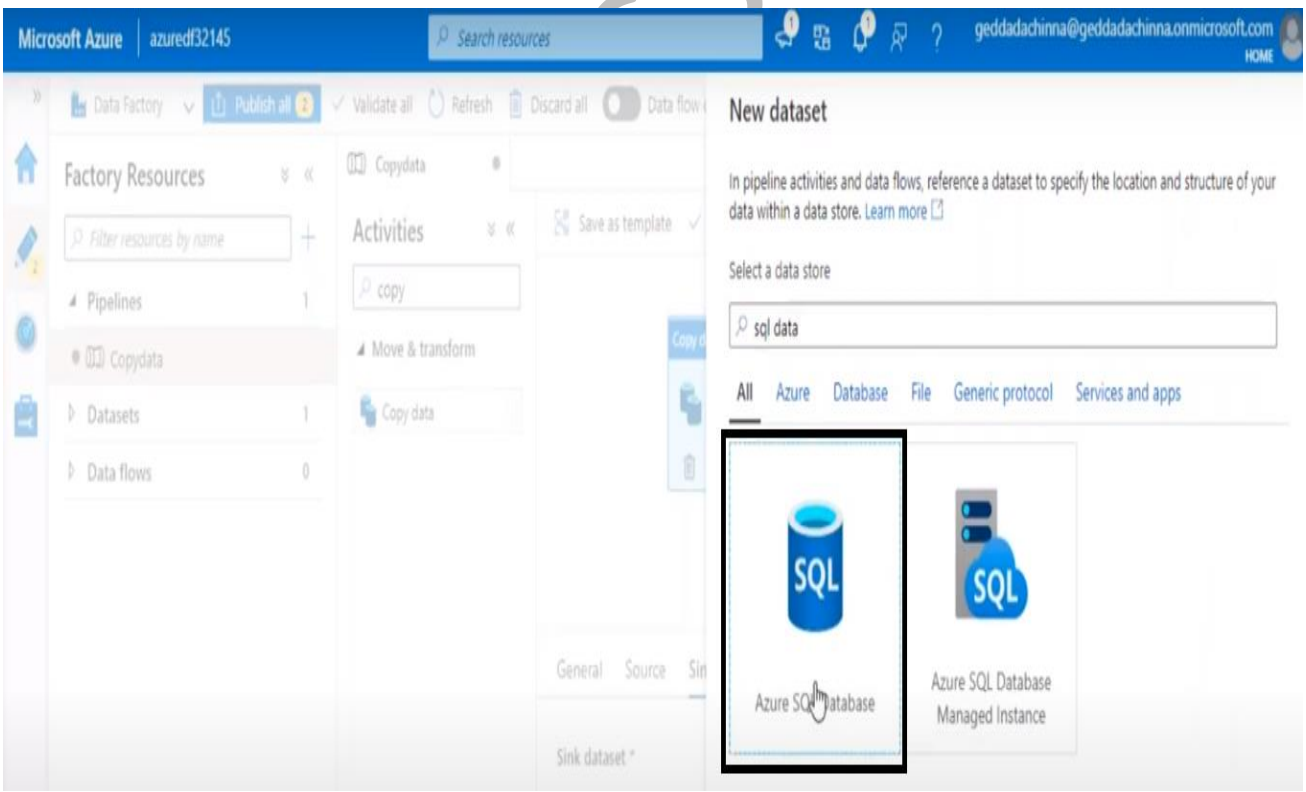
Step 7: Select the dbo.Student table and Click on OK



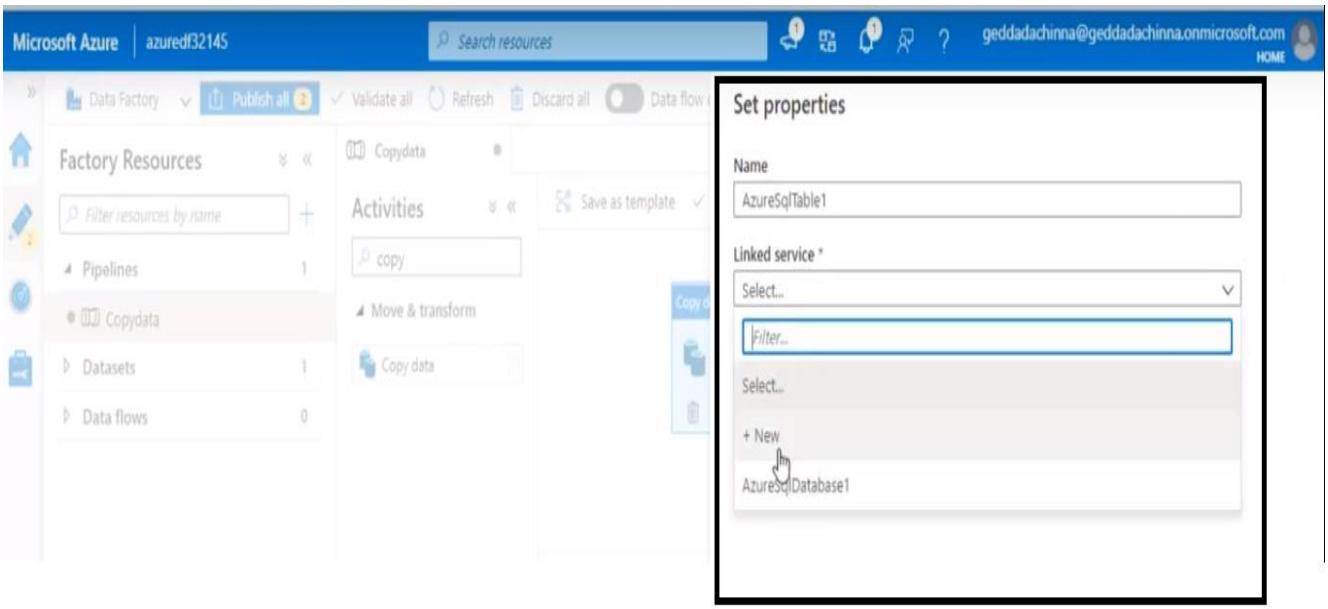
Step 8: Click on Sink---Select New



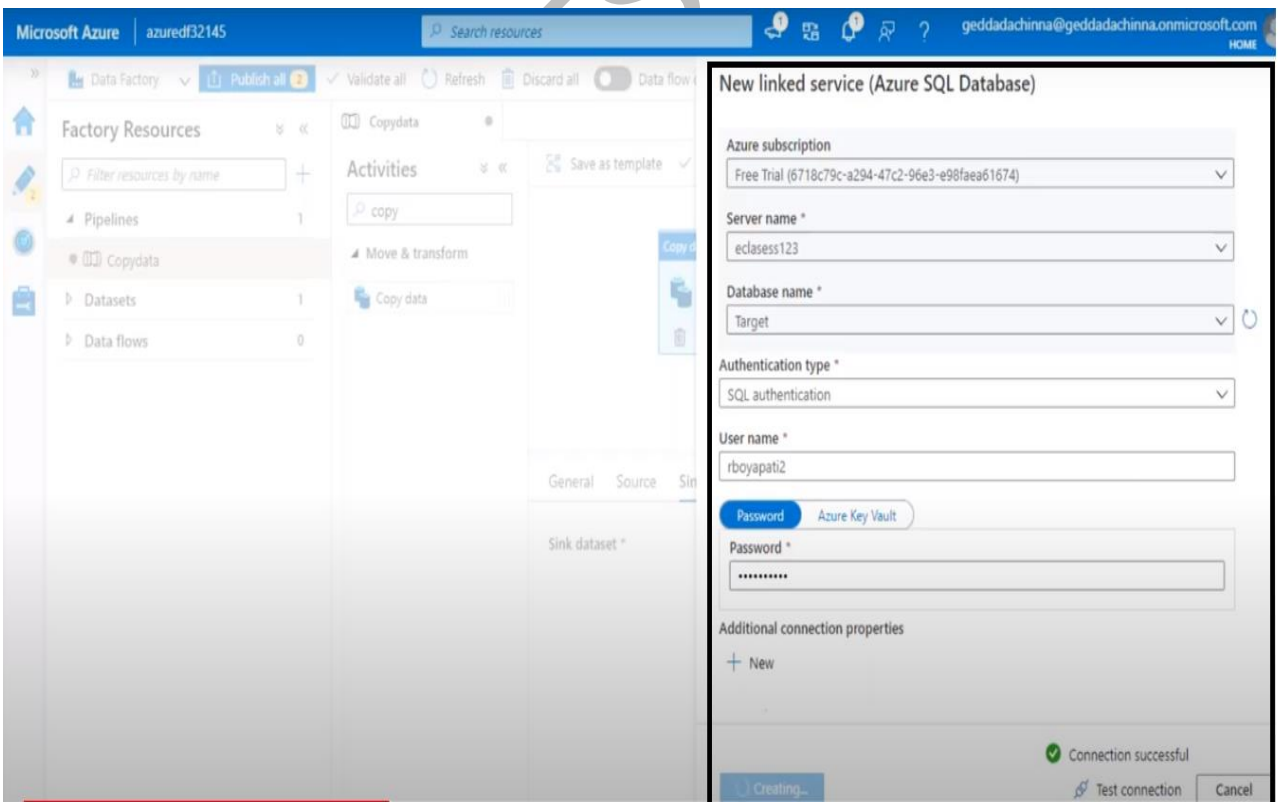
Step 9: Click on Azure SQL Database



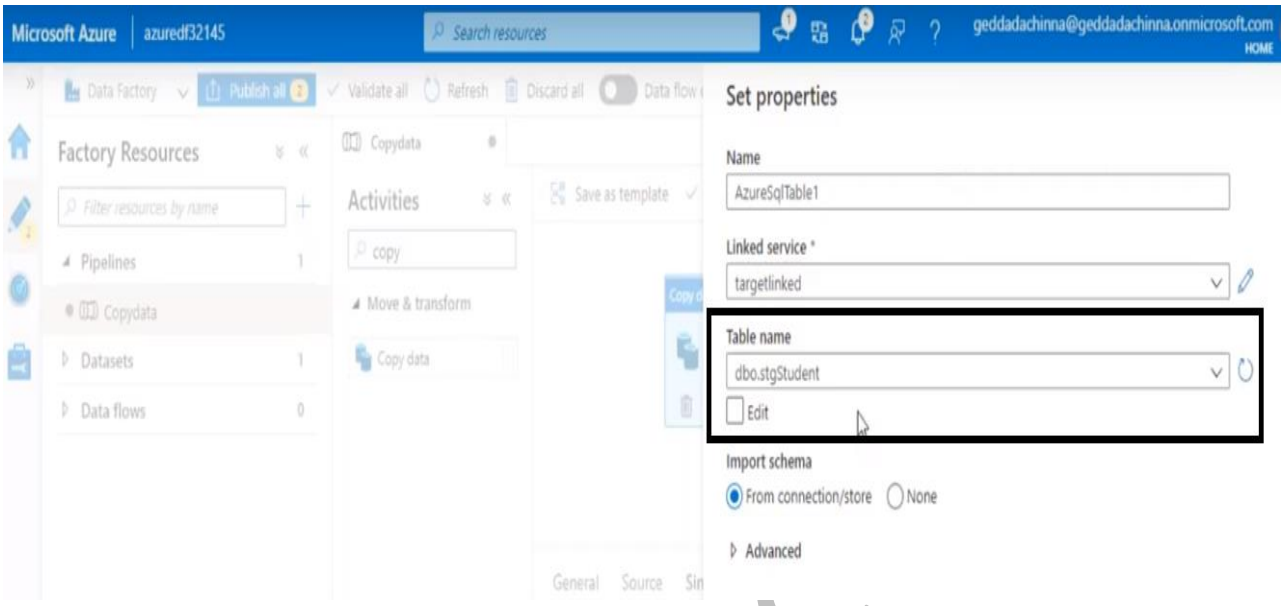
Step 10: Give a proper name for the dataset----Select Linked Service---Click on New



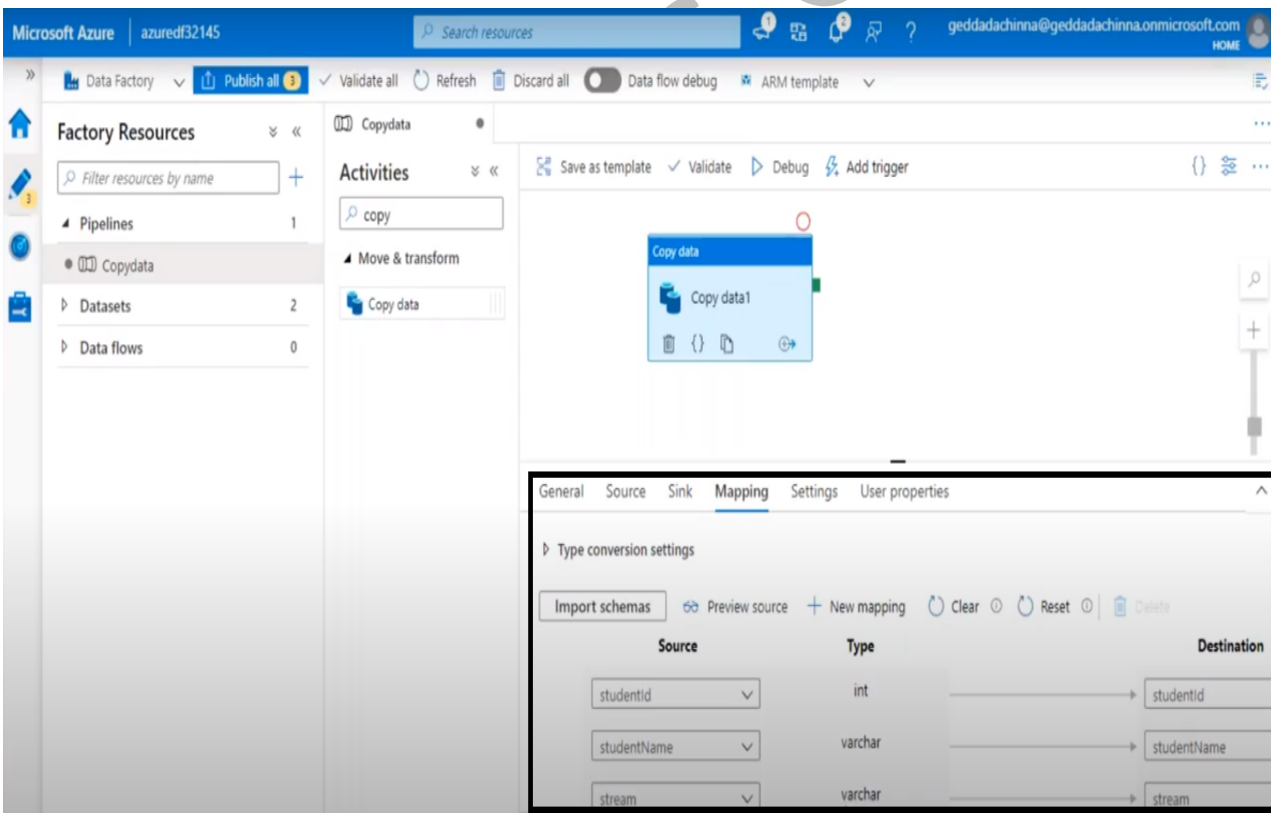
Step 11: Give a proper name for the linked service---Select Subscription-----Select Database name---- Select Authentication type---Give password----Test Connection---Click on Create



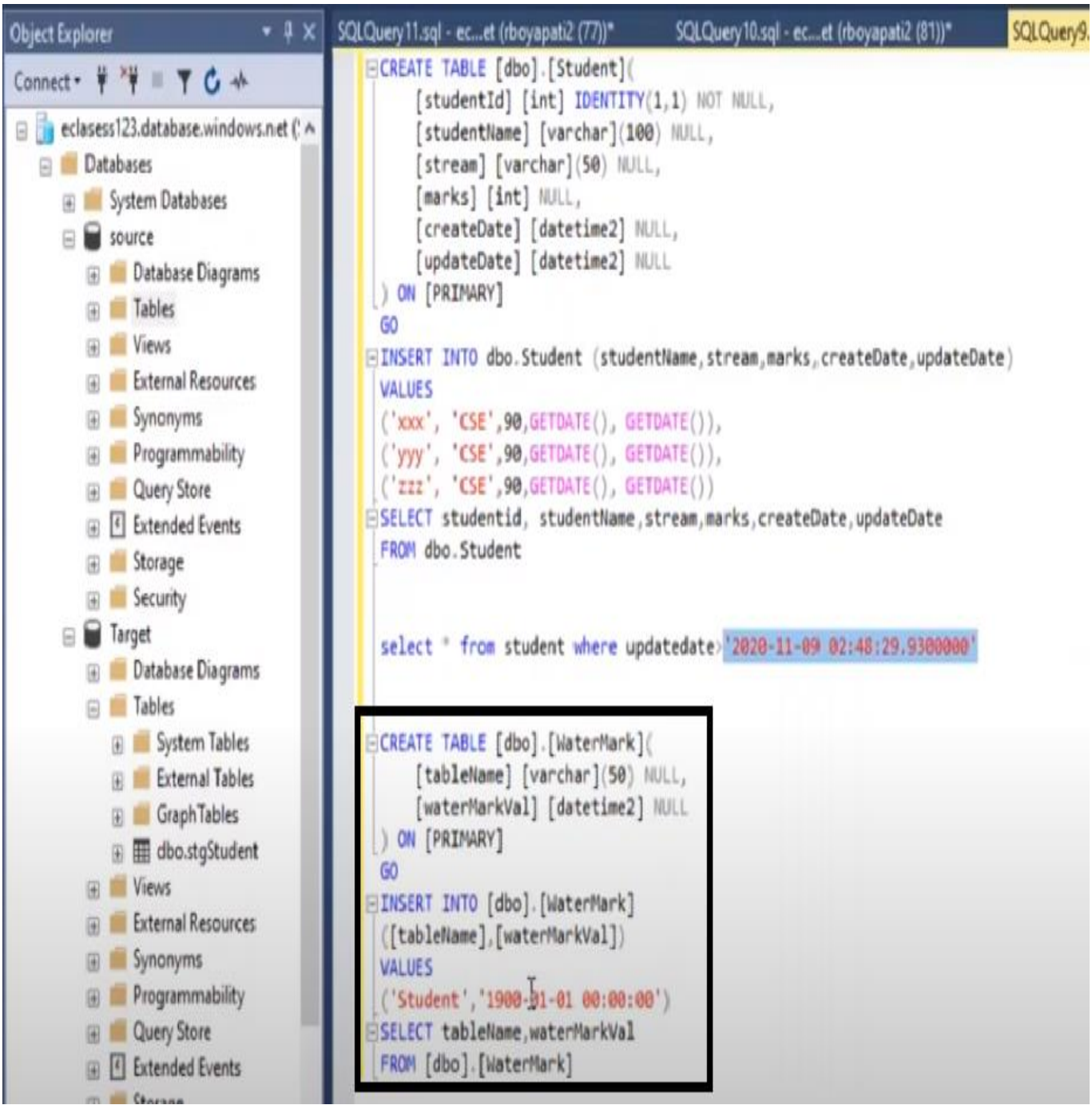
Step 12: Select dbo.StgStudent table and Click on OK



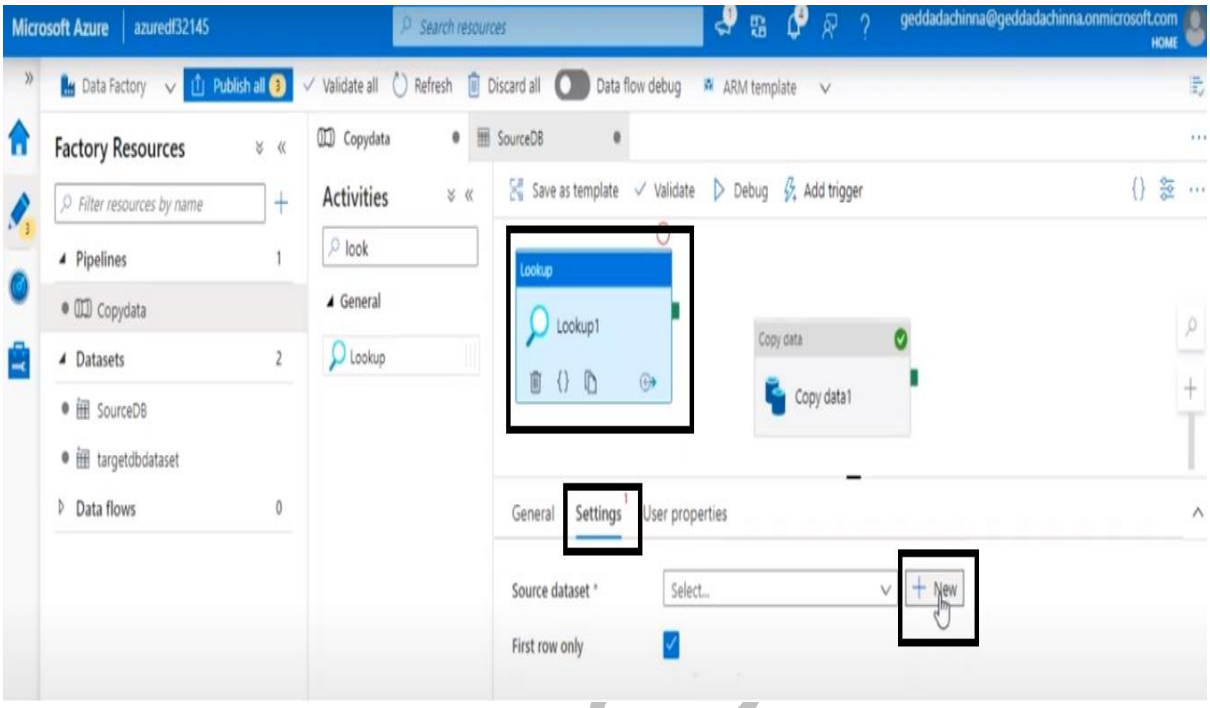
Step 13: Click on Mapping-----Import Schemas----Do Mapping



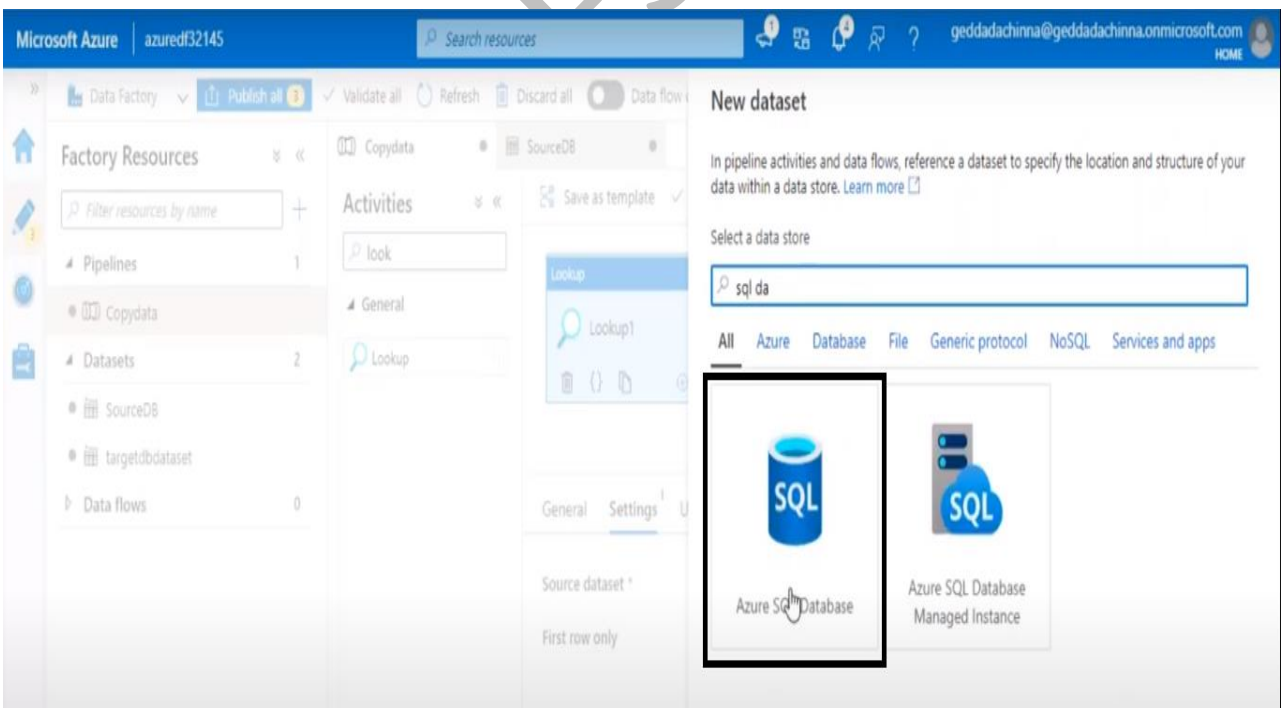
Step 14: Create a dbo.watermark table in source database and insert some dummy value.



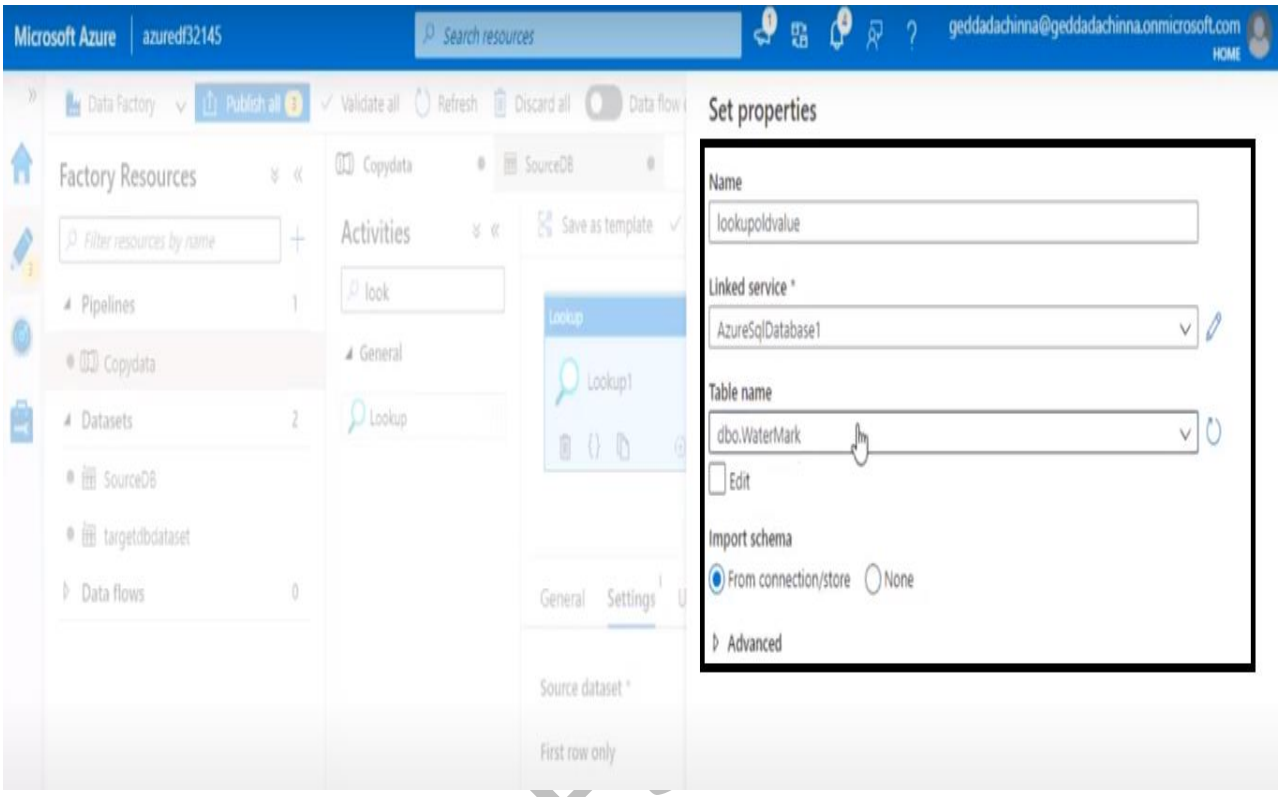
Step 15: We need to get the last updated time of watermark table---For that we need to use Lookup Activity--Drag Lookup activity into the development area---Click on Settings--New



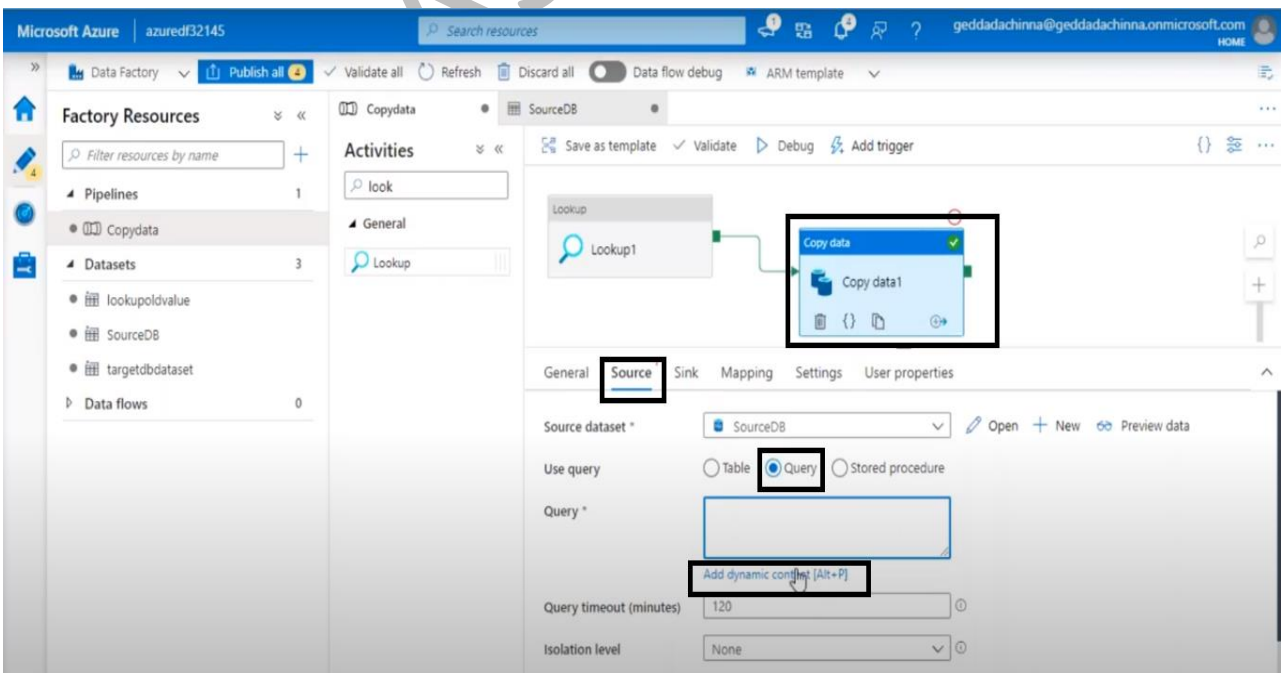
Step 16: Click on Azure SQL Database



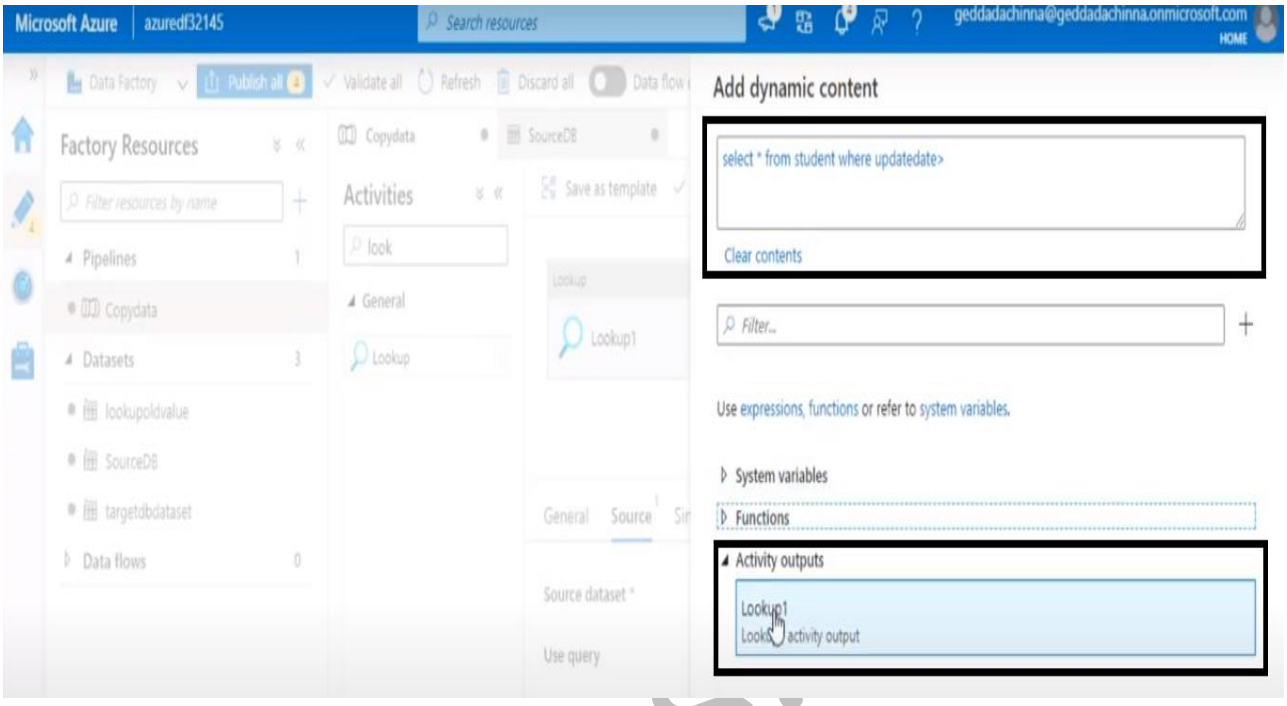
Step 17: Give a proper name for the dataset------Select linked service which has created earlier for the source-----Select Table name as dbo.WaterMark----Click OK



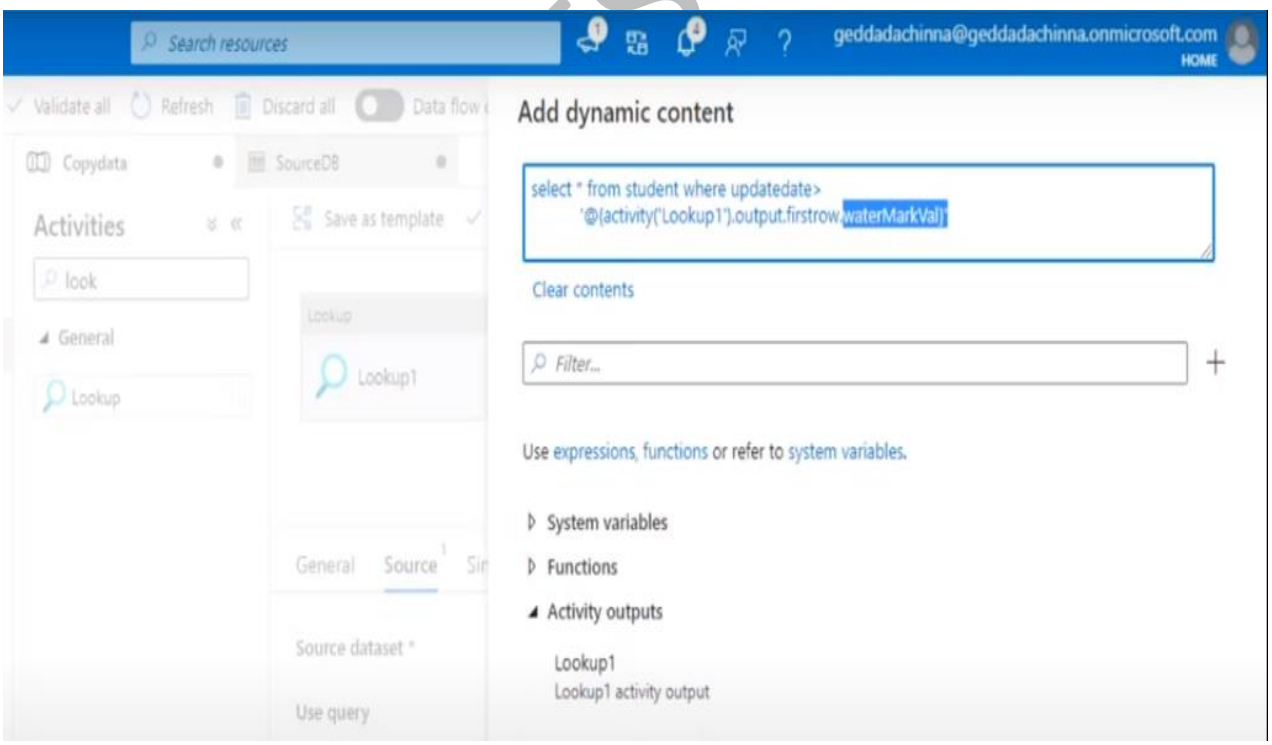
Step 18: Connect Lookup Activity with Copy Activity----Click on Copy Activity----Click on Source-----Select Query----Click on Add dynamic content



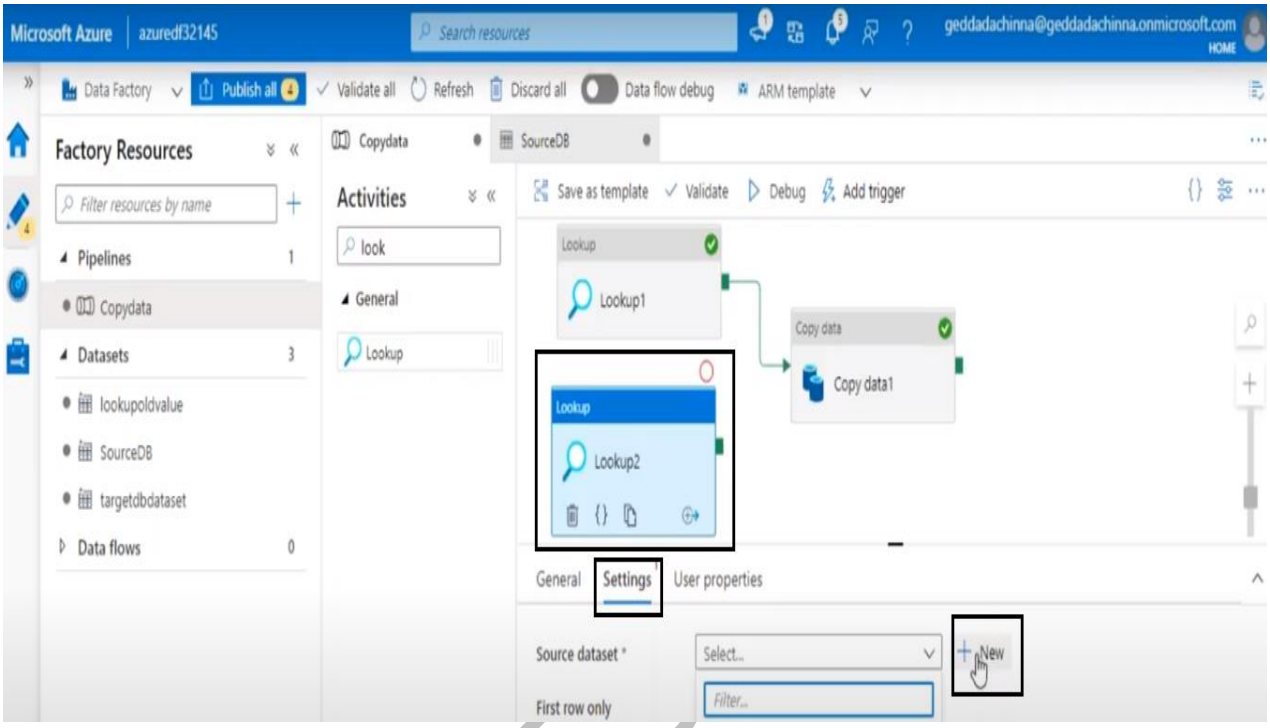
Step 19: Write a query on add dynamic content----Click on Lookup Activity Output



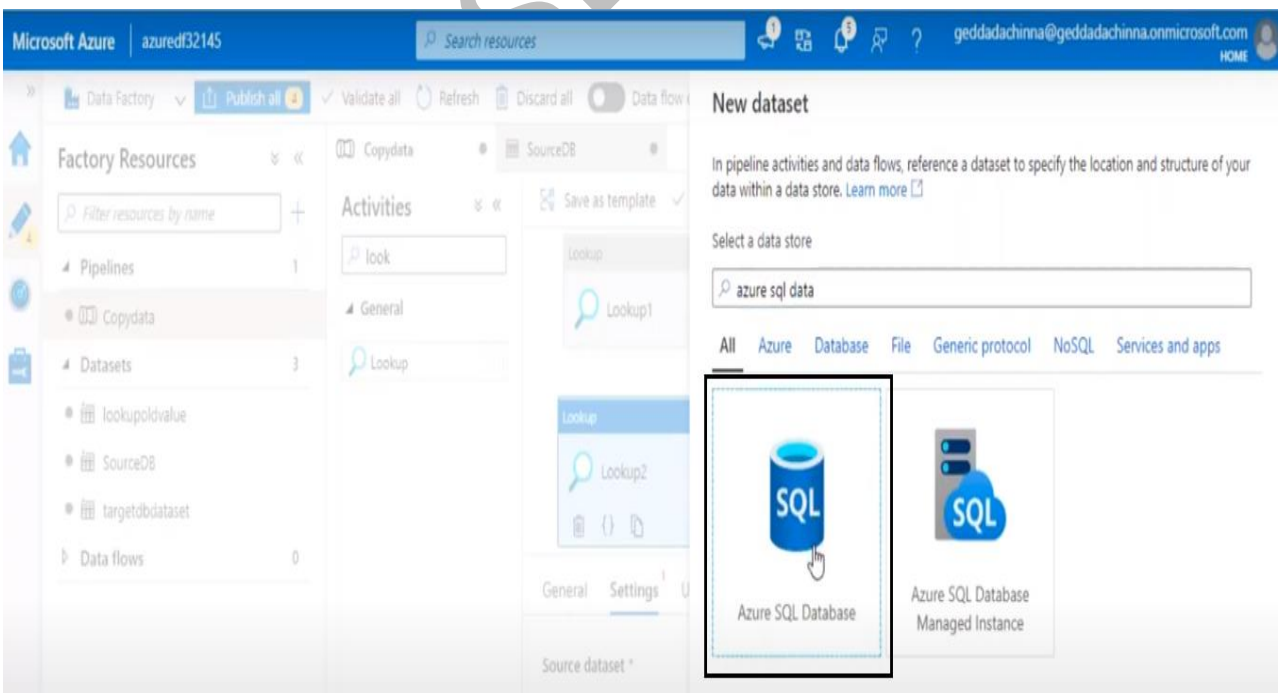
Step 20: add watermark table column name to the Query----Click on Finish



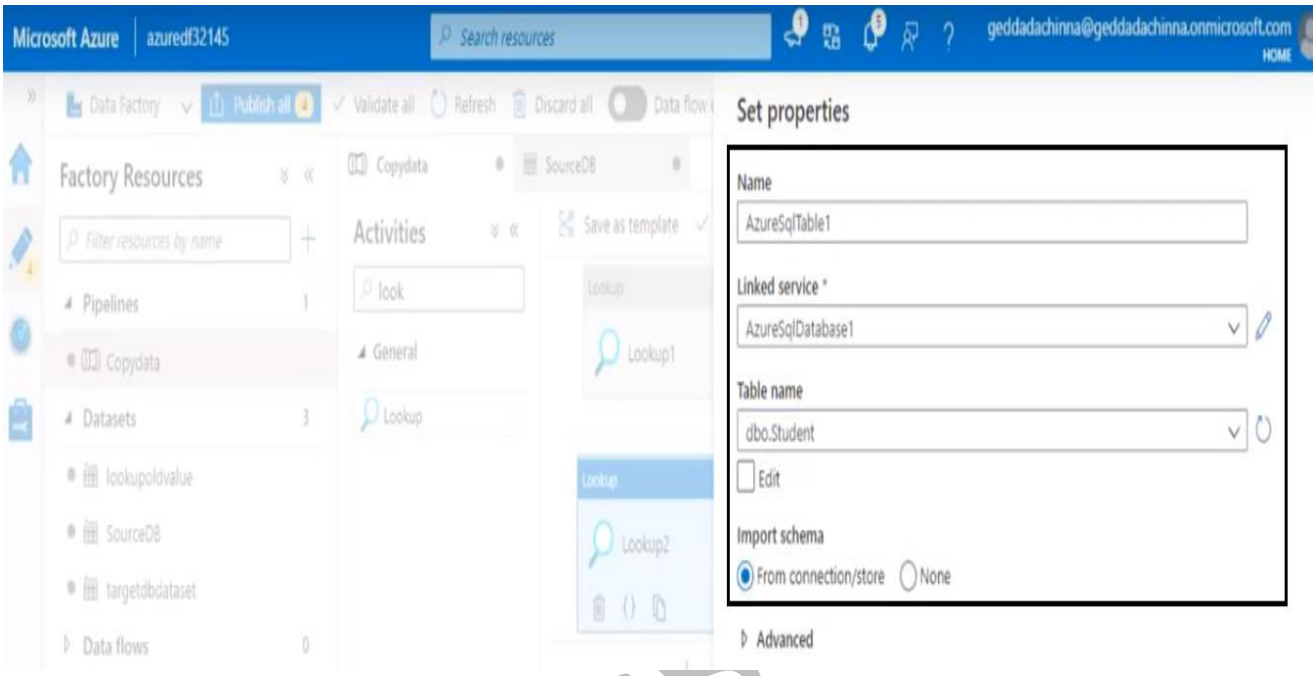
Step 21: We need to get the Maximum updated time of Student table-----For that we need to use Lookup Activity---Drag Lookup Activity into the development area-----Click on Settings---New



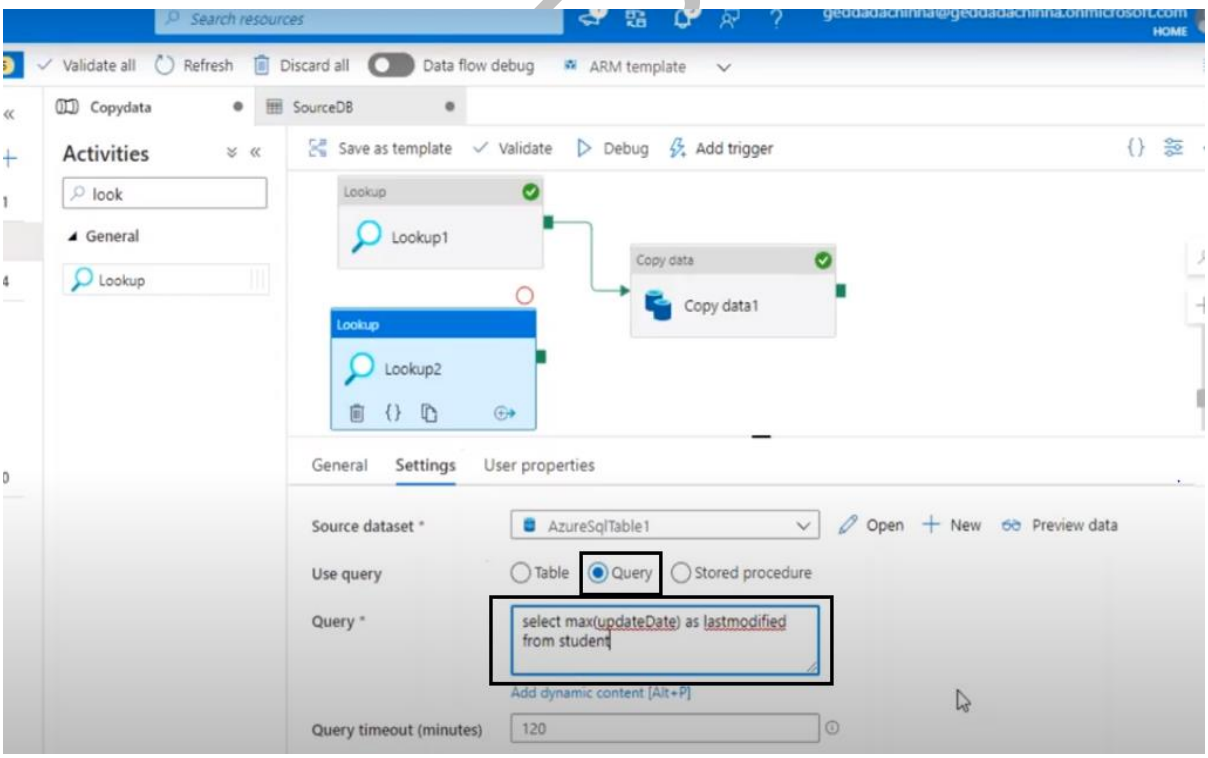
Step 22: Click on Azure SQL Database and Continue



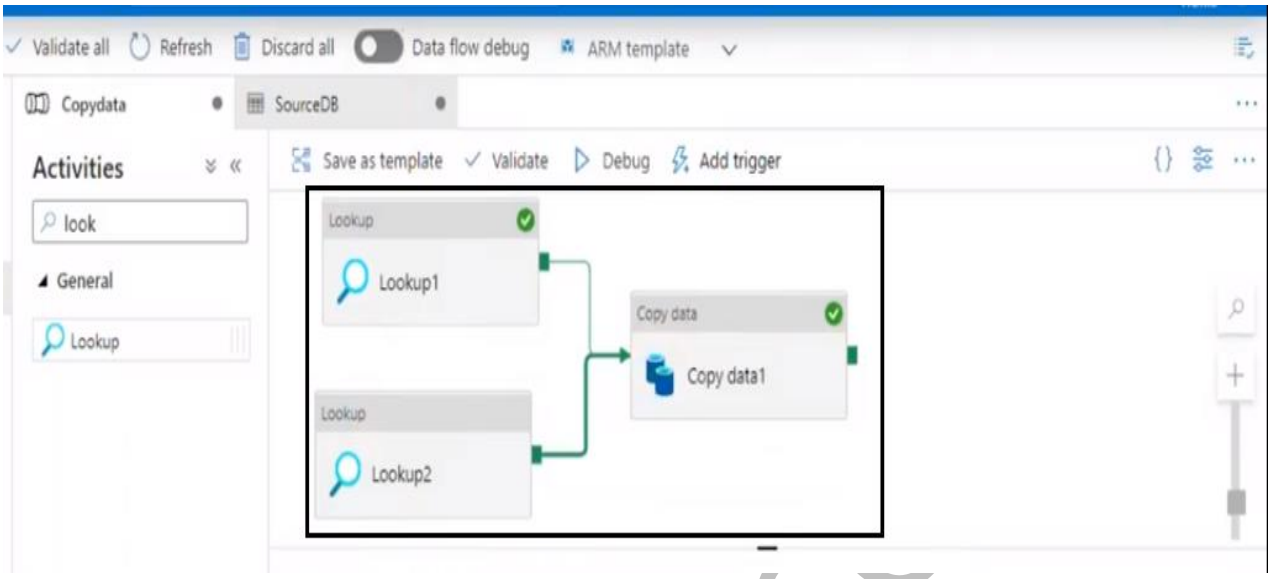
Step 23: Give a proper name for the dataset----Select Linked Service Which has created earlier for the Source table-----Select



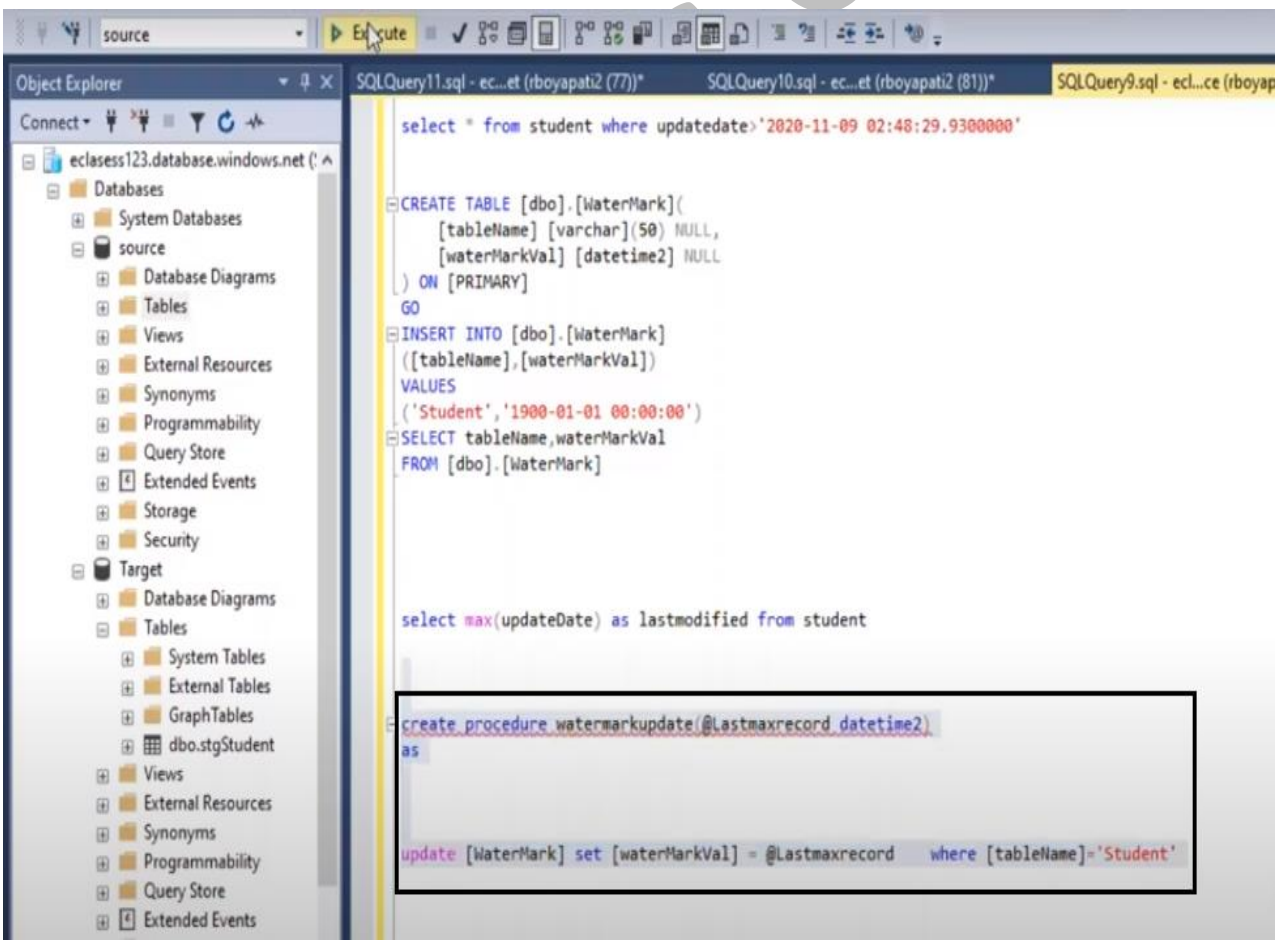
Step 24: Select Query----Write a Query



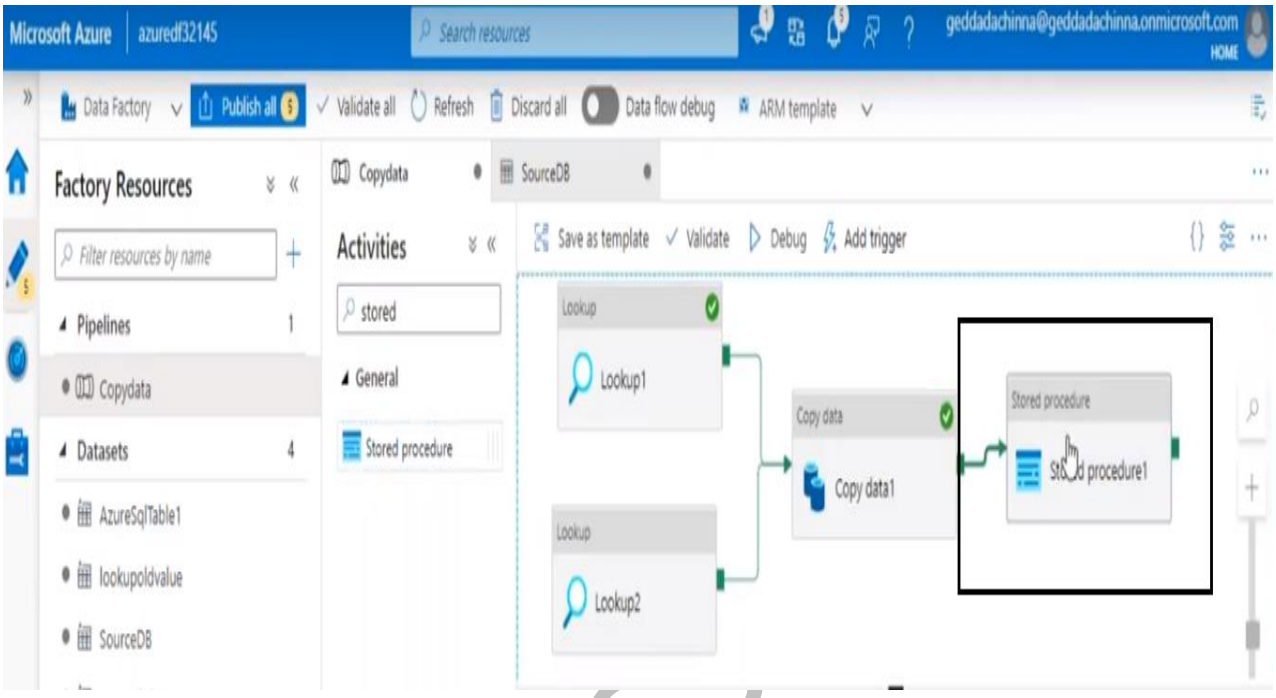
Step 25: Connect Lookup activity2 with Lookup activity1 and Copy Activity



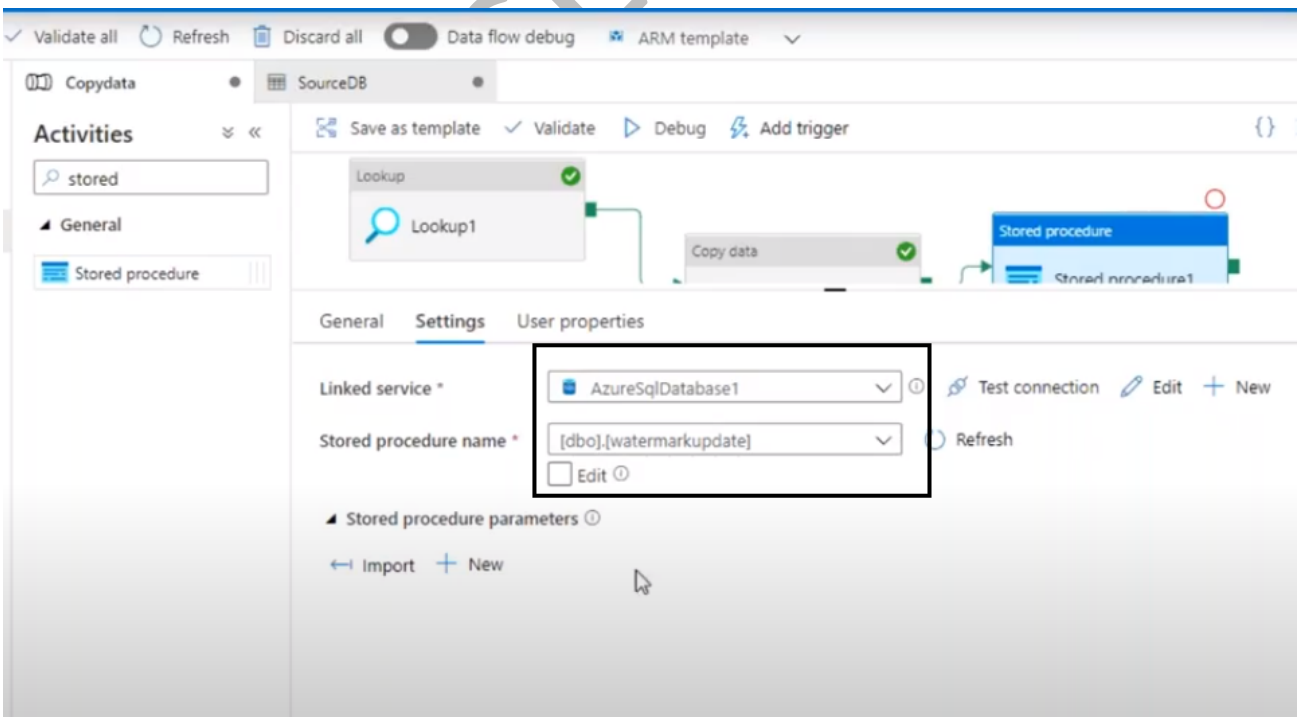
Step 26: We need to create procedure for updating the last updated time of Watermark table



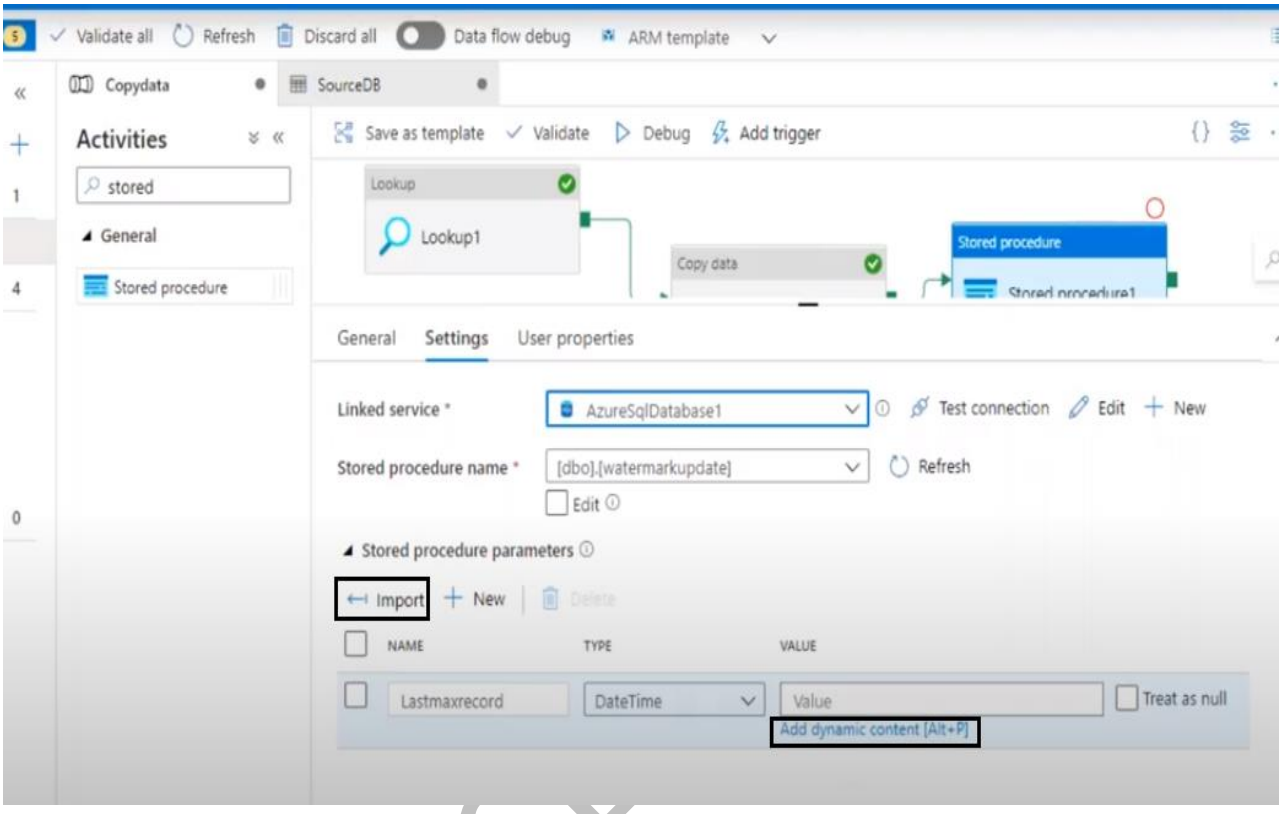
Step 27: Drag Store Procedure Activity into the development area and connect with copy activity which has connected to two Lookup activities.



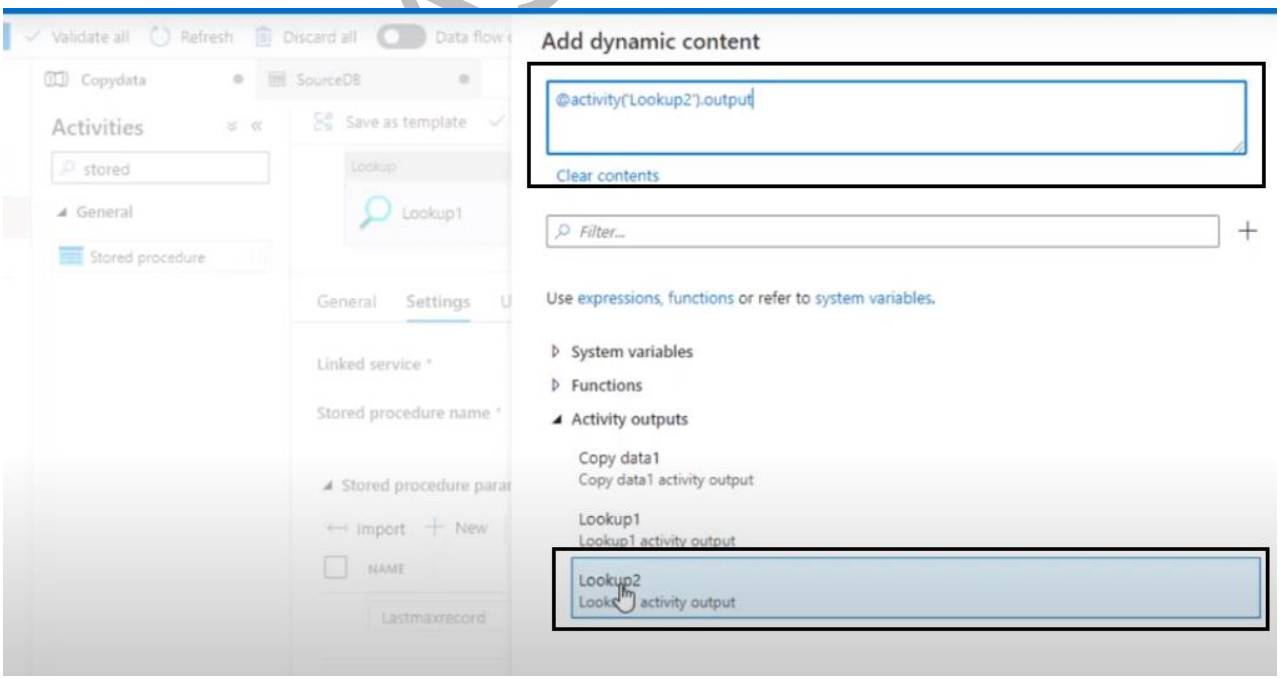
Step 28: Select Store Procedure Activity----Select Linked Service Which has created earlier for the source table----Select Store Procedure name which has created on Step 26.



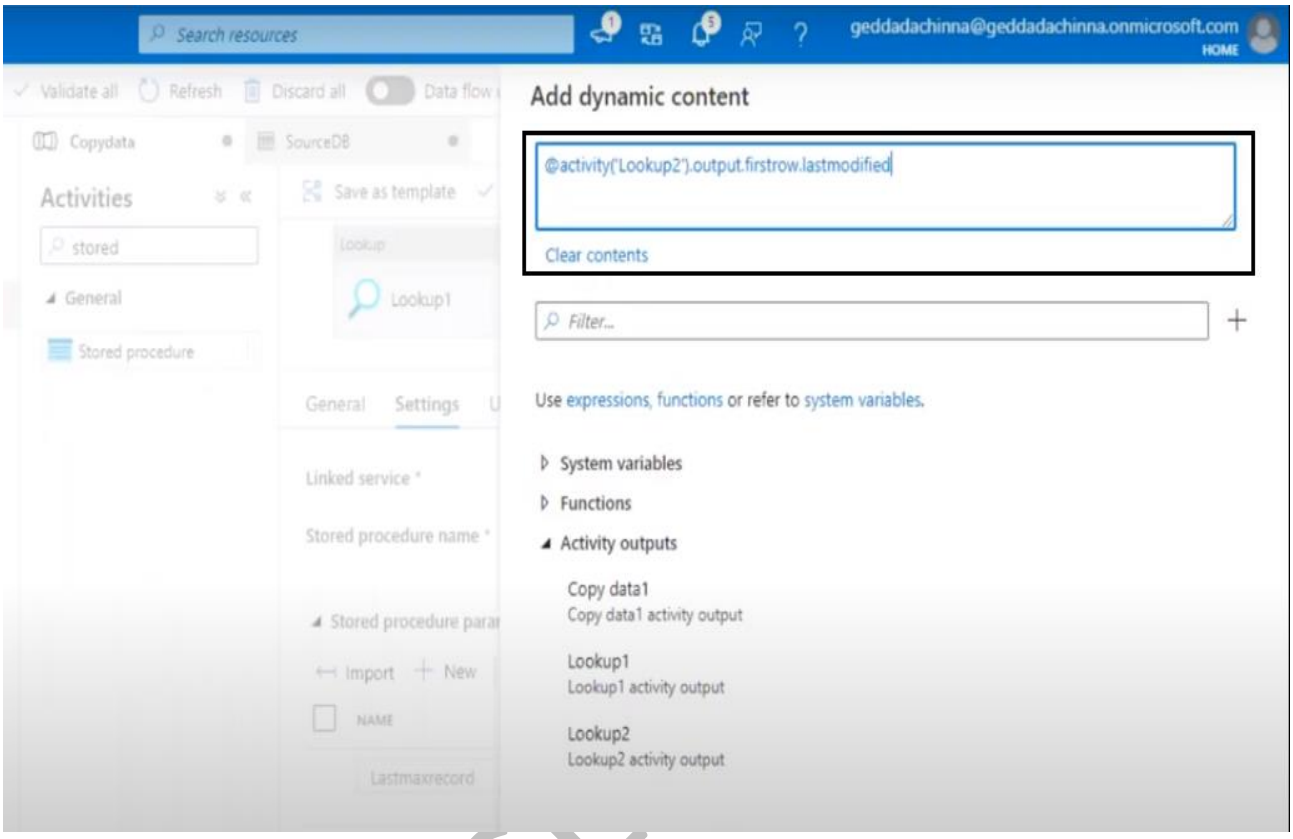
Step 29: Click on Import and Click on Add dynamic content



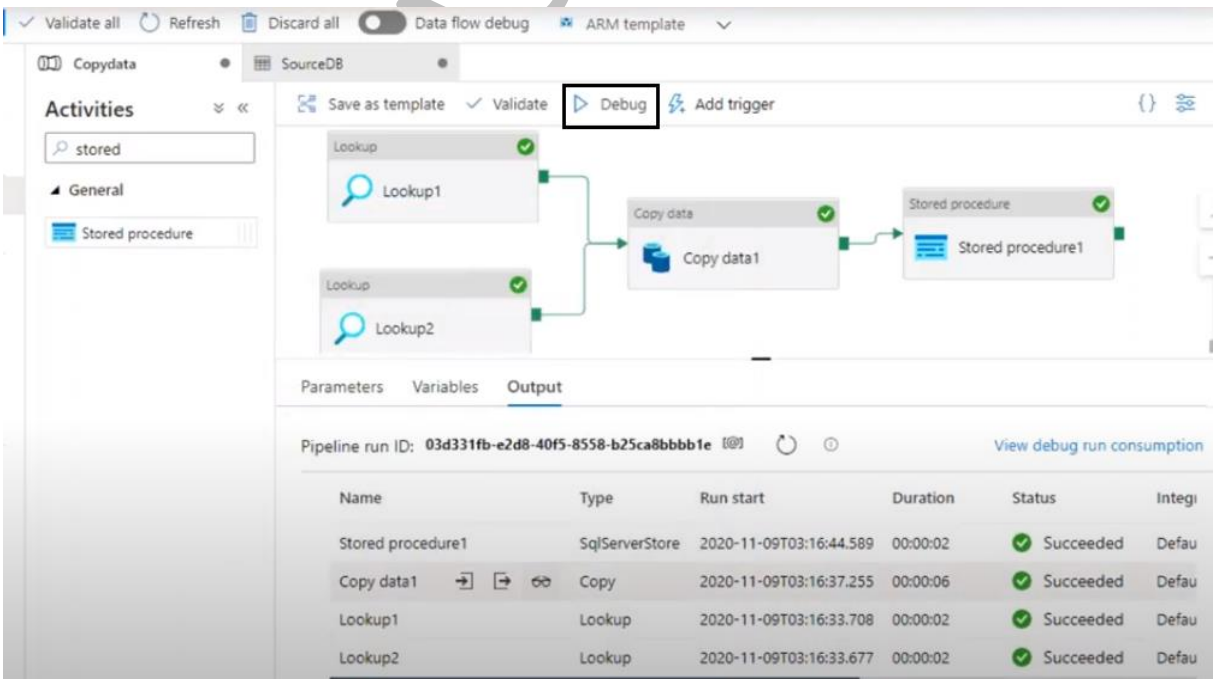
Step 30: Click on Look Activity Output



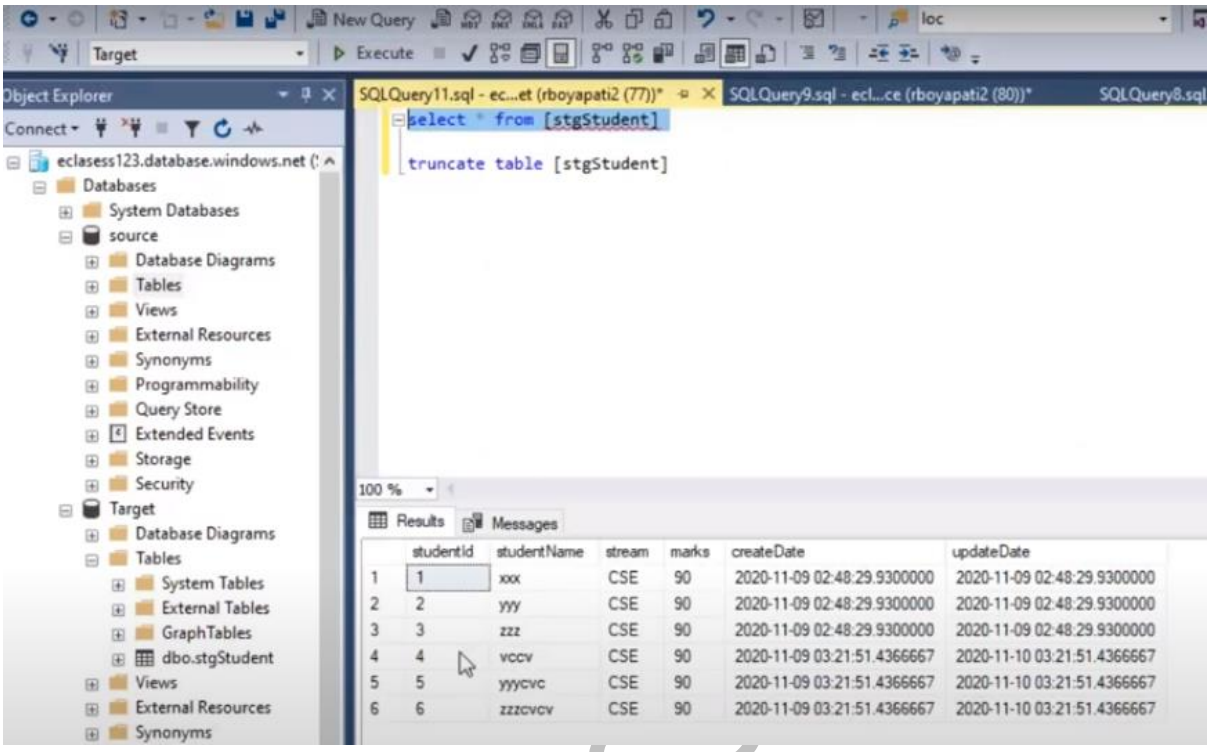
Step 31: Add last modified which is an input to the lookup2 activity (Step 24) at the end of the Query and Click on Finish



Step 32: Click on Debug and Check the Output



Step 33: Write a Query to verify whether the data loaded into dbo.StgStudent table or not



	studentId	studentName	stream	marks	createDate	updateDate
1	1	xxx	CSE	90	2020-11-09 02:48:29.9300000	2020-11-09 02:48:29.9300000
2	2	yyy	CSE	90	2020-11-09 02:48:29.9300000	2020-11-09 02:48:29.9300000
3	3	zzz	CSE	90	2020-11-09 02:48:29.9300000	2020-11-09 02:48:29.9300000
4	4	vccv	CSE	90	2020-11-09 03:21:51.4366667	2020-11-10 03:21:51.4366667
5	5	yyycvc	CSE	90	2020-11-09 03:21:51.4366667	2020-11-10 03:21:51.4366667
6	6	zzzcvcv	CSE	90	2020-11-09 03:21:51.4366667	2020-11-10 03:21:51.4366667

How to Implement SLOWLY CHANGING DIMENSION METHOD – 1

We can insert new records and update old records by using SCD-1 method. In this method

We have to use Data flow activity for transformation.

SCD Type 0 - Inserting New records.

SCD Type 1 - Inserting new records and Overwriting the old values if any update available.

SCD Type 2 - Creating a new additional record without over writing existing record by using Flag or start date and end date.

Prerequisites:

Create Azure SQL Database name it as –Source

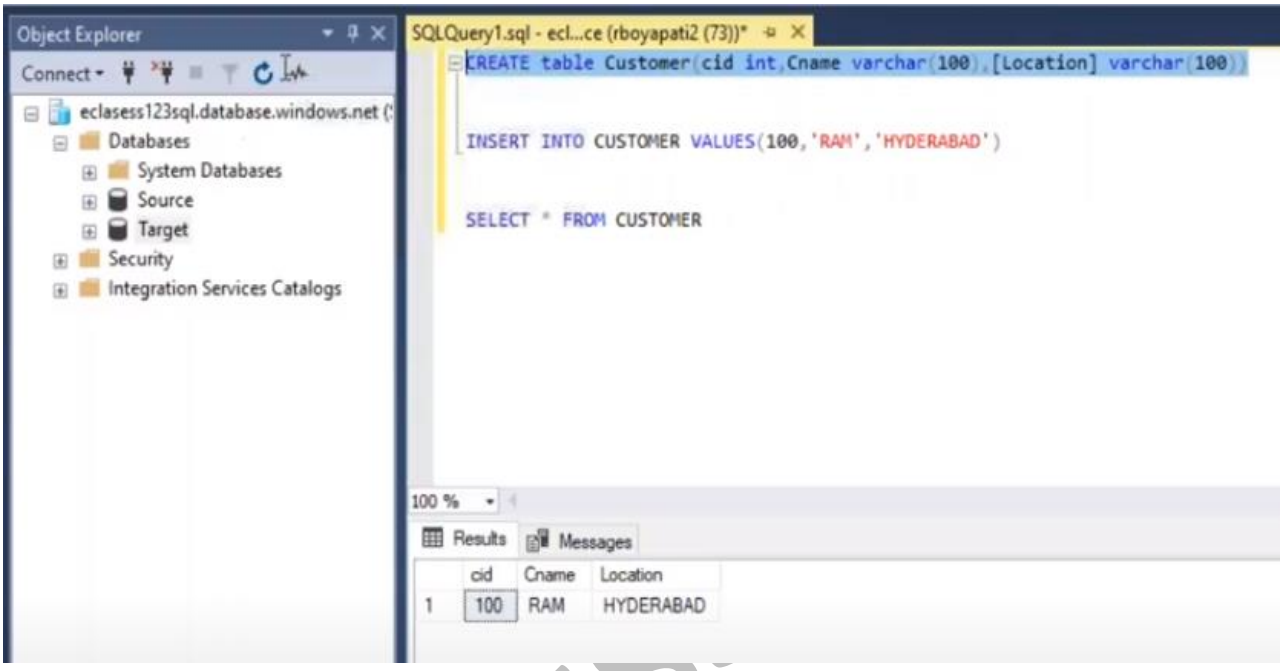
Create table Customer (Cid int, Cname Varchar (100), Location Varchar (100))

Create Another Azure SQL Database name it as –Target

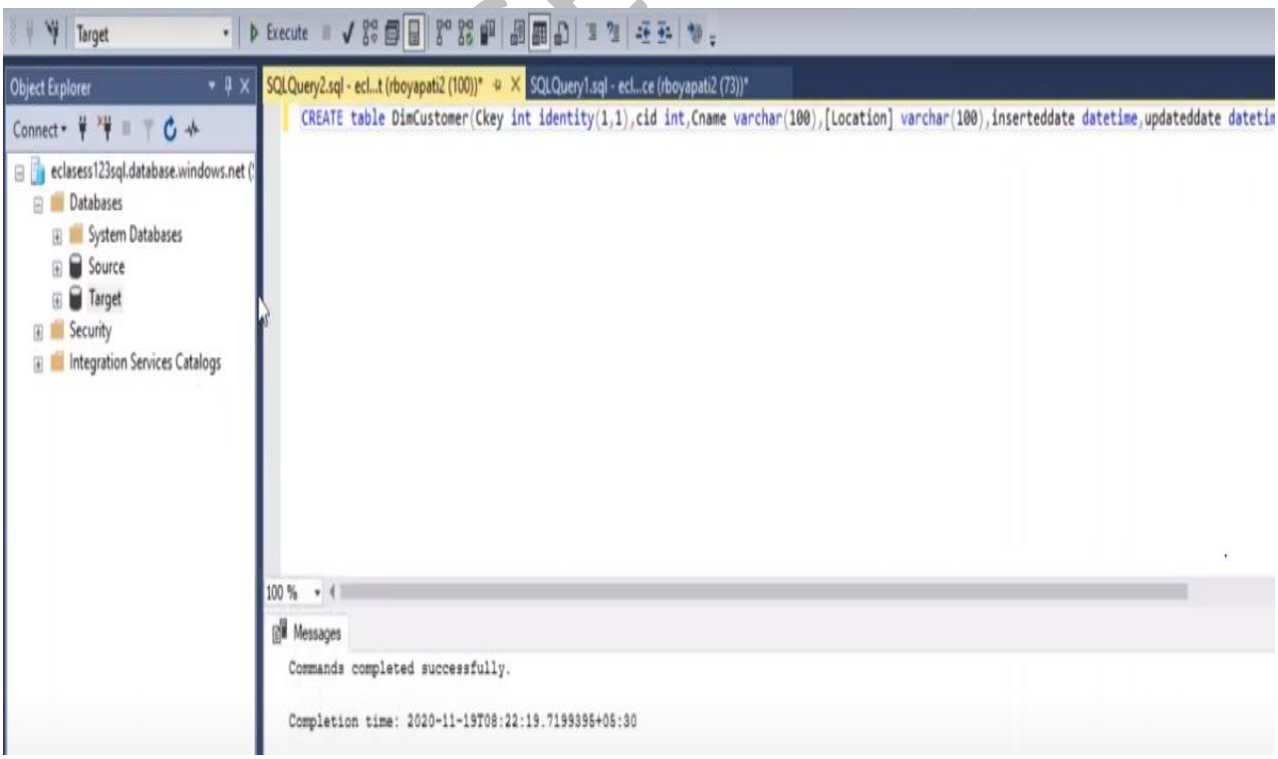
Create table DimCustomer (Ckey int identity (1, 1), Cid int, Cname Varchar (100), Location Varchar (100), inserteddate datetime, Updateddate Datetime)

For example: Loading data from dbo.customer table to dbo.dimcustomer table

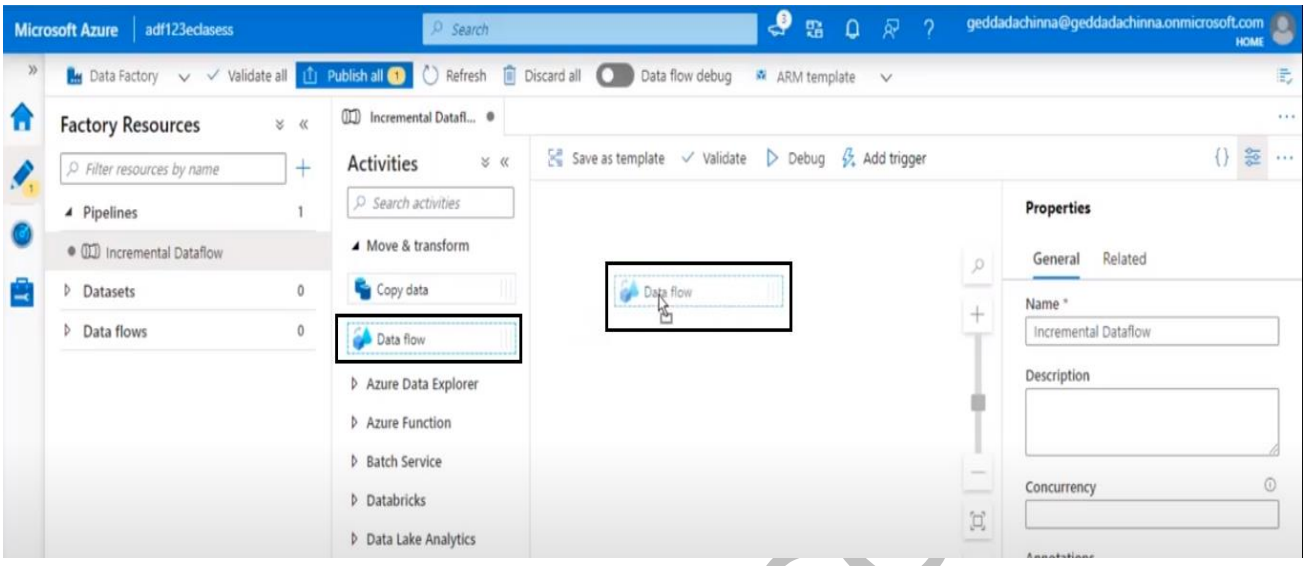
Step 1: Create a dbo.customer table inside Source(azure SQL database) and insert some records



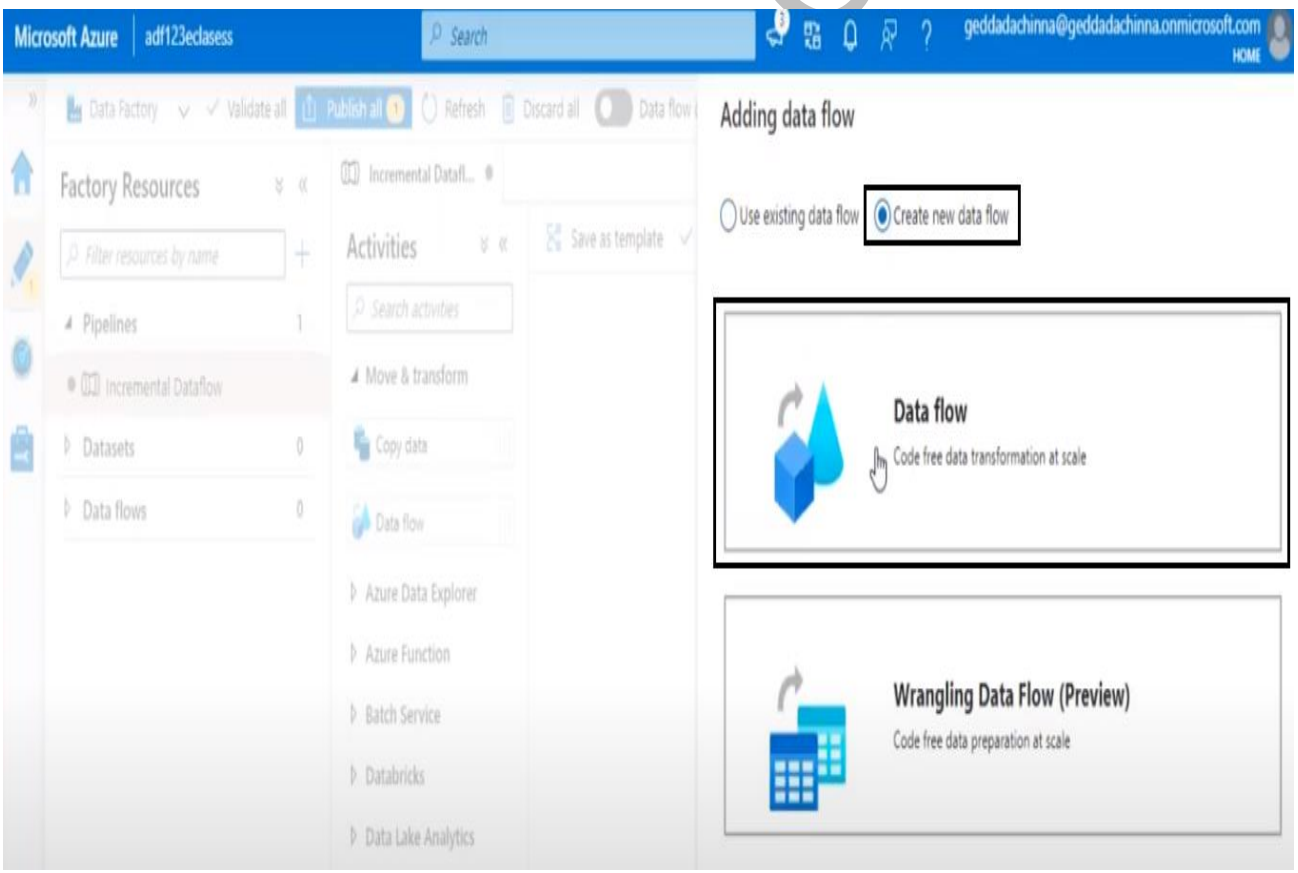
Step 2: Create a dbo.dimcustomer table in Target database (azure SQL database)



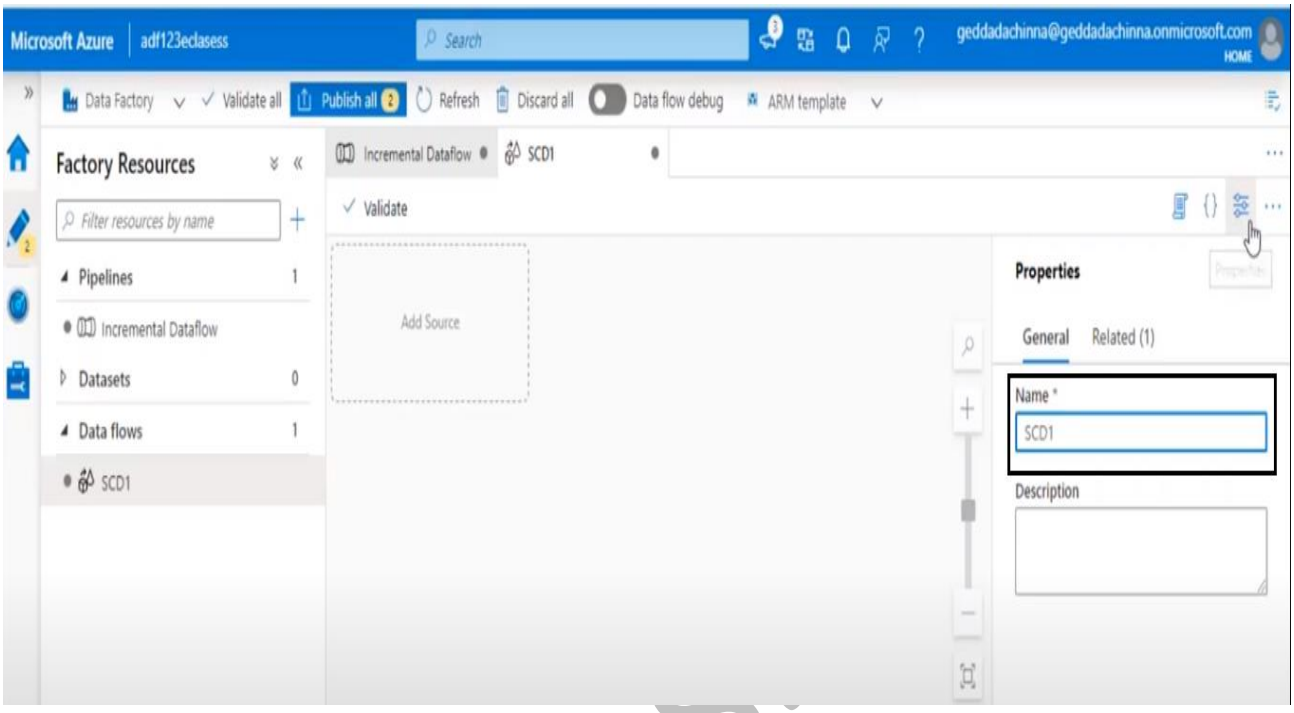
Step 3: Drag dataflow activity into the development area



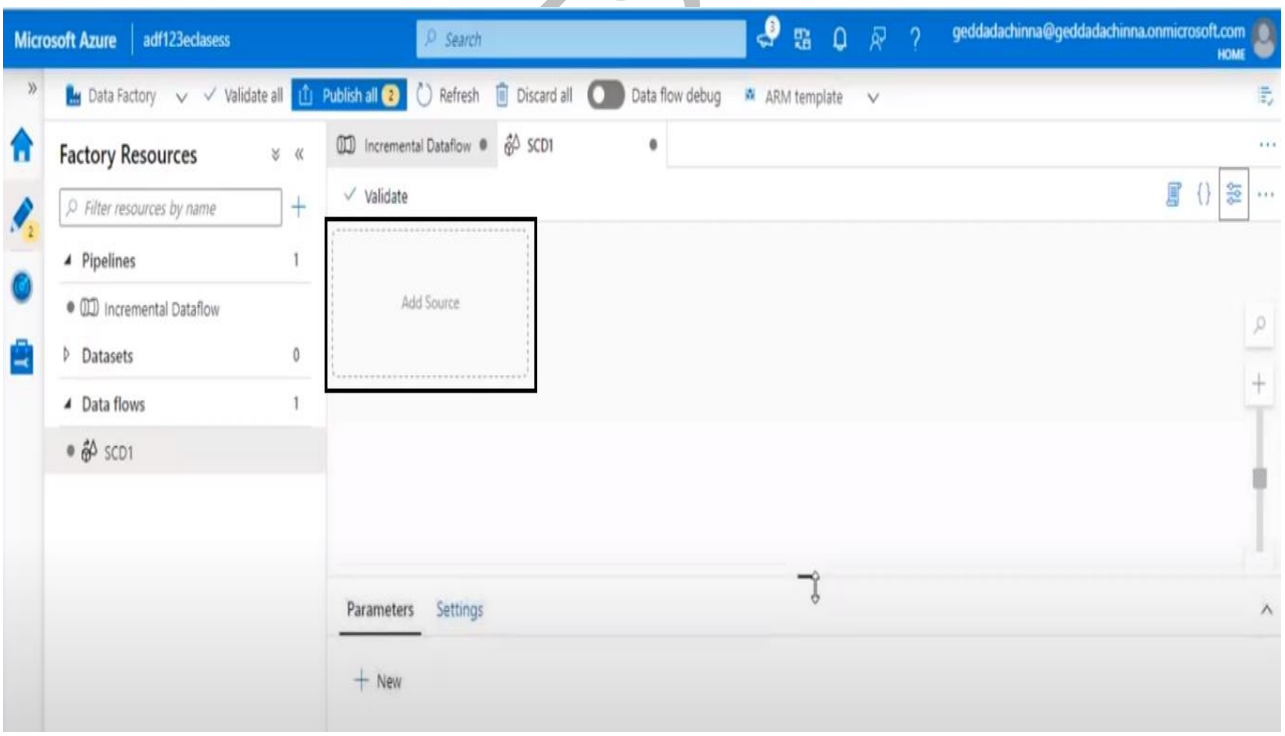
Step 4: Click on Create new dataflow and Select Data flow and Click on OK



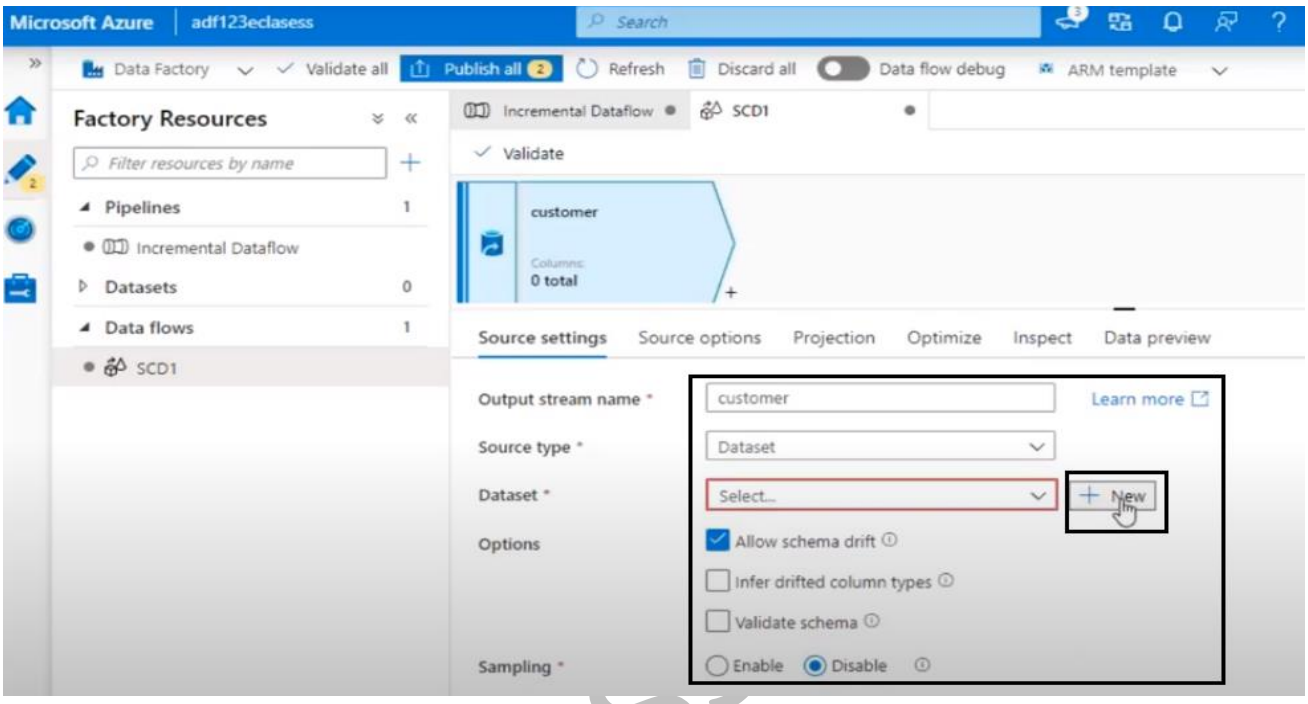
Step 5: Give a proper name for the dataflow



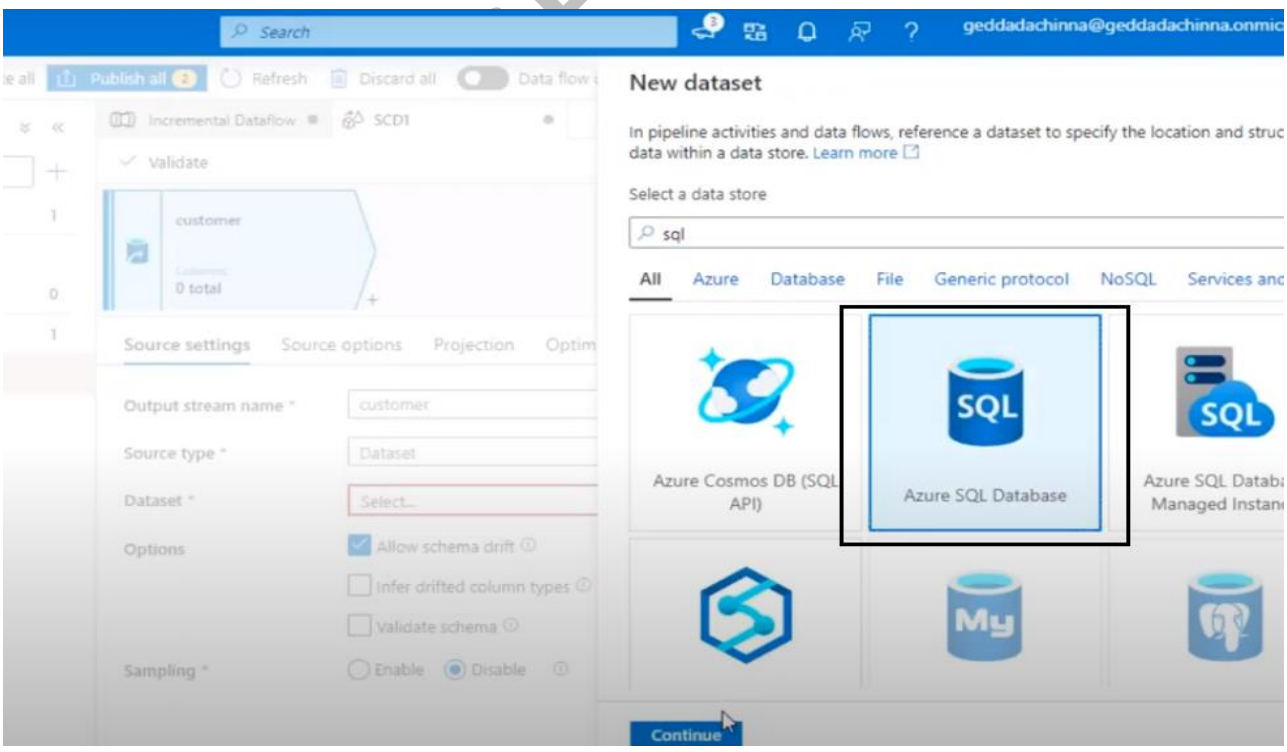
Step 6: Click on add source



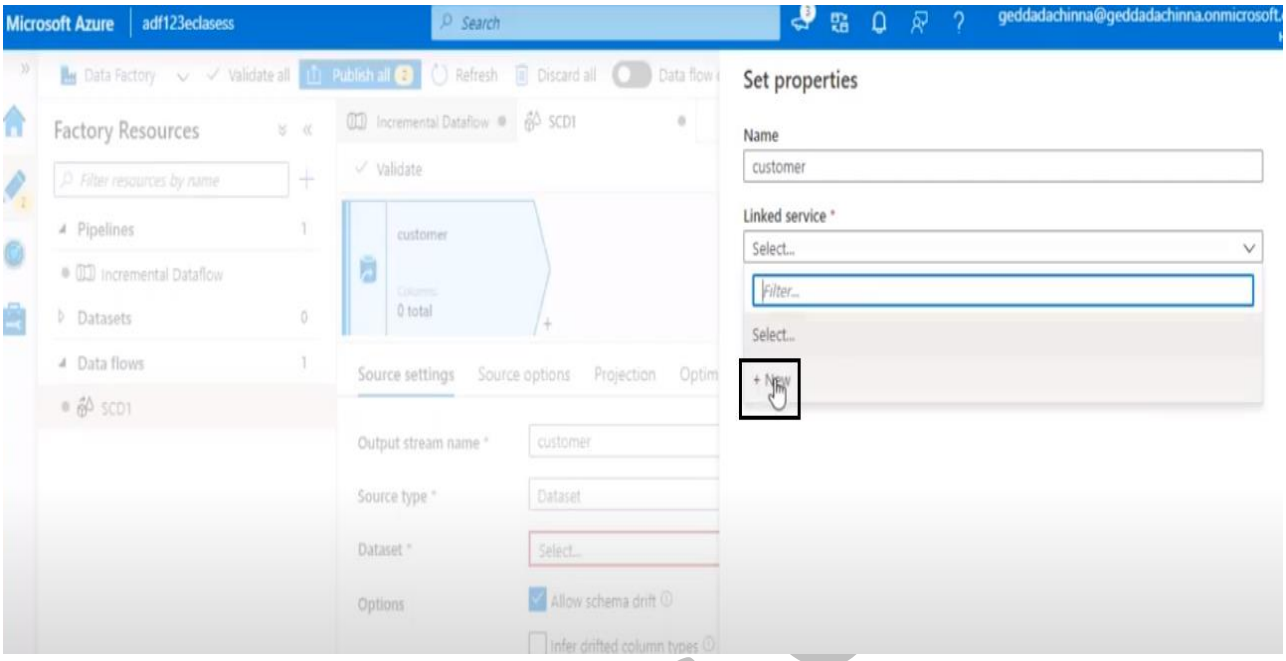
Step 7: Give Output stream name---Select source type as dataset-----Click on New



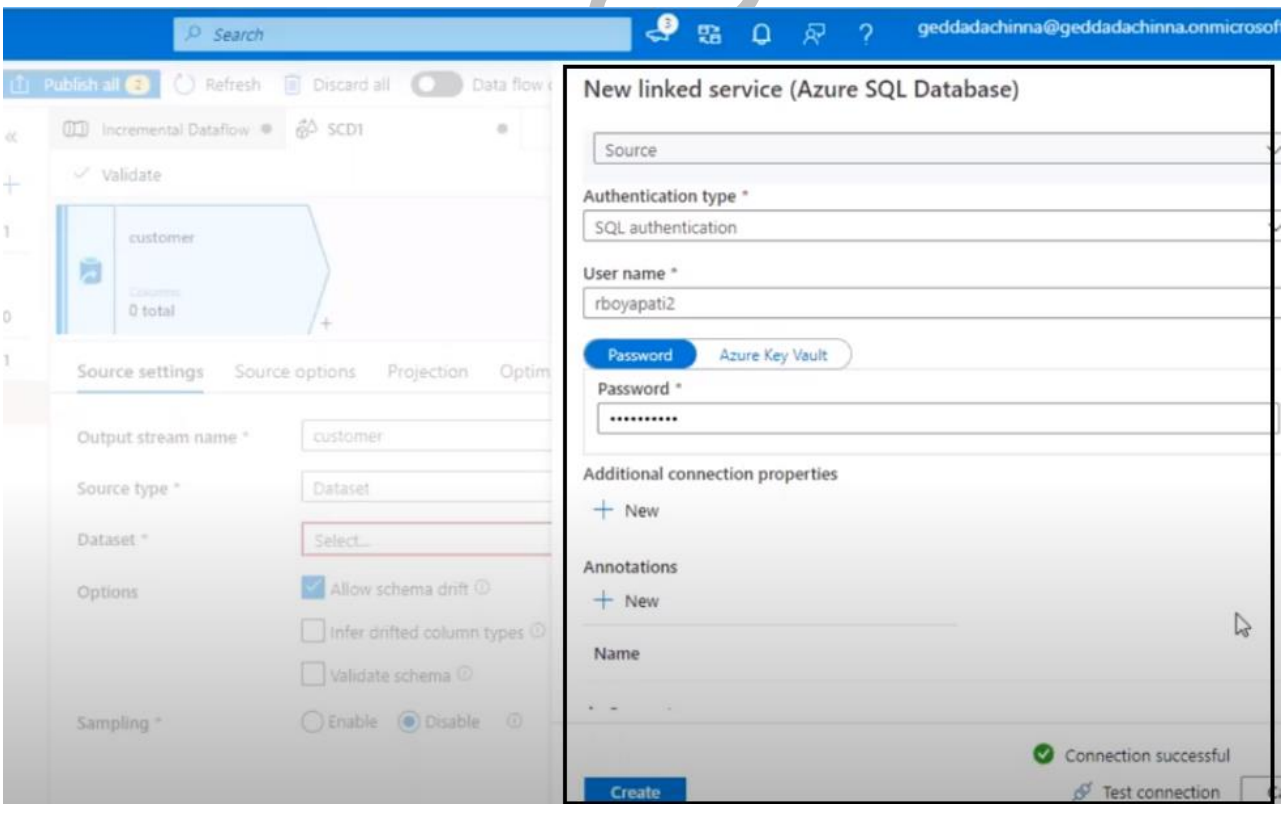
Step 8: Click on Azure SQL database and Continue



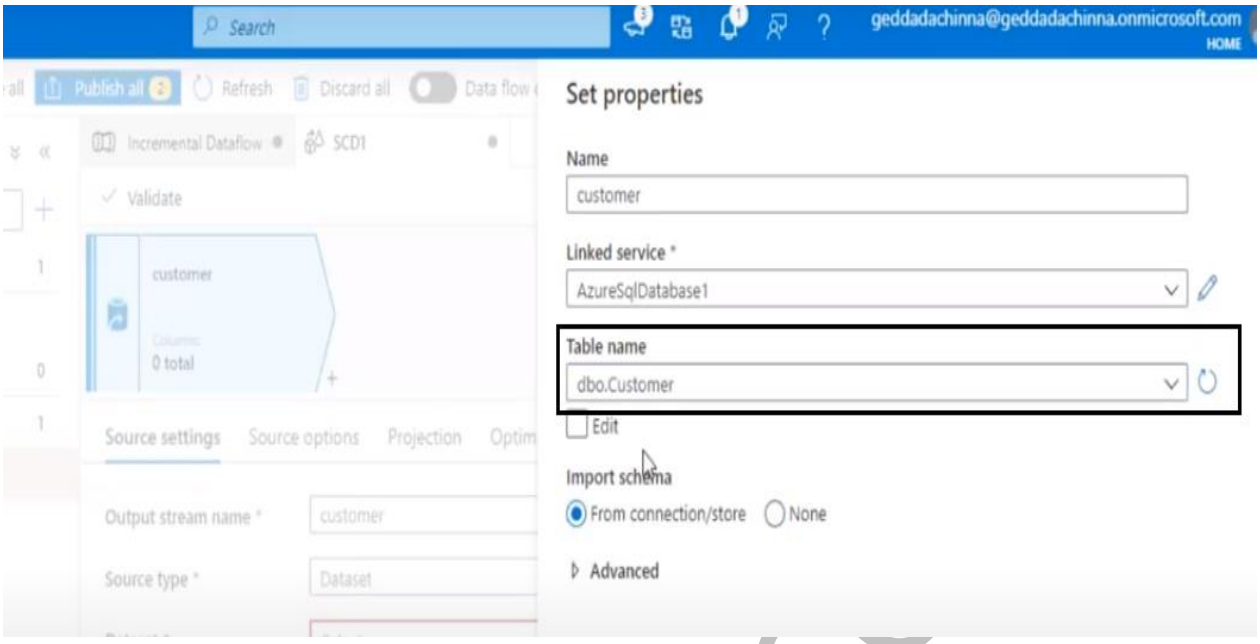
Step 9: Give a proper name for the source dataset---Click on Linked Service---Click on New



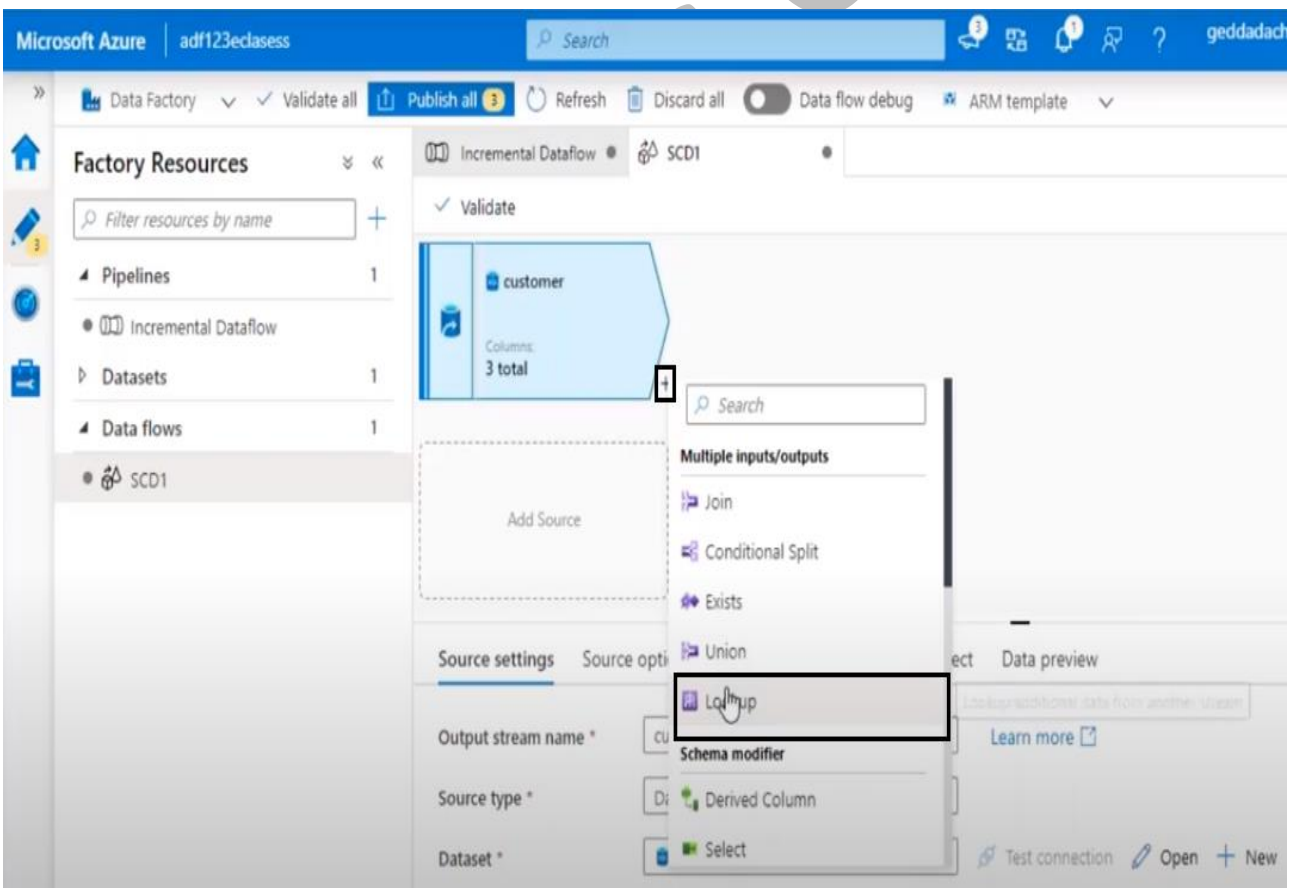
Step 10: Give a proper name---Select subscription---Select database name---Select Authentication type---Give user and password----Test Connection----Click on Create



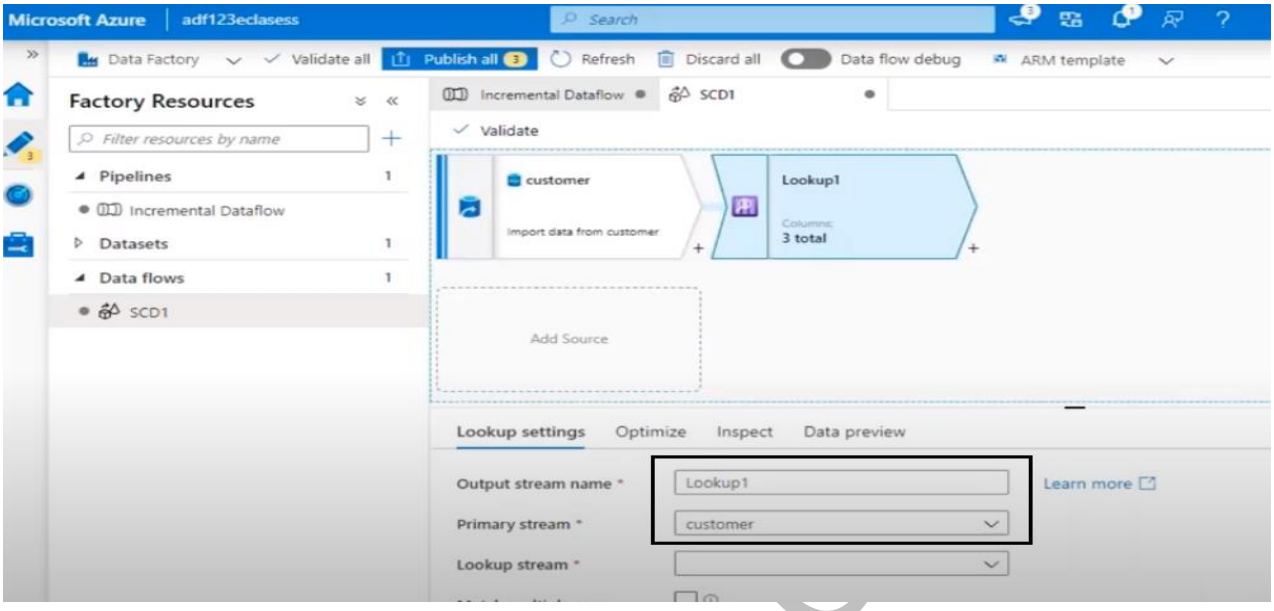
Step 11: Select table name as dbo.customer and click on OK.



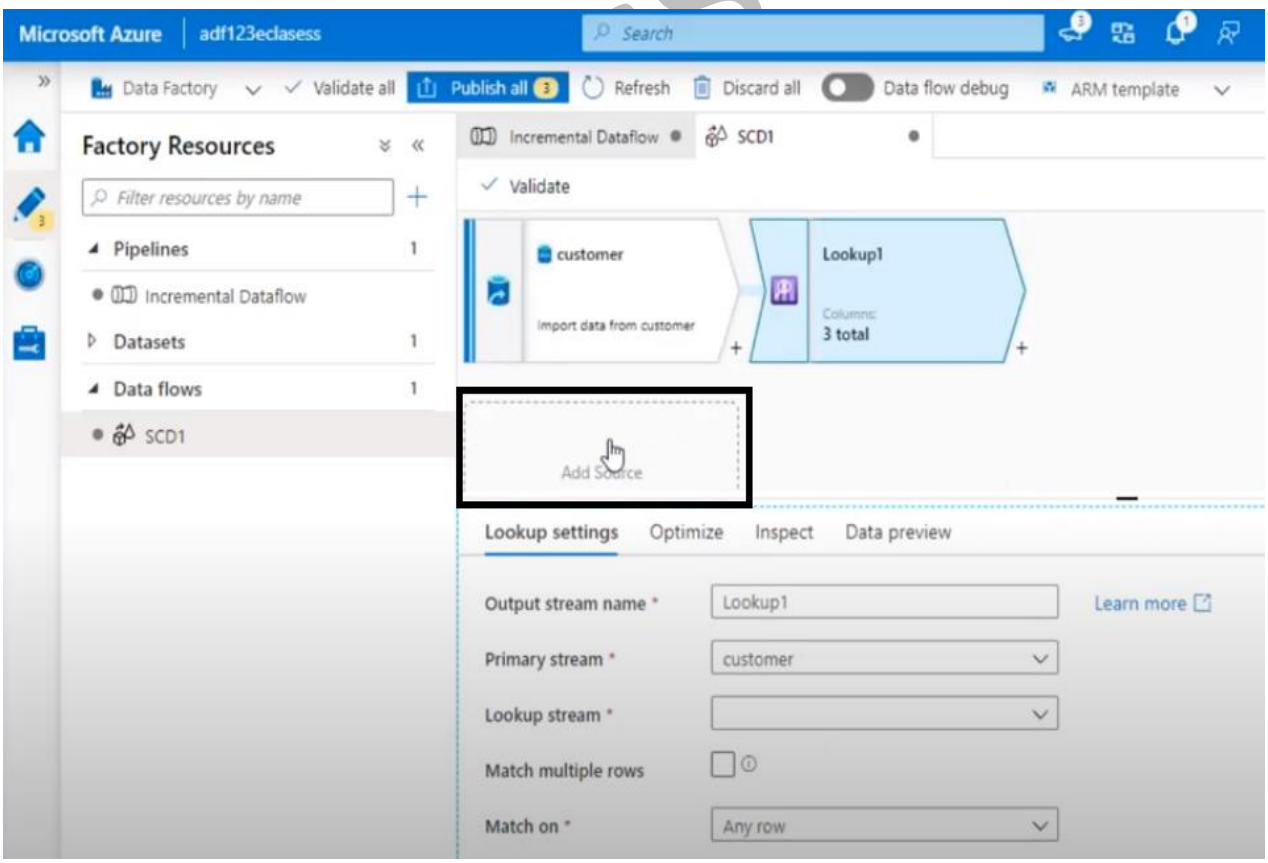
Step 12: Click on '+' symbol and Click on lookup transformation



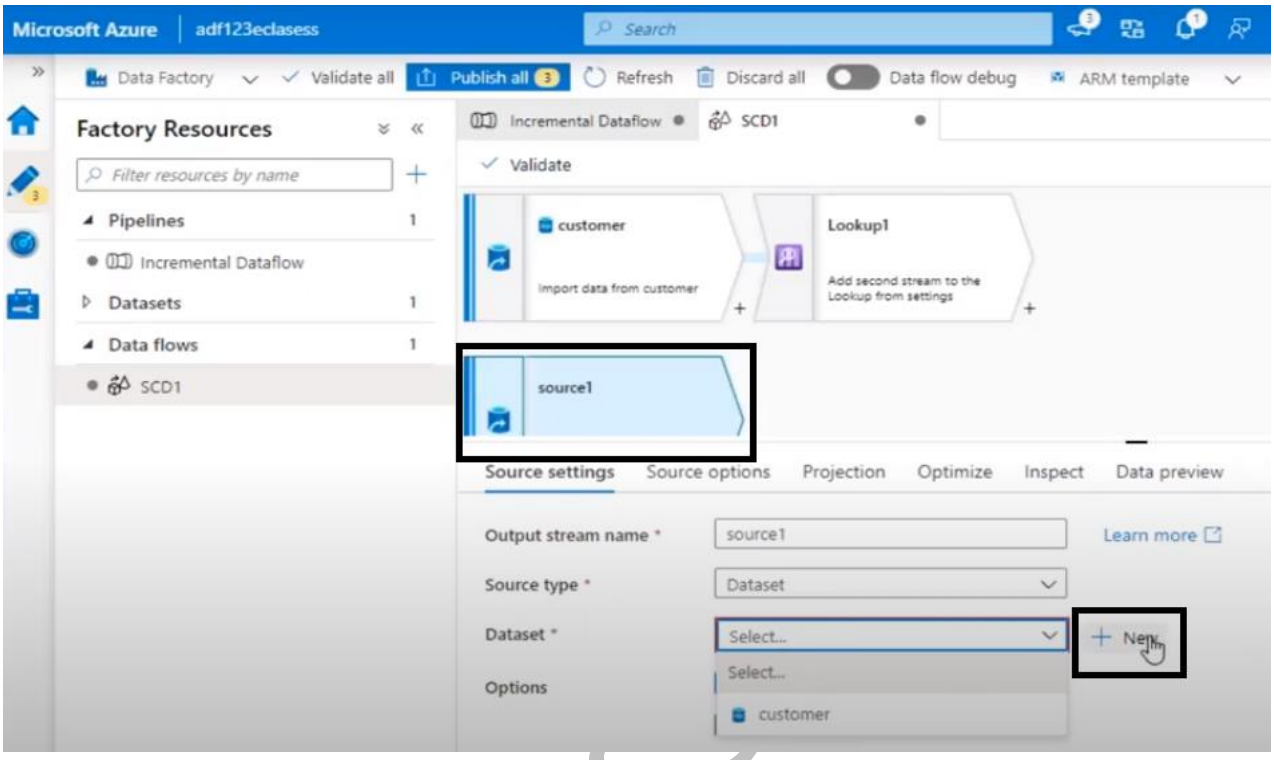
Step 13: Give a proper name for Output stream name in Lookup transformation and Select primary stream as customer.



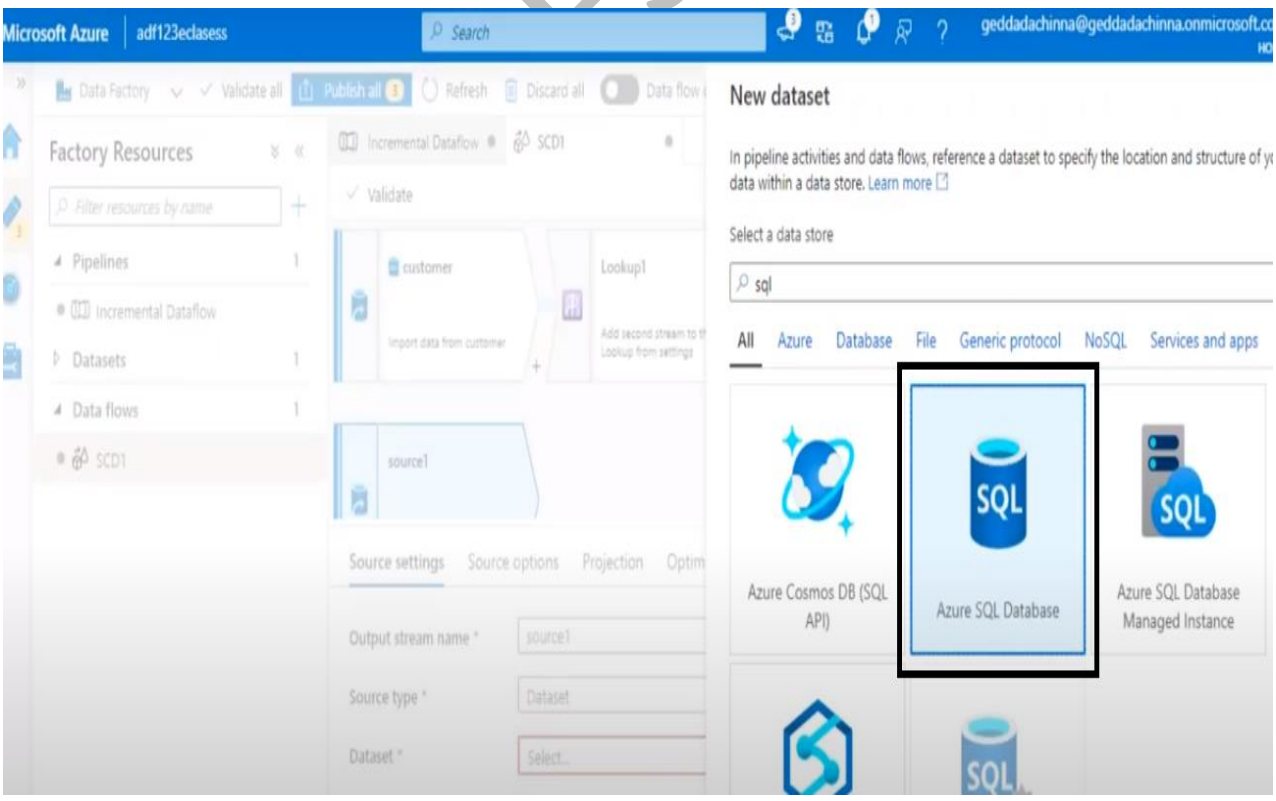
Step 14: Click on Add Source



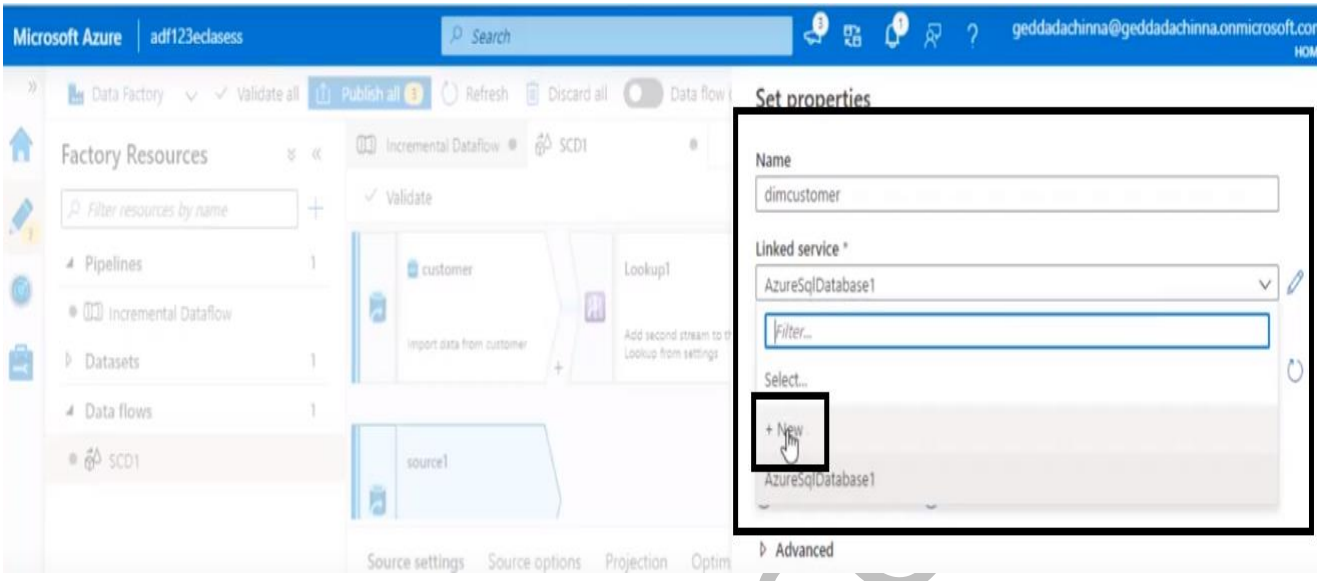
Step 15: Click on New



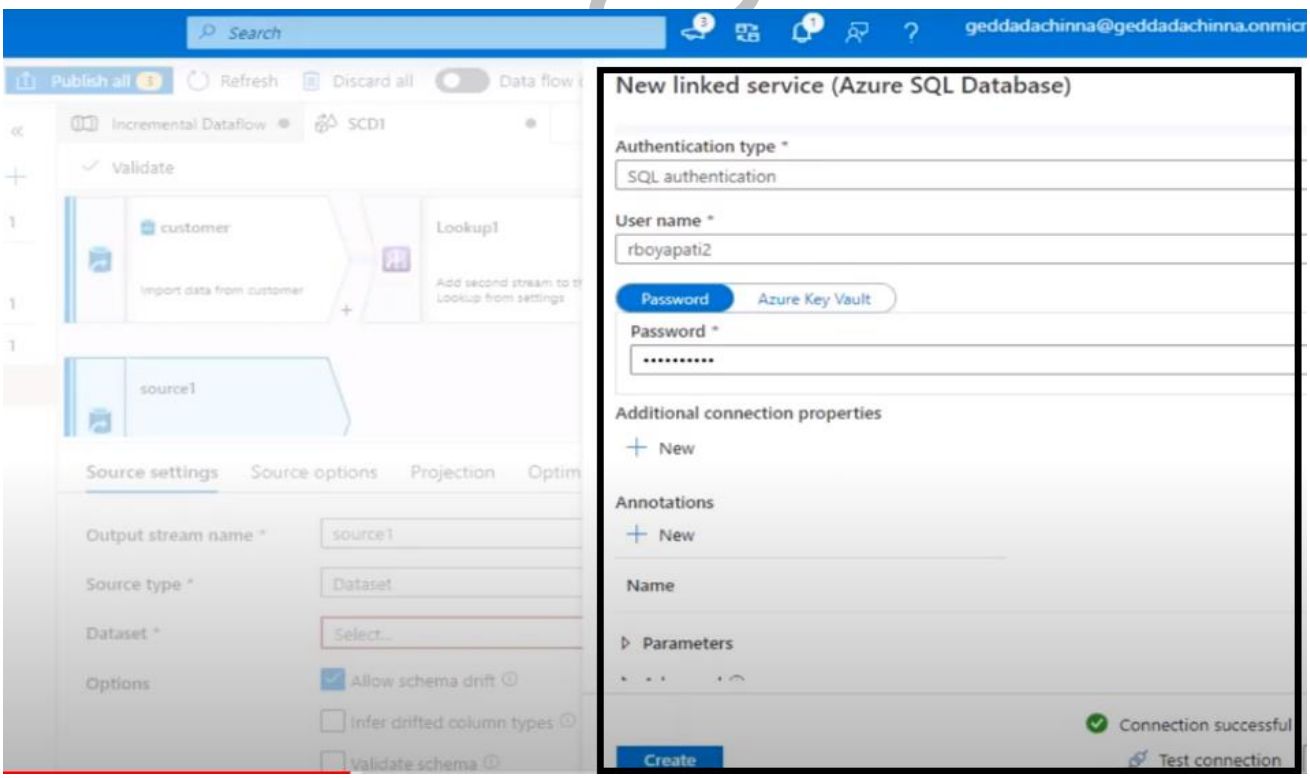
Step 16: Click on Azure SQL Database and Continue



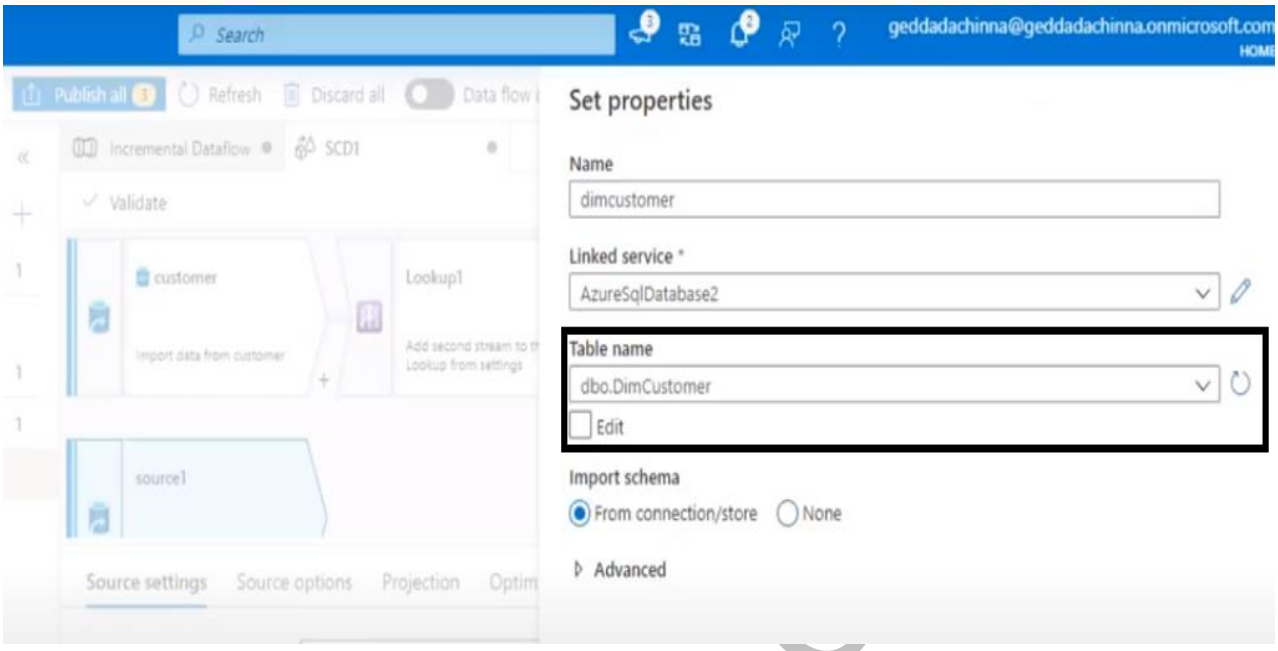
Step 17: Give a proper name for the target dataset----Click on Linked Service----Select New



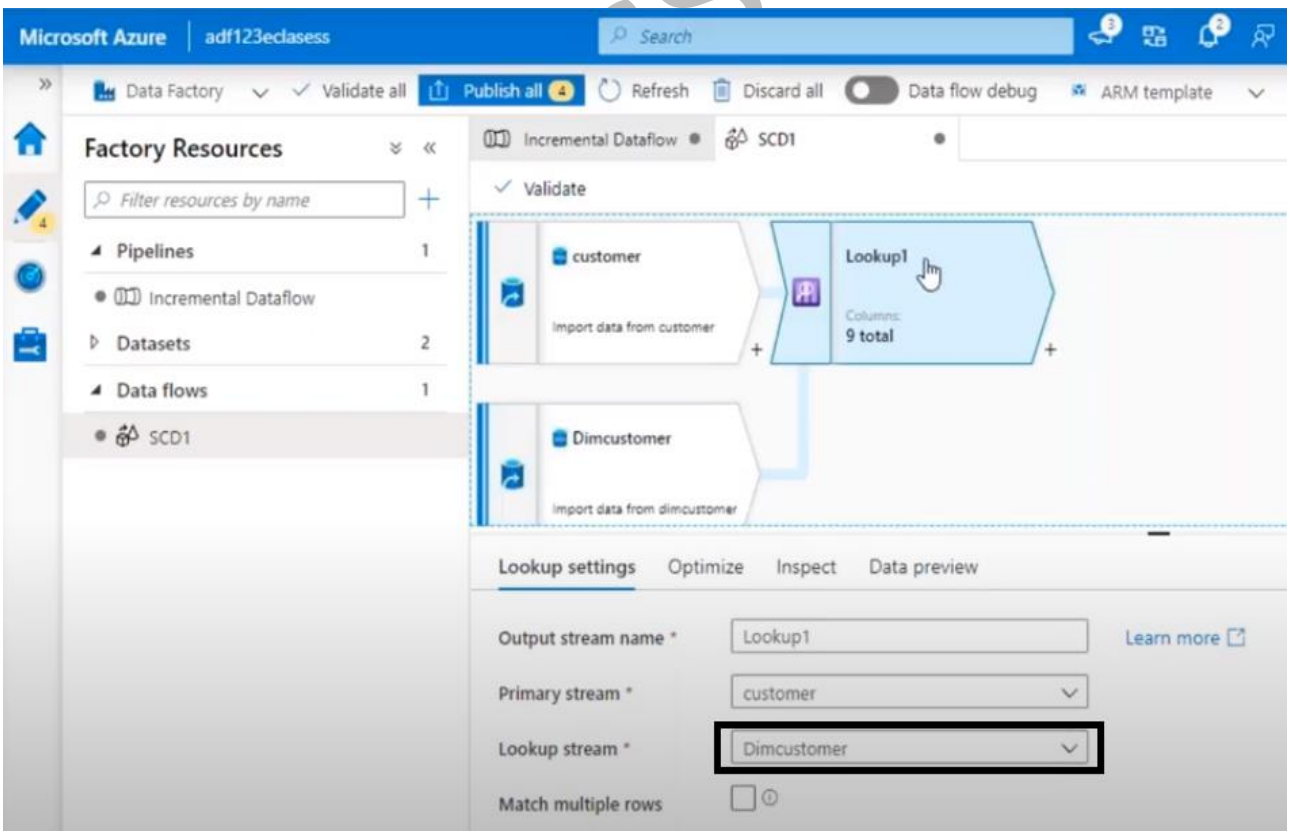
Step 18: Give a proper name---Select subscription---Select database name---Select Authentication type---Give user and password----Test Connection----Click on Create



Step 19: Select dbo.Dimcustomer table and Click on OK

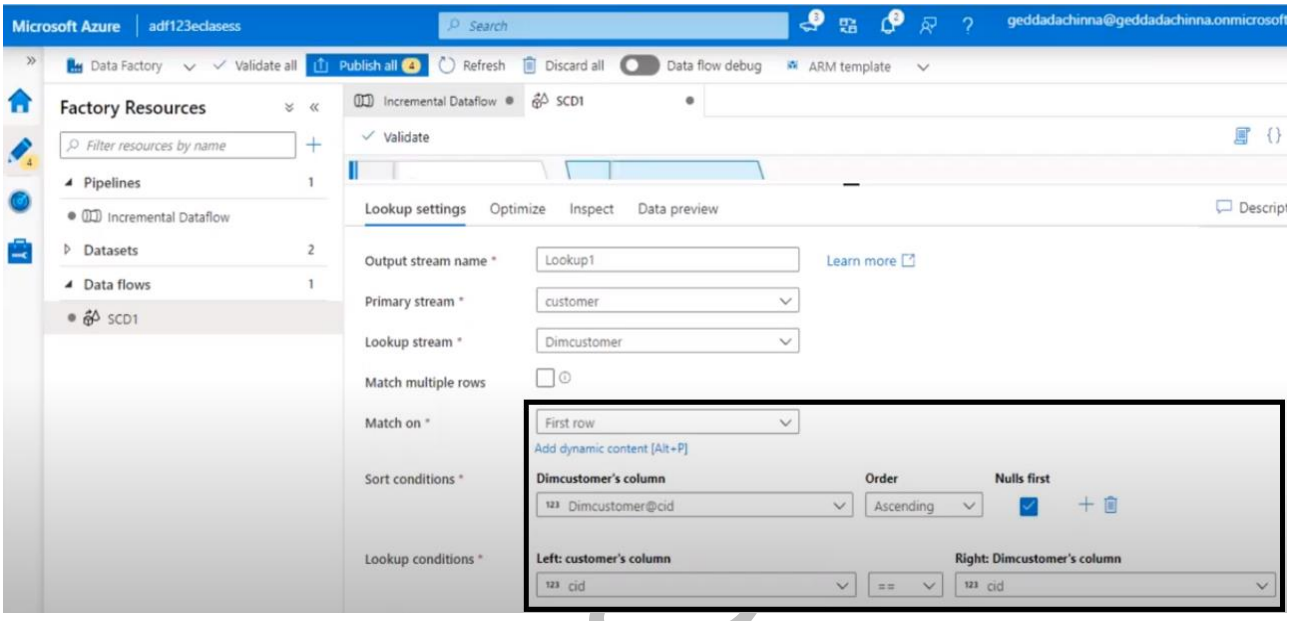


Step 20: Click on Lookup Stream----Select Lookup stream as dim customer

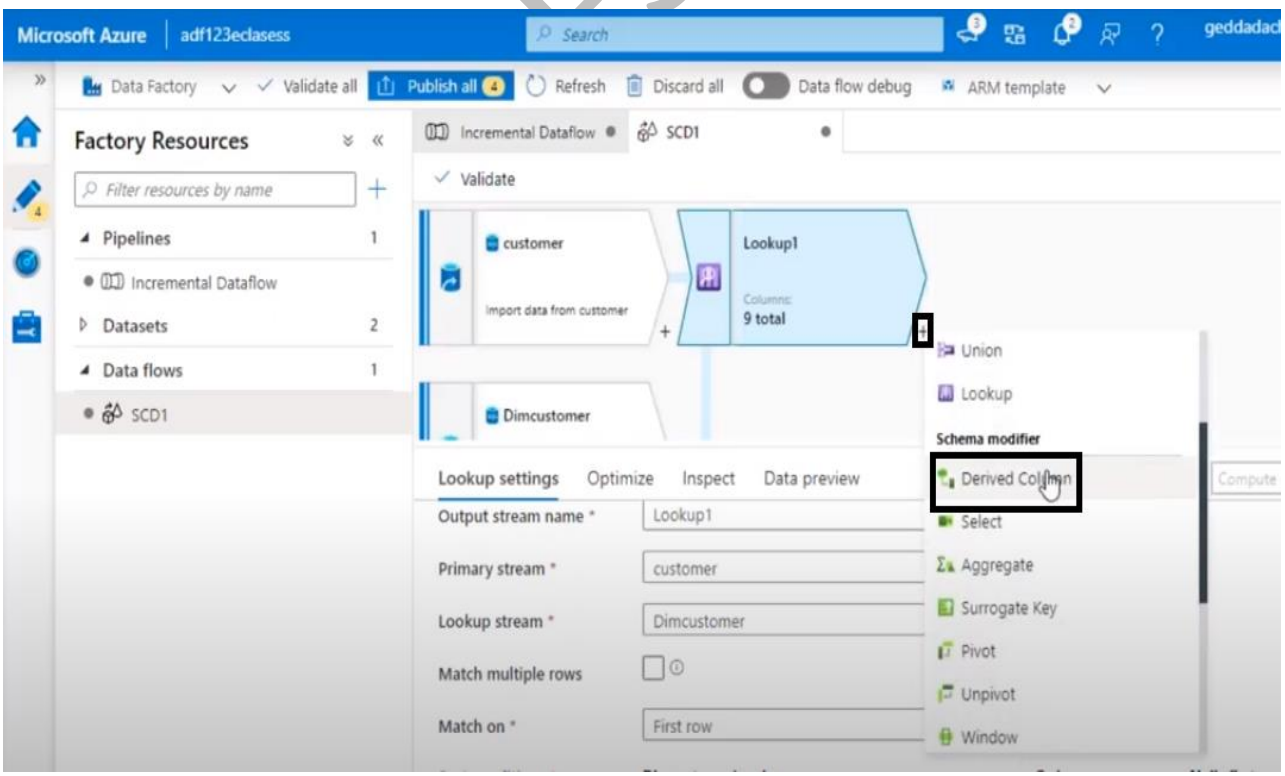


Step 21: Go to Lookup conditions---- Select cid column from the customer table on the left side-----Select cid column from the dimcustomer column on the right side.

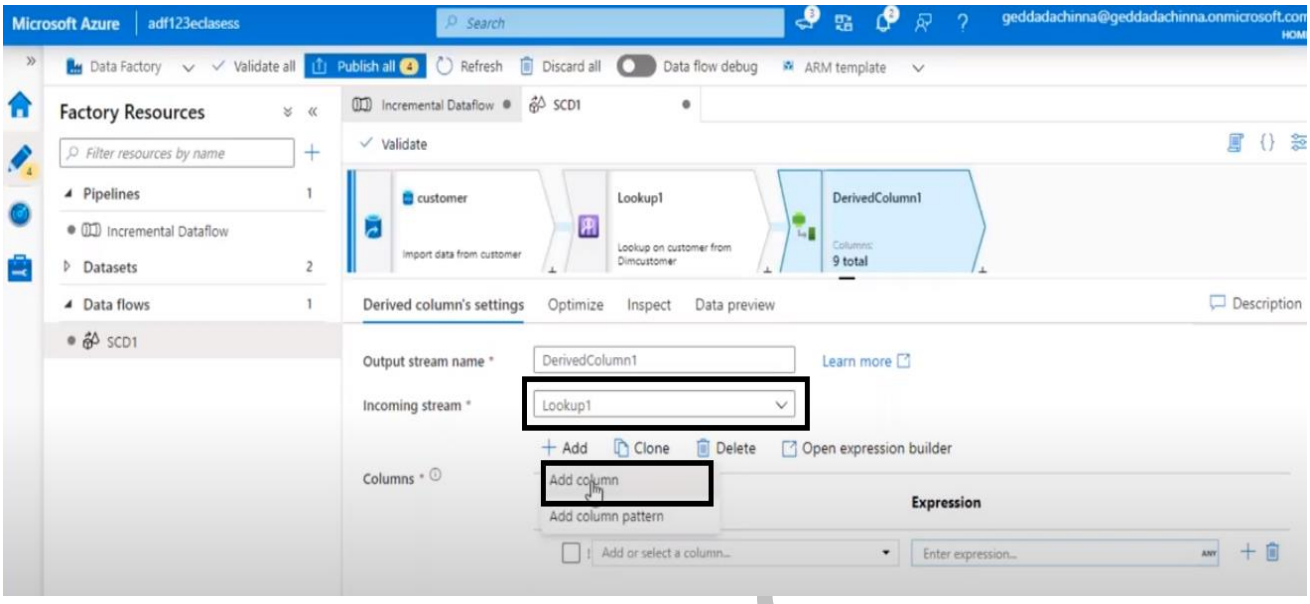
Go to Match on---Select First row-----Go to Sort conditions----Select Dimcustomer@cid----Select Order as Ascending



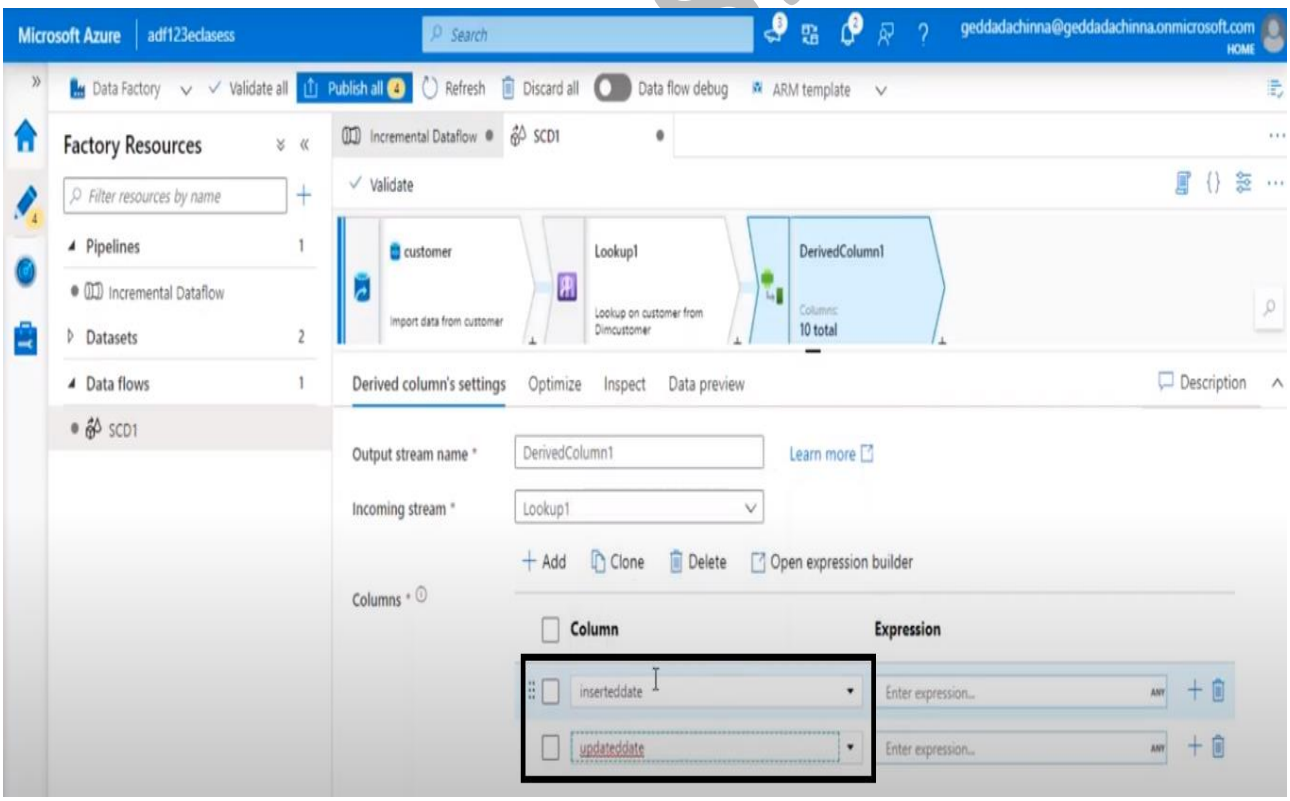
Step 22: Click on '+' symbol---Click on Derived Column



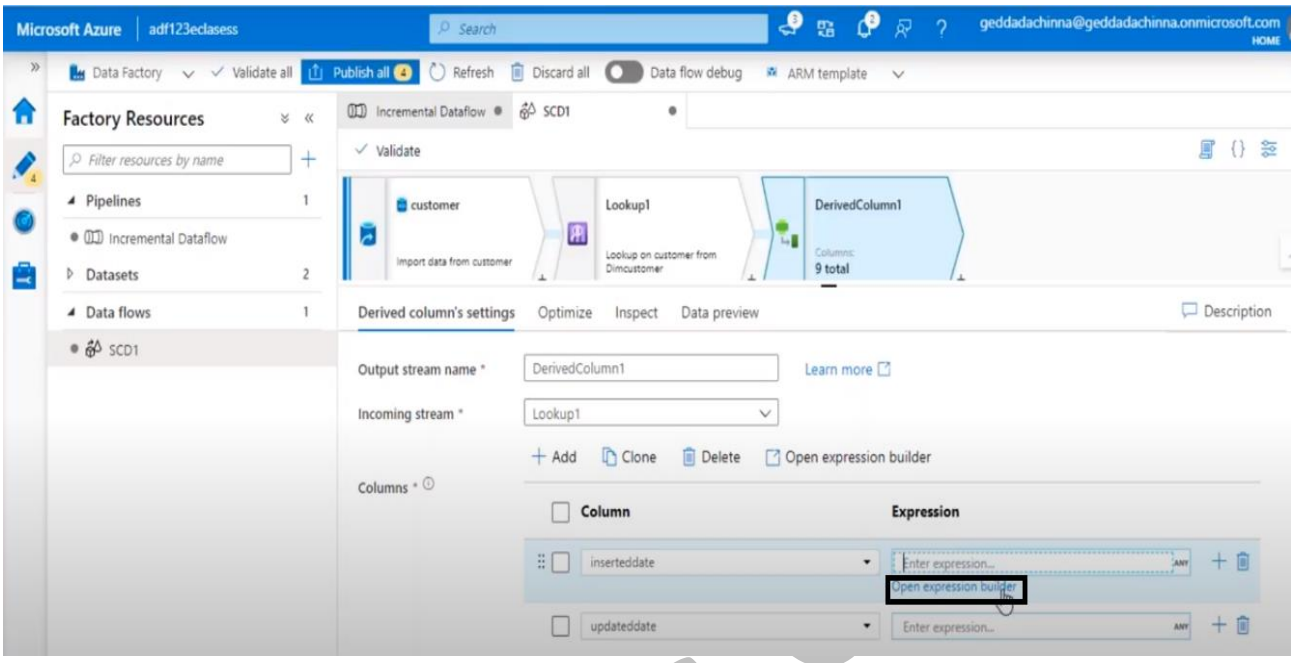
Step 23: Select Incoming stream---Go to Columns----Click on Add Column



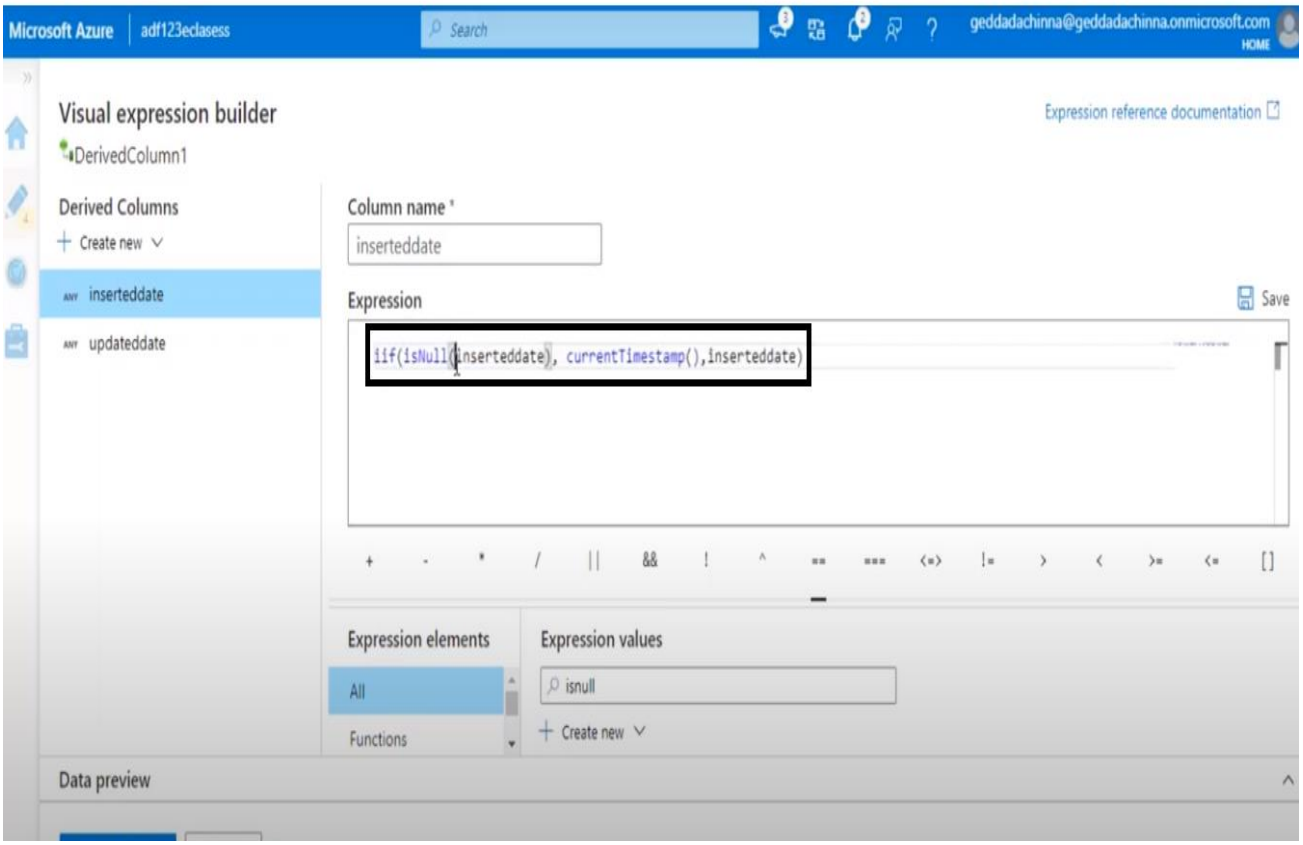
Step 24: Select inserteddate in the first row and updateddate in the second row (inserteddate and updateddate are present in the dimcustomer table)



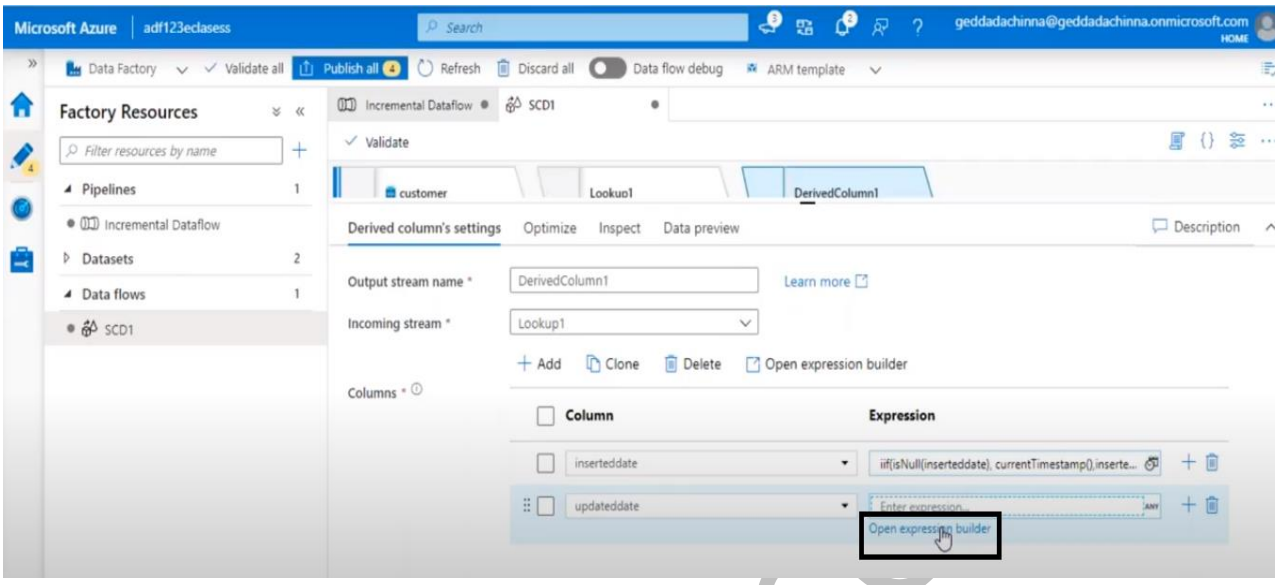
Step 25: Click on Open Expression Builder of the inserteddate



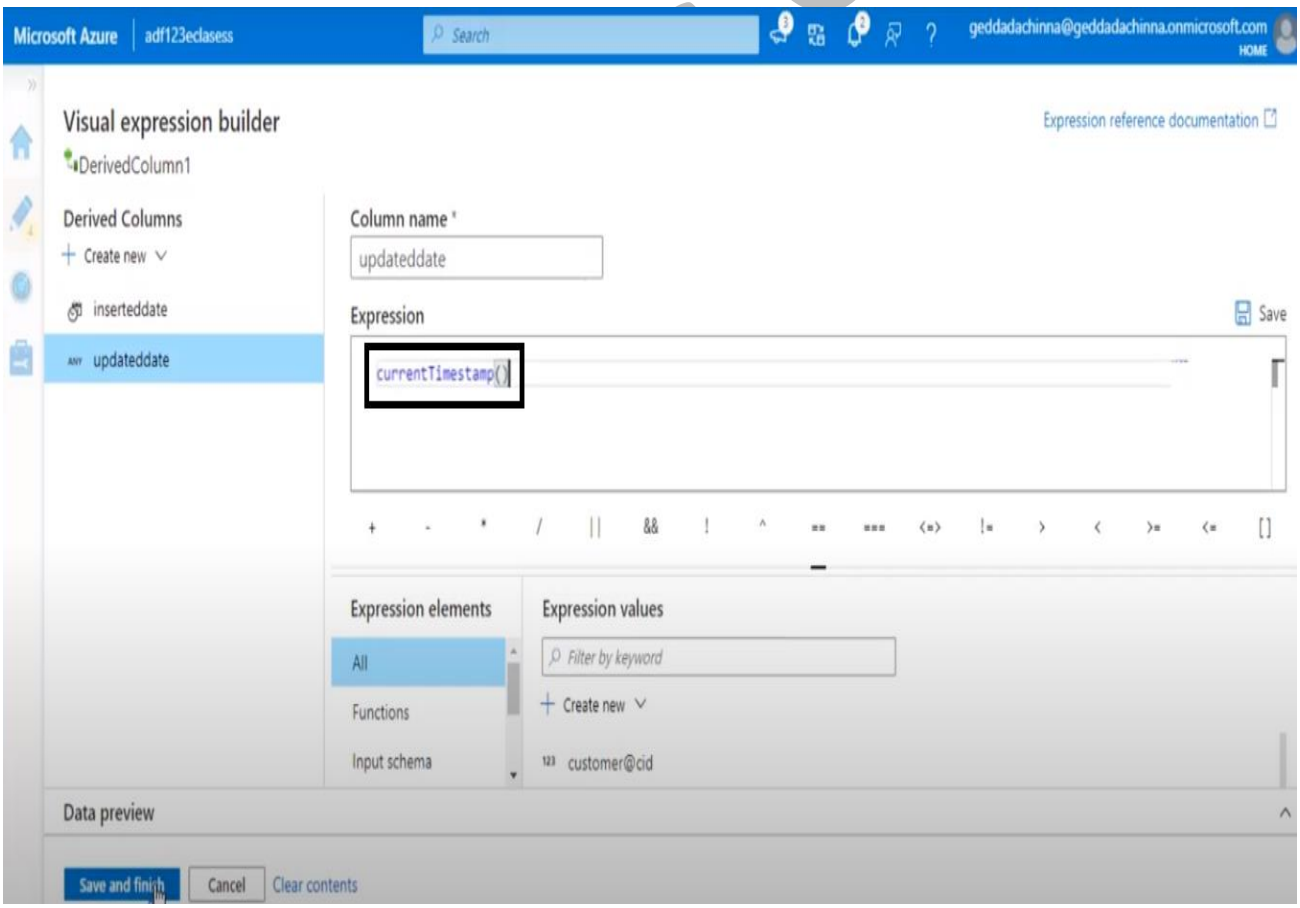
Step 26: Write an Expression----Click on Save and Finish



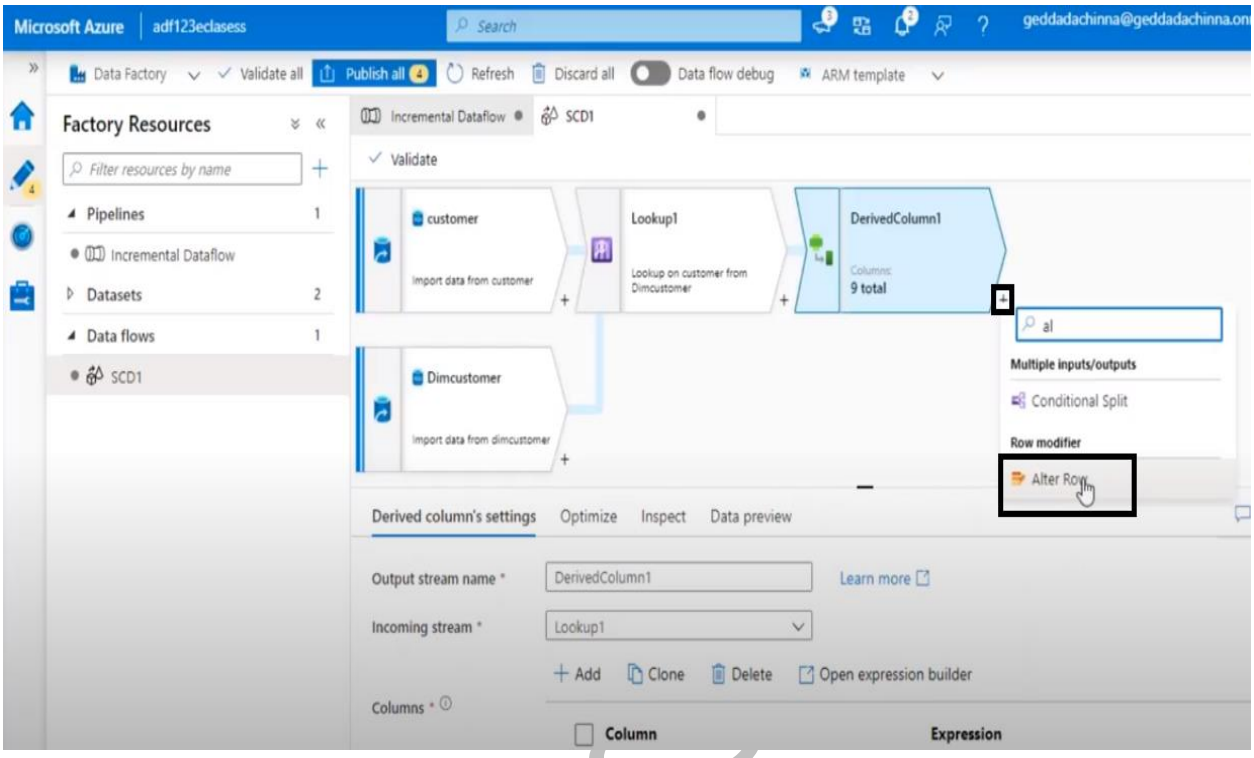
Step 27: Click on open expression builder of the updateddate



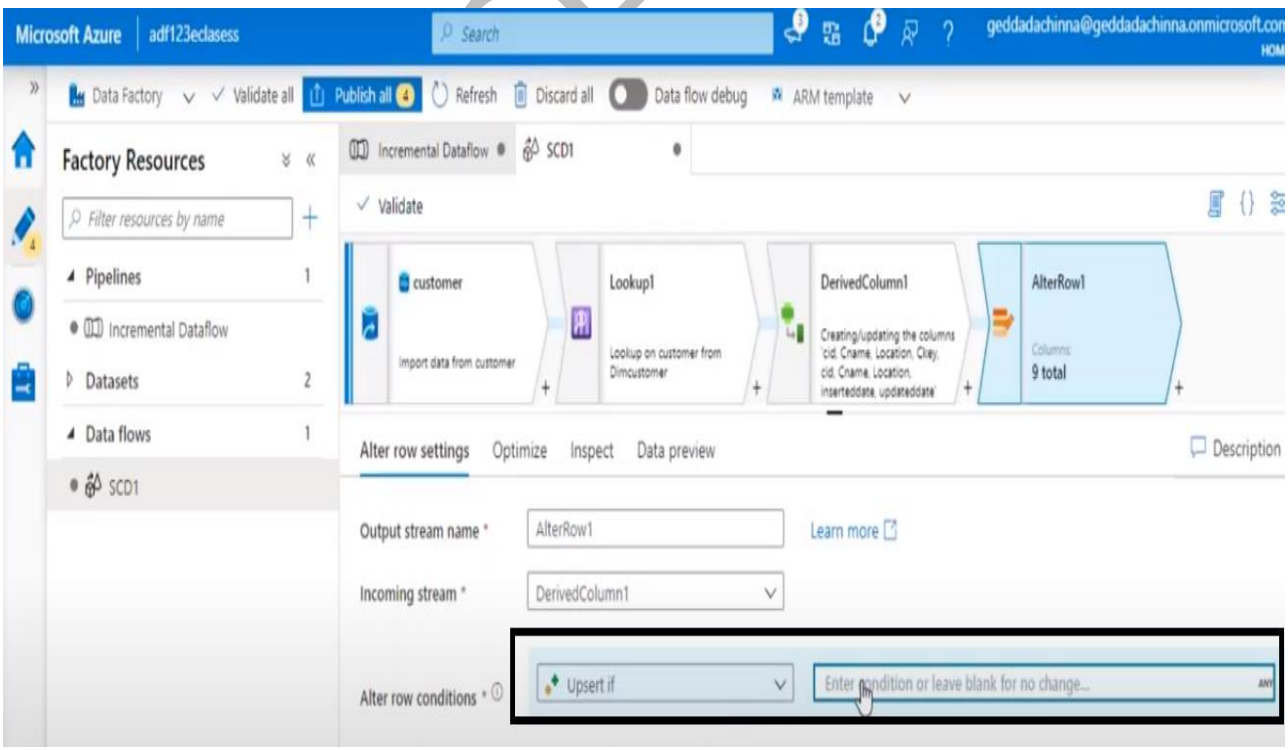
Step 28: Write an expression---Click on Save and Finish



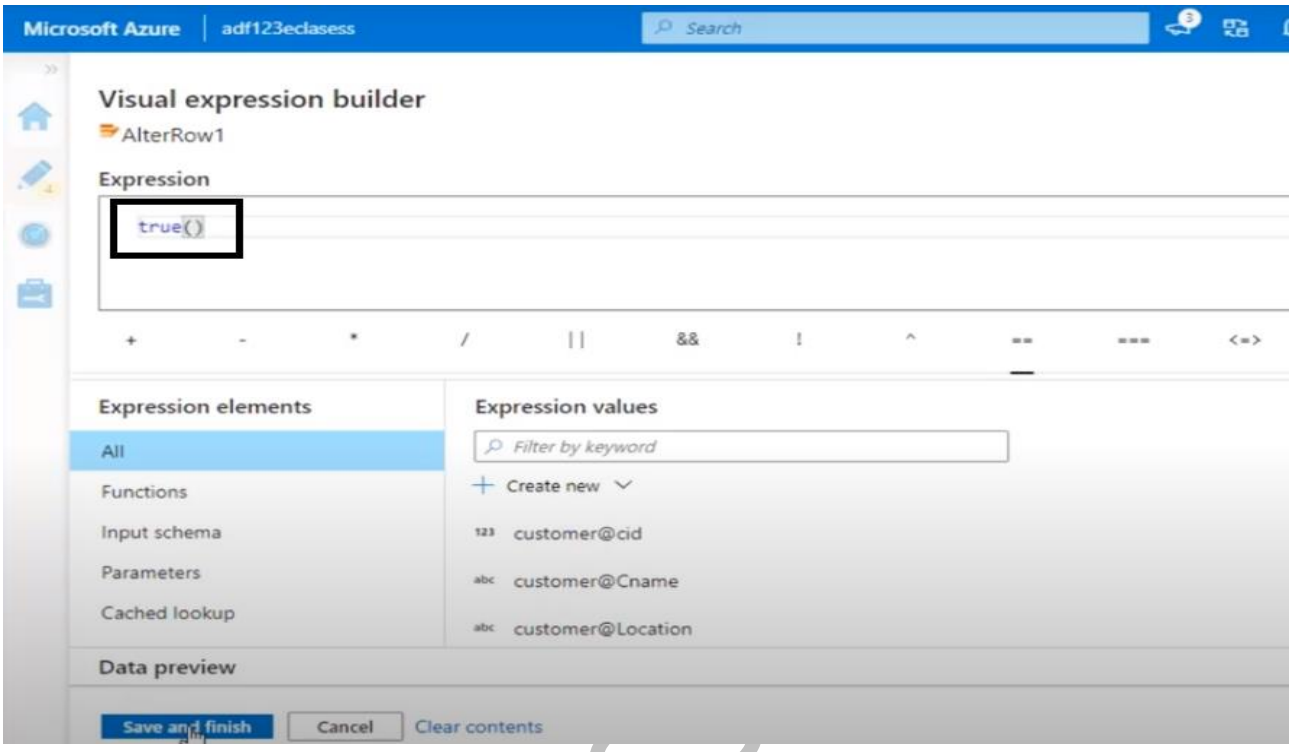
Step 29: Click on '+' symbol---Click on Alter Row transformation



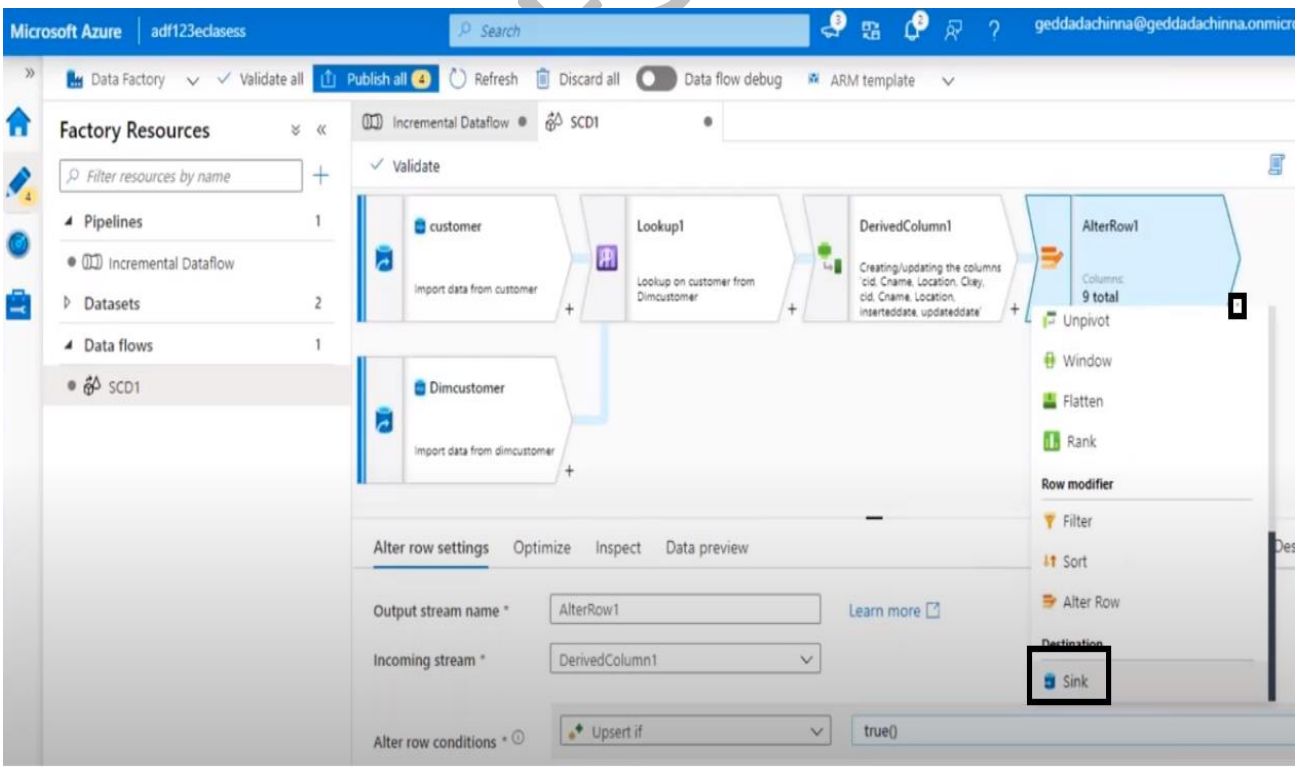
Step 30: Go to Alter row conditions in Alter row transformation----Select upsertif----Click on Enter Condition



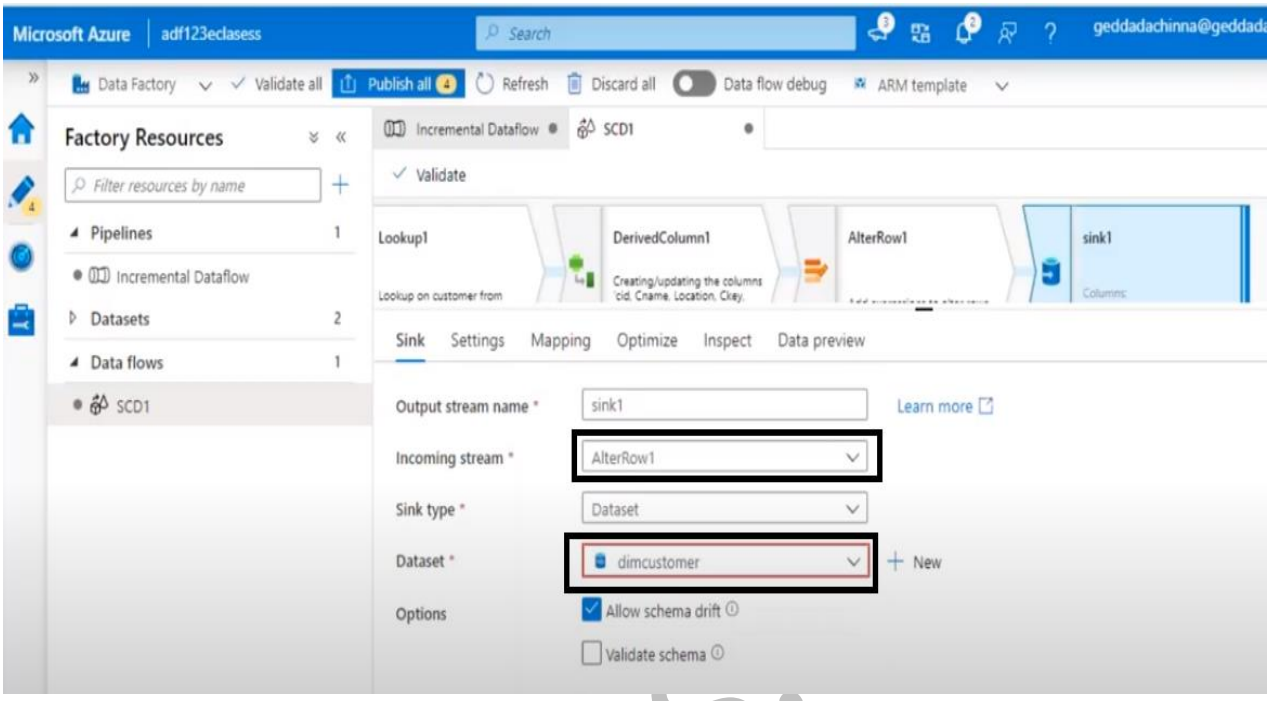
Step 31: Write an expression----Click on Save and Finish



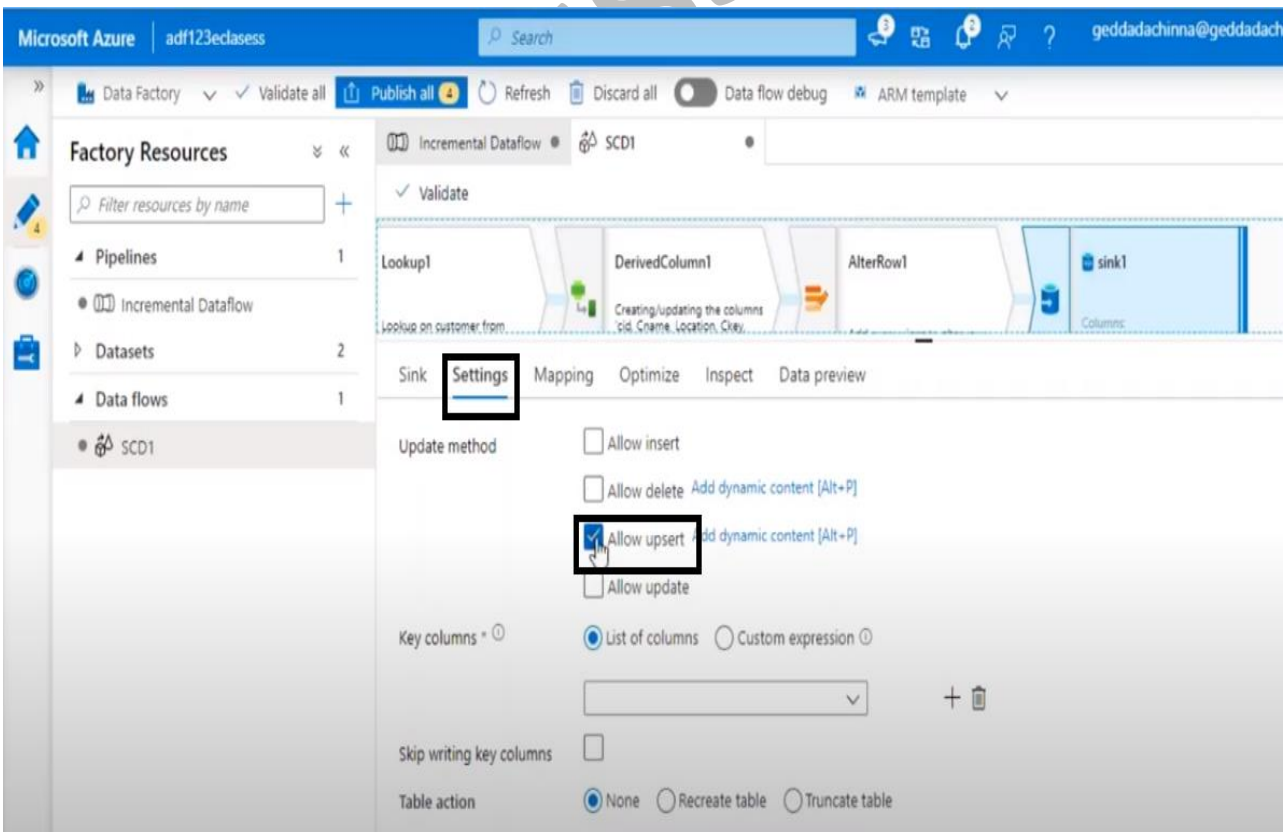
Step 32: Click on '+' symbol and Click on Sink



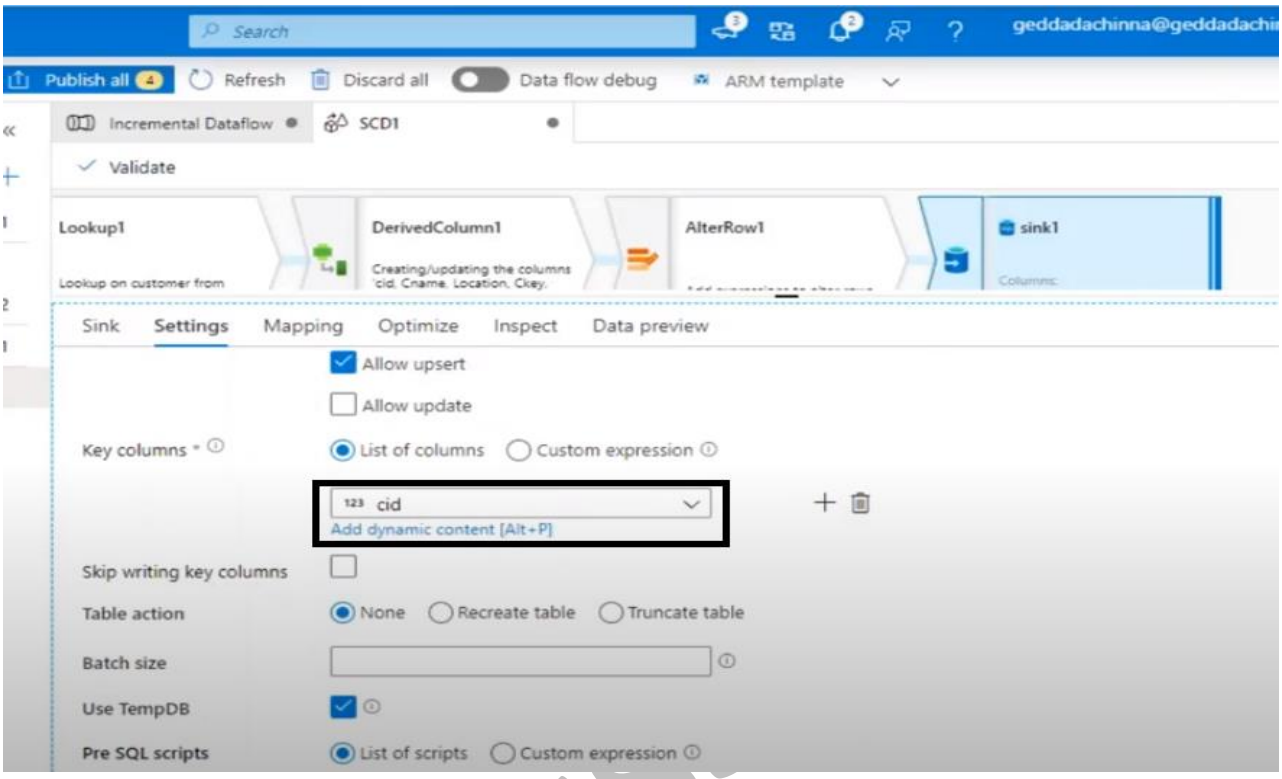
Step 33: Select Incoming stream in sink transformation-----Select target dataset as dimcustomer



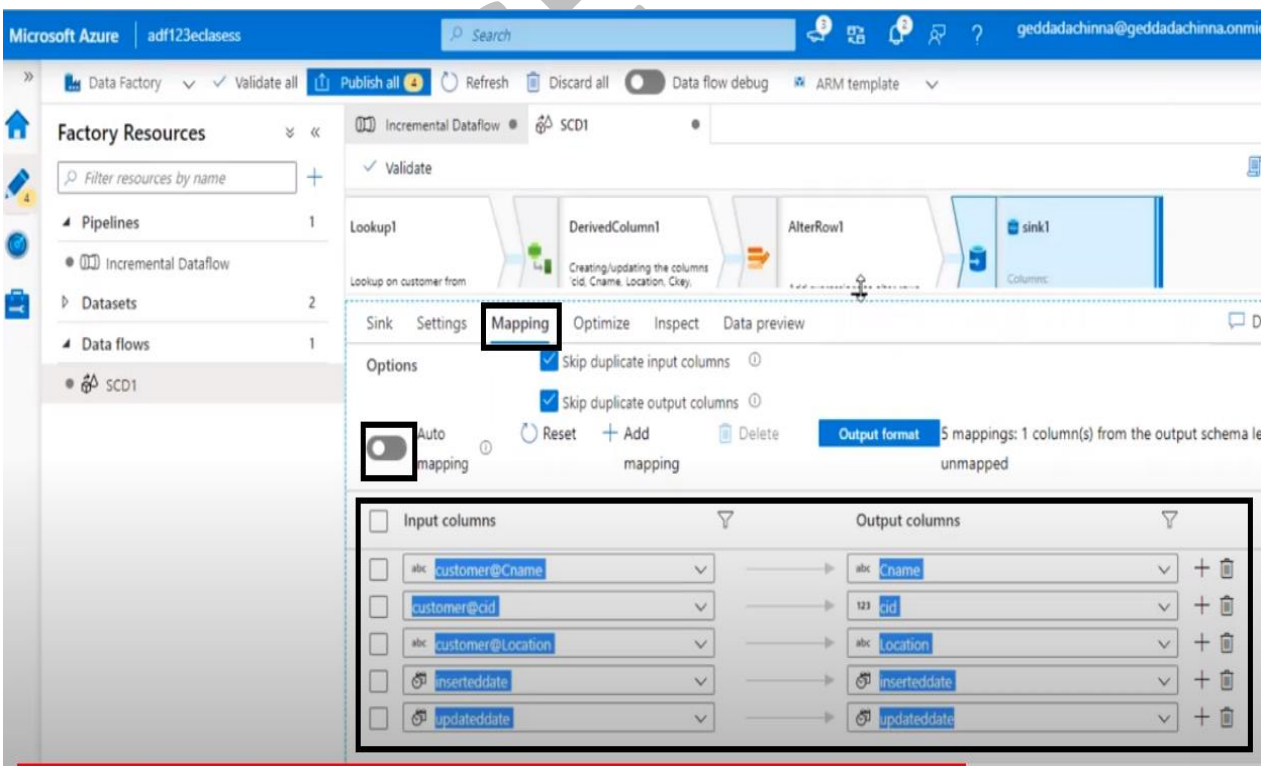
Step 34: Click on Settings and Select Allow upset.



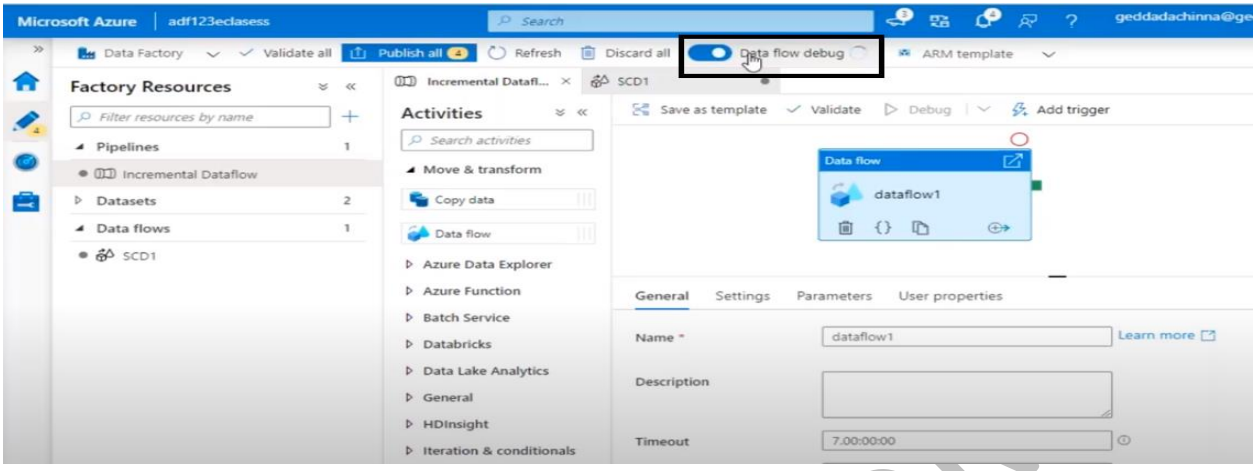
Step 35: Go to Key columns and Select cid column



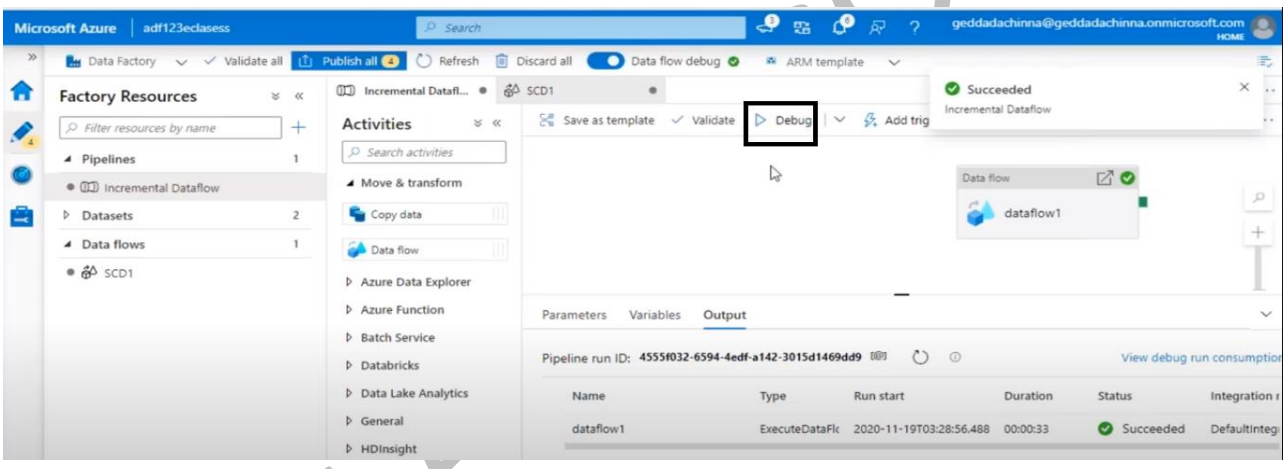
Step 36: Click on Mapping----Disable Auto Mapping----Do Mapping manually



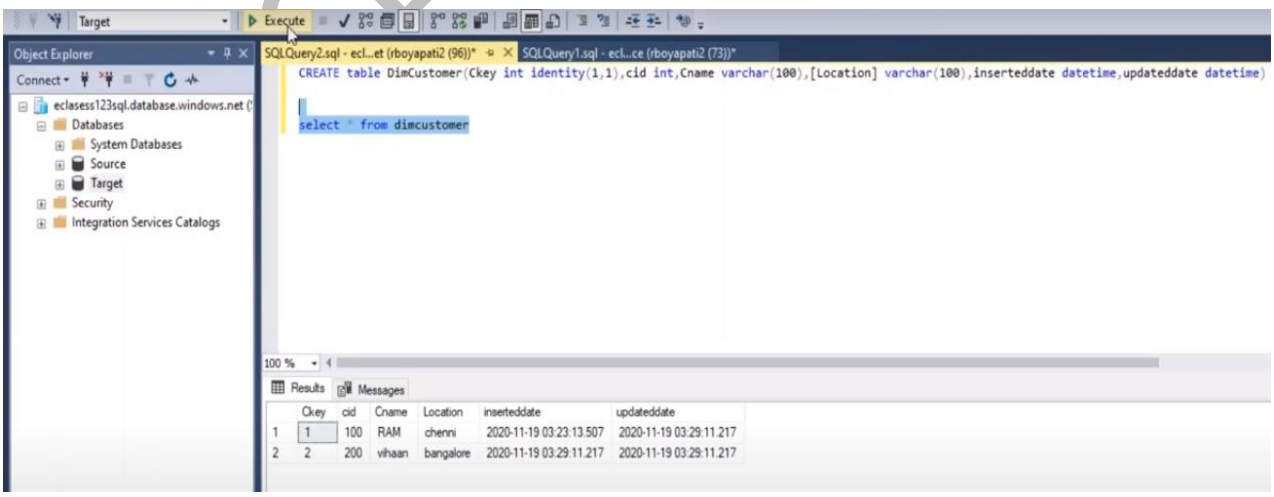
Step 37: Click on Data flow debug



Step 38: Click on Debug and Check the Status

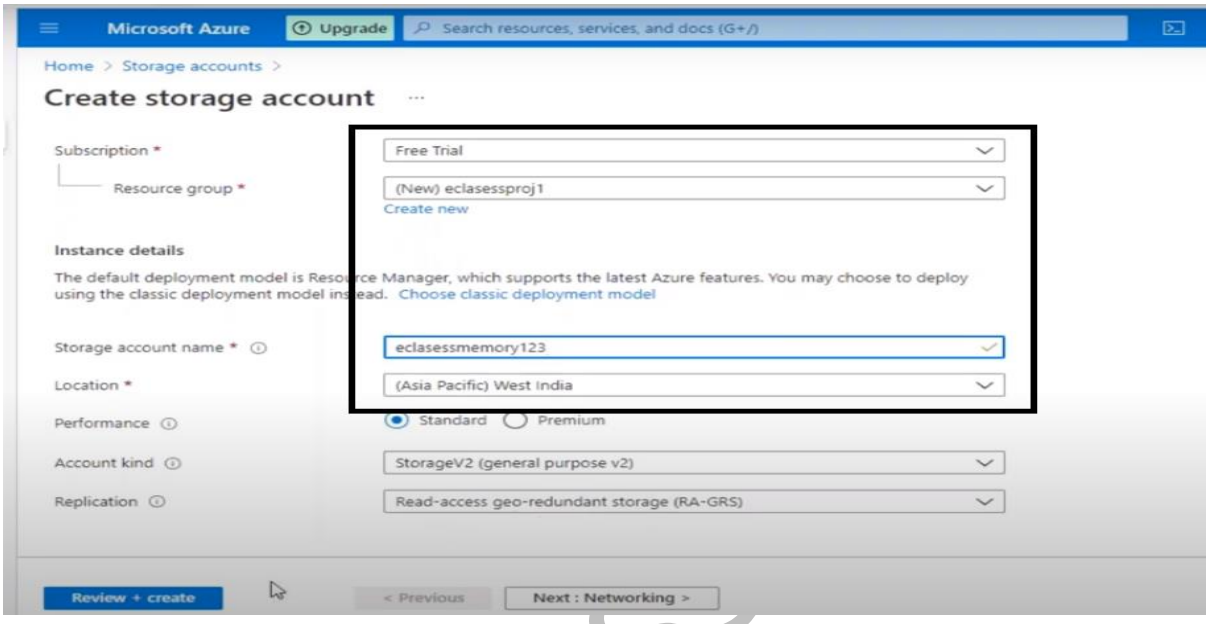


Step 39: Write a Query in the target table to check whether the data is loaded or not

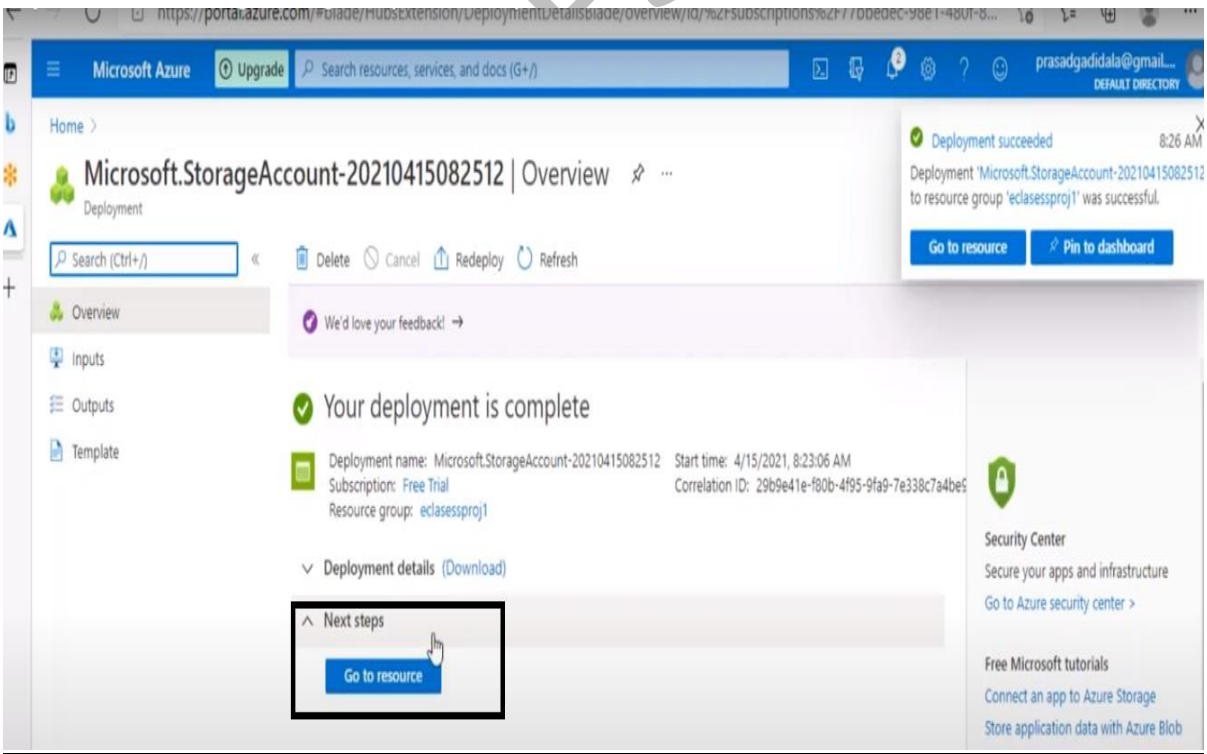


How to Copy data from multiple containers of blob storage in to Azure SQL database.

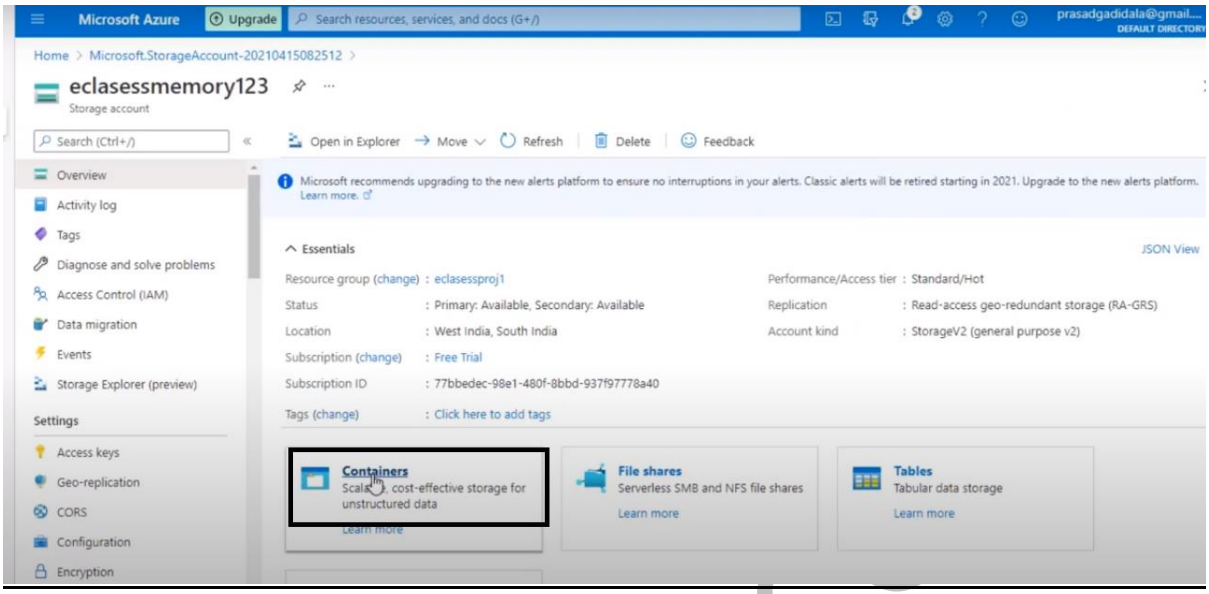
Step 1: Create a storage account



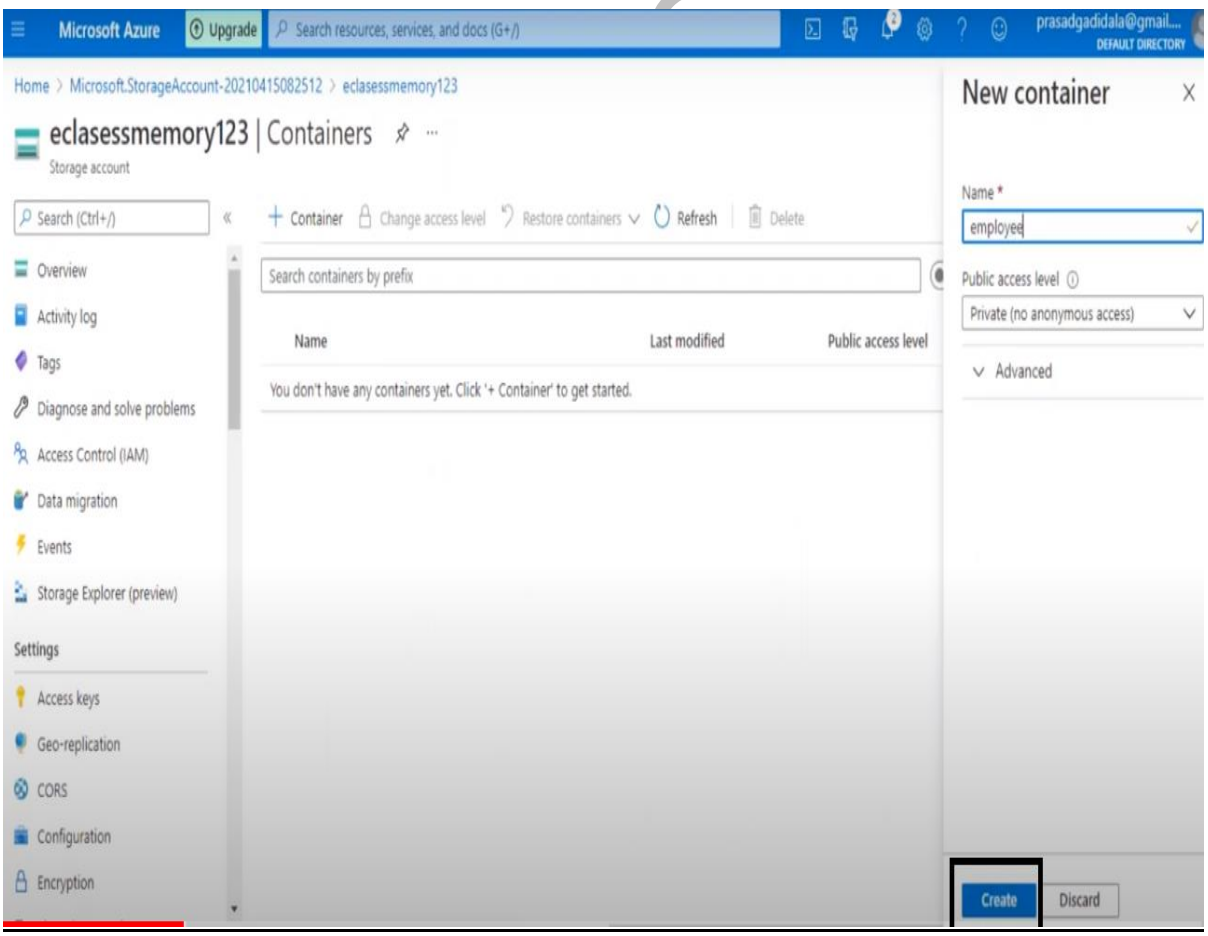
Step 2: Click on go to resource



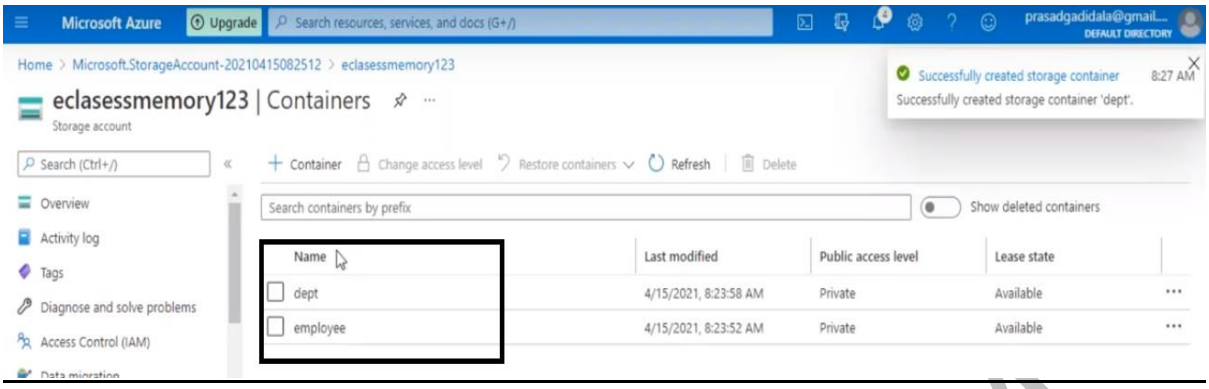
Step 3: Click on container



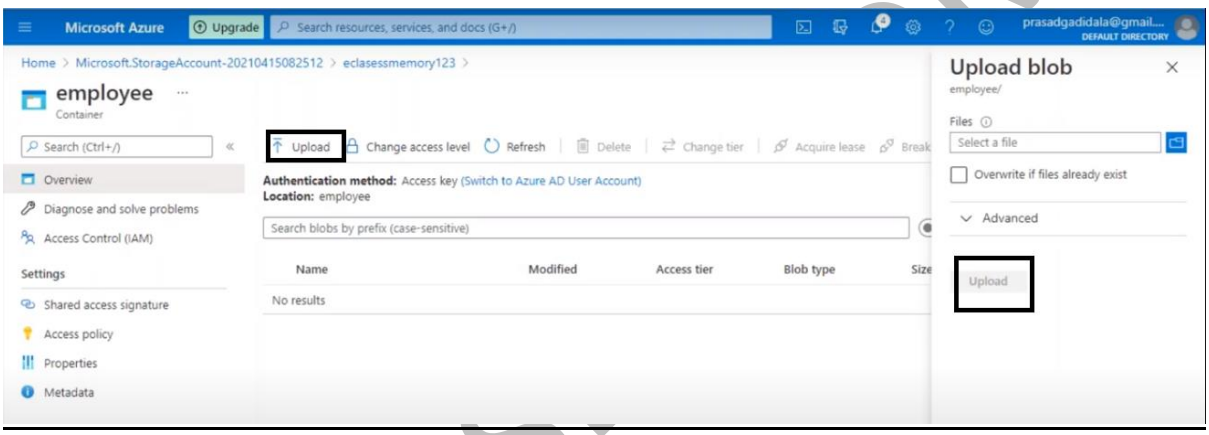
Step 4: Create new container



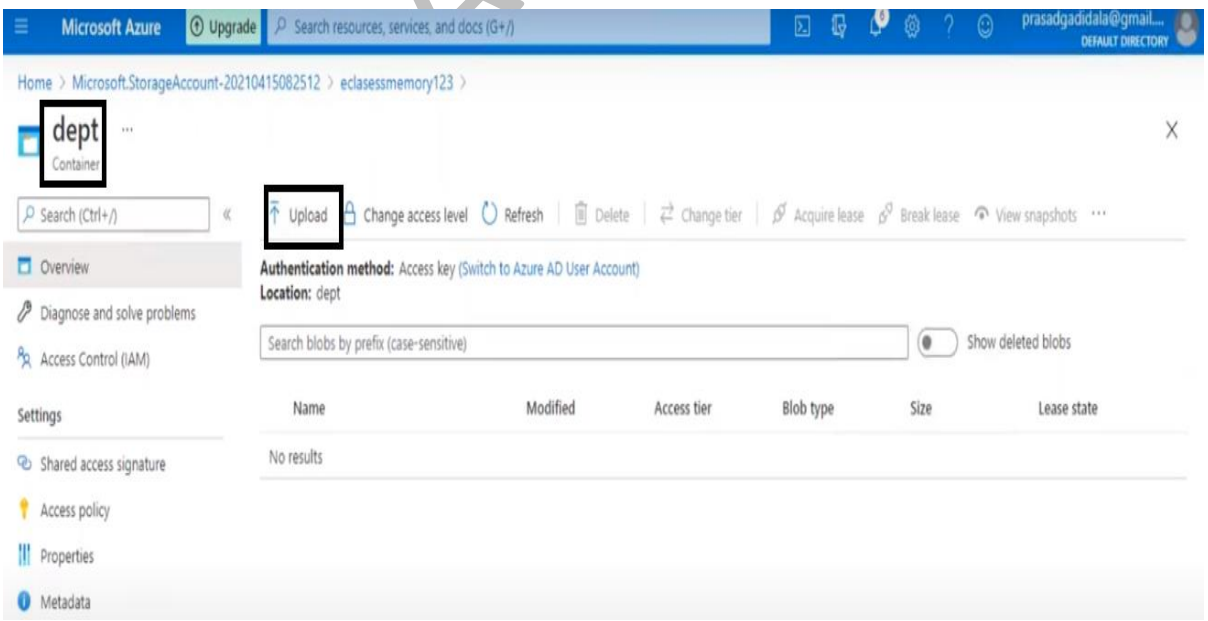
Step 5: Create two containers



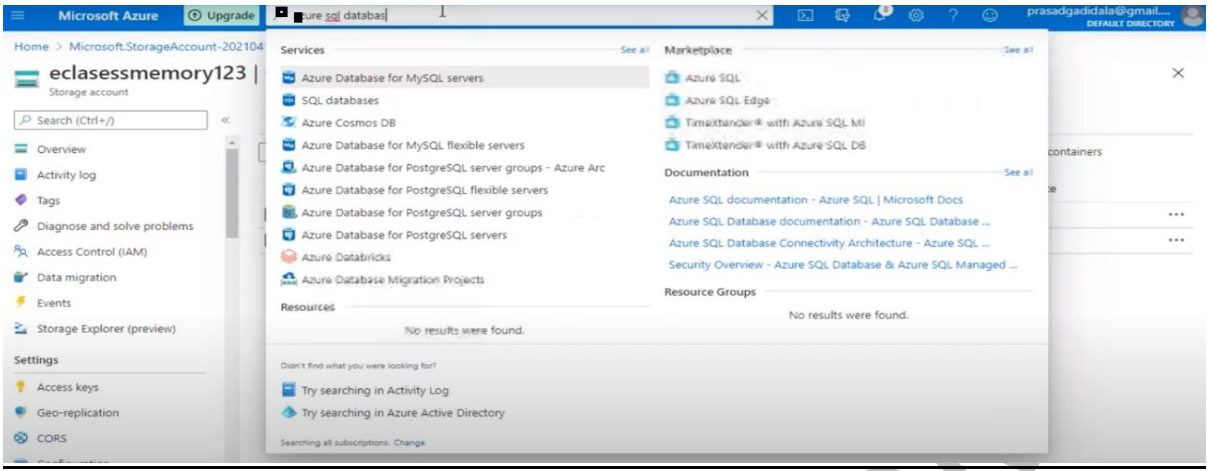
Step 6: Go inside the employee container and upload files into that



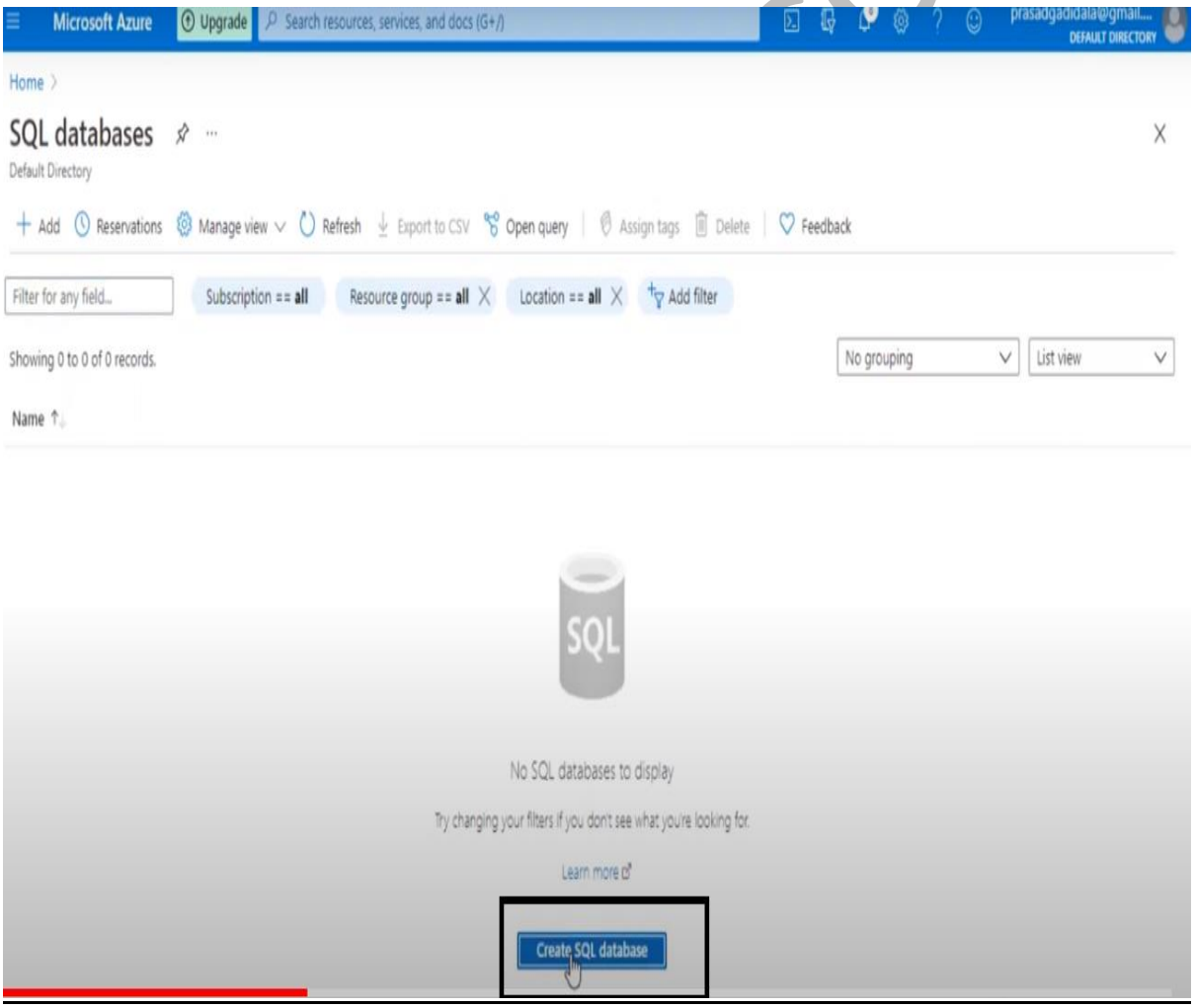
Step 7: Again we have to upload the files into dept container



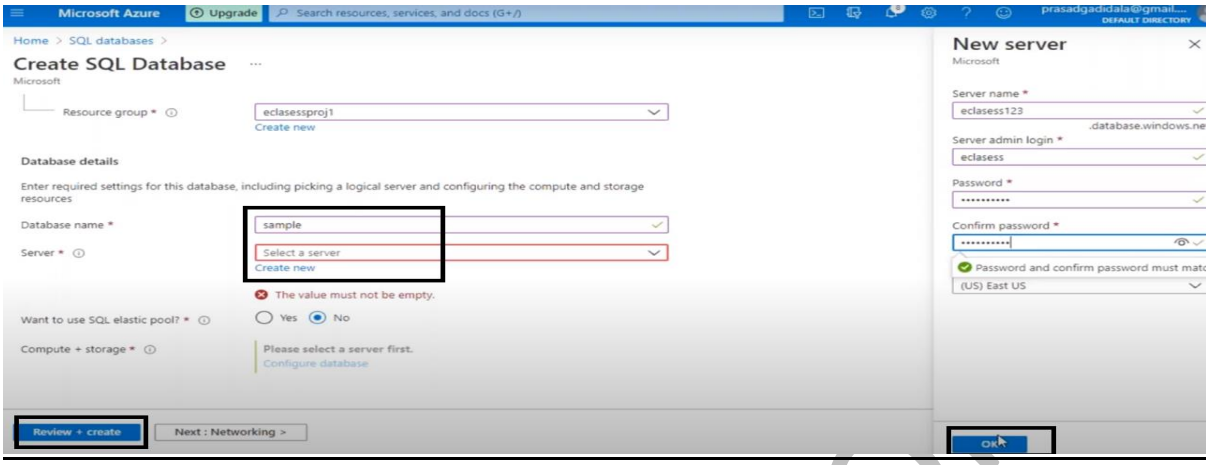
Step 8: Then we need to setup azure sql databases



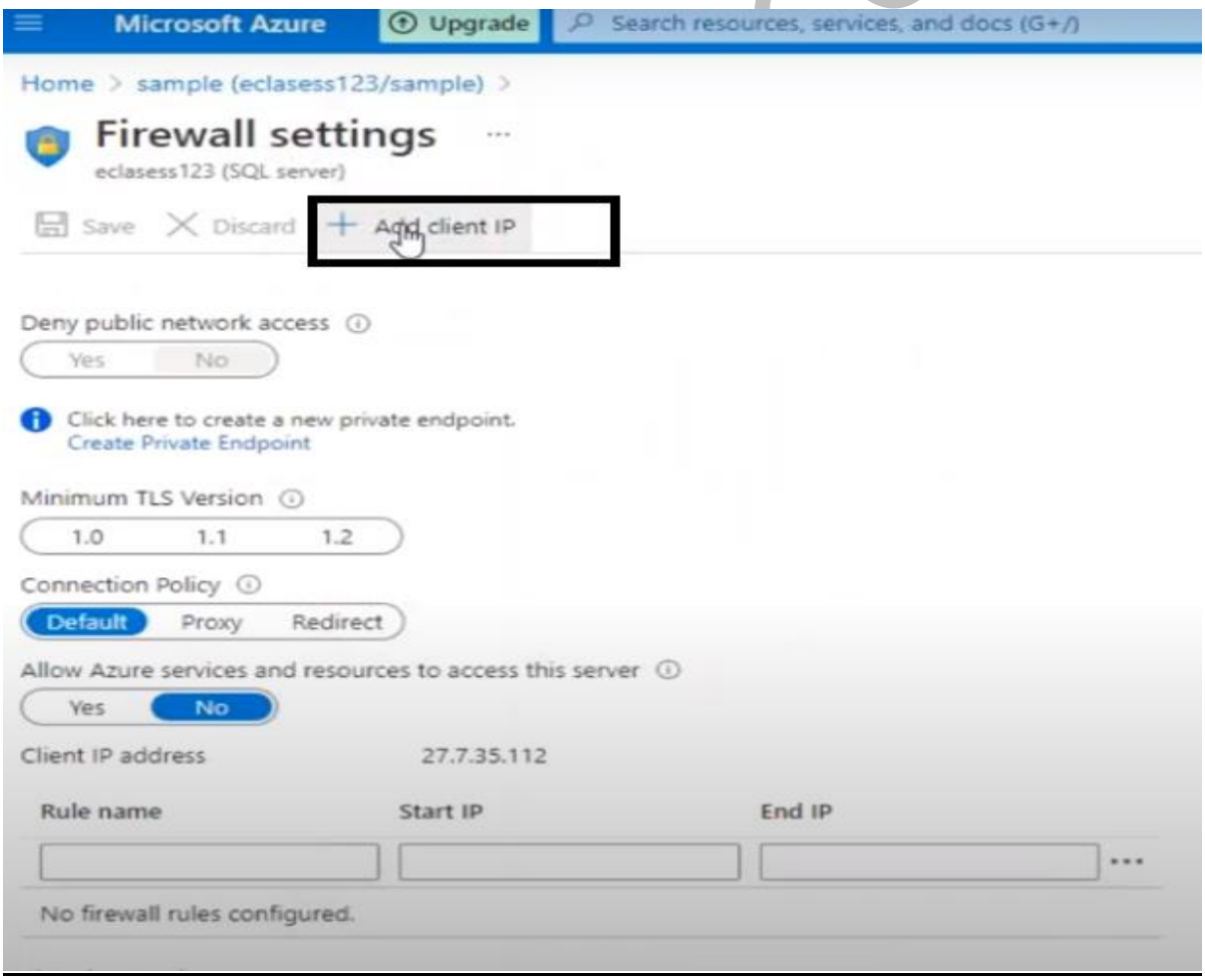
Step 9: Click on create sql datbases

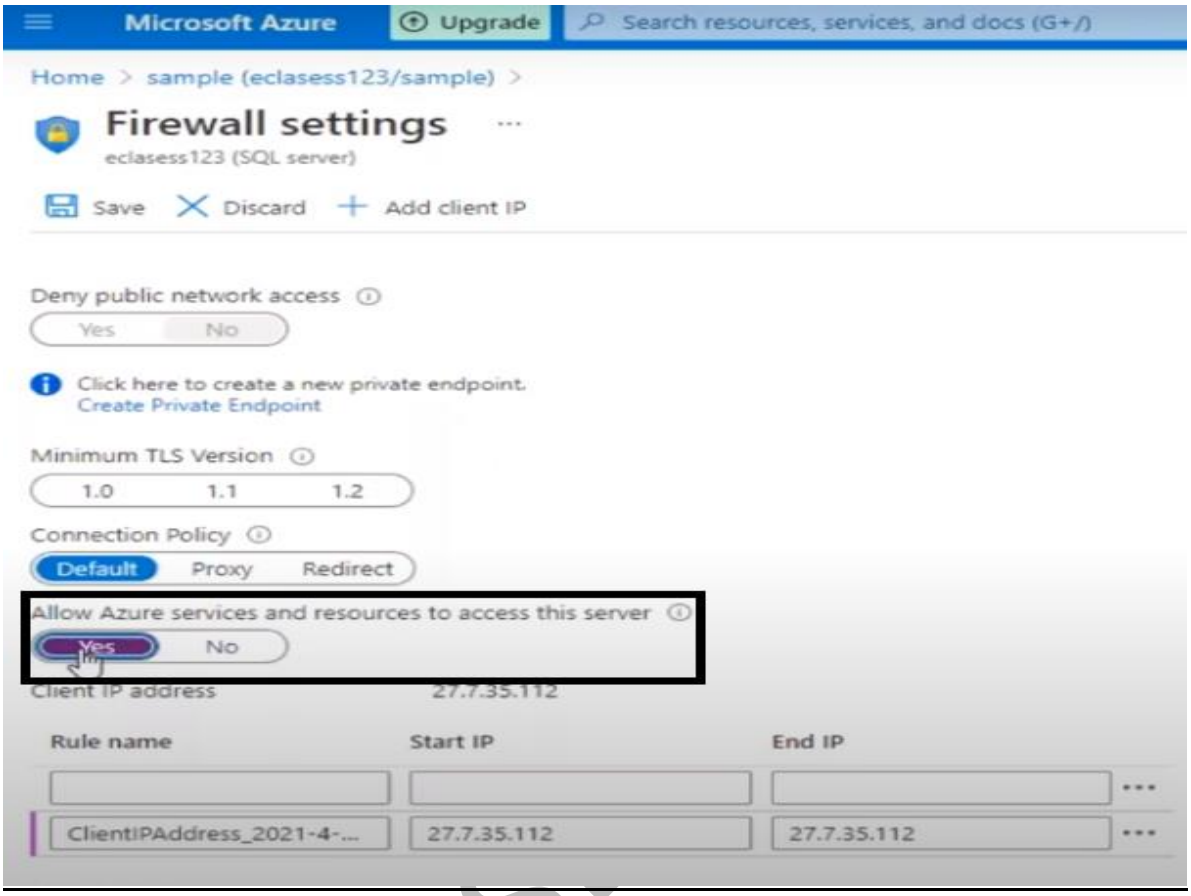


Step 10: While creating sql databases we have to create new server and then review and create

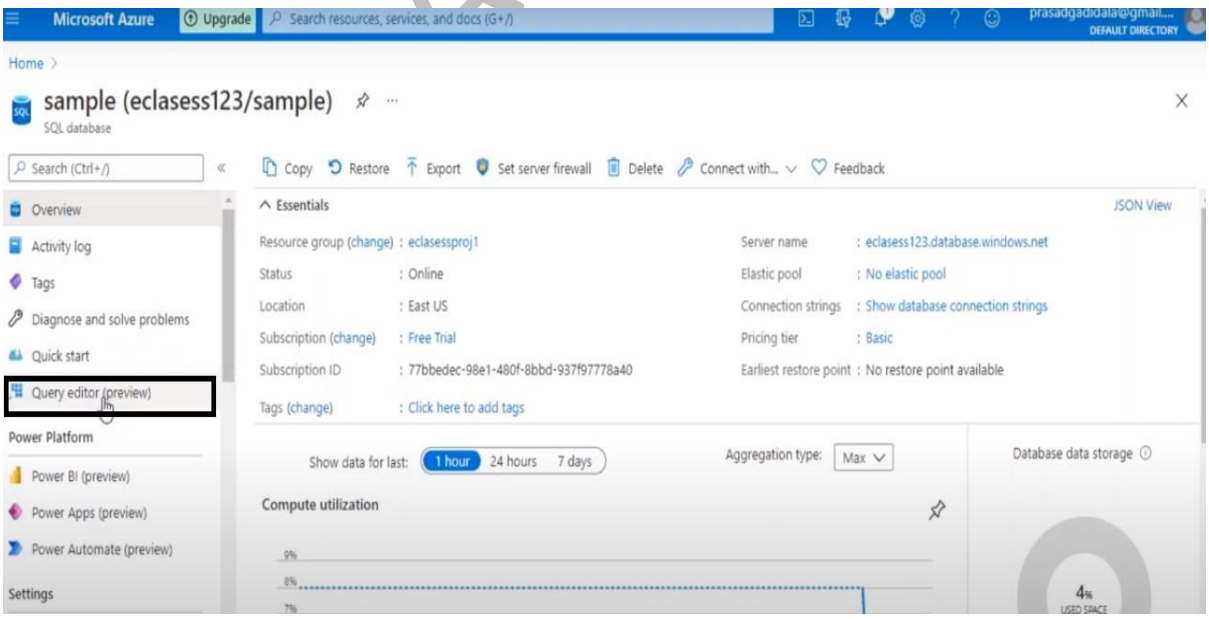


Step 11: We need to setup server firewall

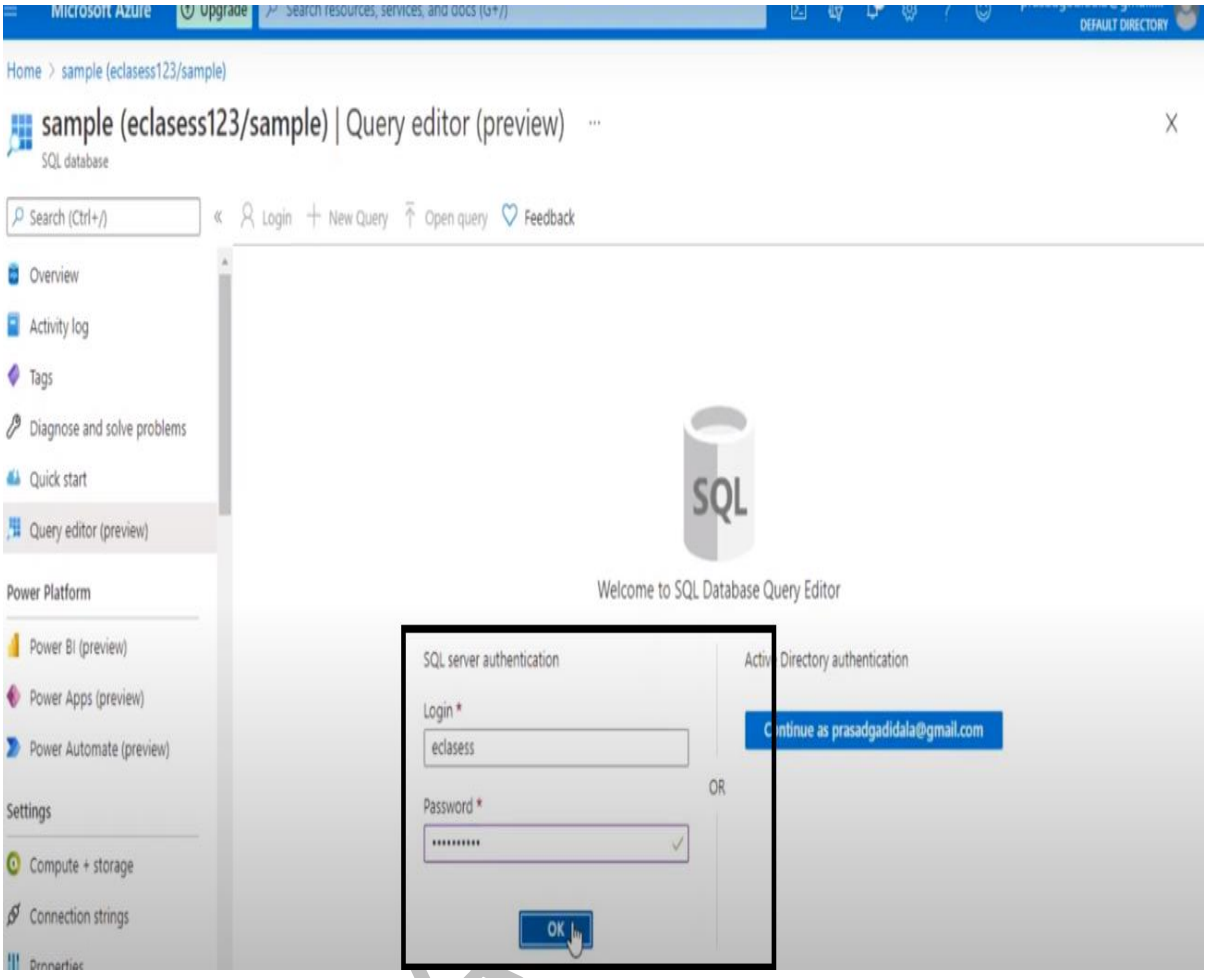




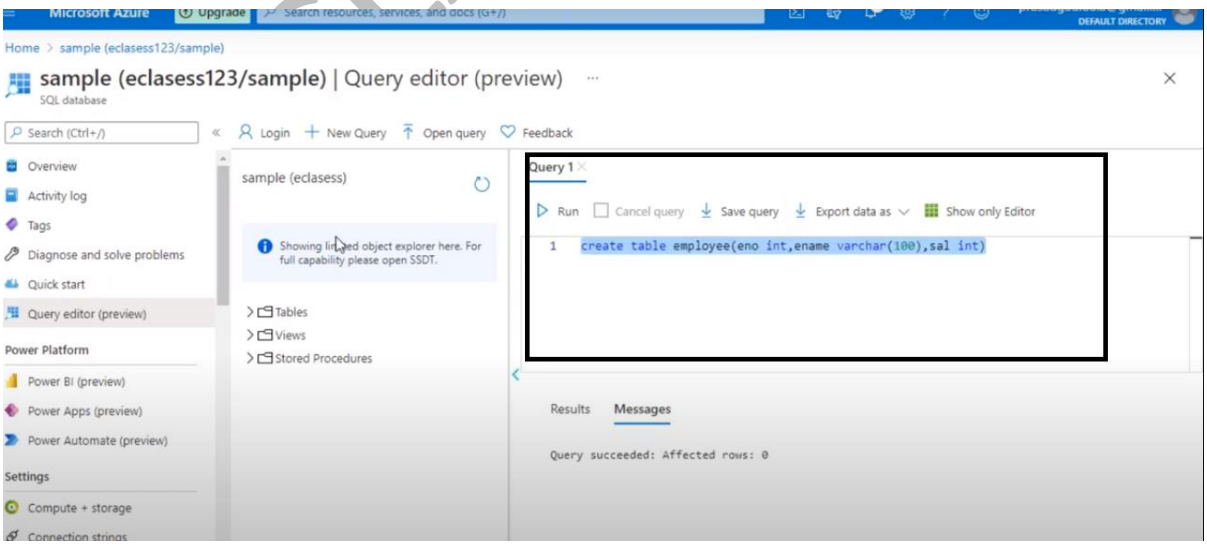
Step 12: Create two tables and then click on query editor



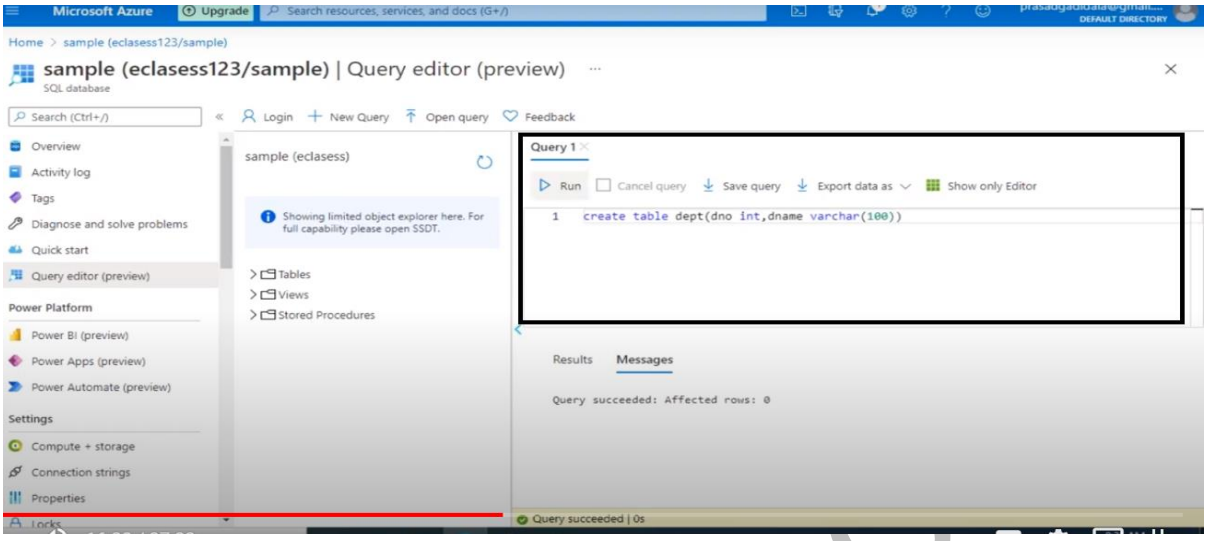
Step 13: Login with new server credentials



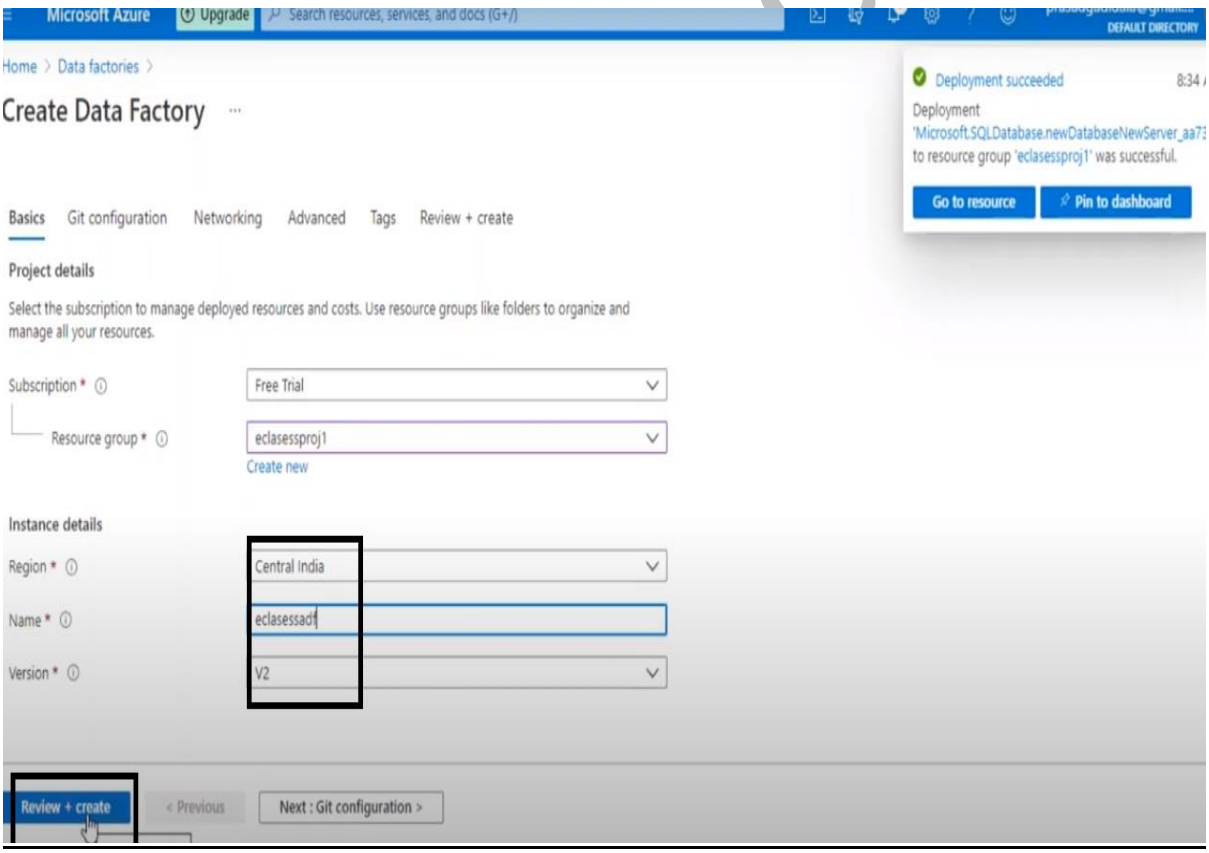
Step 14: Create table for employee and run it



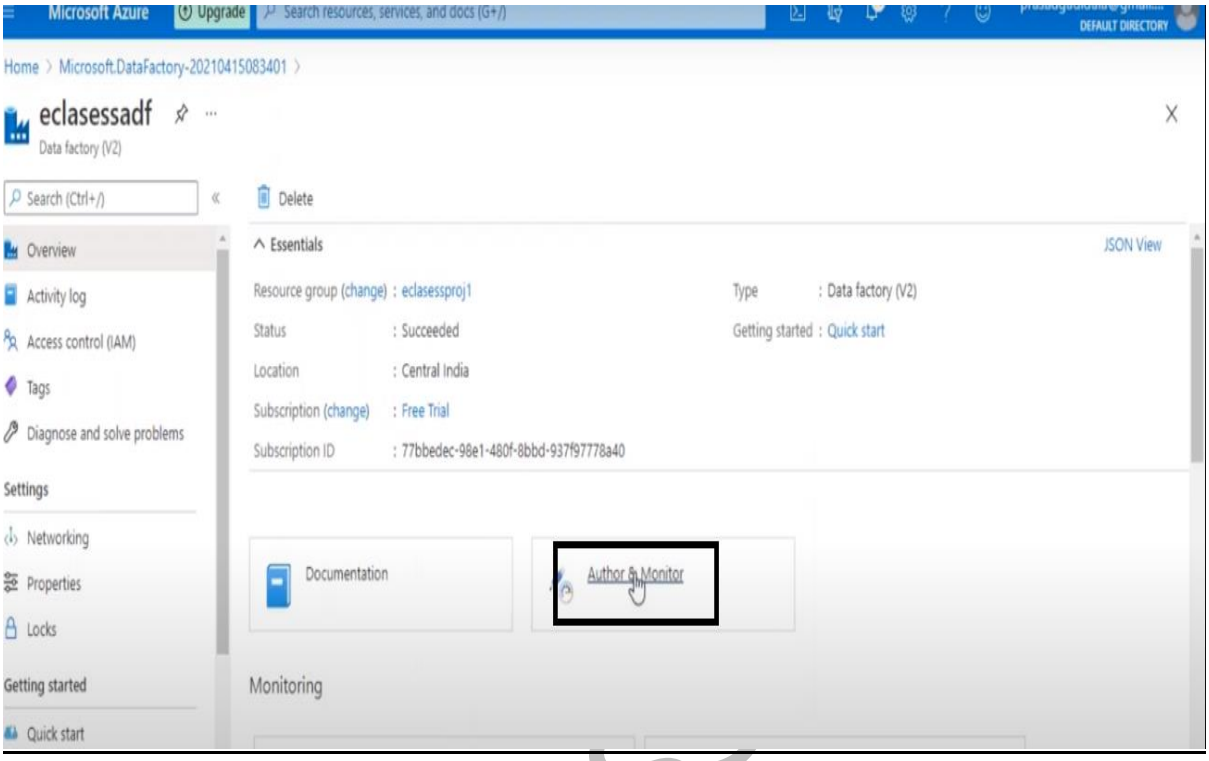
Step 15: Create table for department and run it

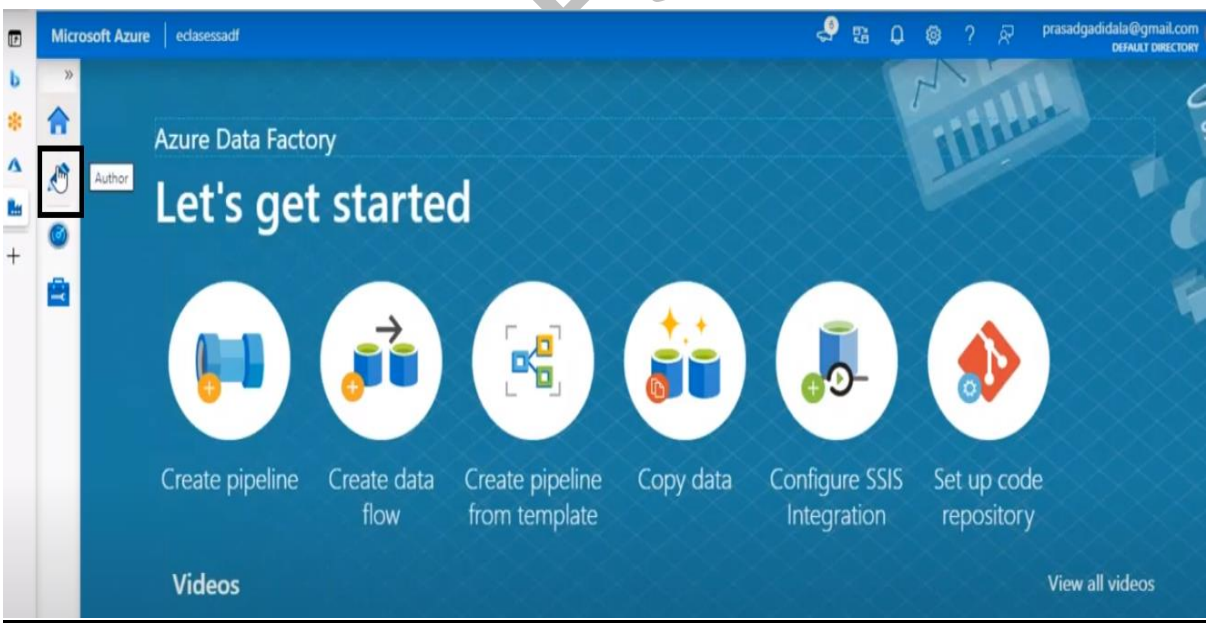


Step 16: Create adf

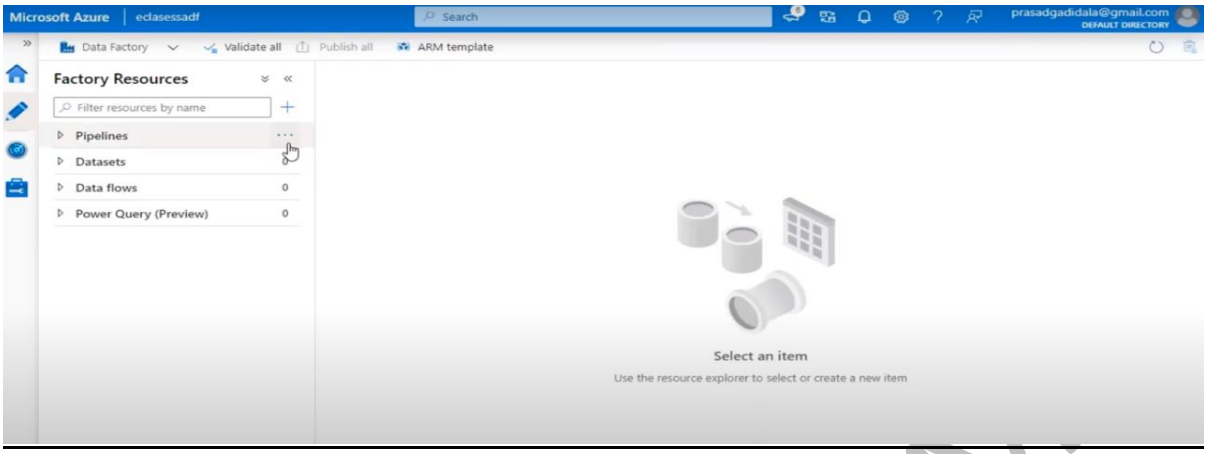


Step 17: Click on author & monitor

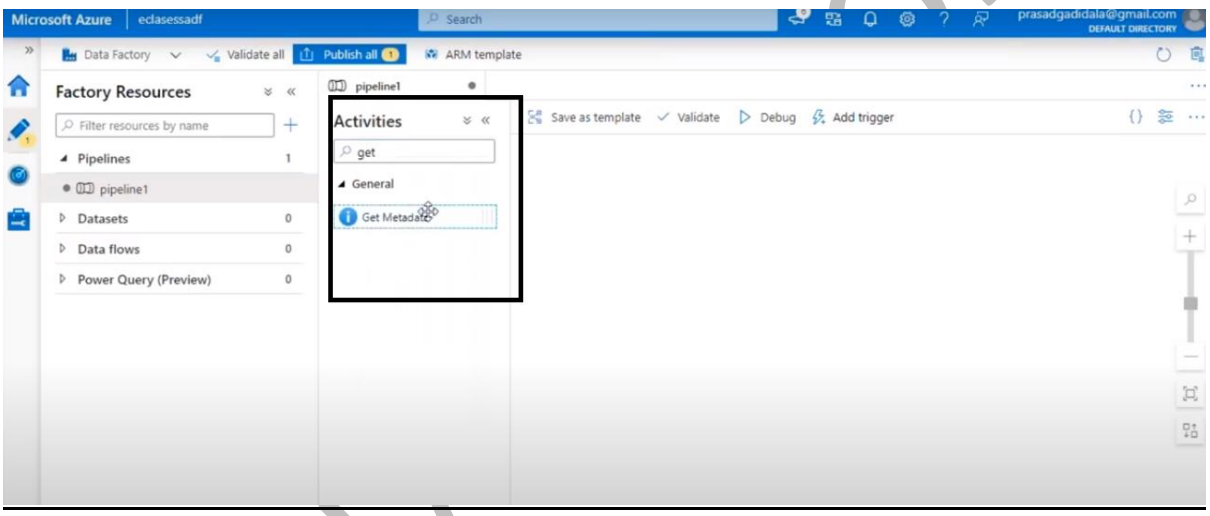




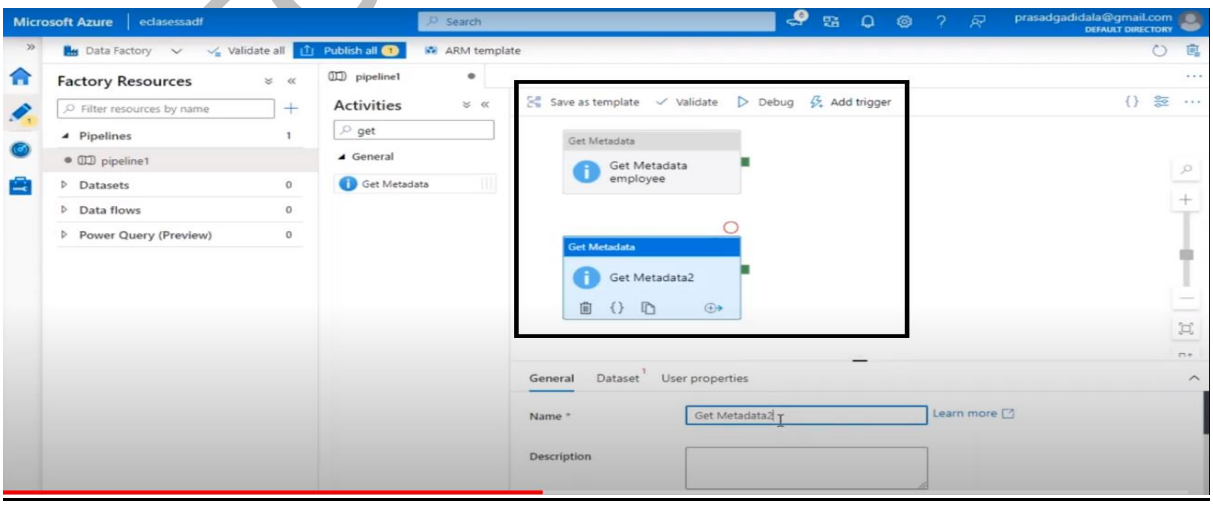
Step 18: We have to load two containers data into adf



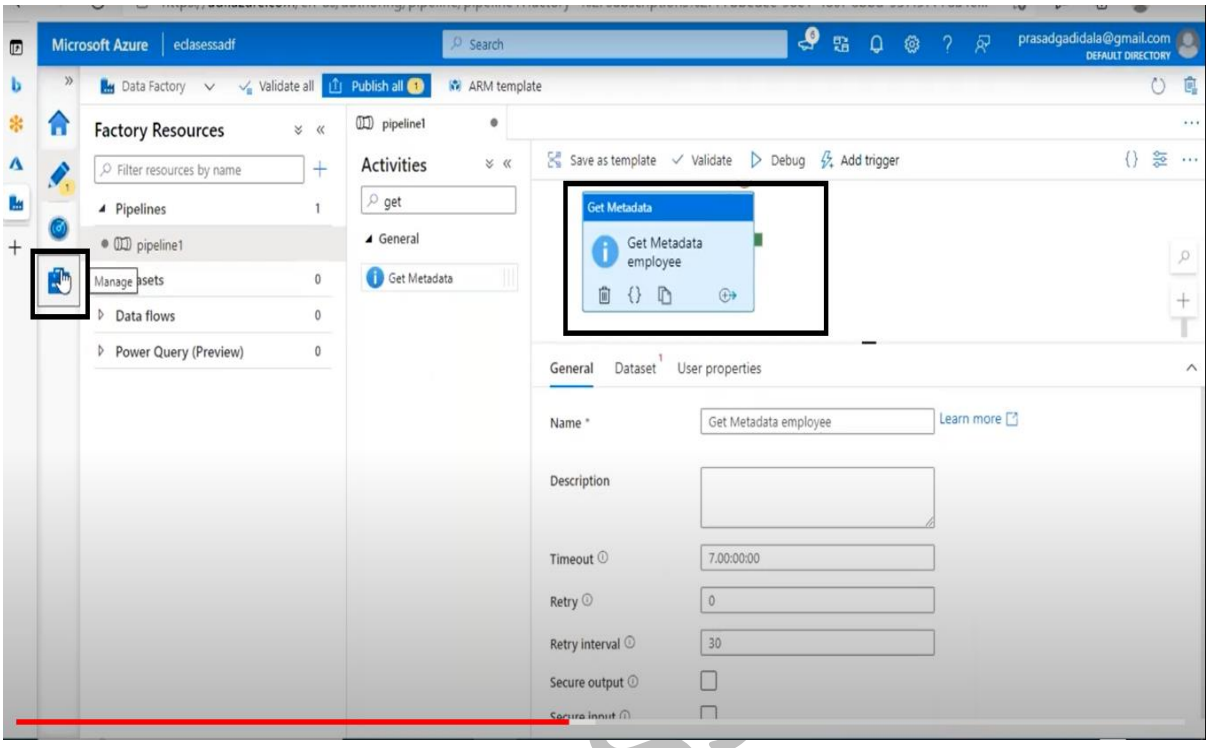
Step 19: Click on pipeline and create it then select get metadata activity



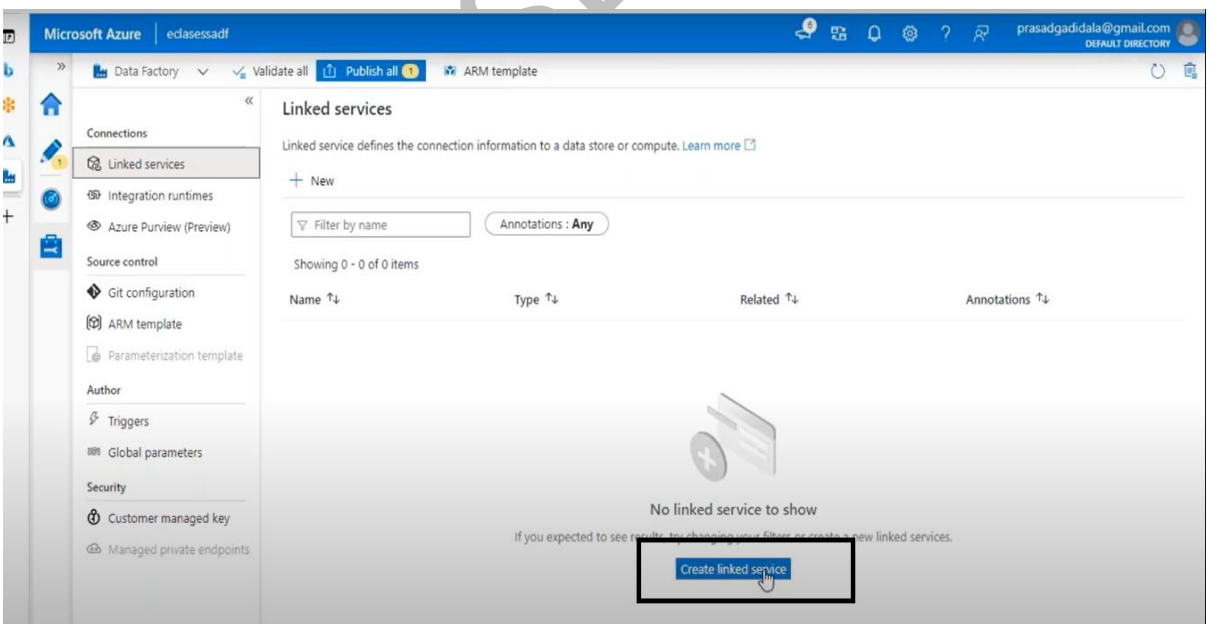
Step 20: Take two get metadata activities for which we created two containers and give the naming for those

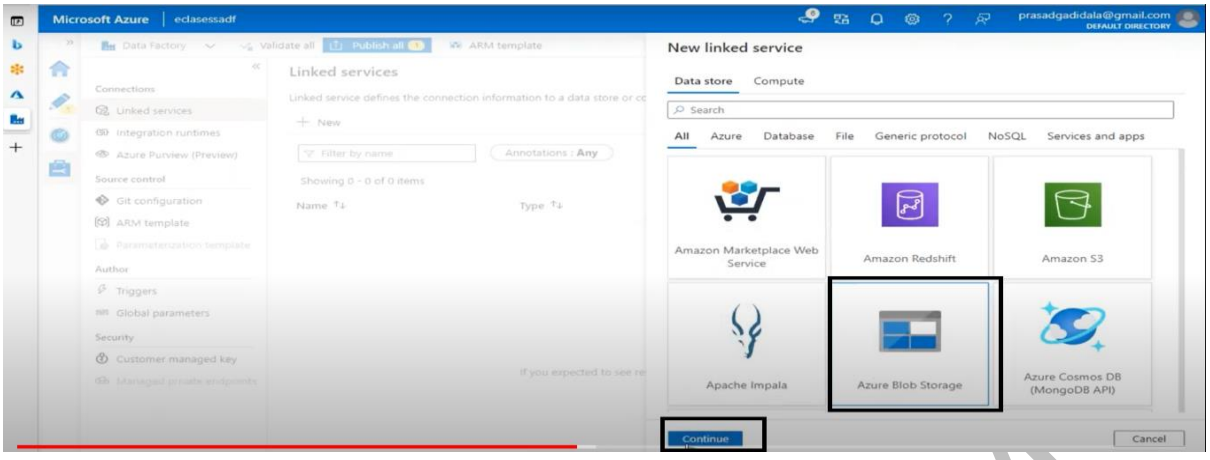


Step 21: Create dataset then before going to create dataset we need to create linked service and click on get metadata then click on linked service

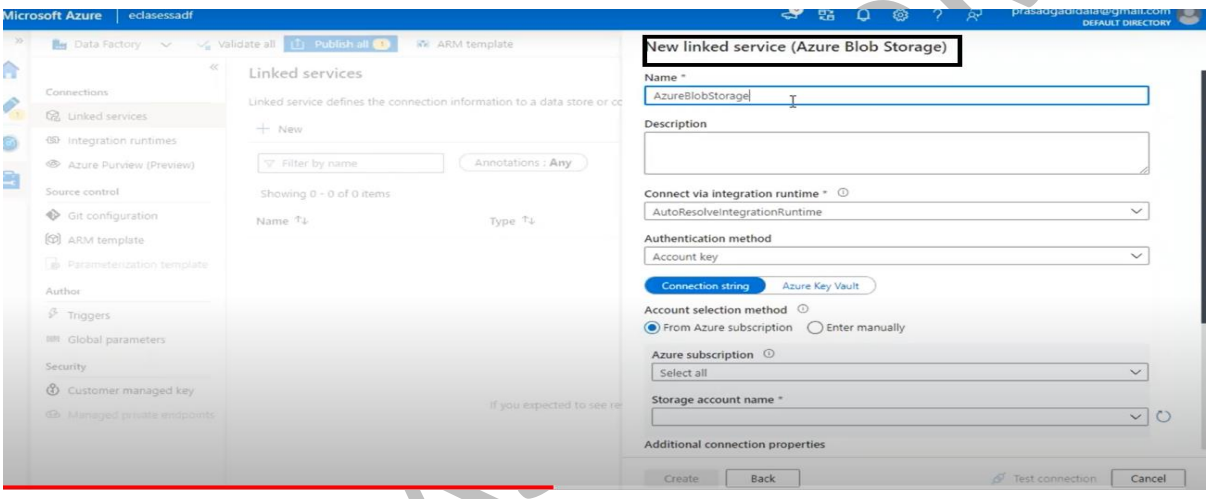


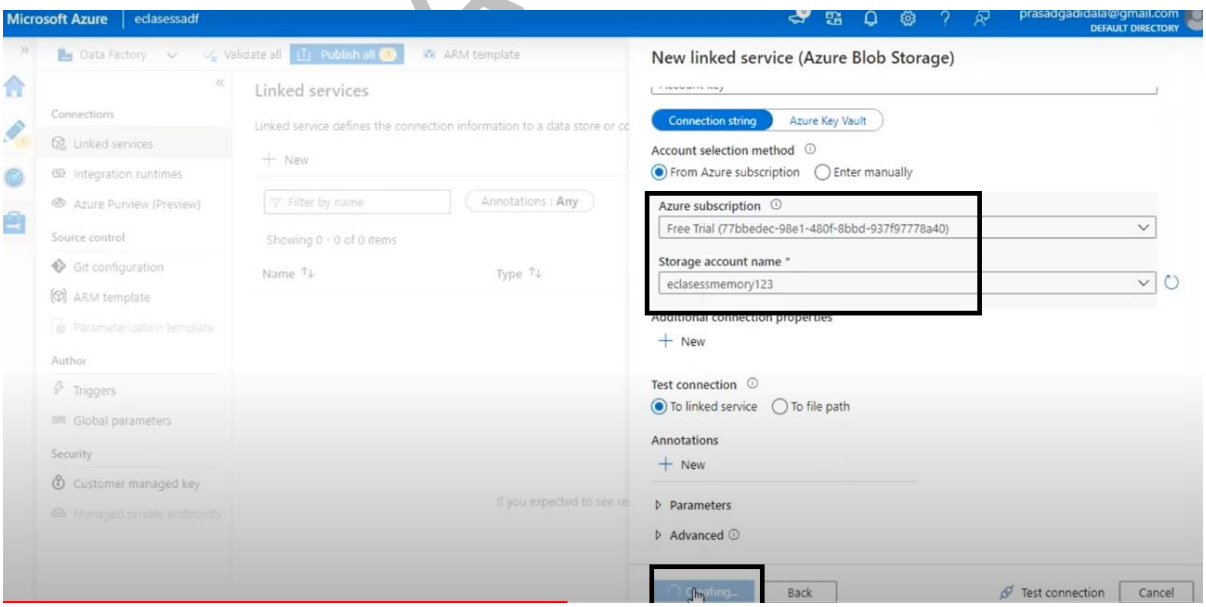
Step 22: We need to create two linked services one for blob storage and another one for sql database simply click on create liked service



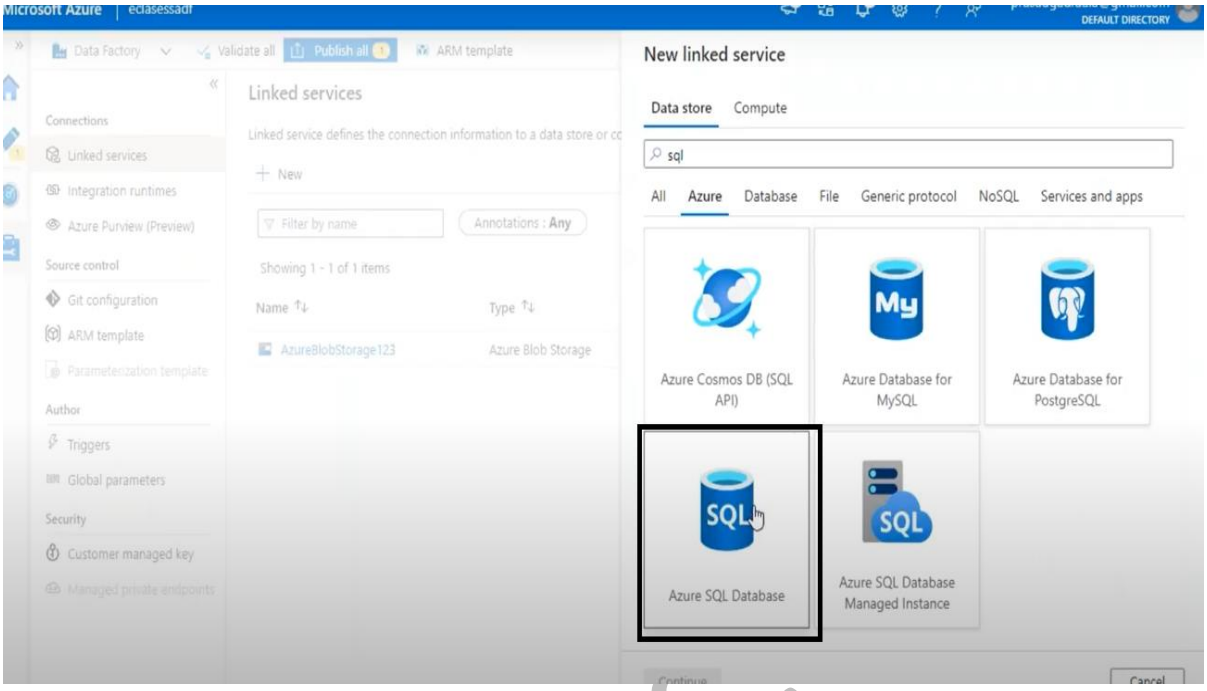


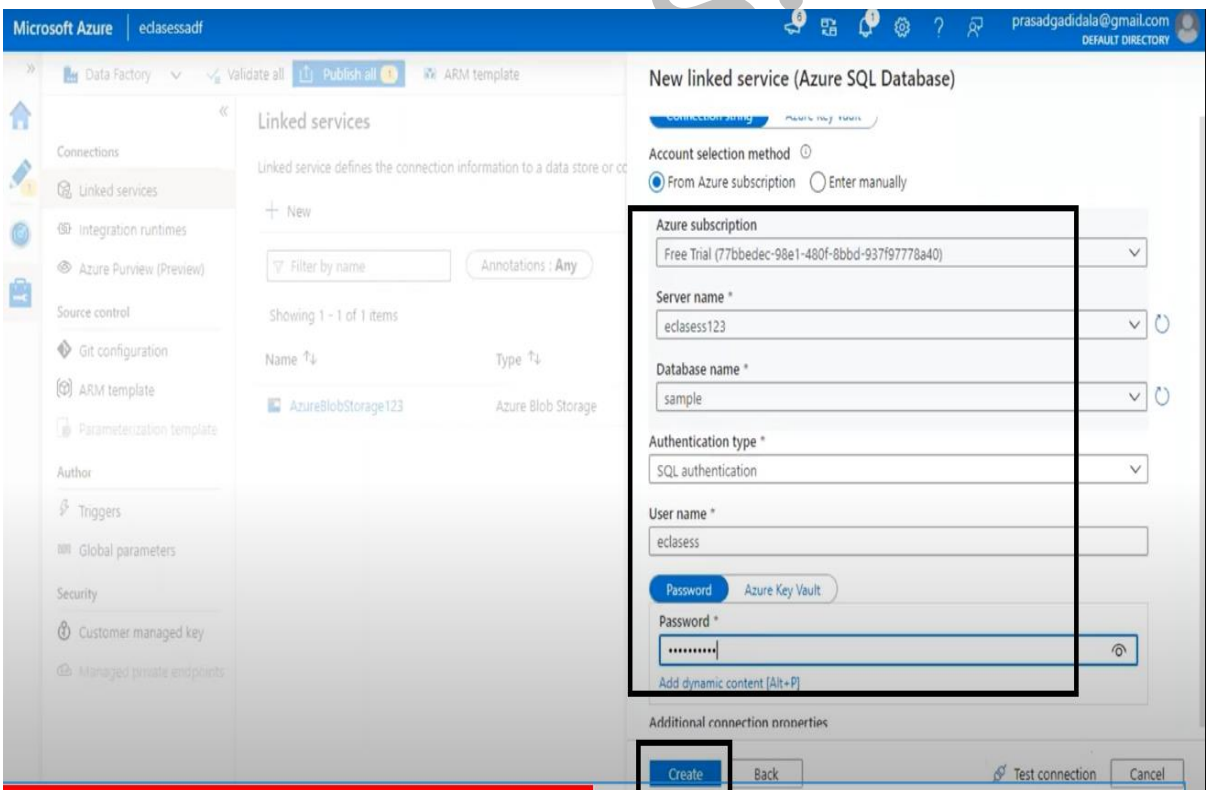
Step 23: Create new linked service



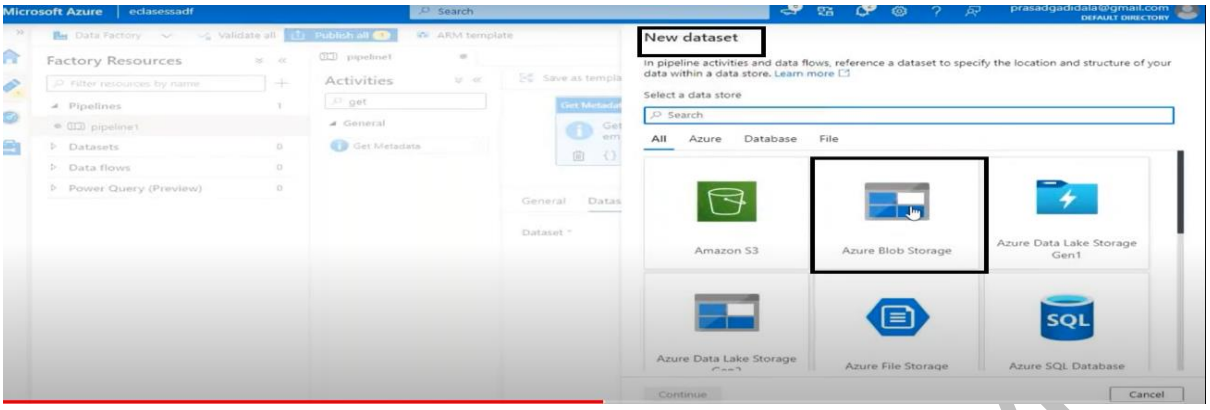


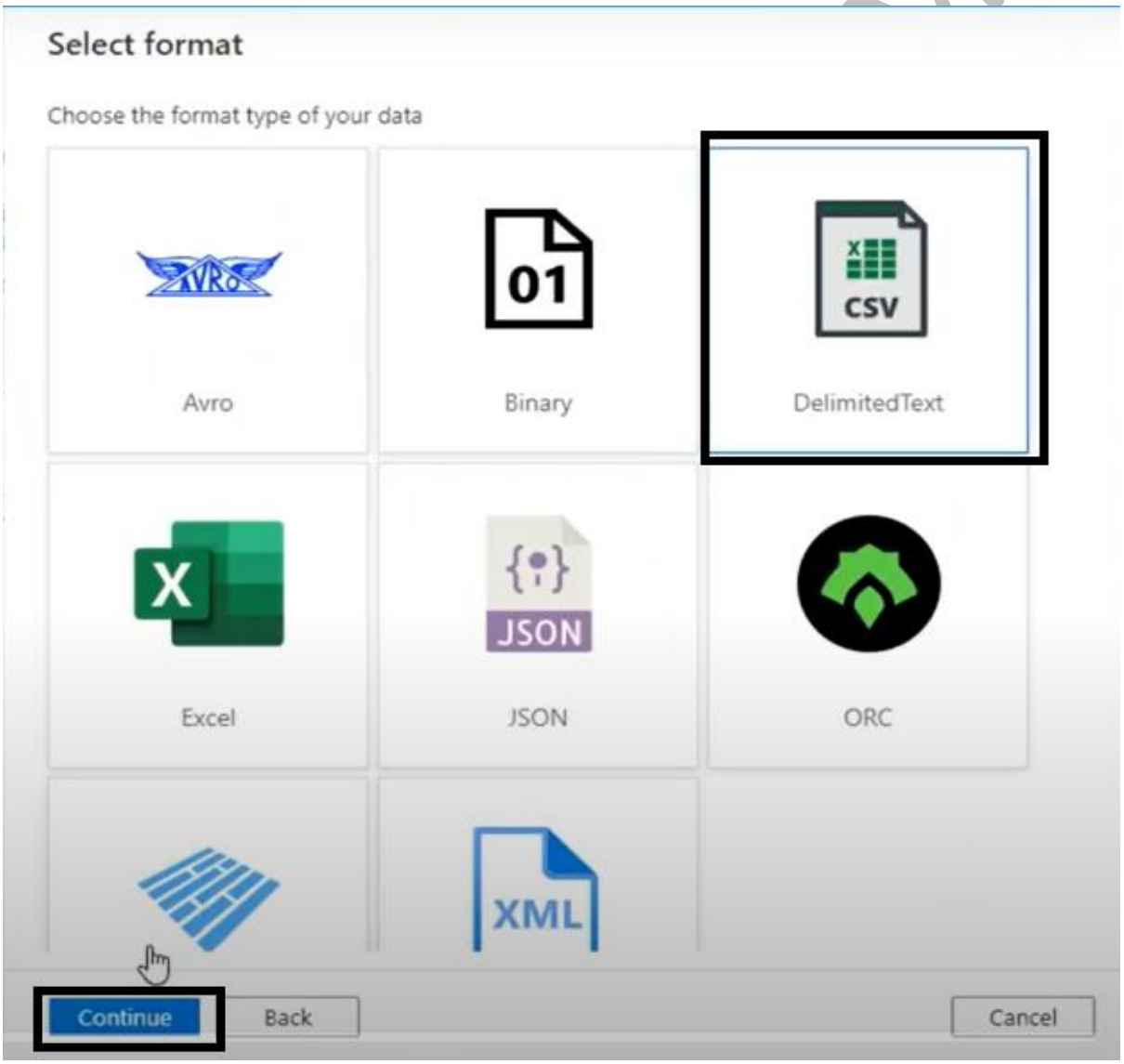
Step 24: Another liked service for azure sql database

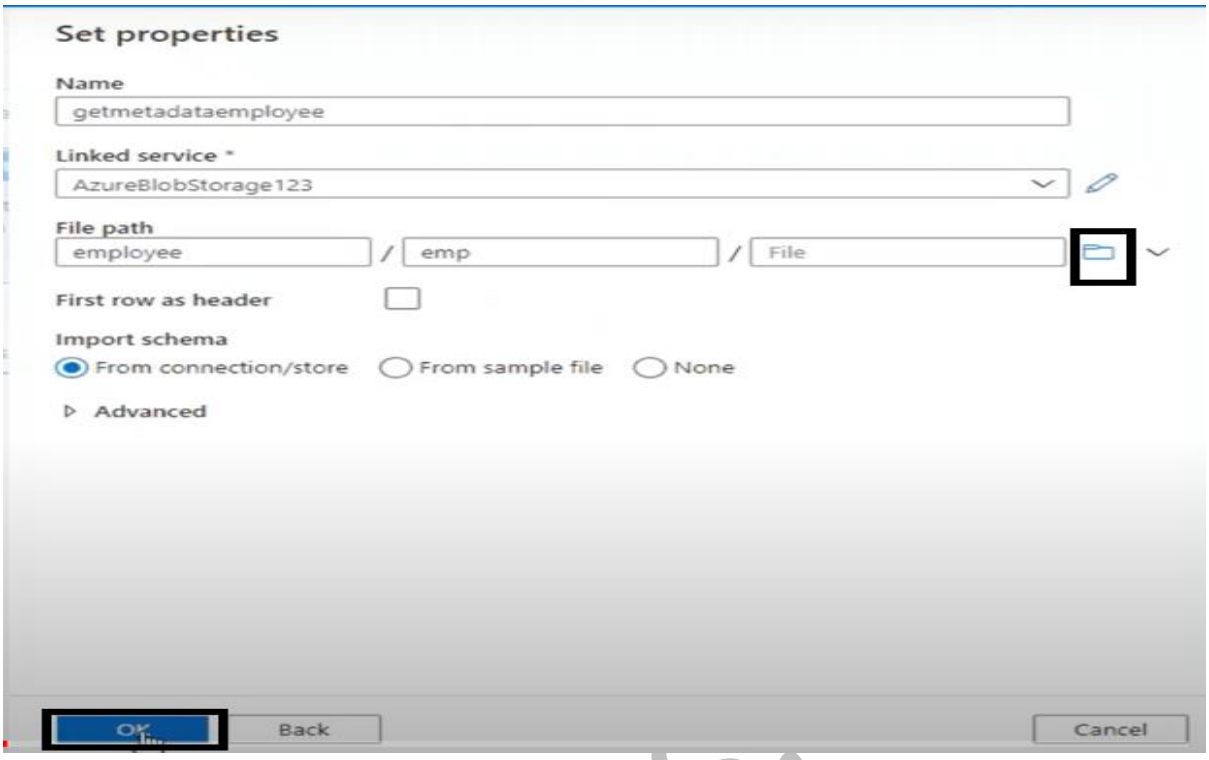




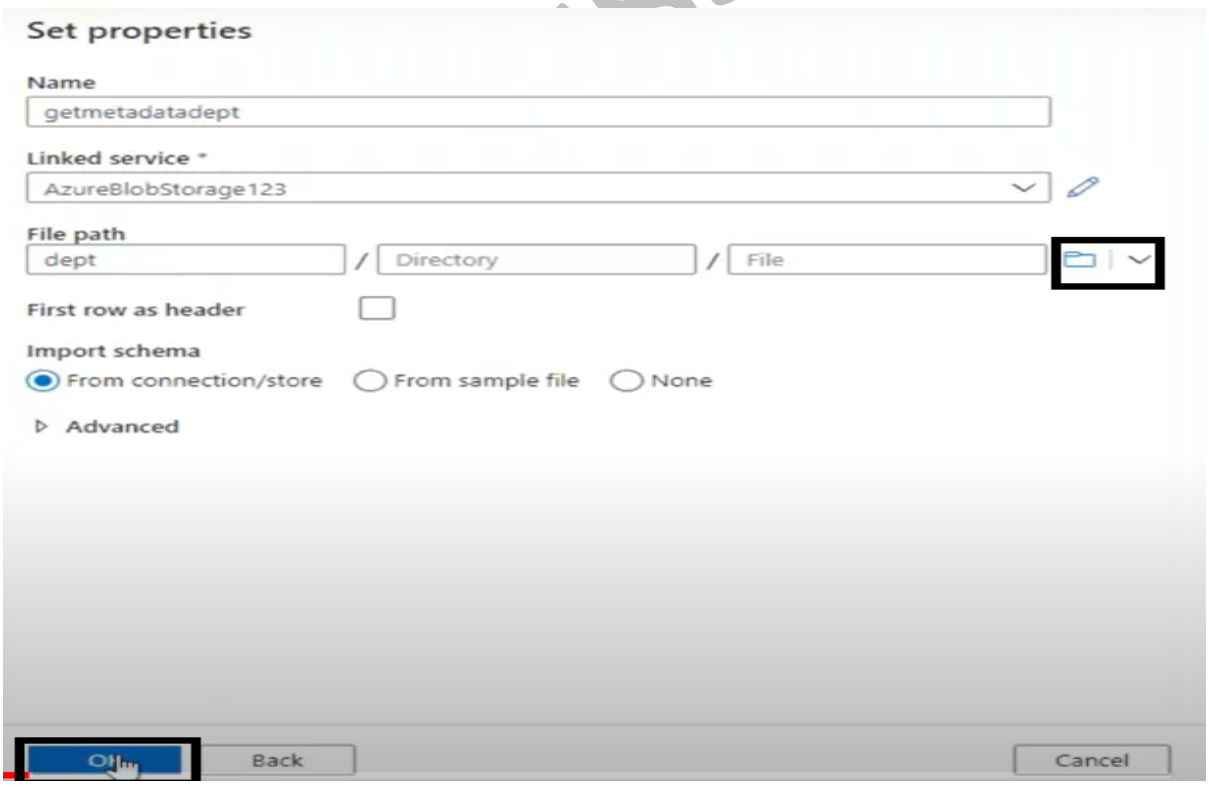
Step 25: Create new dataset for employee



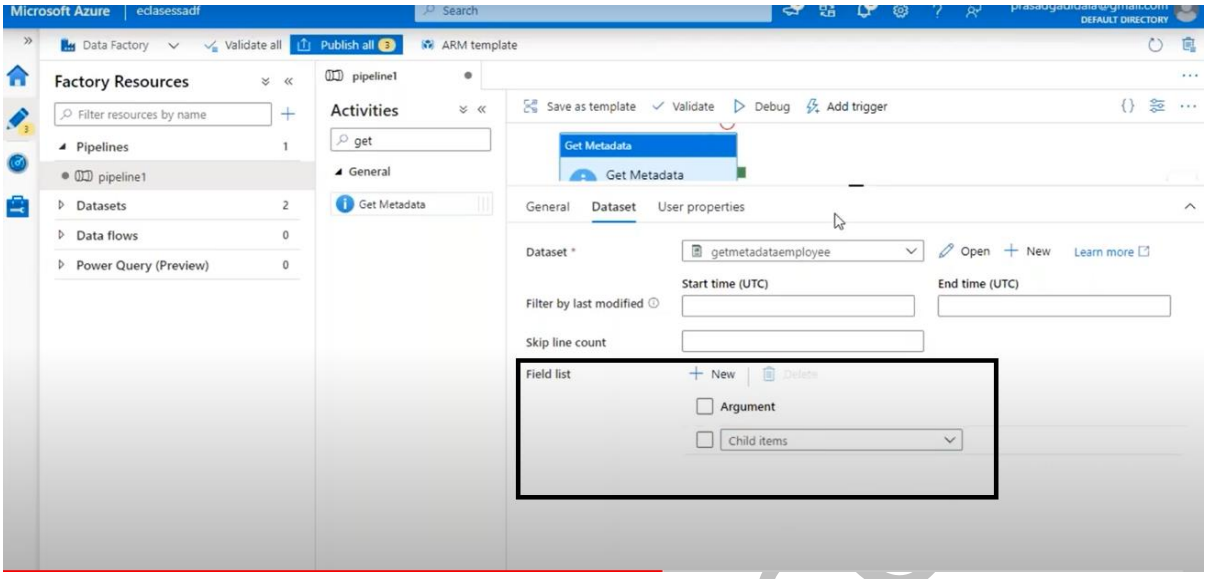




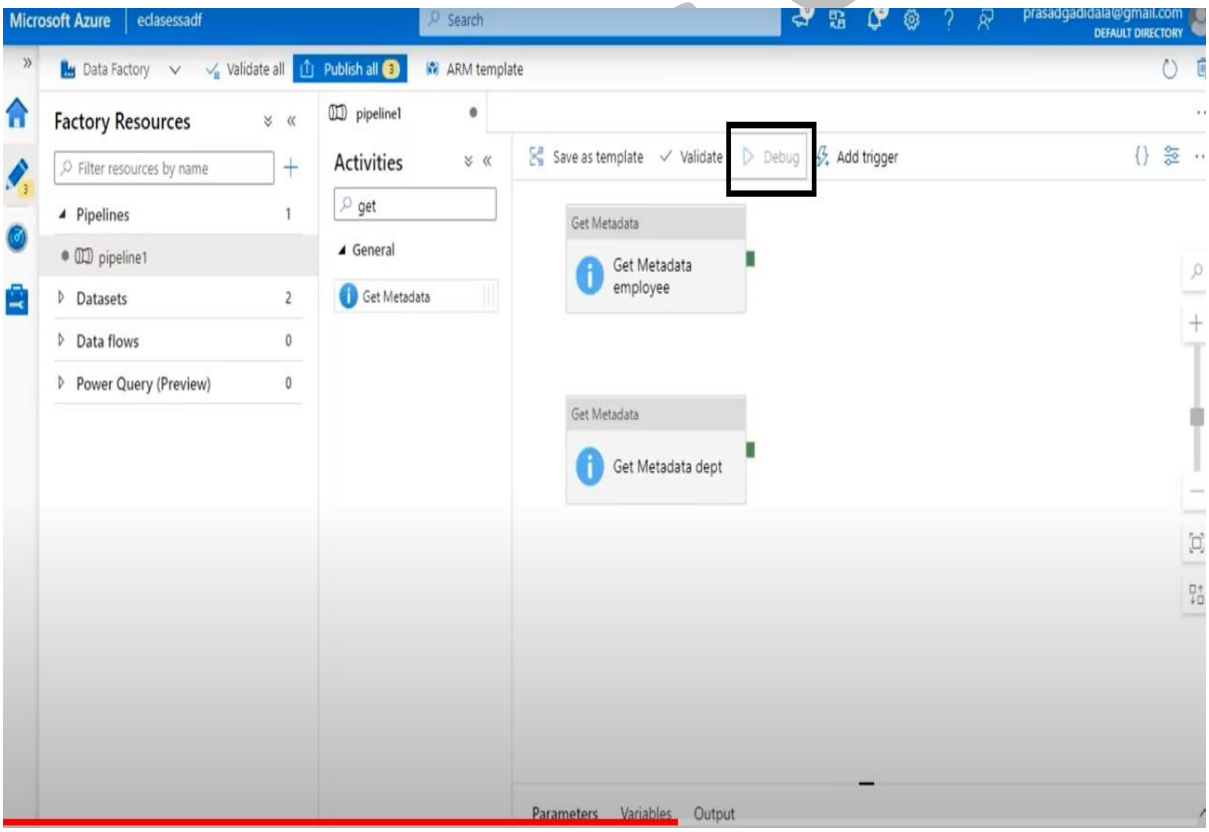
Step 26: Simply click on get metadata for dept and create new dataset for this



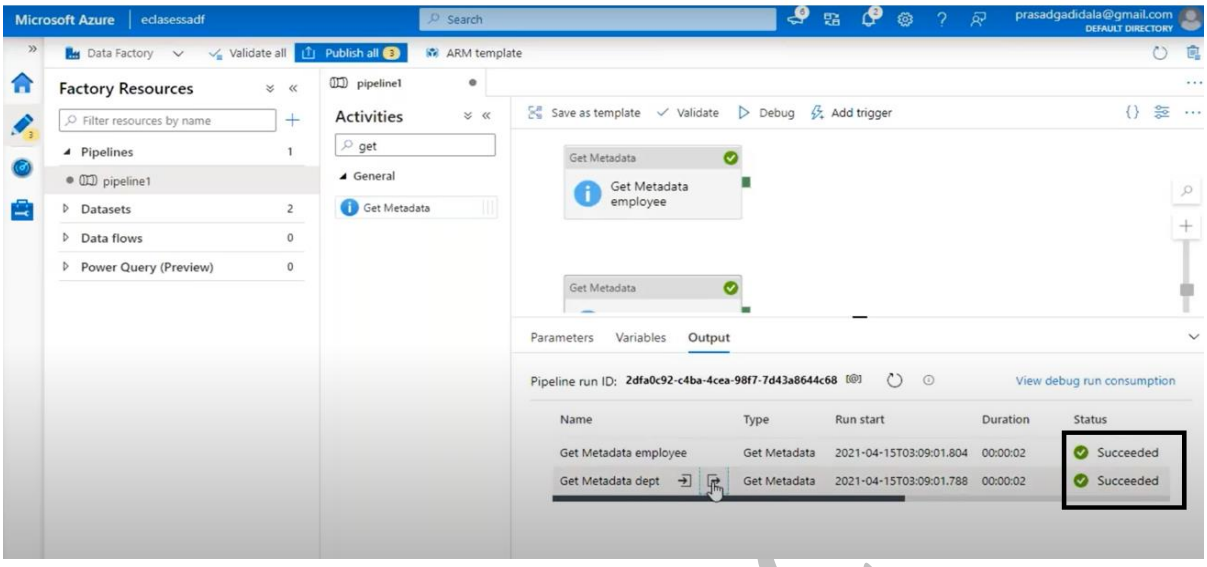
Step 27: Click on field list and select child items same as for get metadata dept also



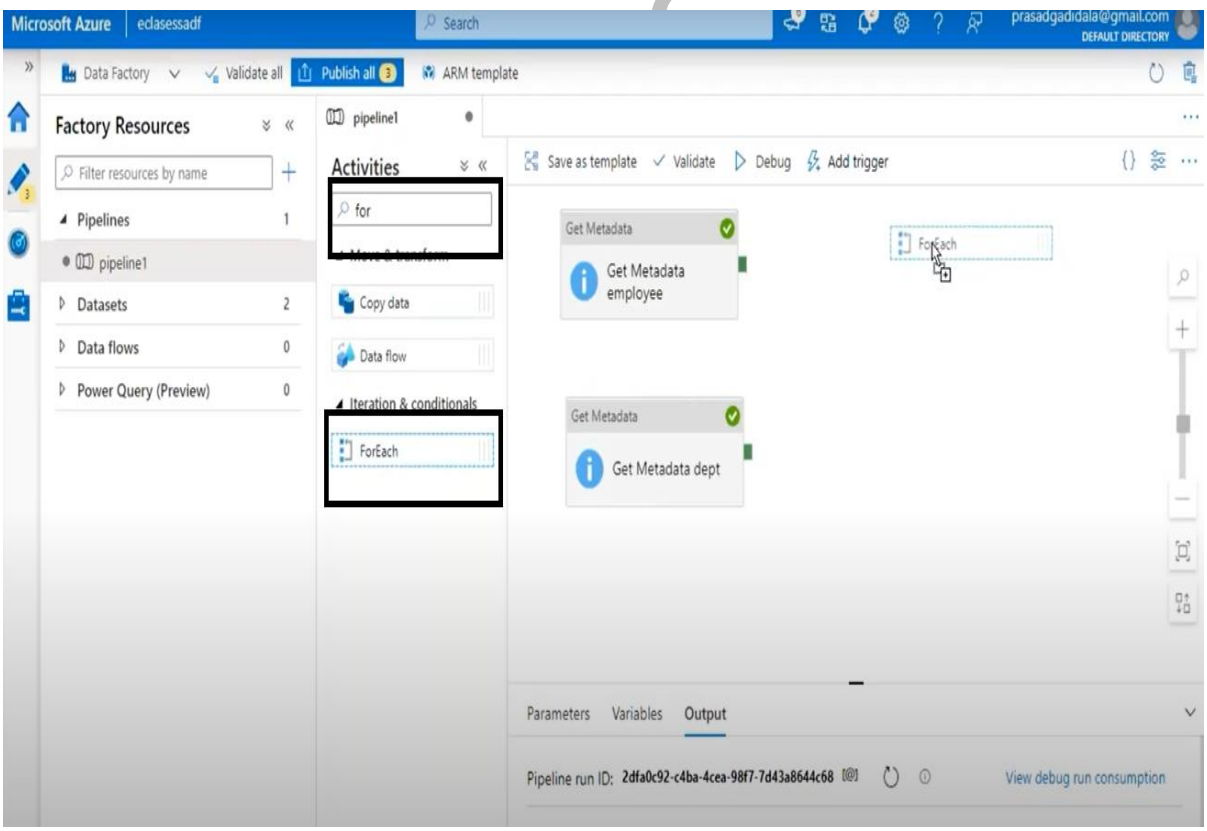
Step 28: Later on debug it and click on debug



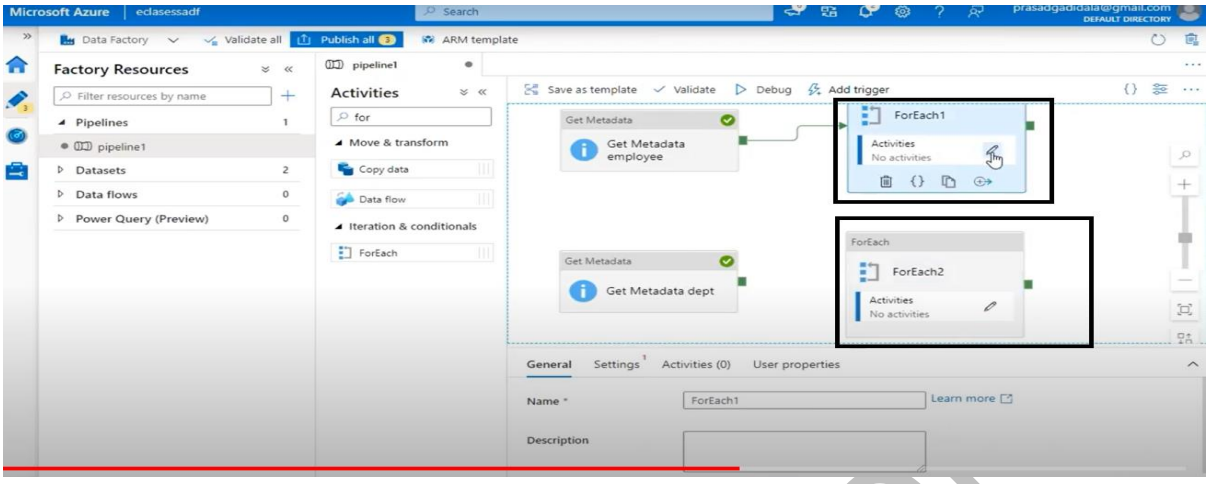
Debug succeeded



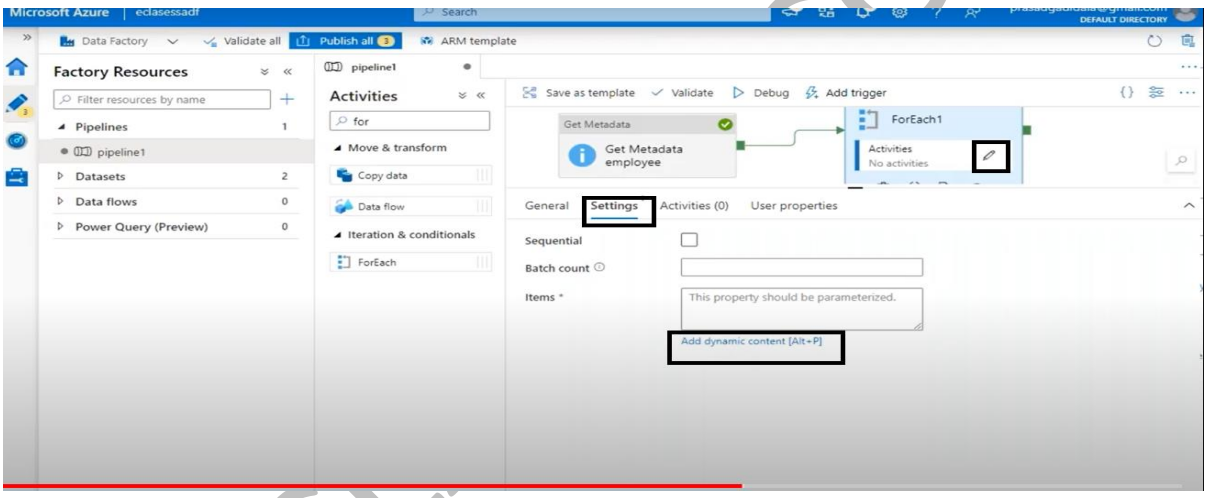
Step 29: We have to loop the get metadata activity to target table by using foreach activity

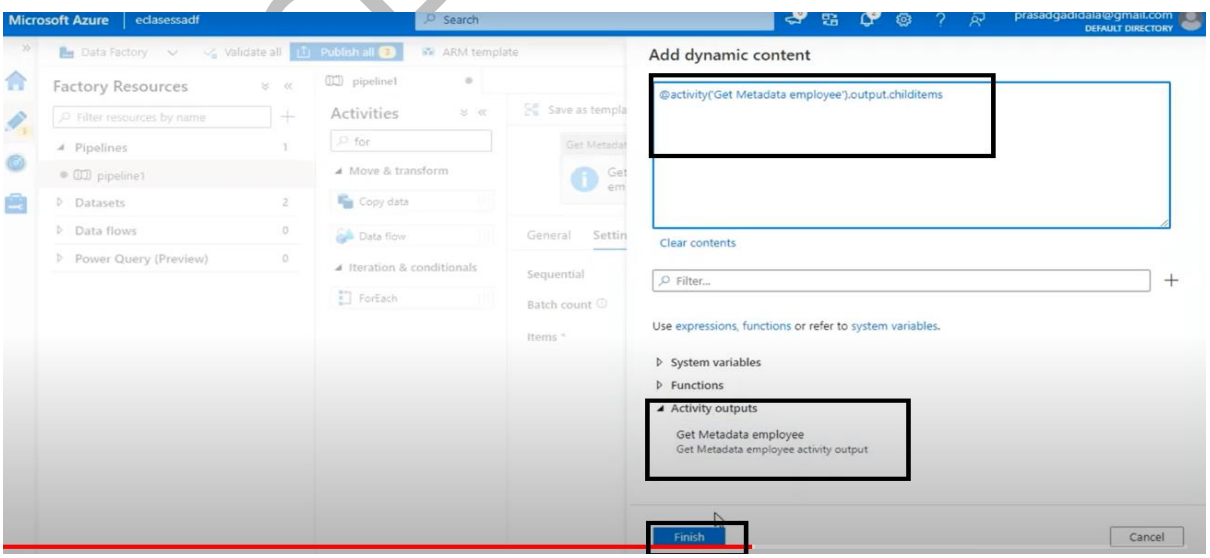


One for each activity for employee and another one for department

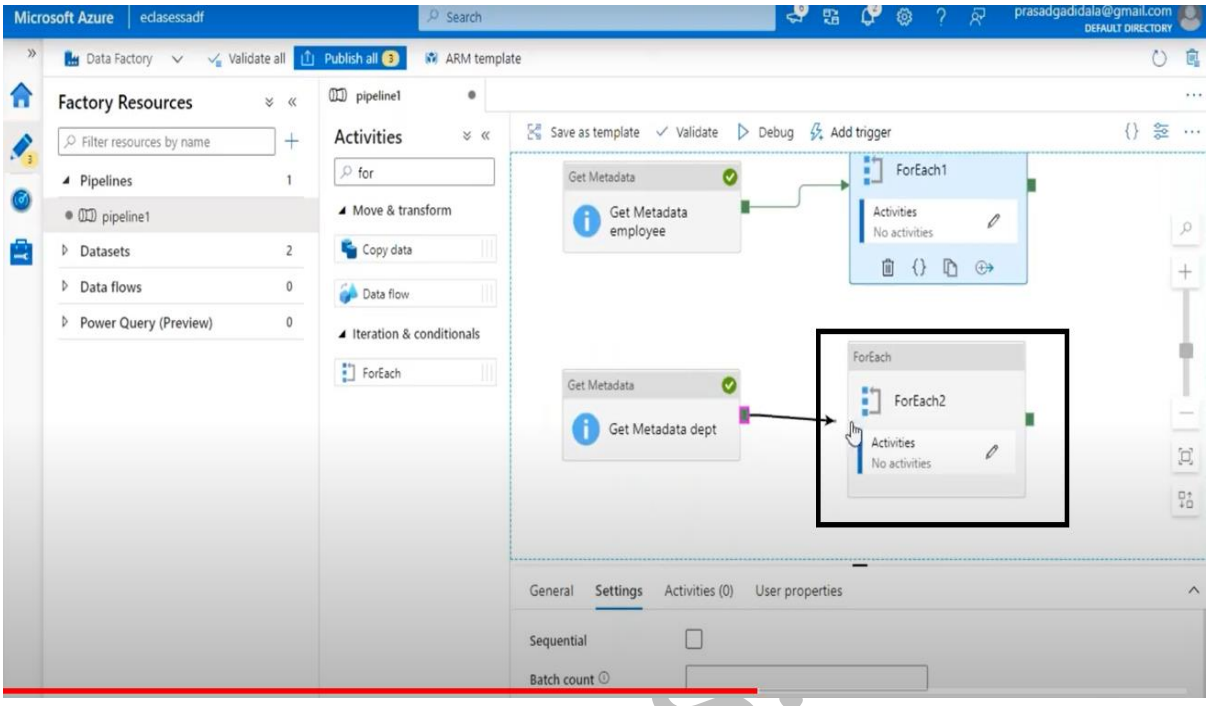


Simply click on edit on foreach for employee and set the settings click on add dynamic content

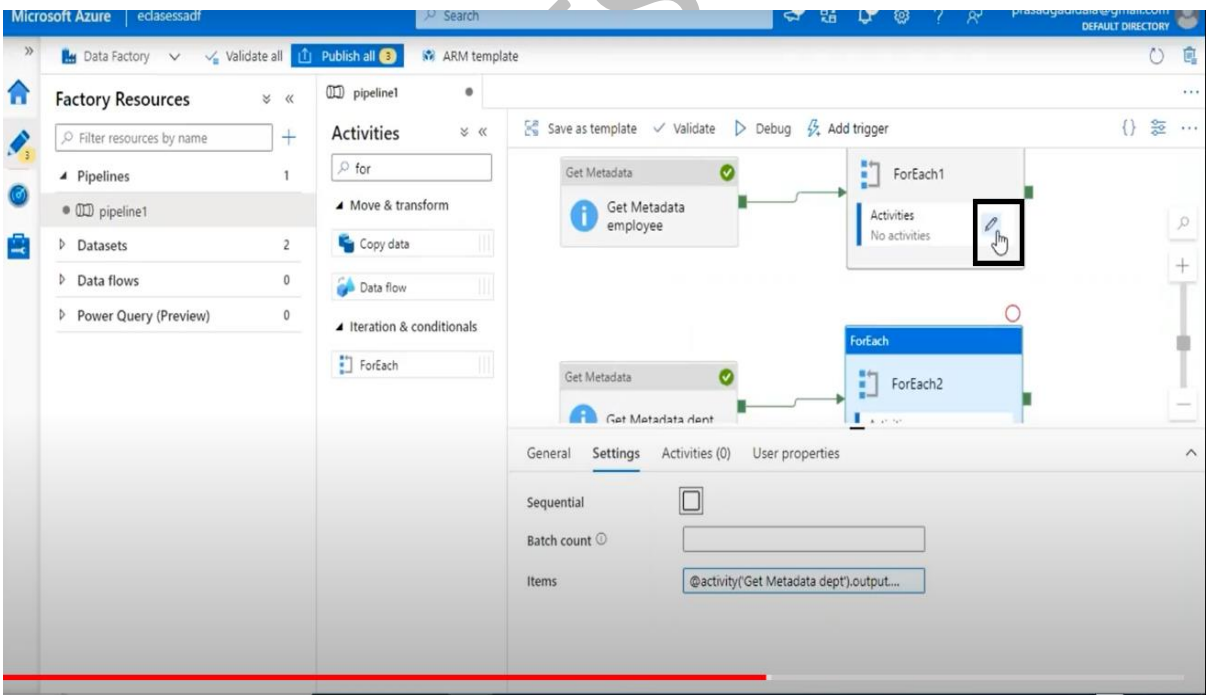




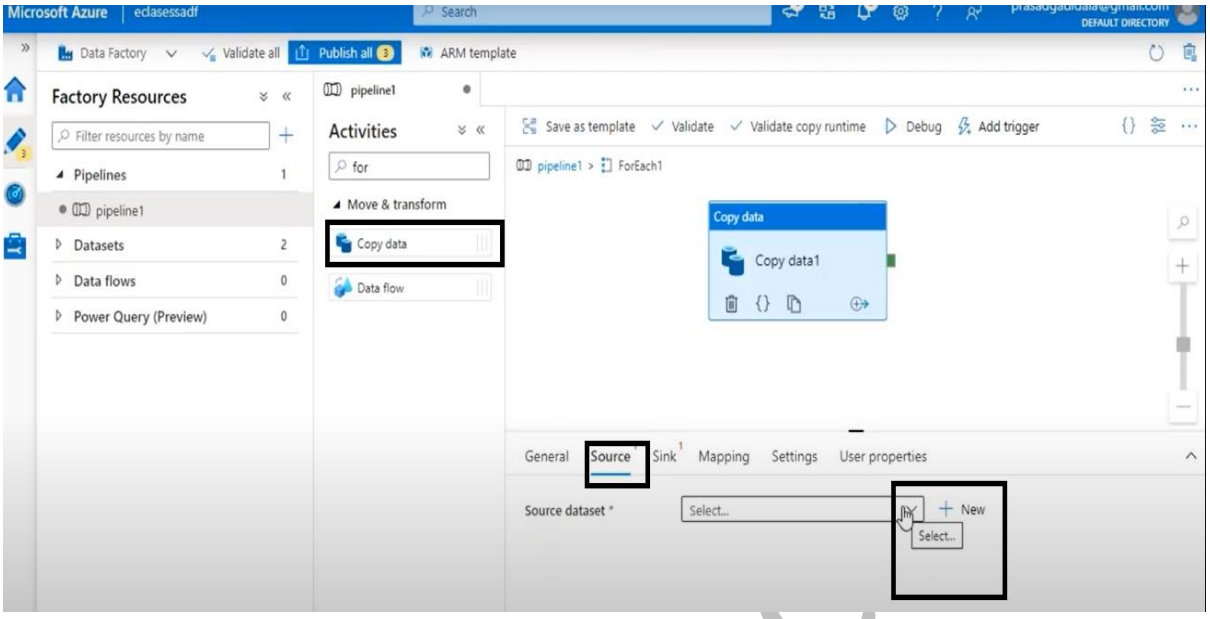
Same as for dept also add dynamic content



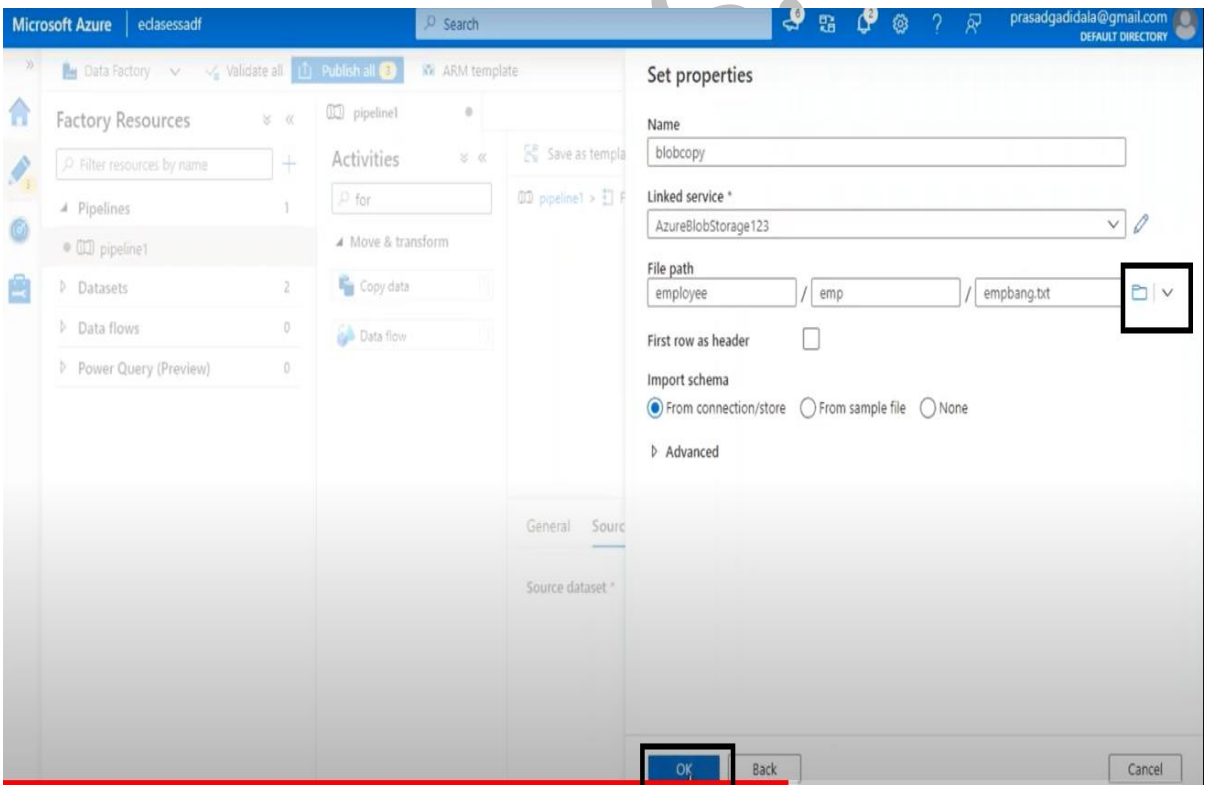
Step 30: Now click on employee foreach activity



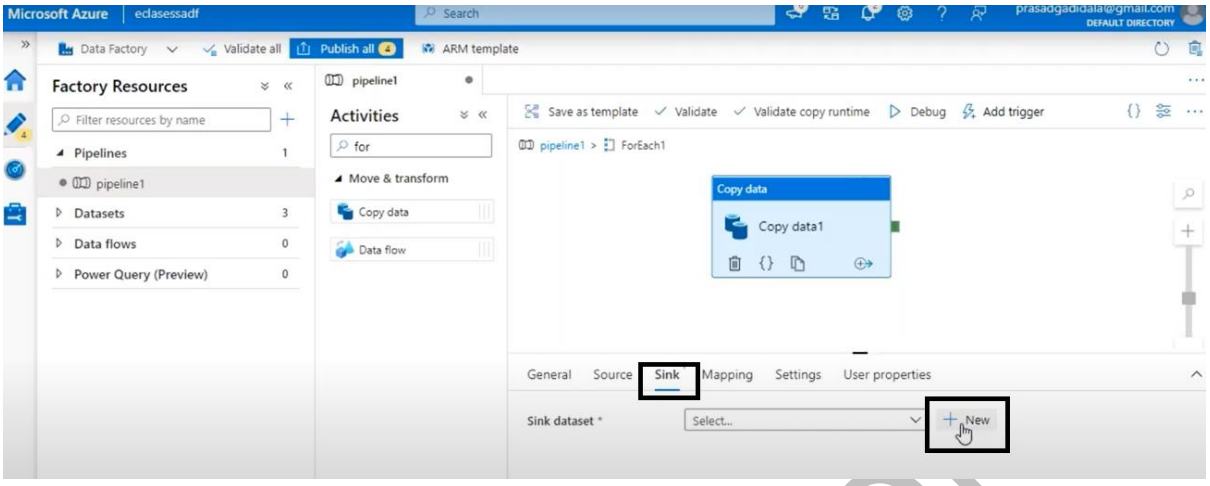
Take one copy data activity for copying data from source to destination

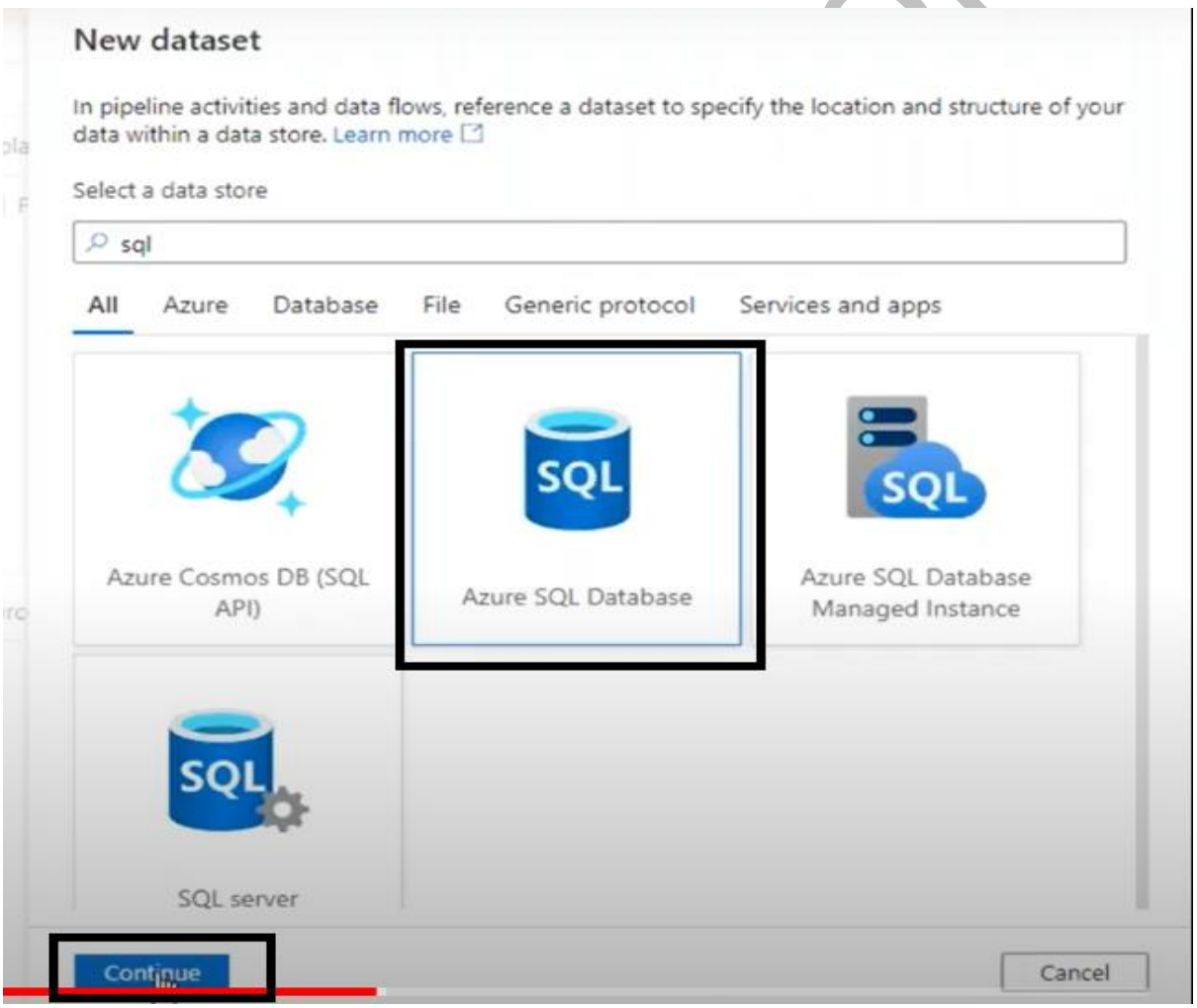


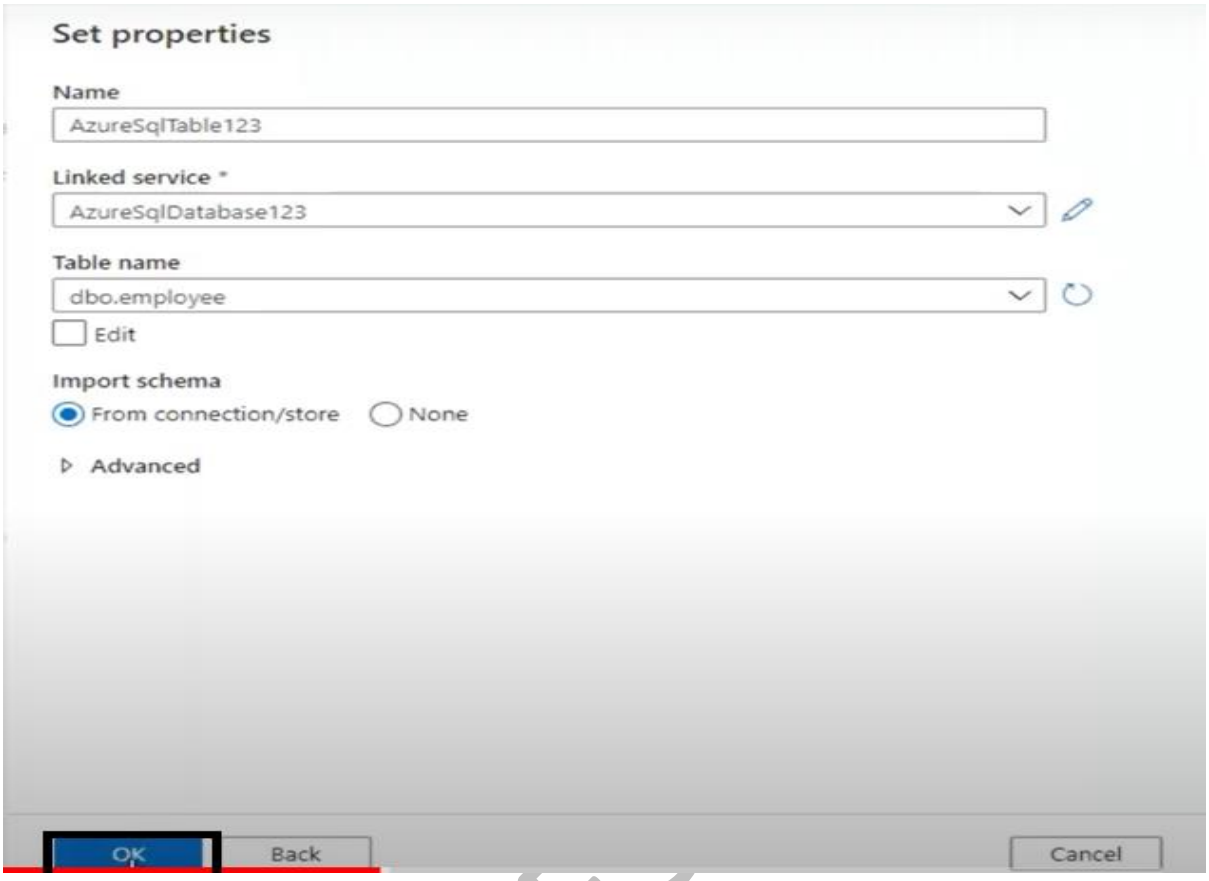
Click on new set properties for dataset



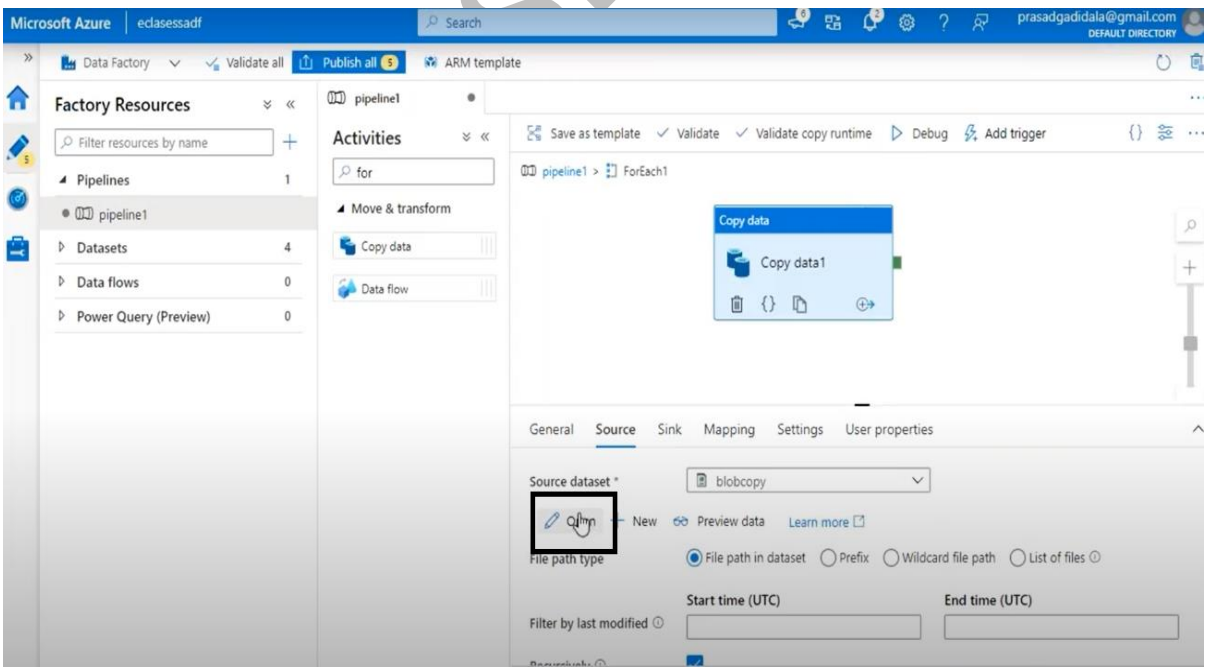
Similarly do for destination

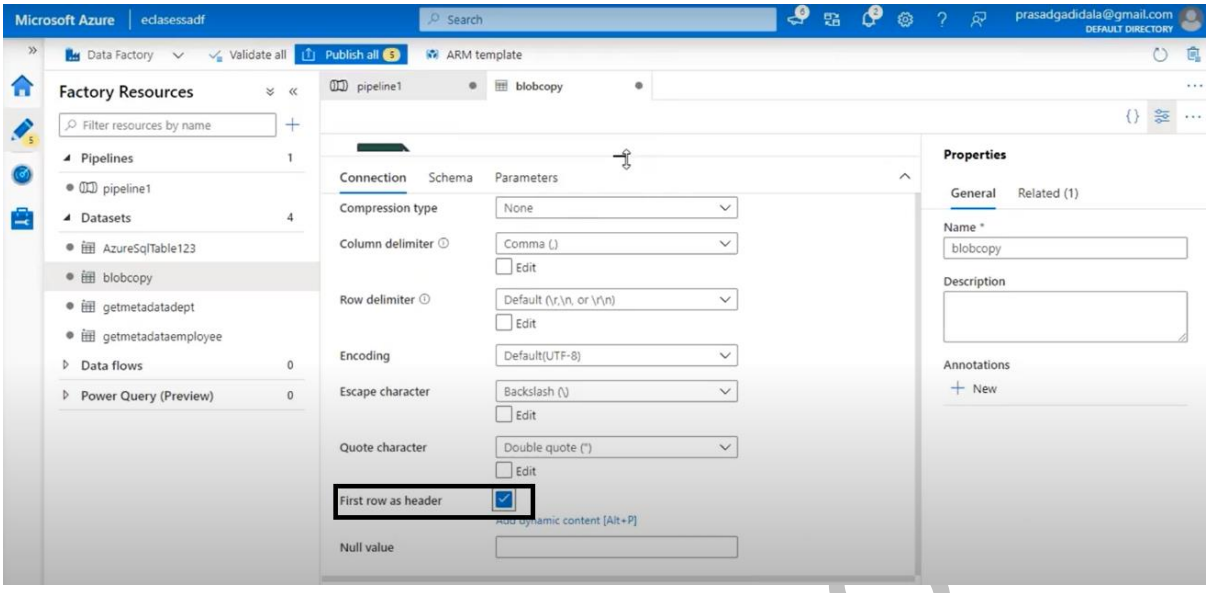




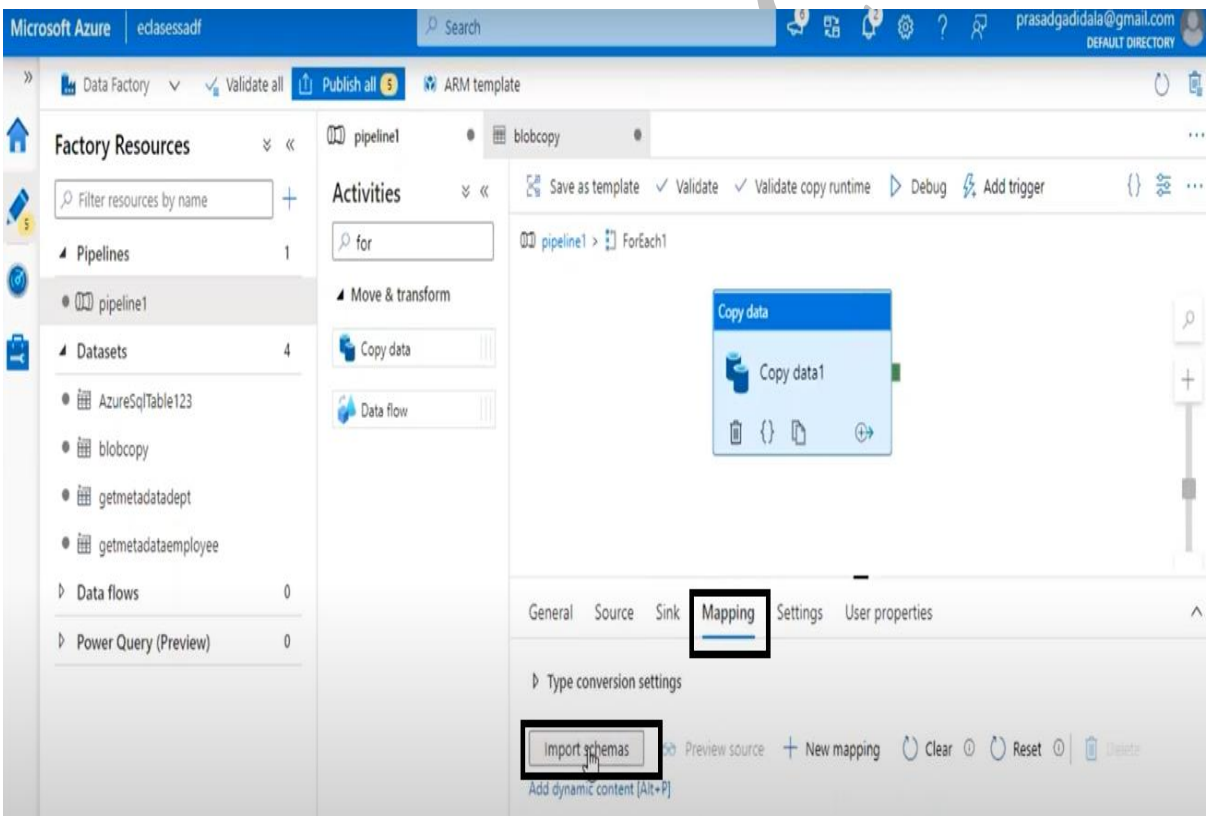


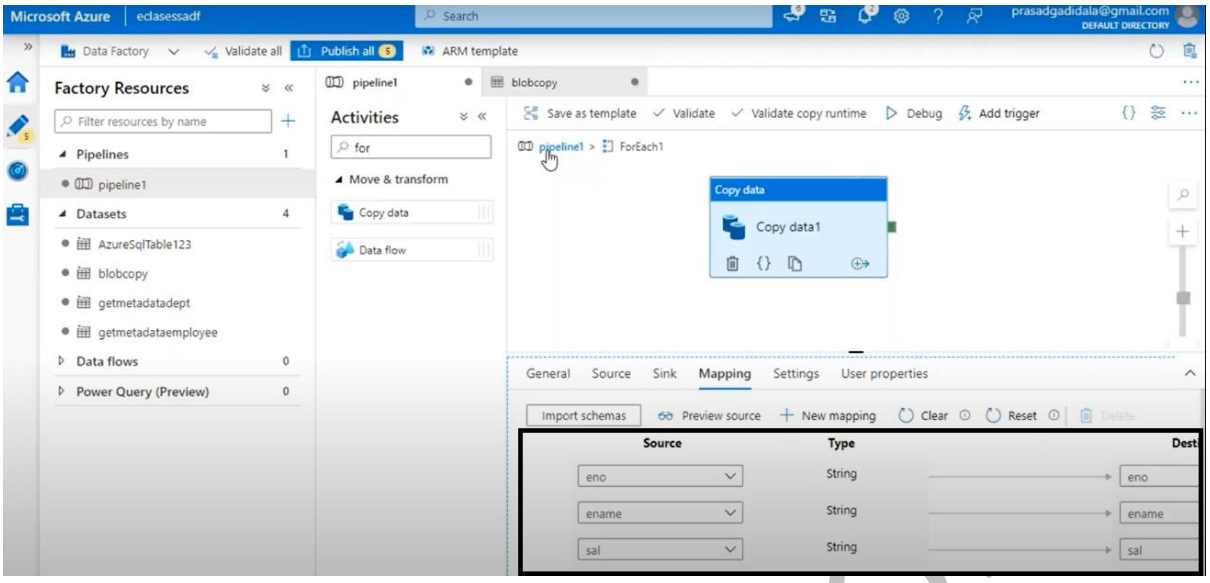
Before mapping we need to select first row as column and first row as header





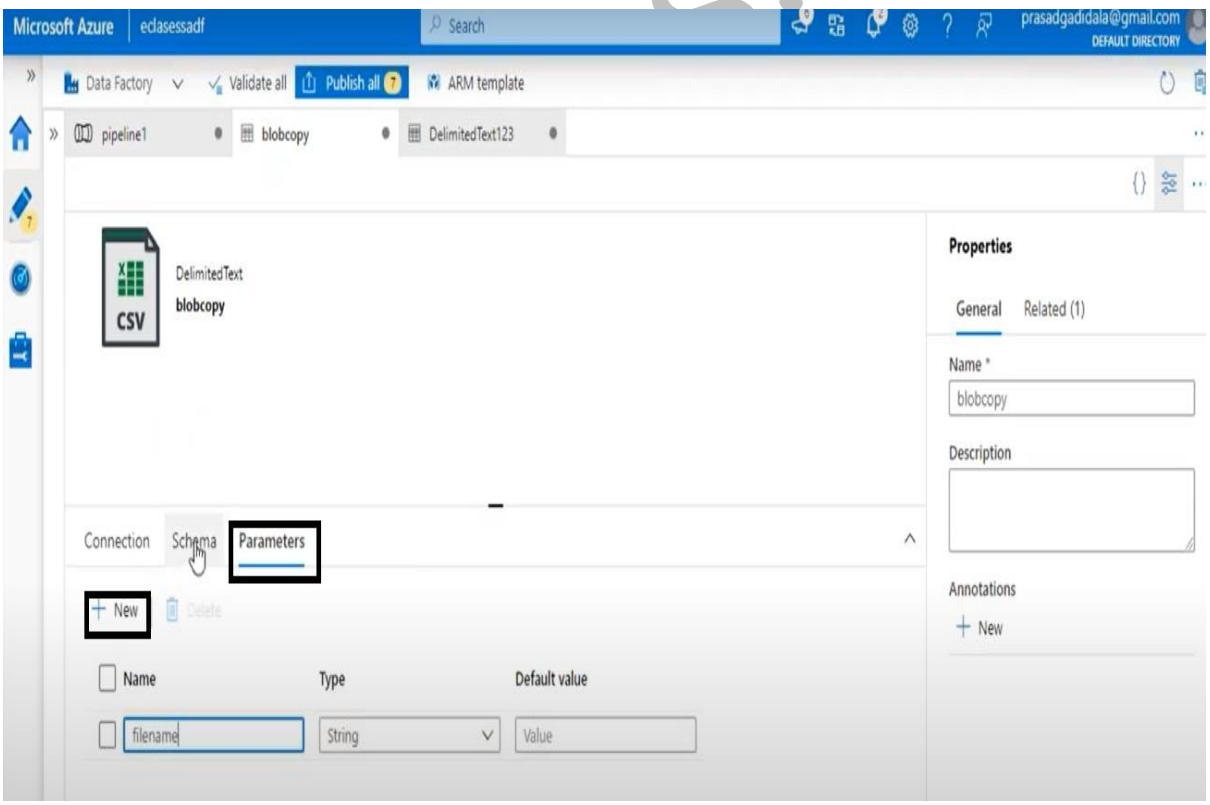
Click on mapping and click on import schemas

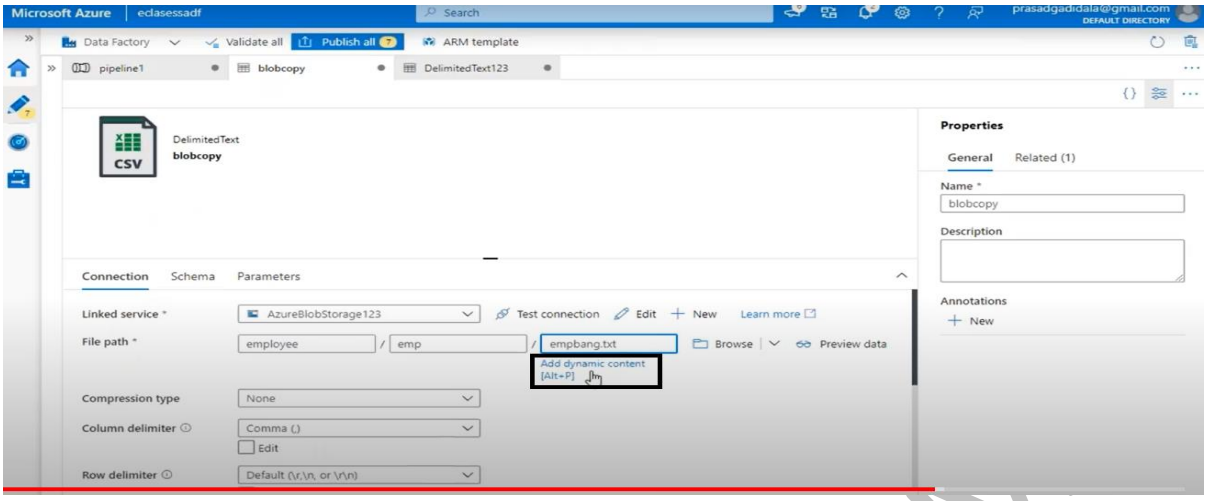


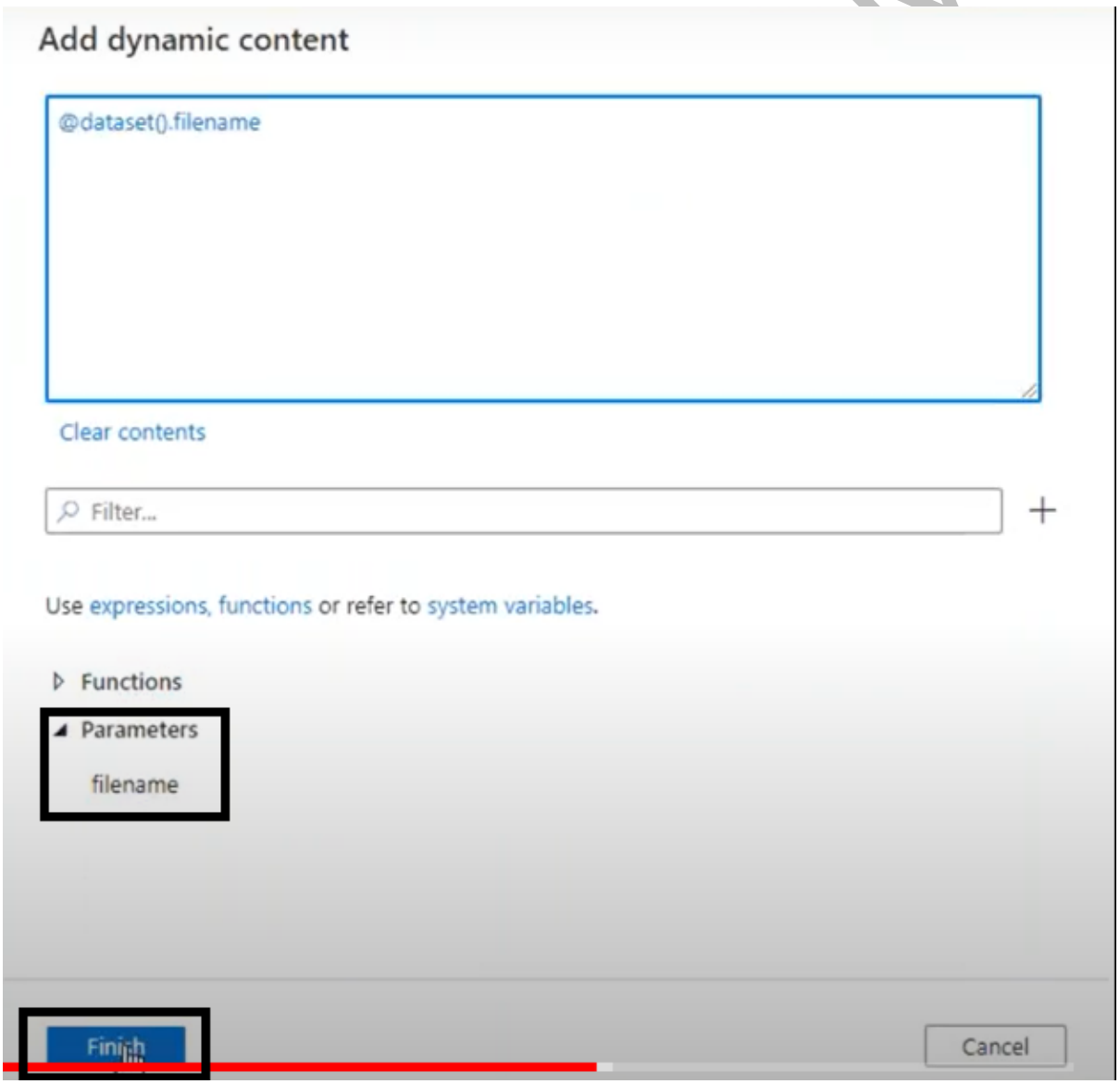


Step 31: Do same for department foreach activity

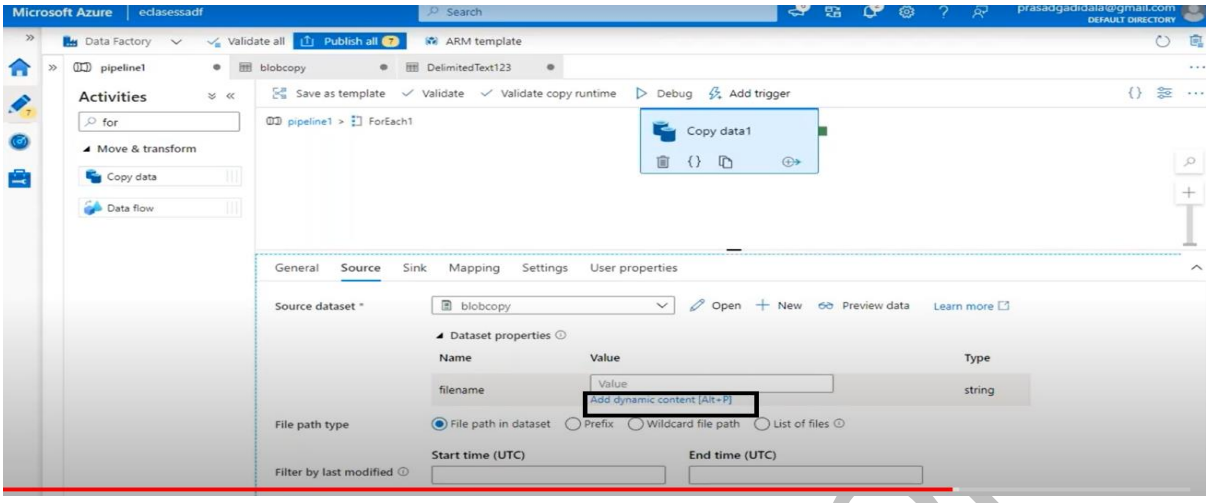
Step 32: Click on employee foreach activity and add parameters adding the filename dataset level

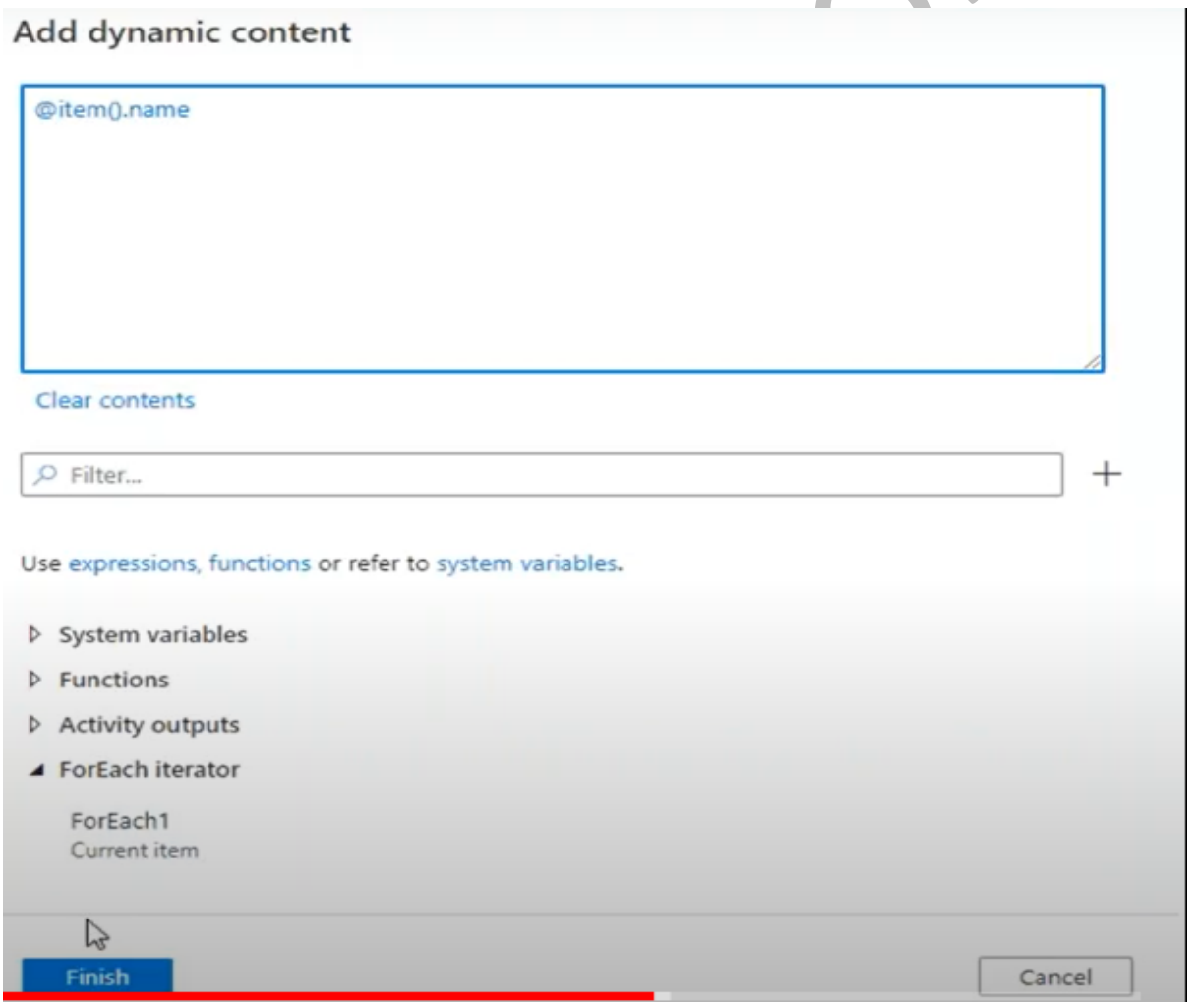



Microsoft Azure
eclasessadf
Search
prasadgadidala@gmail.com
DEFAULT DIRECTORY
Data Factory
Validate all
Publish all
ARM template
pipeline1
blobcopy
DelimitedText123
DelimitedText
blobcopy
Properties
General
Related (1)
Name *
blobcopy
Description
Annotations
New
Connection
Schema
Parameters
Linked service *
AzureBlobStorage123
Test connection
Edit
New
Learn more
File path *
employee
emp
empbang.txt
Browse
Preview data
Add dynamic content [Alt+P]
Compression type
None
Column delimiter
Comma (,)
Edit
Row delimiter
Default (\r,\n, or \r\n)


Add dynamic content
@dataset().filename
Clear contents
Filter...
Use expressions, functions or refer to system variables.
Functions
Parameters
filename
Finish
Cancel

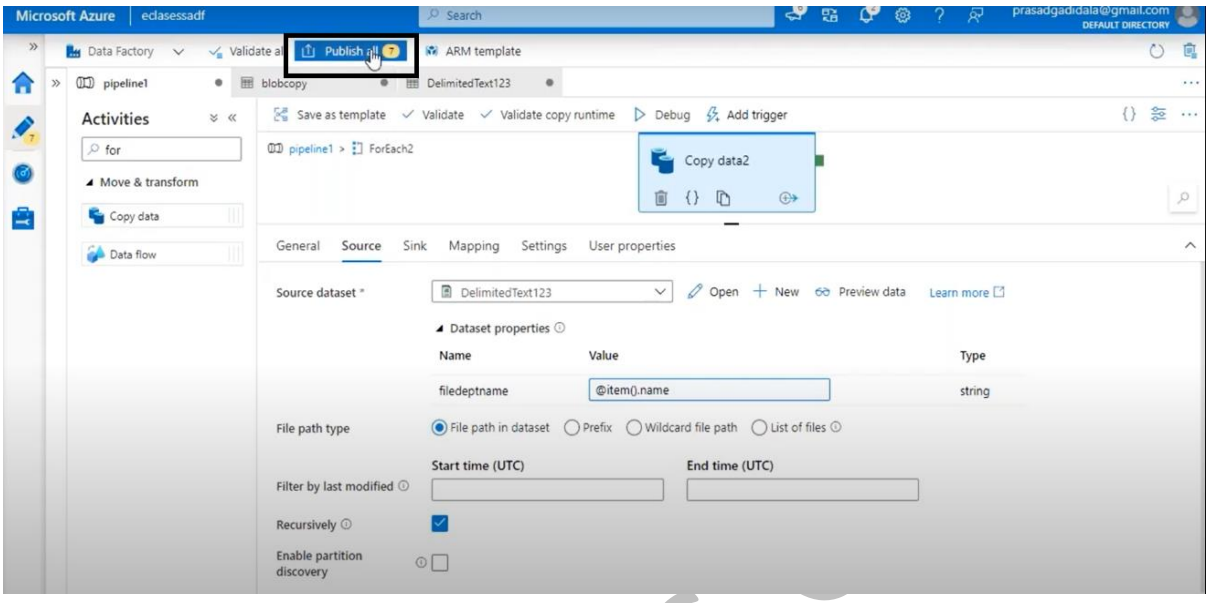
We need to pass the value dynamically



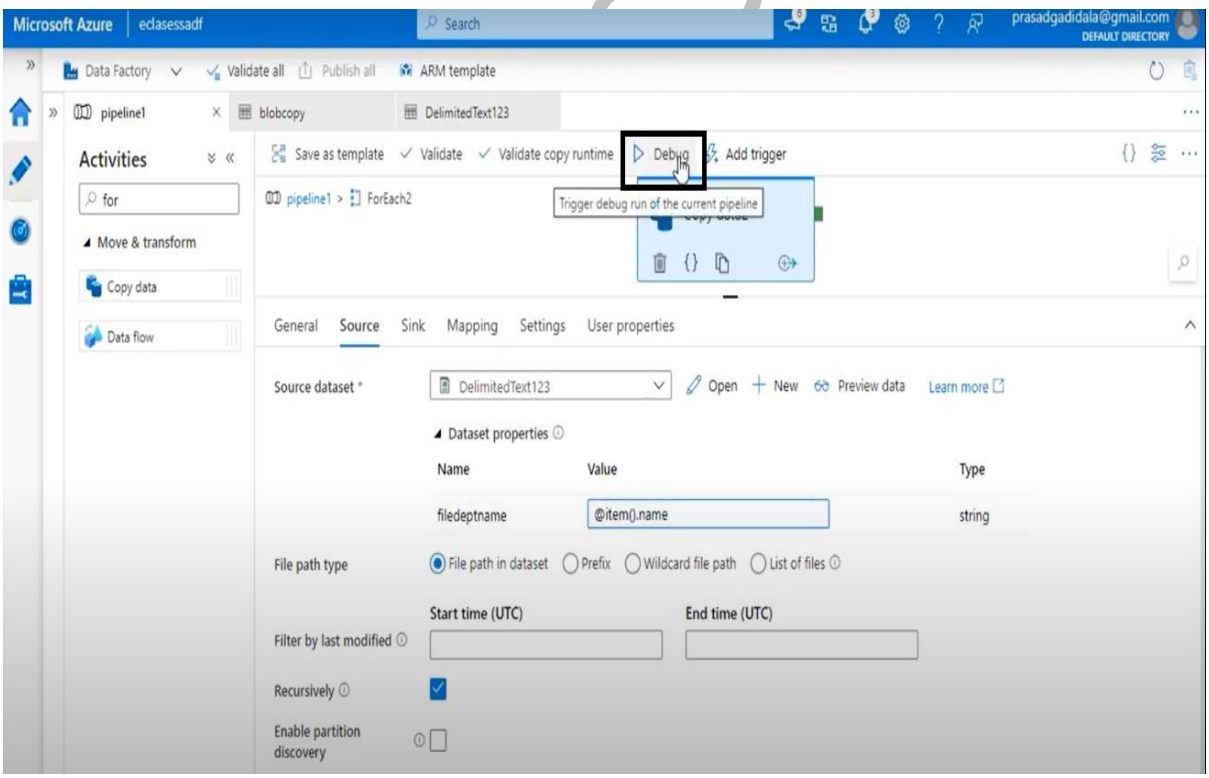


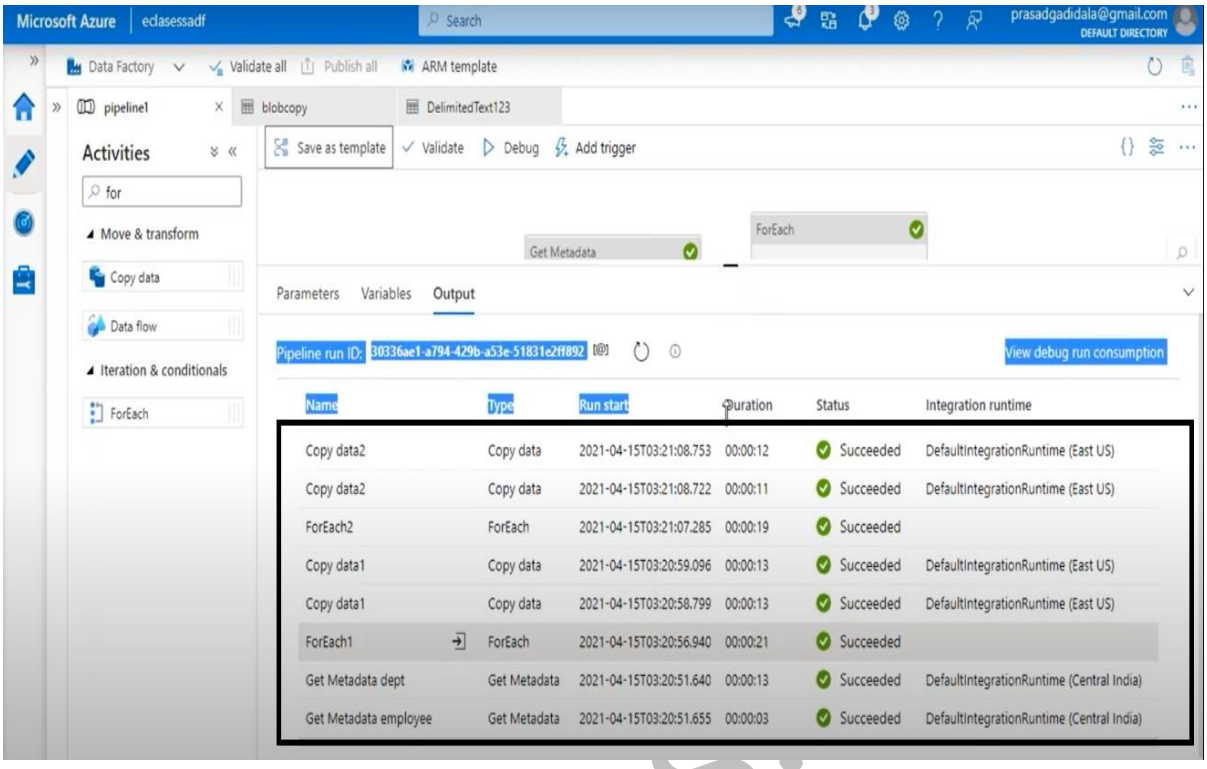
Similarly we can do other one as well

Step 33: We need to publish all those things

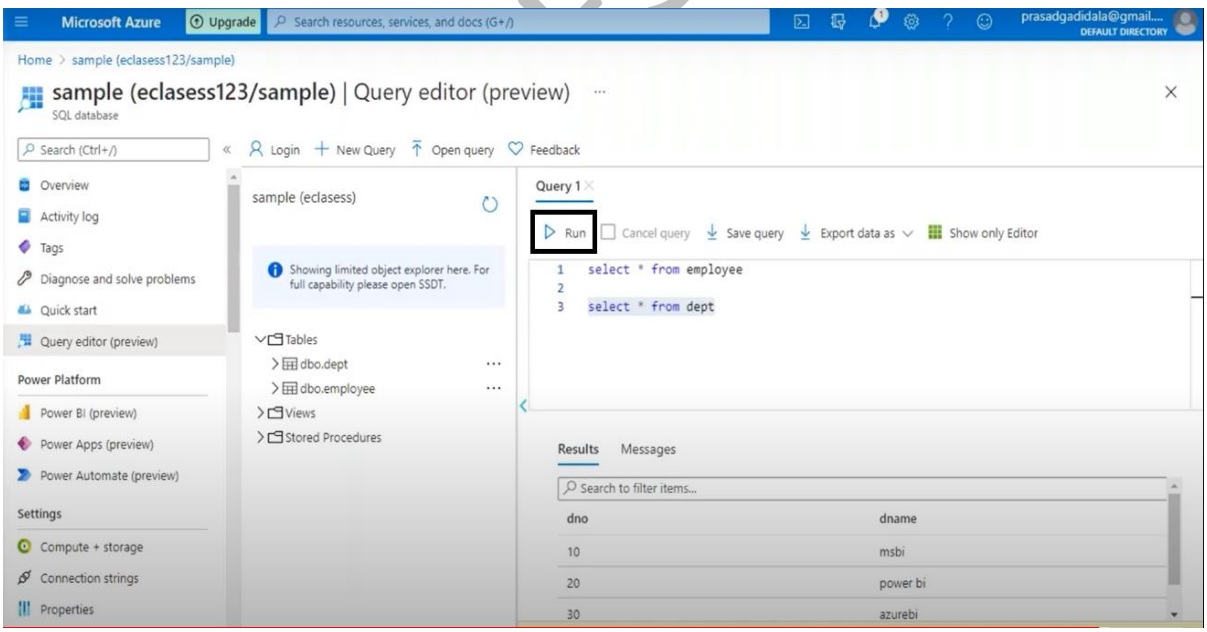


Step 34: Then we need to do debug



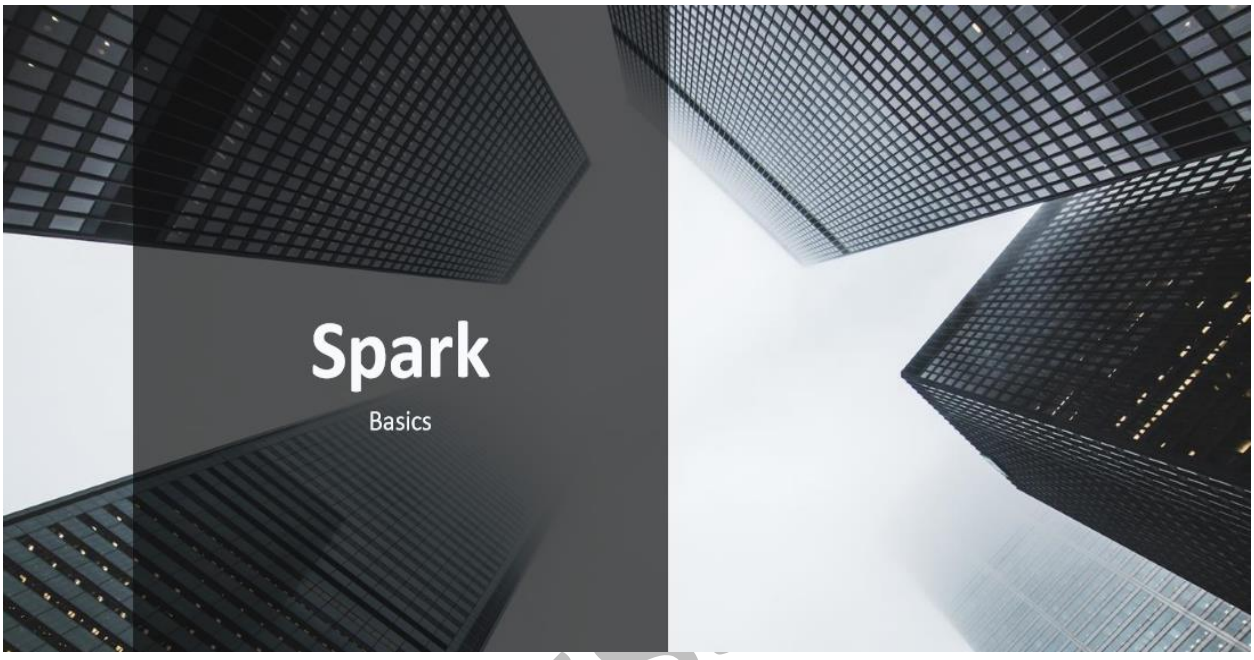


Step 35: Check the data loaded or not open the azure sql database which we created and click on query editor



Azure Data Bricks

Azure Data Bricks build up on Top of Spark Engine frame work to processing of huge volume of data for big data analytics.



Traditional Enterprise systems are used for low volume of data processing.

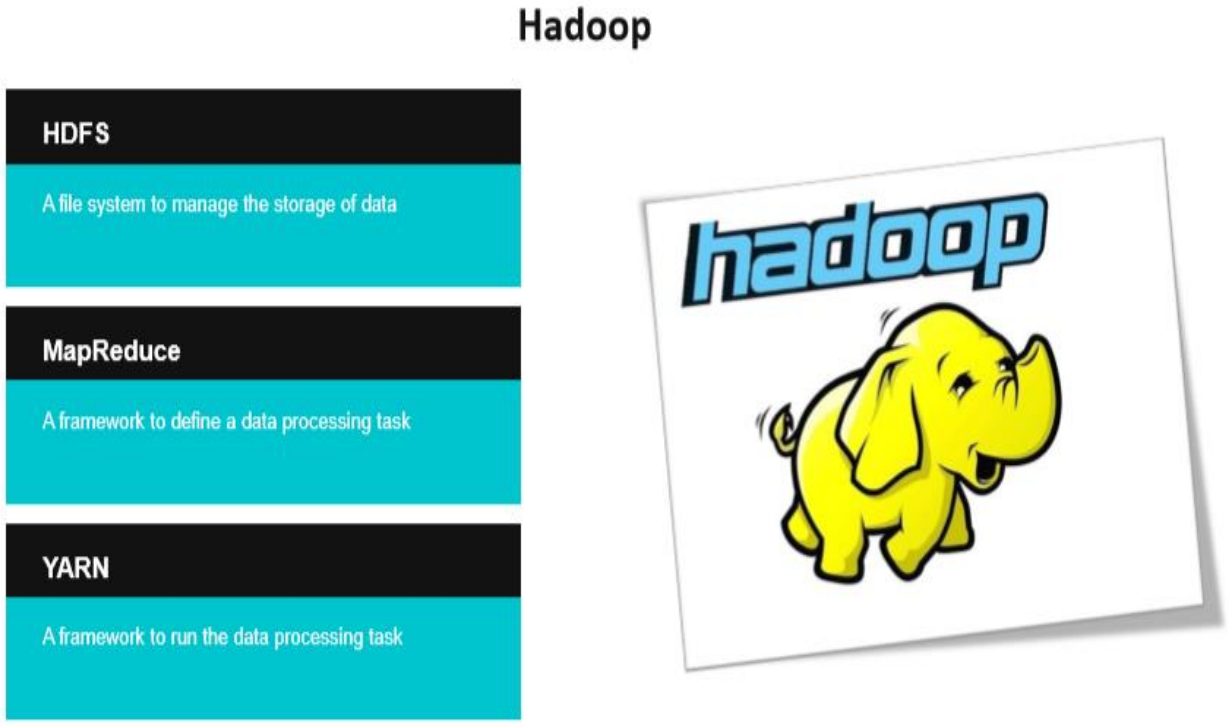
To resolve this issue google has introduced algorithm Map Reduced Program.

Hadoop: It is an open source, Java based framework used for storing and processing big data. The data is stored on inexpensive commodity servers. Data can be stored in Hadoop Distributed File System (HDFS)

Map Reduce: It is a software framework and programming model used for processing huge amounts of data.

Yarn: YARN is the resource management and job scheduling technology. It allows the data stored in HDFS (Hadoop Distributed File System) to be processed and run by various data processing engines such as batch processing, stream processing, interactive processing, graph processing.

ECLASESS



Hadoop Map Reduce dividing program into small programs and assign them into multiple clusters, finally all results from different clusters combined into one computer and returning result.

But there are some pitfalls in map reduce program potentially this is using hard disk to read and write data while processing.

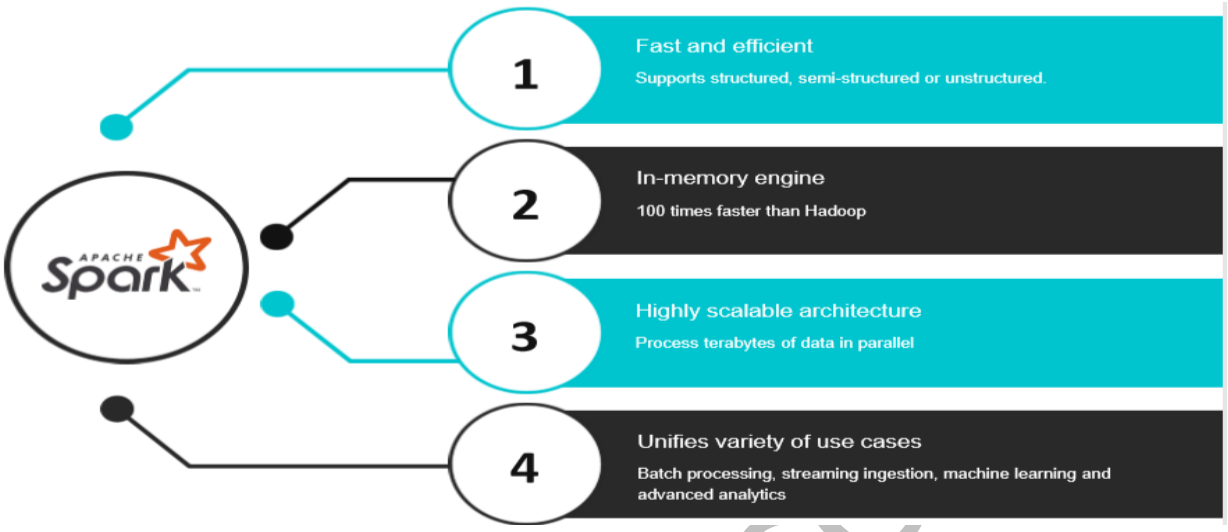
When to use Hadoop MapReduce:

Linear processing of huge data sets. Hadoop MapReduce allows parallel processing of huge amounts of data.

Economical solution, if no immediate results are expected. MapReduce a good solution if the speed of processing is not critical. If data processing can be done during night hours, consider using Hadoop MapReduce.

Apache Spark: It is a data processing framework that can quickly perform processing tasks on very large data sets, and can also distribute data processing tasks across multiple computers.

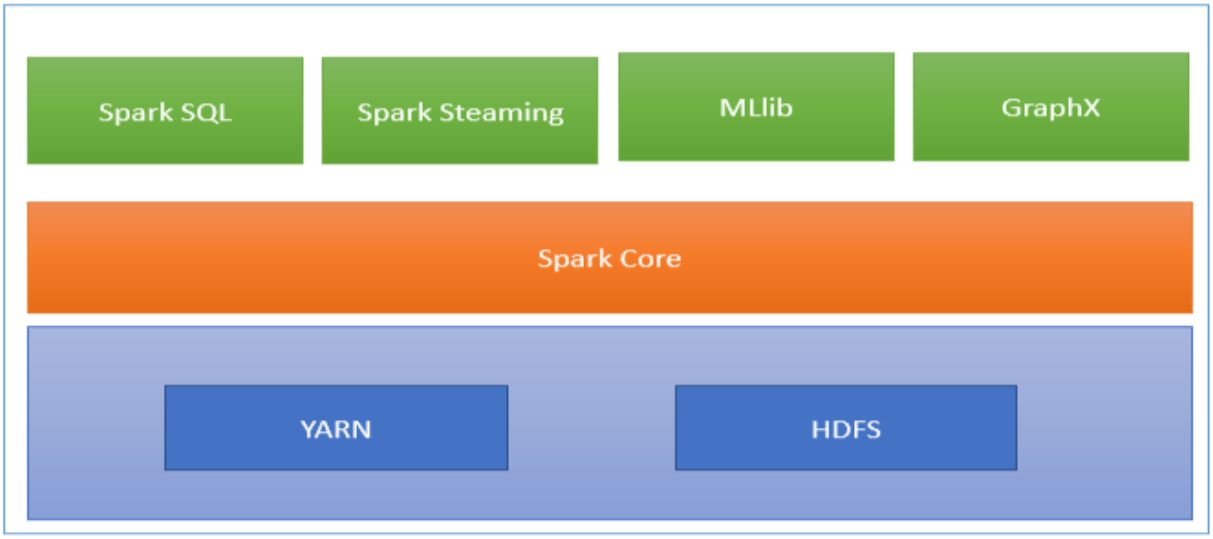
When to use Spark:



1. **Spark has its unique features:**

    1. Fast and efficient to process structured, unstructured and semi structured data.
    2. In memory technology, processing data through RAM instead of Hard disk. It's faster 100 times then map reduce.
    3. Highly Scalable and processing terabytes of data in parallel
    4. Real-time processing. If a business needs immediate insights, then they should opt for Spark and its in-memory processing.
    5. Iterative processing. Spark's Resilient Distributed Datasets (RDDs) enable multiple map operations in memory, while Hadoop MapReduce has to write interim results to a disk.
    6. Graph processing. Spark's computational model is good for iterative computations that are typical in graph processing. And Apache Spark has GraphX – an API for graph computation.
    7. Machine learning. Spark has MLlib – a built-in machine learning library, while Hadoop needs a third-party to provide it. MLlib has out-of-the-box algorithms that also run in memory.
    8. Joining datasets. Due to its speed, Spark can create all combinations faster, though Hadoop may be better if joining of very large data sets that requires a lot of shuffling and sorting is needed

ECLASESS.COM



There are 6 components in Apache Spark Ecosystem

1. Apache Spark- Spark Core,

   Spark Core is embedded with a special collection called RDD (resilient distributed dataset).

   RDD are building blocks of spark

   An RDD is a collection of entities (Rows and Records)

   There are two operations performed on RDDs: Transformation and Action-

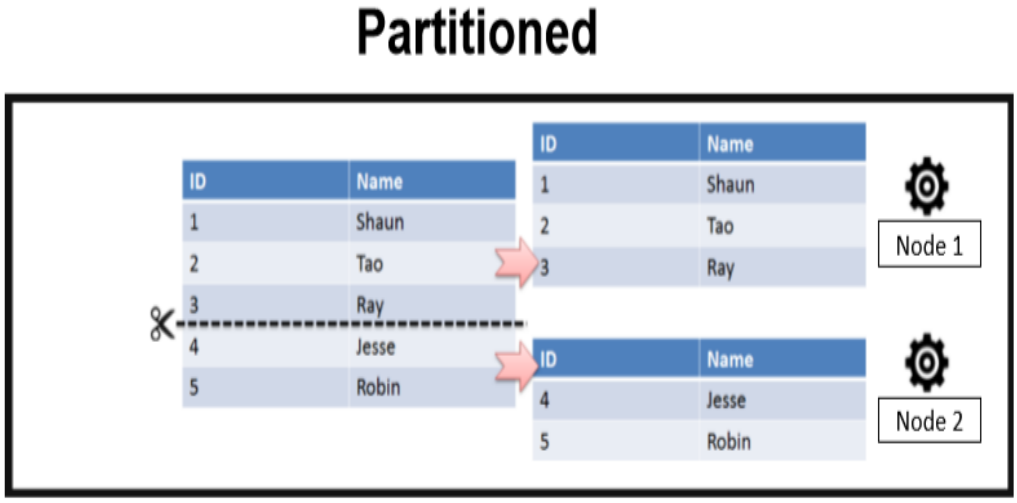
   Transformation: It is a function that produces new RDD from the existing RDDs.

   Action: In Transformation, RDDs are created from each other. But when we want to work with the actual dataset, then, at that point we use Action.

ECLASESS.

Characteristics of RDD:

Partitioned: Split data parallel across multiple nodes in cluster.



Immutable: RDDs once created can't be changed.

- A data set loaded in to RDD
- The user may define a chain of transformations on the dataset.

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	Itotha	28

Example: Transformation

1. Load Data
2. Pick only the 3rd column
3. Sort the values

Example: Action

1. The first 10 rows
2. A count
3. A Sum

Request a result using an action

Transformation is executed only when a result is requested

Resilient: Can be reconstructed if any of node crashes when it got failed.



Allows RDDs to be reconstructed when nodes crash

Allows RDDs to be Lazily instantiated (materialized) when accessing the results

Spark keeps a record of the series of transformations requested by the use

RDD can be created in 2 ways

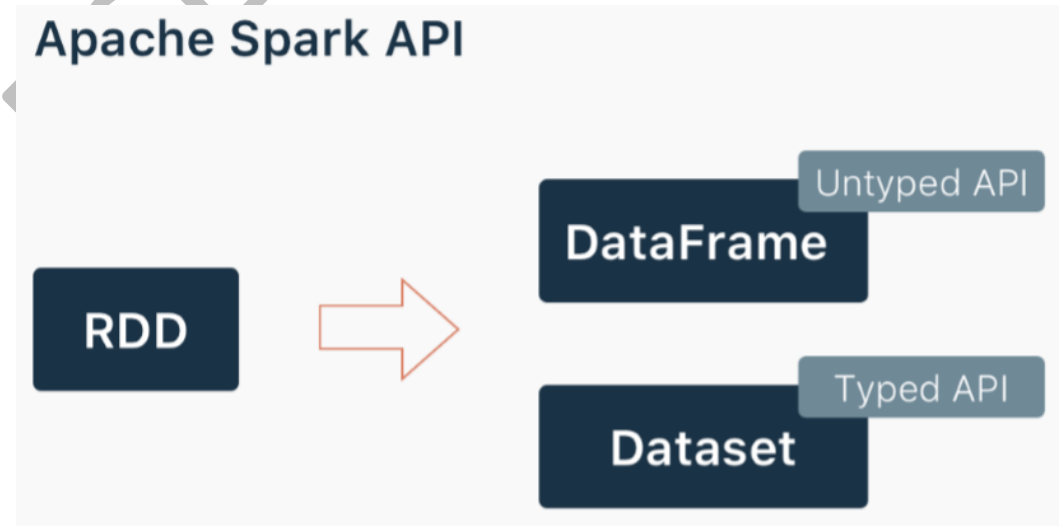
• Reading a file

• Transforming another RDD

RDD turns to be Data Frame or Data Set based on Type of API used to Query.

Data Sets are Present in Scala and Java and R.

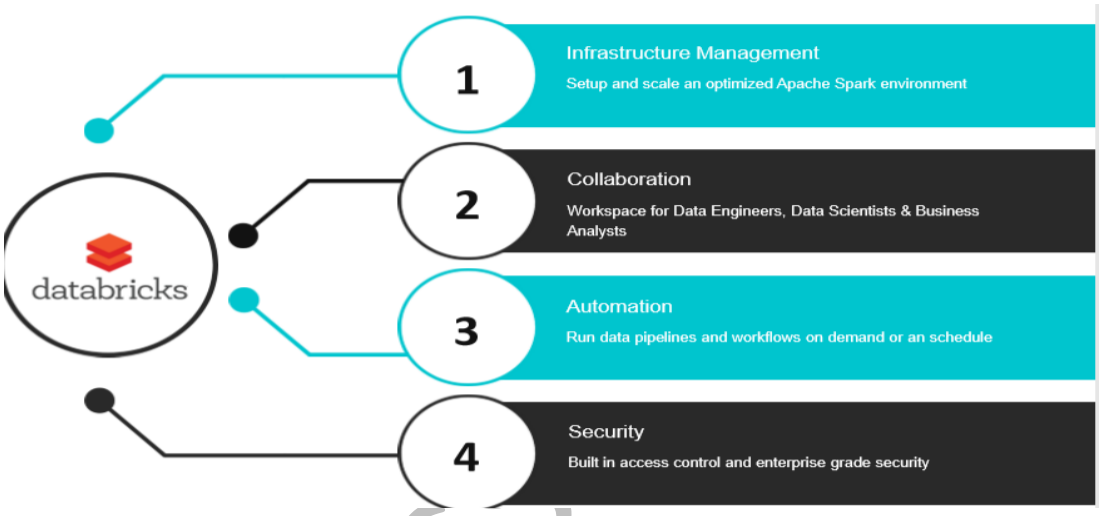
Data Frames are Present in All Languages.



Data Bricks:

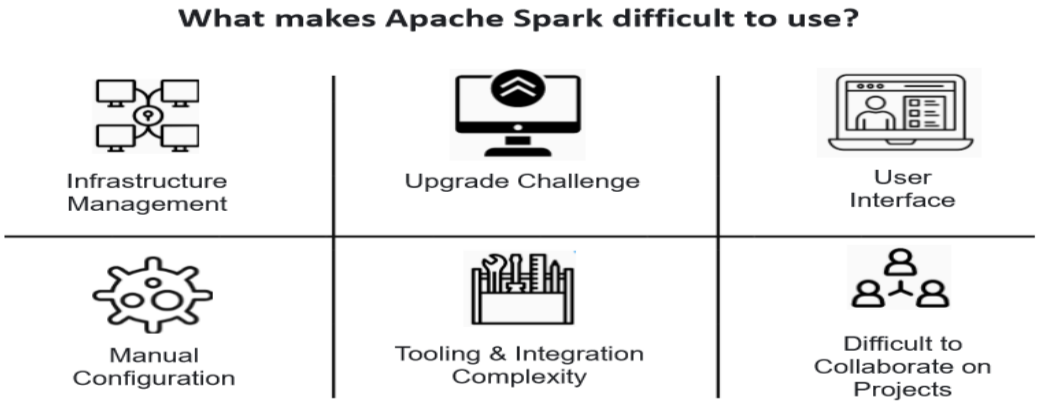
This is distributed processing of data with In-memory technology also supporting languages Scala, Python, SQL and R & Java.

This is fully managed User Interface with Strong Security.



Why Data Bricks introduced on Top of Apache Spa

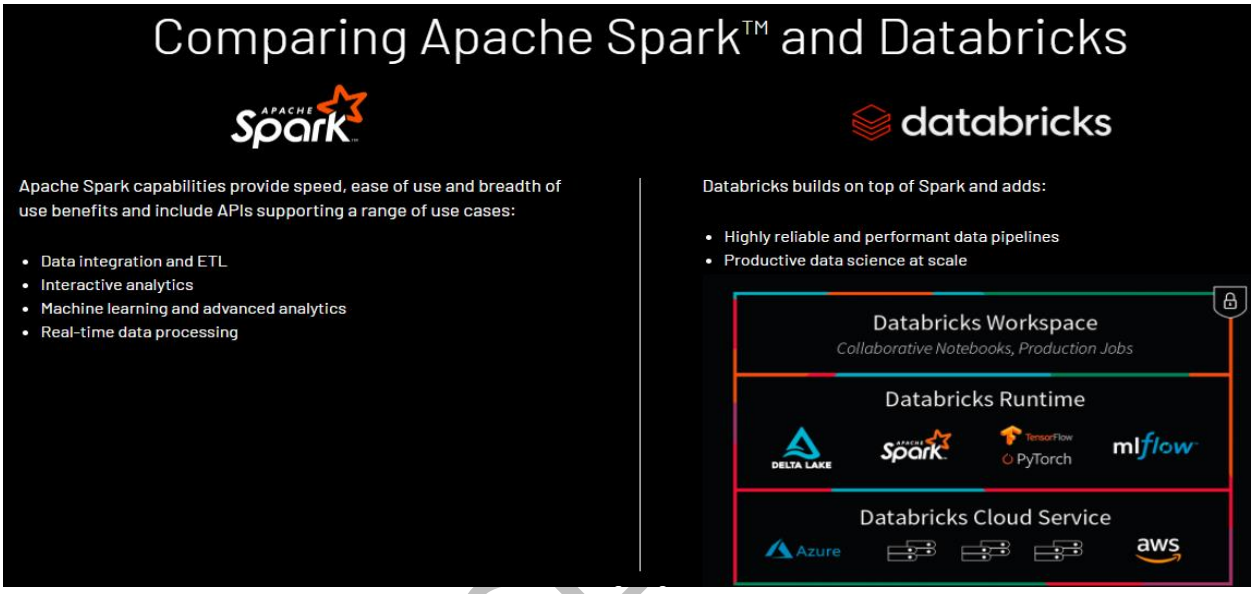
There are some difficulties in Apache Spark for Developers:



- Infrastructure Management overhead on IT Team.
- Upgrading of newer version of Spark engine is difficult.
- User interface is not that much good.
- Manual configurations
- Integrating other tools is difficult
- Collaboration on projects is difficult

Data Bricks overcoming all difficulties in Apache Spark by containing.

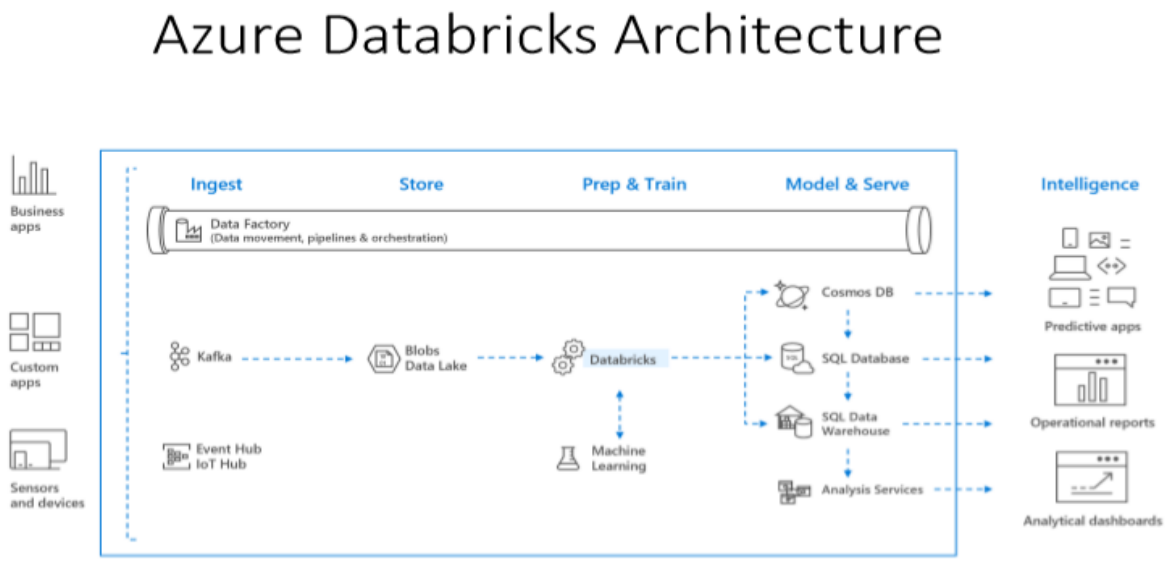
1. Efficient and Interactive Platform.
2. Tools are available
3. Integrated and Interactive Workspace
4. User Interface to Manage Infrastructure Scalability, Failure Recovery and Upgrades).





Why Azure Data bricks?

1. 50x performance for Apache Sparks workloads.
   Deploy auto-scaling compute clusters with highly-optimized Spark that perform up to 50x faster
2. Millions of server hours each day:
   Azure Data bricks is trusted by thousands of customers who run millions of server hours each day across more than 30 Azure regions.
3. Ease of use
   Start with a single click in the Azure Portal, natively integrate with Azure security and data services, and boost productivity by up to 25% with collaborative data engineering and data science



Azure Data bricks Concepts:

Workspace:

The workspace is an environment for accessing all of your Azure Data bricks assets. The workspace organizes objects (notebooks, libraries, dashboards, and experiments) into folders and provides access to data objects and computational resources.

Notebook: A web-based interface to documents that contain runnable commands, visualizations, and narrative text in Different languages Scala, SQL, Python and R.

Cluster: A set of computation resources and configurations on which you run notebooks and jobs. There are two types of clusters:

Interactive: You create an interactive cluster using the UI, CLI, or REST API. You can manually terminate and restart an interactive cluster. Multiple users can share such clusters to do collaborative interactive analysis.

Automated: The Azure Data bricks job scheduler creates an automated cluster when you run a job on a new automated cluster and terminates the cluster when the job is complete. You cannot restart an automated cluster.

Pool: Reduce cluster start and auto-scaling times. When attached to a pool, a cluster allocates its driver and worker nodes from the pool. If the pool does not have sufficient idle resources to accommodate the cluster's request, the pool expands by allocating new instances from the instance provider. When an attached cluster is terminated, the instances it used are returned to the pool and can be reused by a different cluster.

Worker Node: This worker node which runs the application code in the cluster. Choose number of worker nodes.

Driver Node: This is receiving task and distributing task to multiple smaller tasks and coordinating tasks. This is Responsible for converting a user application to smaller execution units called tasks and then schedules them to run with a cluster manager on executors.

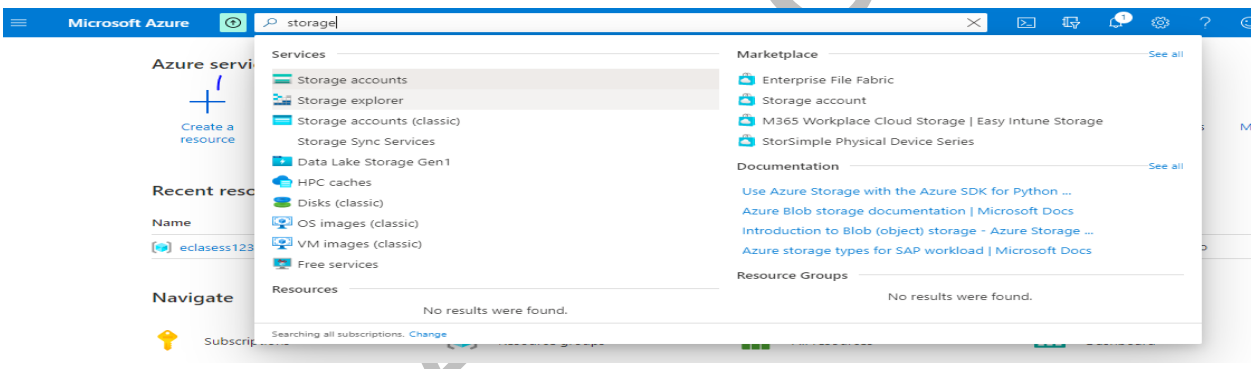
How to Connect to DatalakeGen2 from Data Bricks and performing Data Analysis.

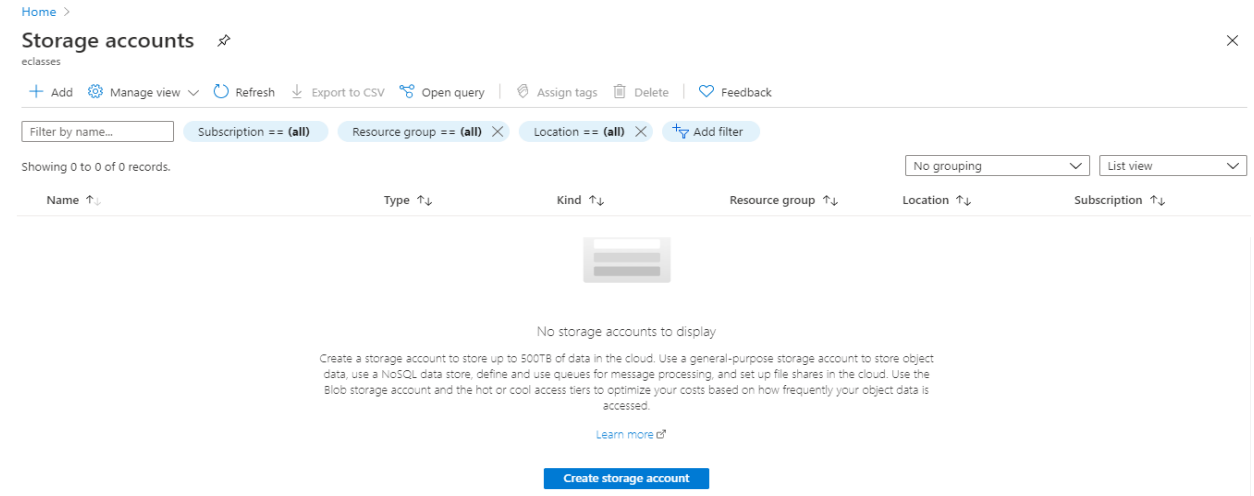
Step-1

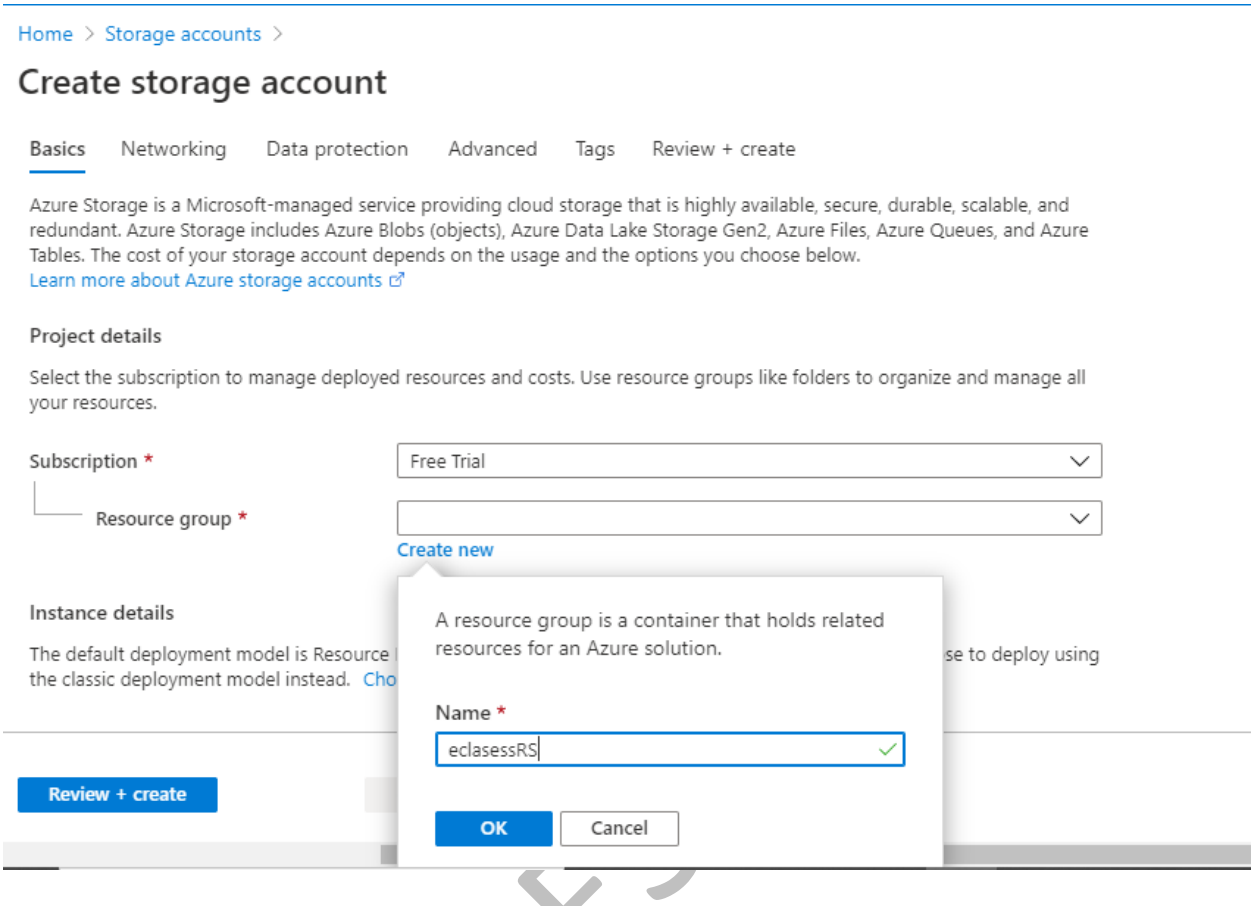
Sign in to the Azure portal.

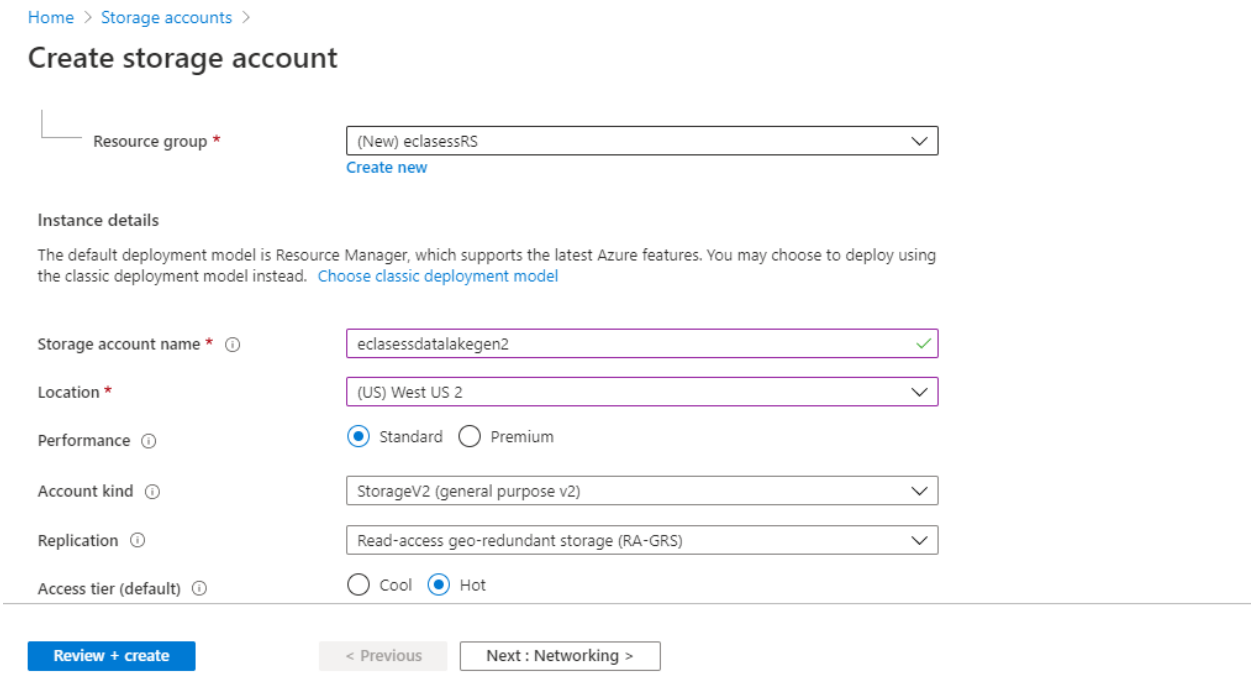
Step-2

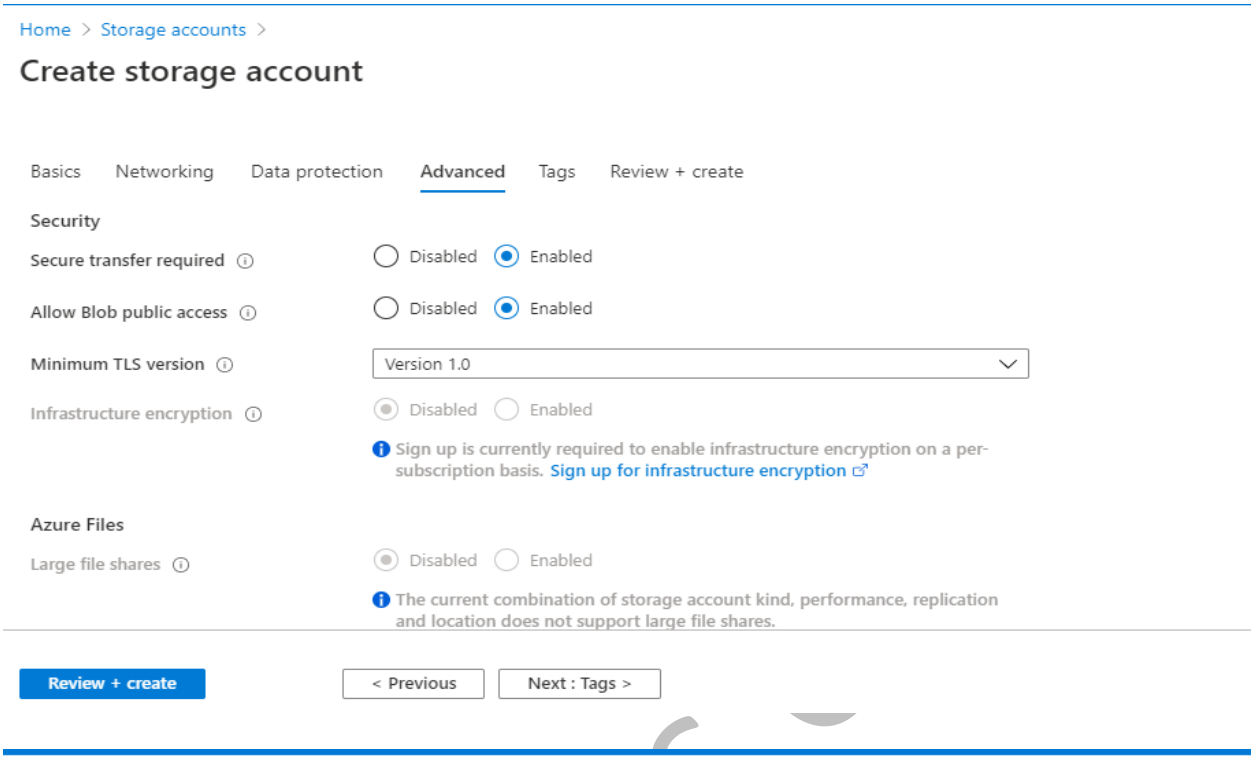
Create Azure Data Lake Gen2 Service:

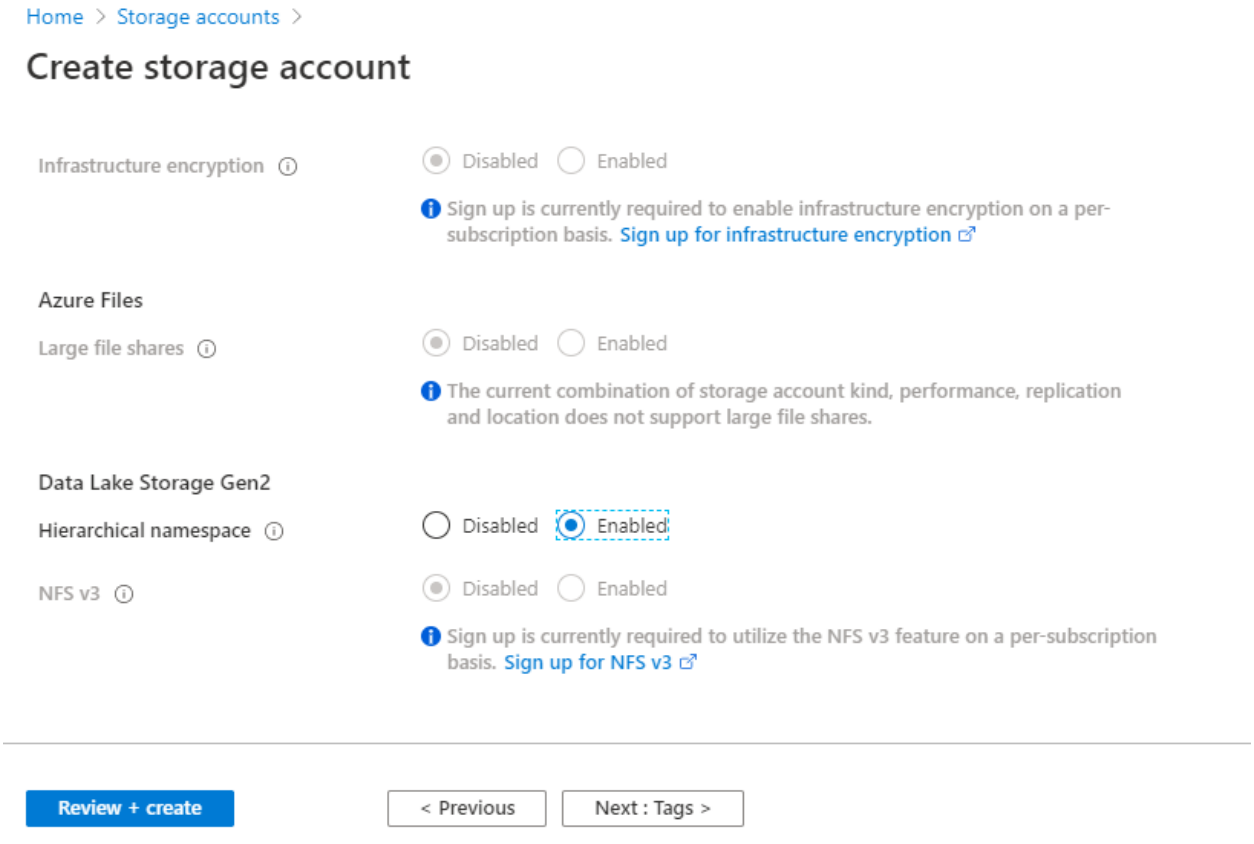


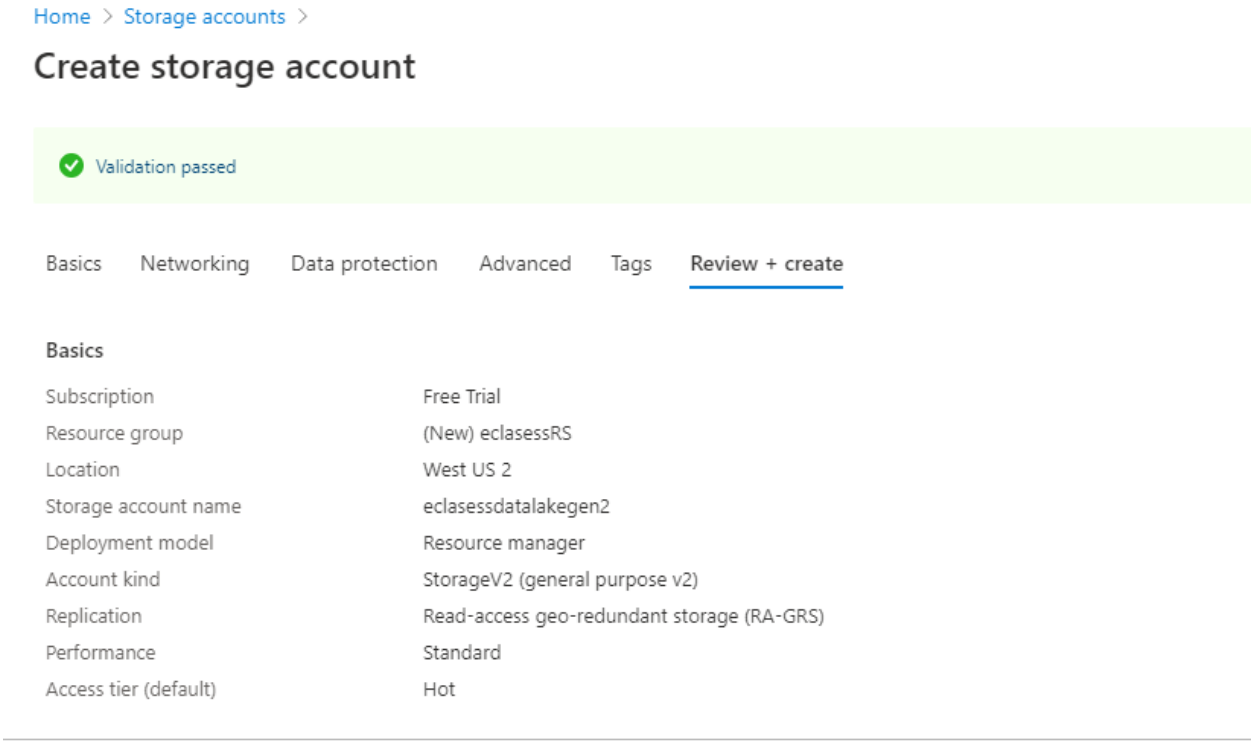


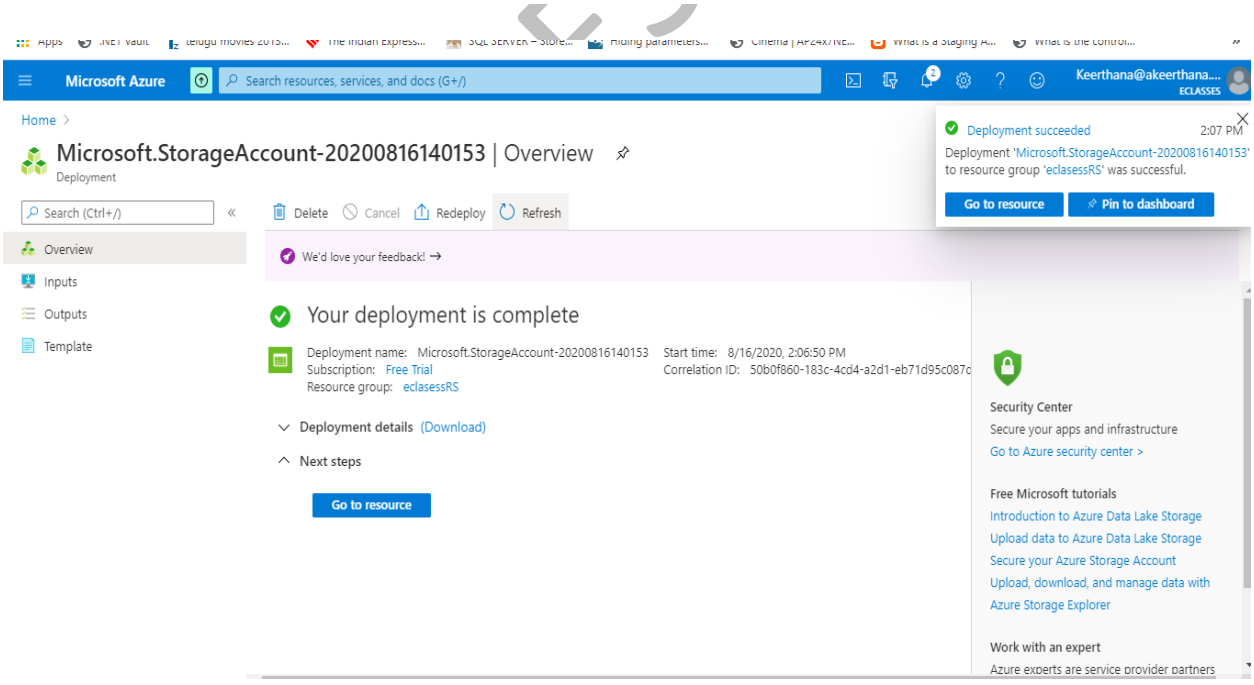

Home > Storage accounts >
Create storage account
Basics
Networking
Data protection
Advanced
Tags
Review + create
Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below.
Learn more about Azure storage accounts
Project details
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.
Subscription *
Free Trial
Resource group *
Create new
Instance details
The default deployment model is Resource
the classic deployment model instead. Cho
se to deploy using
A resource group is a container that holds related resources for an Azure solution.
Name *
eclasessRS
OK
Cancel
Review + create

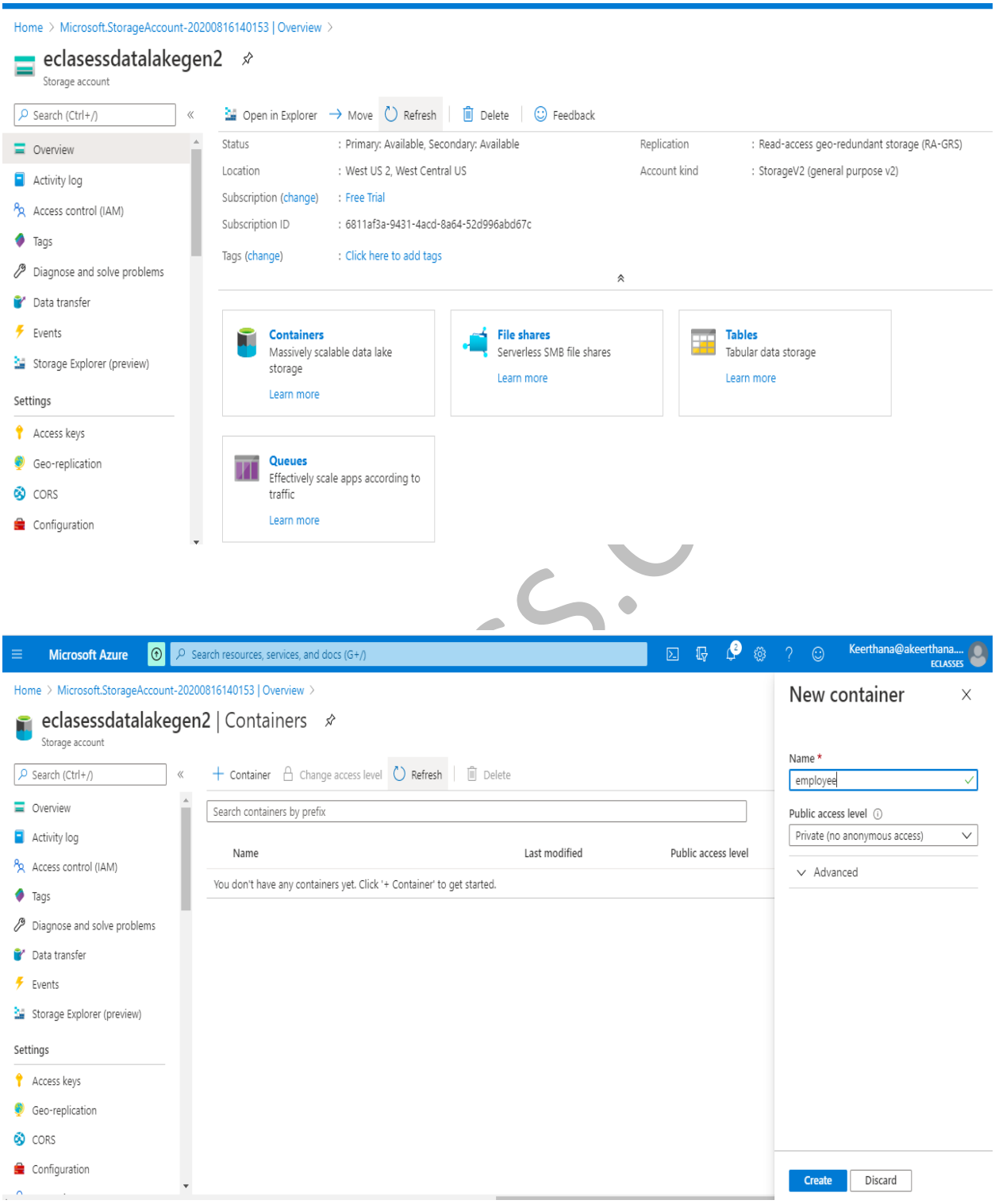

Home > Storage accounts >
Create storage account
Resource group *
(New) eclasessRS
Create new
Instance details
The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. Choose classic deployment model
Storage account name *
eclasessdatalakegen2
Location *
(US) West US 2
Performance
Standard
Premium
Account kind
StorageV2 (general purpose v2)
Replication
Read-access geo-redundant storage (RA-GRS)
Access tier (default)
Cool
Hot
Review + create
< Previous
Next : Networking >

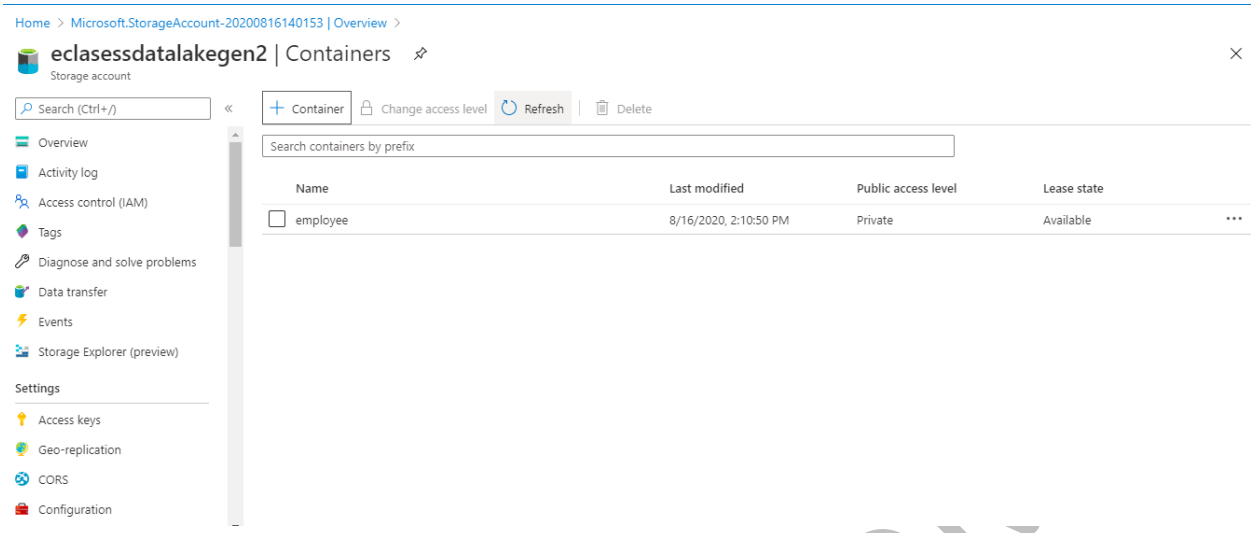

Home > Storage accounts >
Create storage account
Basics Networking Data protection Advanced Tags Review + create
Security
Secure transfer required
Disabled Enabled
Allow Blob public access
Disabled Enabled
Minimum TLS version
Version 1.0
Infrastructure encryption
Disabled Enabled
Sign up is currently required to enable infrastructure encryption on a per-subscription basis. Sign up for infrastructure encryption
Azure Files
Large file shares
Disabled Enabled
The current combination of storage account kind, performance, replication and location does not support large file shares.
Review + create
< Previous
Next : Tags >

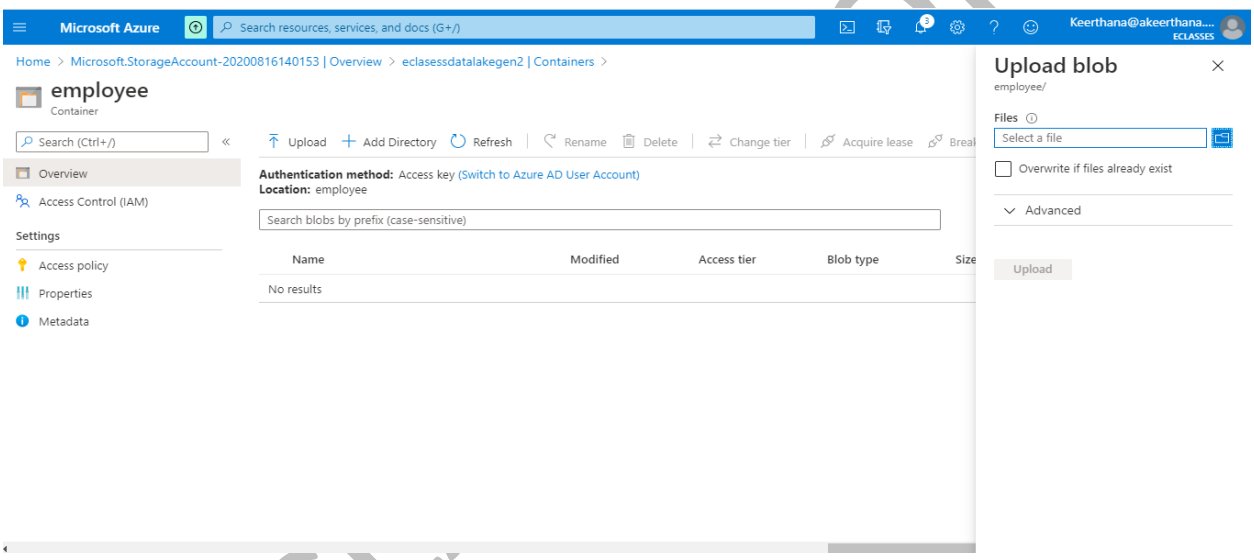

Home > Storage accounts >
Create storage account
Infrastructure encryption
Disabled Enabled
Sign up is currently required to enable infrastructure encryption on a per-subscription basis. Sign up for infrastructure encryption
Azure Files
Large file shares
Disabled Enabled
The current combination of storage account kind, performance, replication and location does not support large file shares.
Data Lake Storage Gen2
Hierarchical namespace
Disabled Enabled
NFS v3
Disabled Enabled
Sign up is currently required to utilize the NFS v3 feature on a per-subscription basis. Sign up for NFS v3
Review + create
< Previous
Next : Tags >

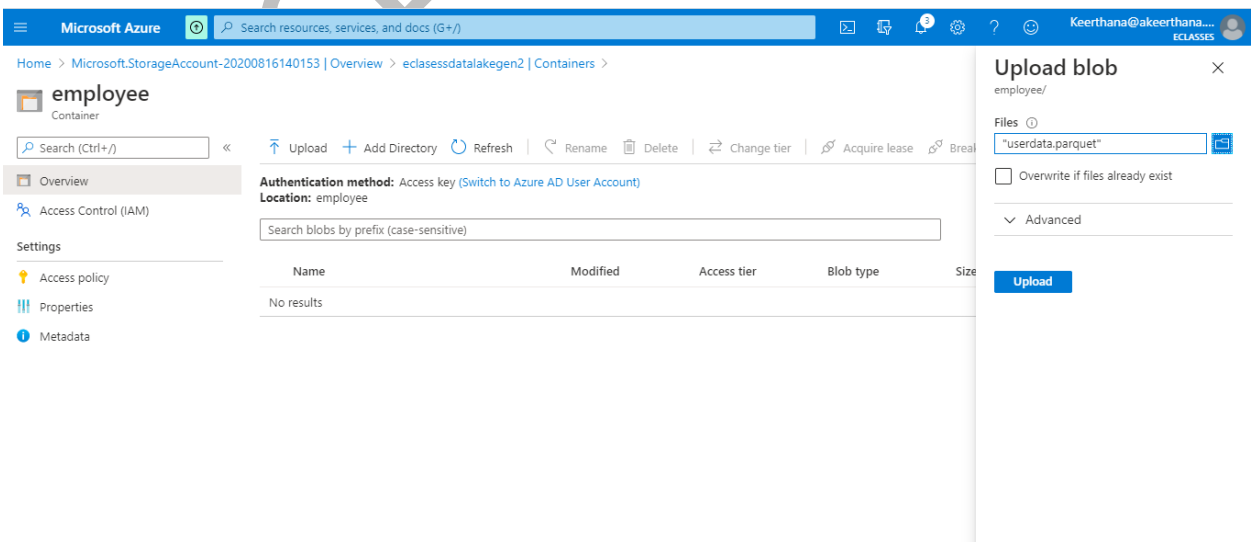

Home > Storage accounts >
Create storage account
Validation passed
Basics Networking Data protection Advanced Tags Review + create
Basics
Subscription Free Trial
Resource group (New) eclasessRS
Location West US 2
Storage account name eclasessdatalakegen2
Deployment model Resource manager
Account kind StorageV2 (general purpose v2)
Replication Read-access geo-redundant storage (RA-GRS)
Performance Standard
Access tier (default) Hot
Create
< Previous
Next >
Download a template for automation

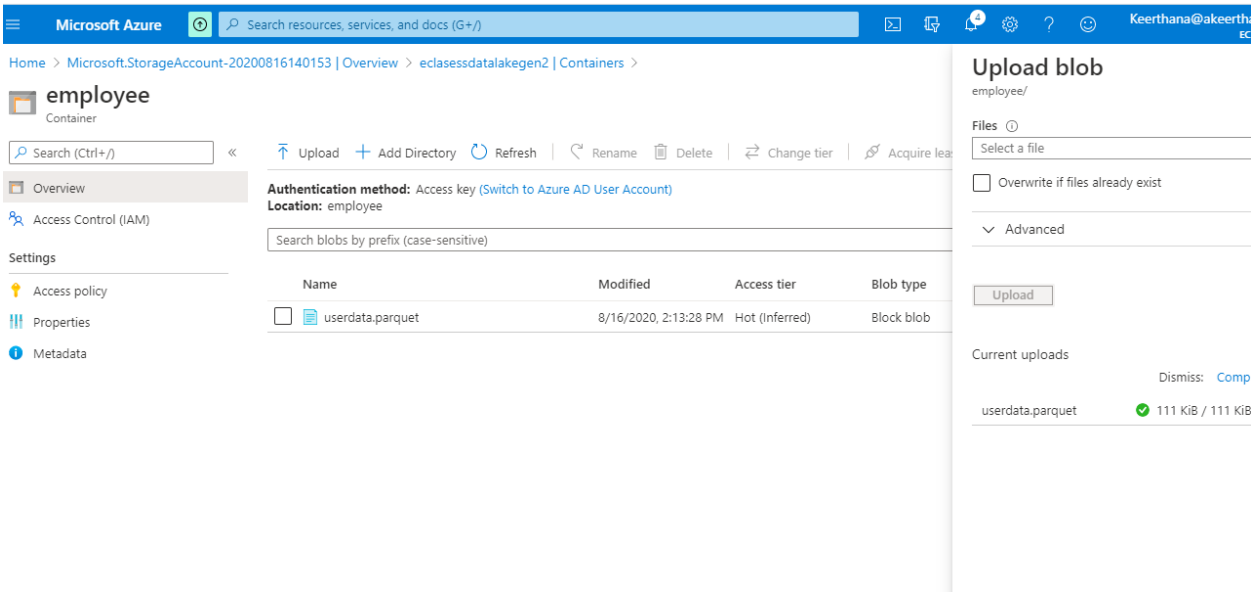

Microsoft Azure
Search resources, services, and docs (G+/)
Keerthana@akeerthana....
ECLASSES
Home
Microsoft.StorageAccount-20200816140153 | Overview
Deployment
Search (Ctrl+/)
Overview
Inputs
Outputs
Template
Delete
Cancel
Redeploy
Refresh
We'd love your feedback!
Deployment succeeded
2:07 PM
Deployment 'Microsoft.StorageAccount-20200816140153' to resource group 'eclasessRS' was successful.
Go to resource
Pin to dashboard
Your deployment is complete
Deployment name: Microsoft.StorageAccount-20200816140153
Subscription: Free Trial
Resource group: eclasessRS
Start time: 8/16/2020, 2:06:50 PM
Correlation ID: 50b0f860-183c-4cd4-a2d1-eb71d95c087c
Deployment details (Download)
Next steps
Go to resource
Security Center
Secure your apps and infrastructure
Go to Azure security center >
Free Microsoft tutorials
Introduction to Azure Data Lake Storage
Upload data to Azure Data Lake Storage
Secure your Azure Storage Account
Upload, download, and manage data with Azure Storage Explorer
Work with an expert
Azure experts are service provider partners


Home > Microsoft.StorageAccount-20200816140153 | Overview >
eclasessdatalakegen2
Storage account
Search (Ctrl+/)
Open in Explorer
Move
Refresh
Delete
Feedback
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Status : Primary: Available, Secondary: Available
Location : West US 2, West Central US
Subscription (change) : Free Trial
Subscription ID : 6811af3a-9431-4acd-8a64-52d996abd67c
Tags (change) : Click here to add tags
Replication : Read-access geo-redundant storage (RA-GRS)
Account kind : StorageV2 (general purpose v2)
Containers
Massively scalable data lake storage
Learn more
File shares
Serverless SMB file shares
Learn more
Tables
Tabular data storage
Learn more
Queues
Effectively scale apps according to traffic
Learn more
Microsoft Azure
Search resources, services, and docs (G+/)
Keerthana@akeerthana....
ECLASSES
Home > Microsoft.StorageAccount-20200816140153 | Overview >
eclasessdatalakegen2 | Containers
Storage account
Container
Change access level
Refresh
Delete
Search containers by prefix
Name
Last modified
Public access level
You don't have any containers yet. Click '+ Container' to get started.
New container
Name *
employee
Public access level
Private (no anonymous access)
Advanced
Create
Discard

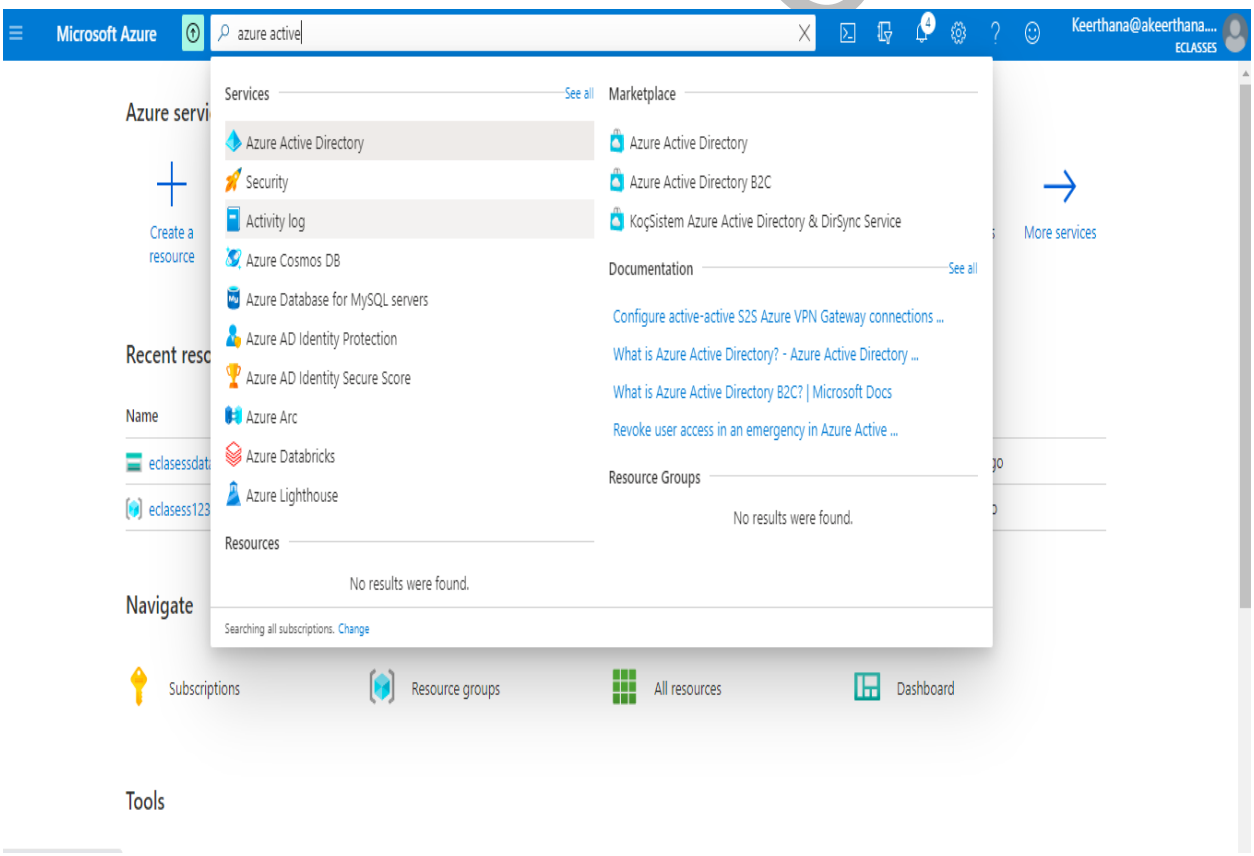

Home > Microsoft.StorageAccount-20200816140153 | Overview >
eclasessdatalakegen2 | Containers
Storage account
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
+ Container
Change access level
Refresh
Delete
Search containers by prefix
Name
Last modified
Public access level
Lease state
employee
8/16/2020, 2:10:50 PM
Private
Available

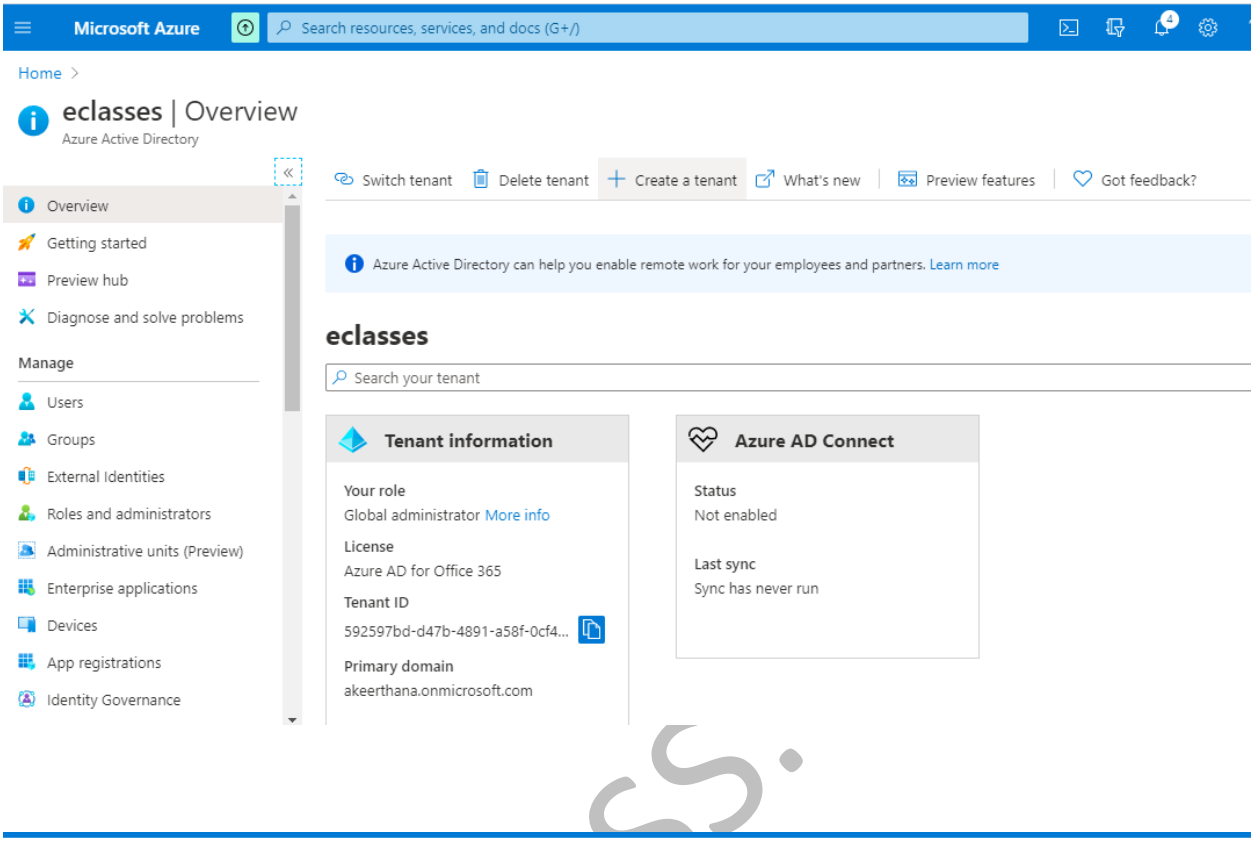

Microsoft Azure
Search resources, services, and docs (G+/)
Keerthana@akeerthana....
ECLASSES
Home > Microsoft.StorageAccount-20200816140153 | Overview > eclasessdatalakegen2 | Containers >
employee
Container
Search (Ctrl+/)
Overview
Access Control (IAM)
Settings
Access policy
Properties
Metadata
Upload
Add Directory
Refresh
Rename
Delete
Change tier
Acquire lease
Authentication method: Access key (Switch to Azure AD User Account)
Location: employee
Search blobs by prefix (case-sensitive)
Name
Modified
Access tier
Blob type
No results
Upload blob
employee/
Files
Select a file
Overwrite if files already exist
Advanced
Upload

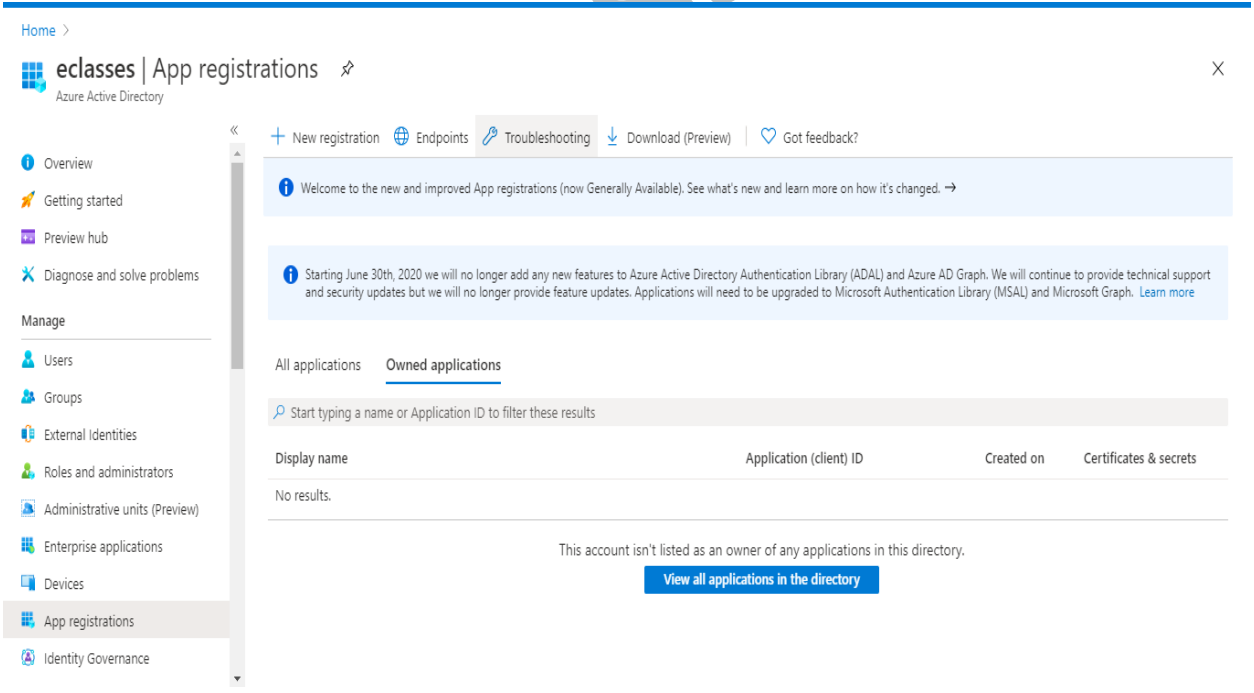

Microsoft Azure
Search resources, services, and docs (G+/)
Keerthana@akeerthana....
ECLASSES
Home > Microsoft.StorageAccount-20200816140153 | Overview > eclasessdatalakegen2 | Containers >
employee
Container
Search (Ctrl+/)
Overview
Access Control (IAM)
Settings
Access policy
Properties
Metadata
Upload
Add Directory
Refresh
Rename
Delete
Change tier
Acquire lease
Authentication method: Access key (Switch to Azure AD User Account)
Location: employee
Search blobs by prefix (case-sensitive)
Name
Modified
Access tier
Blob type
No results
Upload blob
employee/
Files
"userdata.parquet"
Overwrite if files already exist
Advanced
Upload

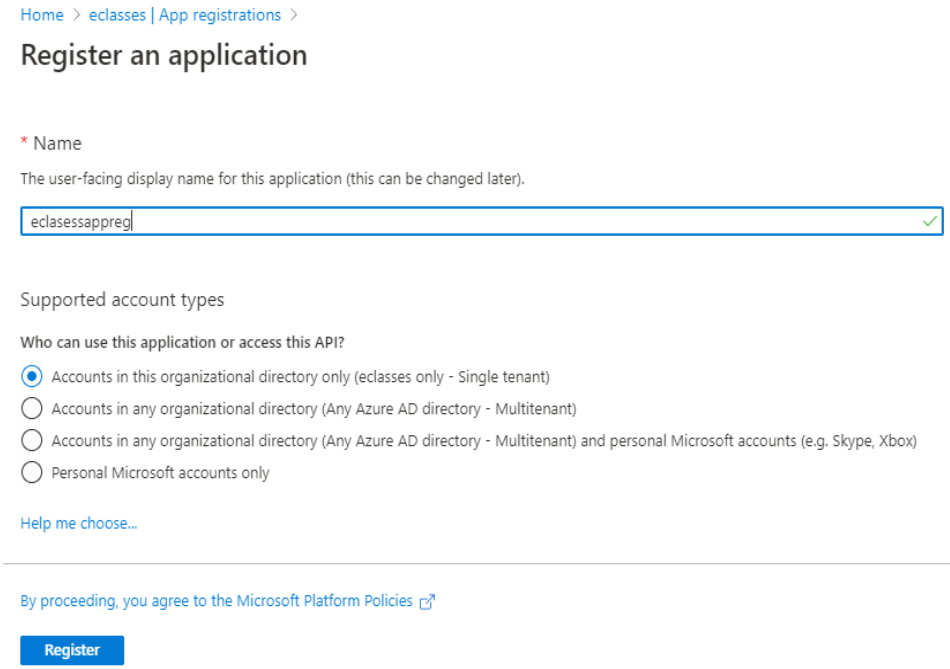


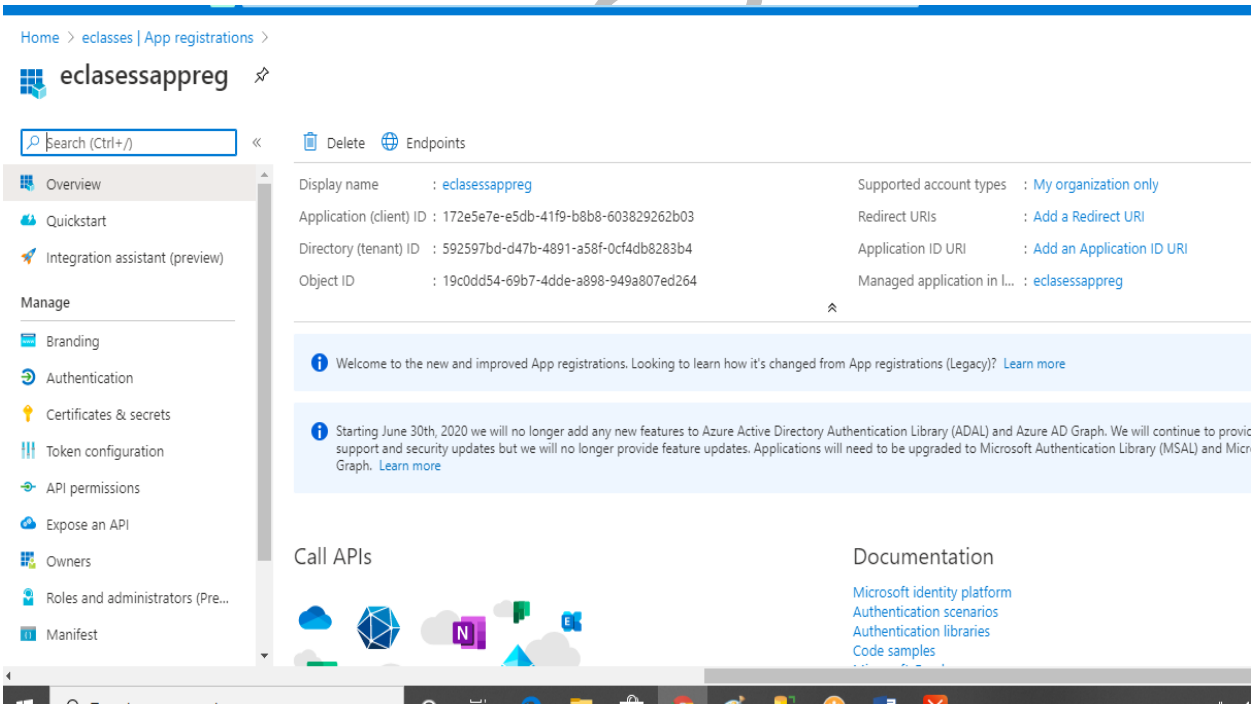
Step-3: Create Azure Active Directory App Registration to Authenticate Services Using Service Principle.




Microsoft Azure
Search resources, services, and docs (G+/)
Home
eclasses | Overview
Azure Active Directory
Overview
Getting started
Preview hub
Diagnose and solve problems
Manage
Users
Groups
External Identities
Roles and administrators
Administrative units (Preview)
Enterprise applications
Devices
App registrations
Identity Governance
Switch tenant
Delete tenant
Create a tenant
What's new
Preview features
Got feedback?
Azure Active Directory can help you enable remote work for your employees and partners. Learn more
eclasses
Search your tenant
Tenant information
Your role
Global administrator More info
License
Azure AD for Office 365
Tenant ID
592597bd-d47b-4891-a58f-0cf4...
Primary domain
akeerthana.onmicrosoft.com
Azure AD Connect
Status
Not enabled
Last sync
Sync has never run


Home
eclasses | App registrations
Azure Active Directory
Overview
Getting started
Preview hub
Diagnose and solve problems
Manage
Users
Groups
External Identities
Roles and administrators
Administrative units (Preview)
Enterprise applications
Devices
App registrations
Identity Governance
New registration
Endpoints
Troubleshooting
Download (Preview)
Got feedback?
Welcome to the new and improved App registrations (now Generally Available). See what's new and learn more on how it's changed.
Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. Learn more
All applications
Owned applications
Start typing a name or Application ID to filter these results
Display name
Application (client) ID
Created on
Certificates & secrets
No results.
This account isn't listed as an owner of any applications in this directory.
View all applications in the directory

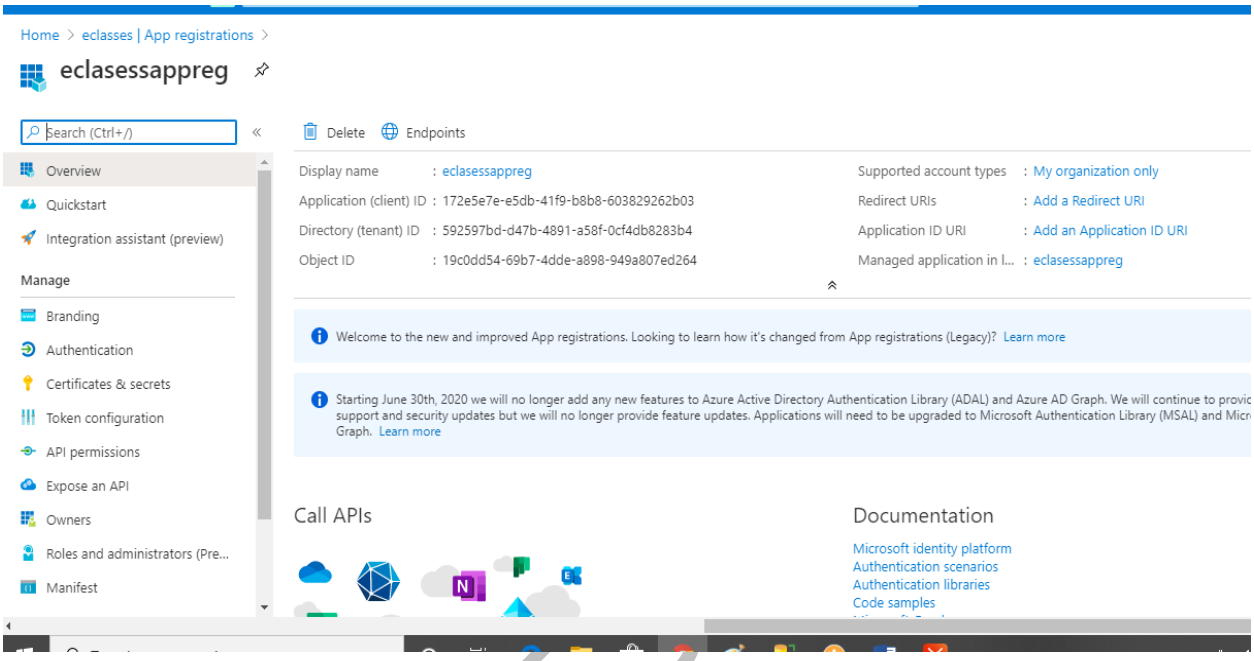

Home > eclasses | App registrations >
Register an application
* Name
The user-facing display name for this application (this can be changed later).
eclasessappreg
Supported account types
Who can use this application or access this API?
Accounts in this organizational directory only (eclasses only - Single tenant)
Accounts in any organizational directory (Any Azure AD directory - Multitenant)
Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
Personal Microsoft accounts only
Help me choose...
By proceeding, you agree to the Microsoft Platform Policies
Register

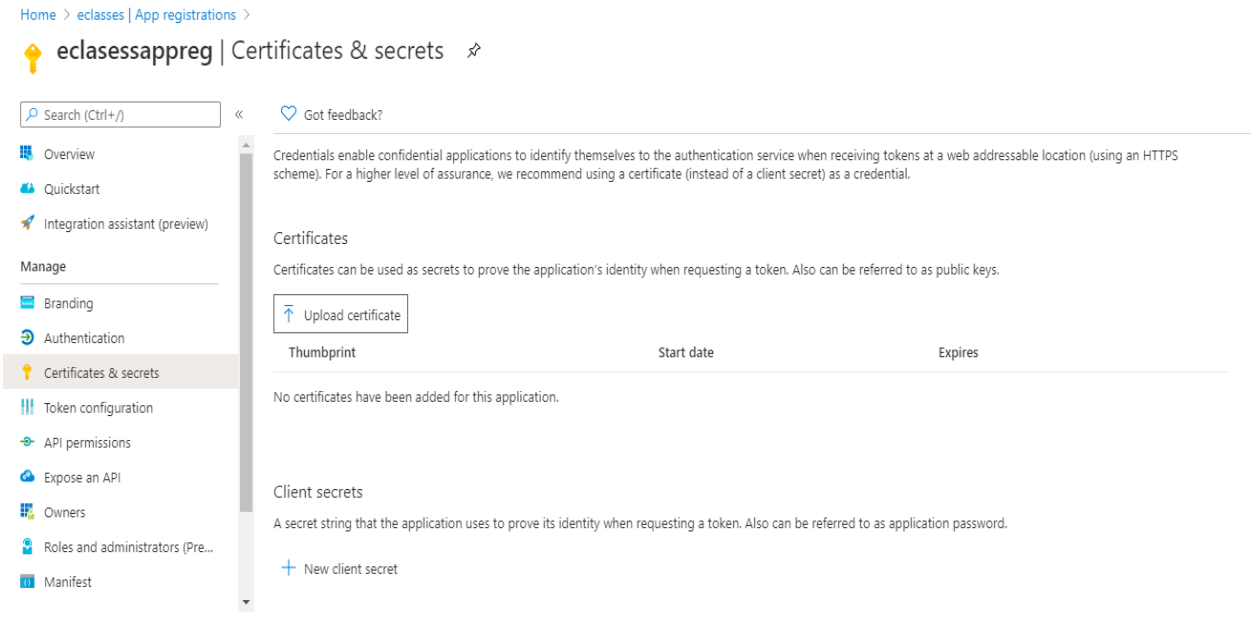

Home > eclasses | App registrations >
eclasessappreg
Search (Ctrl+/)
Delete
Endpoints
Overview
Quickstart
Integration assistant (preview)
Manage
Branding
Authentication
Certificates & secrets
Token configuration
API permissions
Expose an API
Owners
Roles and administrators (Pre...
Manifest
Display name : eclasessappreg
Application (client) ID : 172e5e7e-e5db-41f9-b8b8-603829262b03
Directory (tenant) ID : 592597bd-d47b-4891-a58f-0cf4db8283b4
Object ID : 19c0dd54-69b7-4dde-a898-949a807ed264
Supported account types : My organization only
Redirect URIs : Add a Redirect URI
Application ID URI : Add an Application ID URI
Managed application in l... : eclasessappreg
Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? Learn more
Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provid
support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Micr
Graph. Learn more
Call APIs
Documentation
Microsoft identity platform
Authentication scenarios
Authentication libraries
Code samples

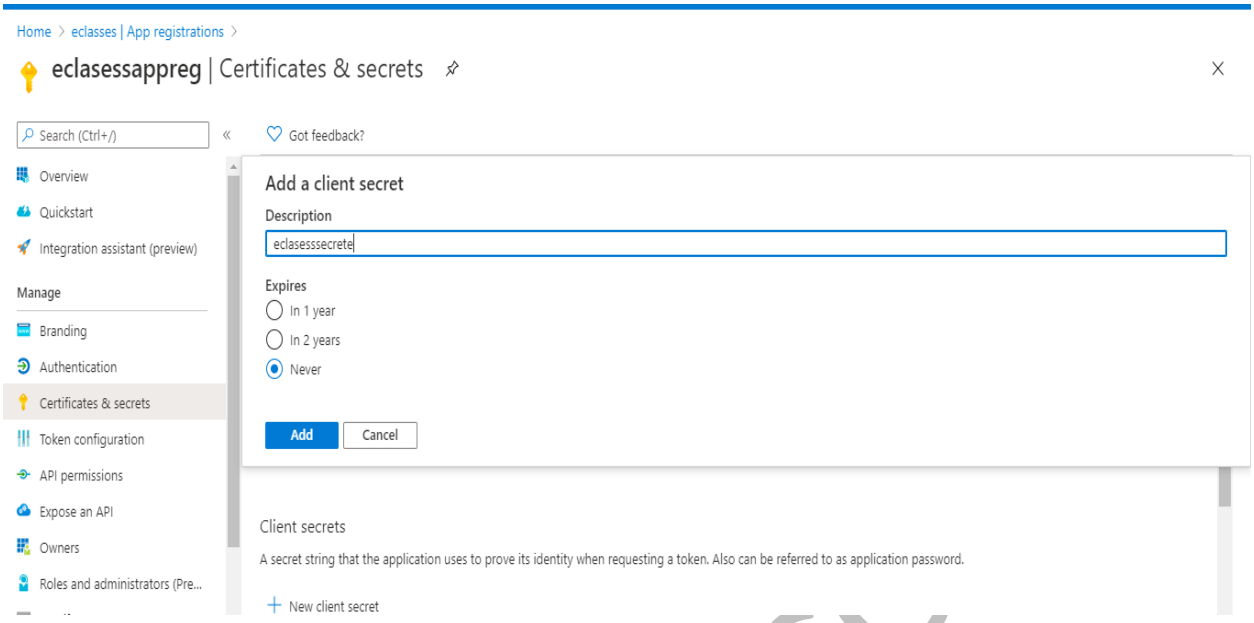
Application(client)ID:172e5e7e-e5db-41f9-b8b8-603829262b03

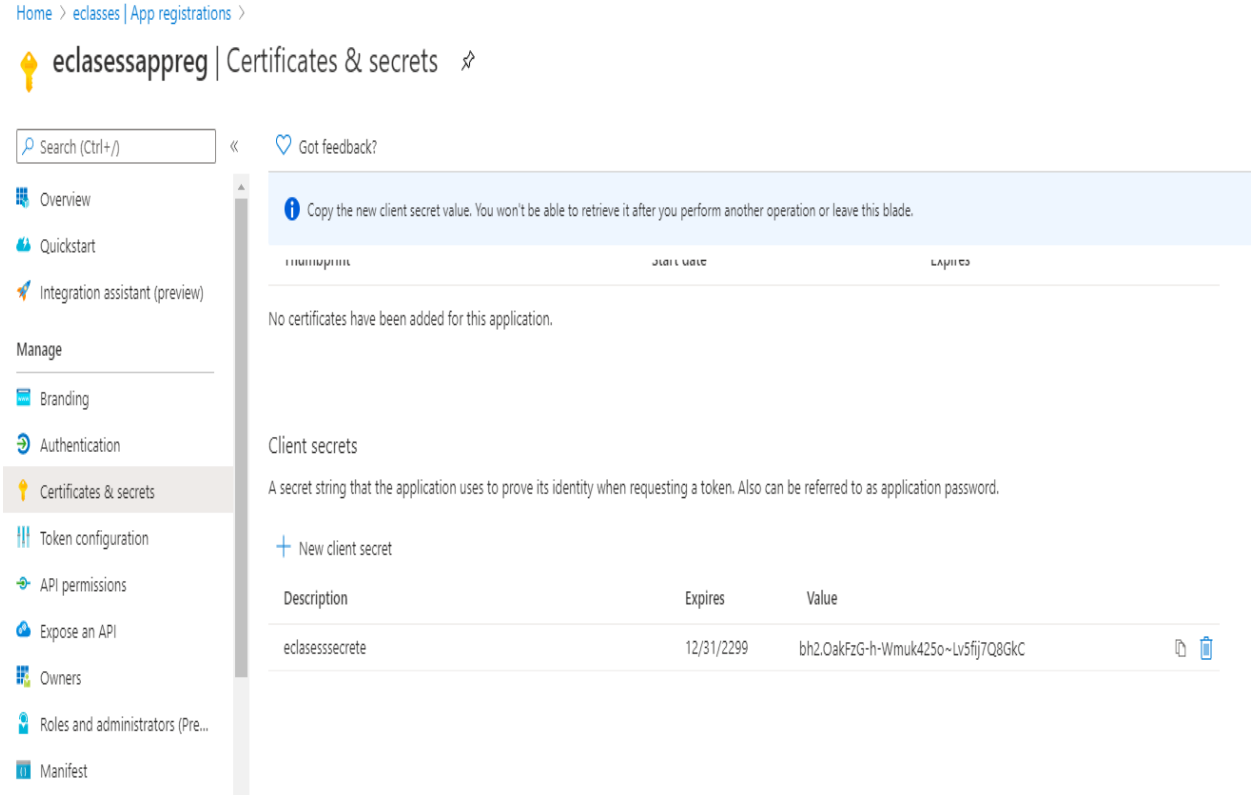
Directory (tenant) ID:592597bd-d47b-4891-a58f-0cf4db8283b4

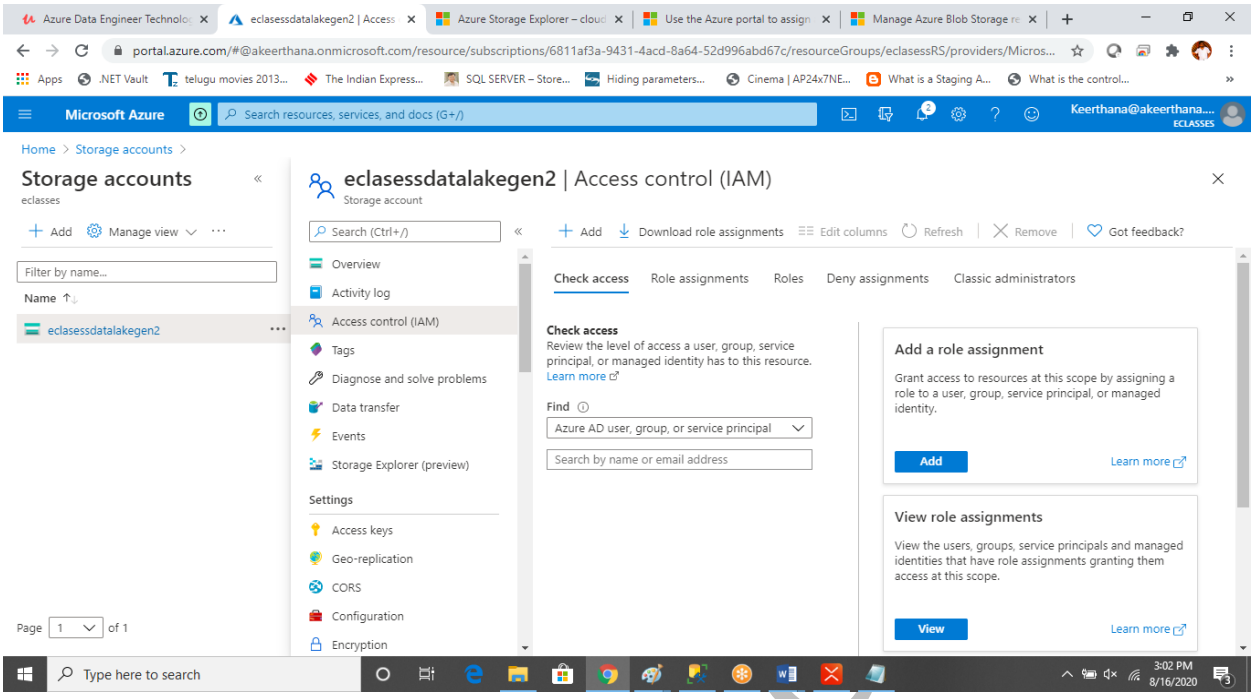
Secrete key: bh2.OakFzG-h-Wmuk425o~Lv5fij7Q8GkC

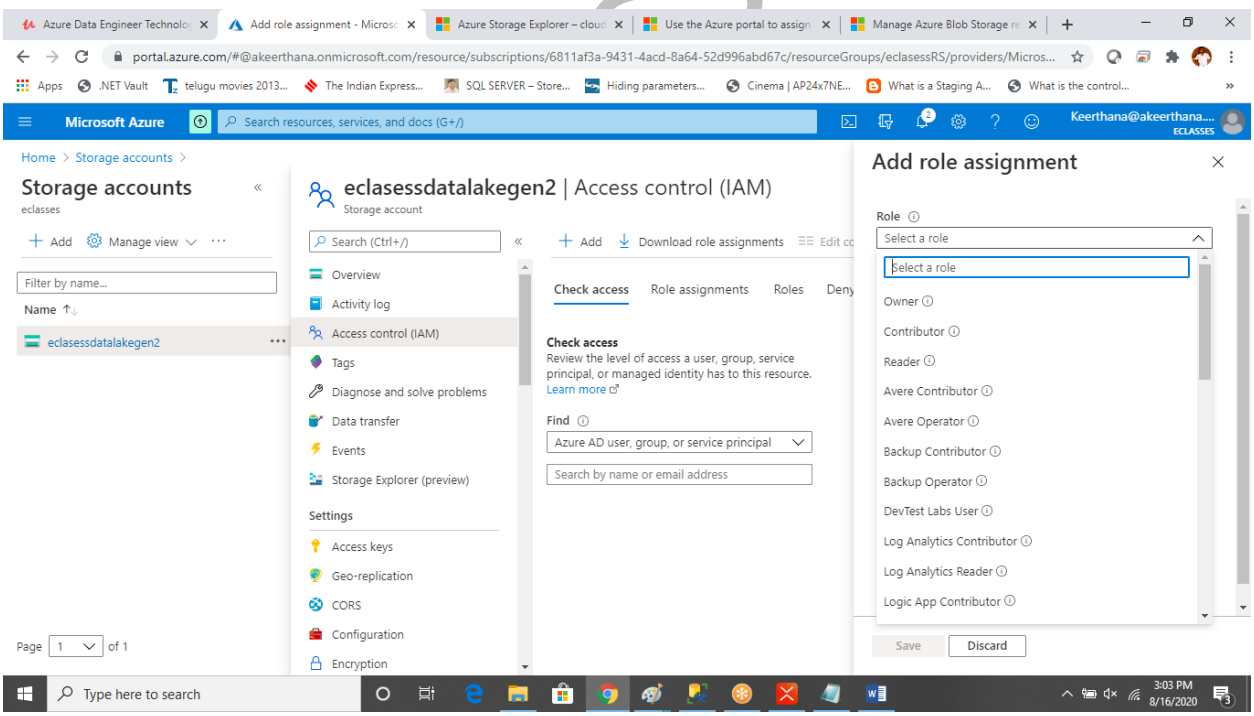


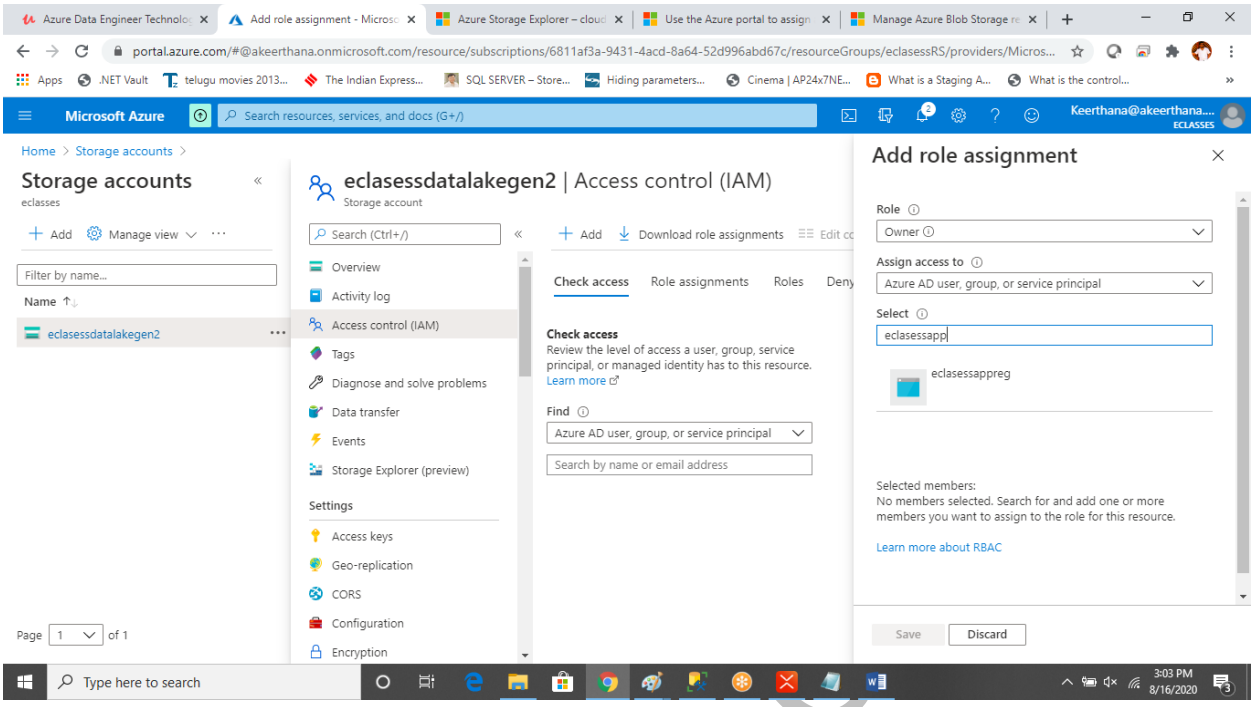


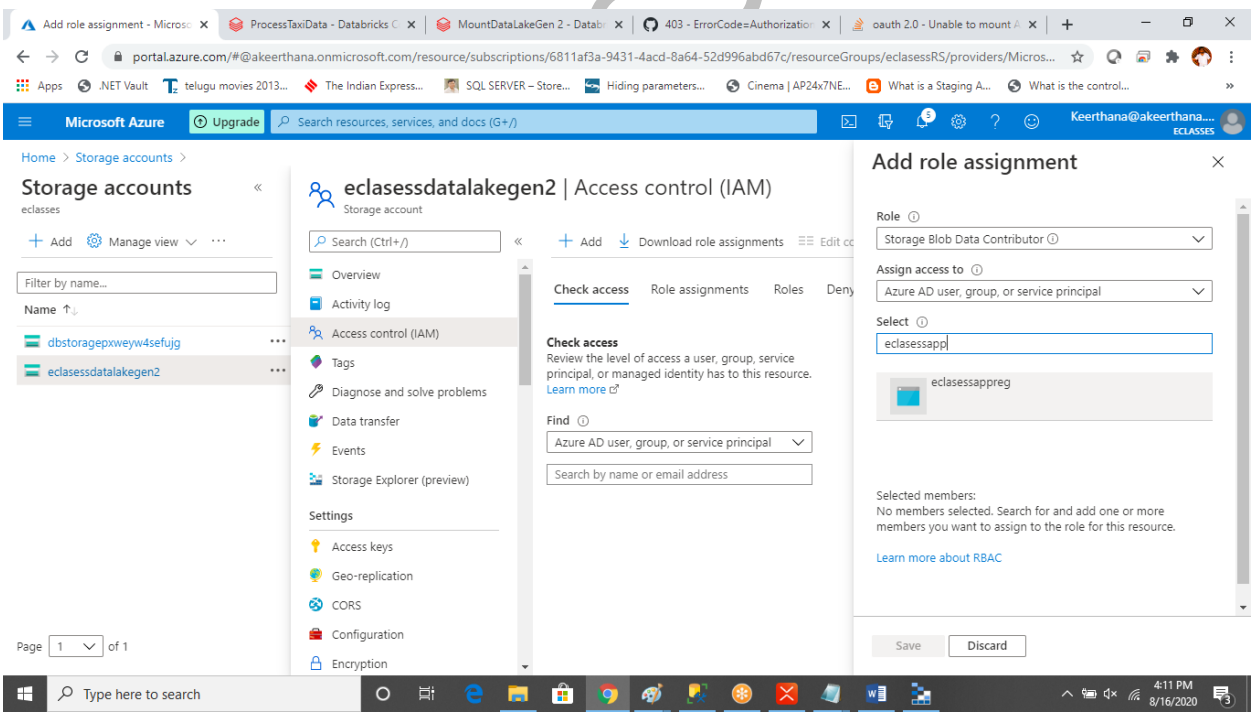

Home > eclasses | App registrations >
eclasessappreg | Certificates & secrets
Search (Ctrl+/)
Got feedback?
Overview
Quickstart
Integration assistant (preview)
Manage
Branding
Authentication
Certificates & secrets
Token configuration
API permissions
Expose an API
Owners
Roles and administrators (Pre...
Add a client secret
Description
eclasesssecrete
Expires
In 1 year
In 2 years
Never
Add
Cancel
Client secrets
A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.
New client secret

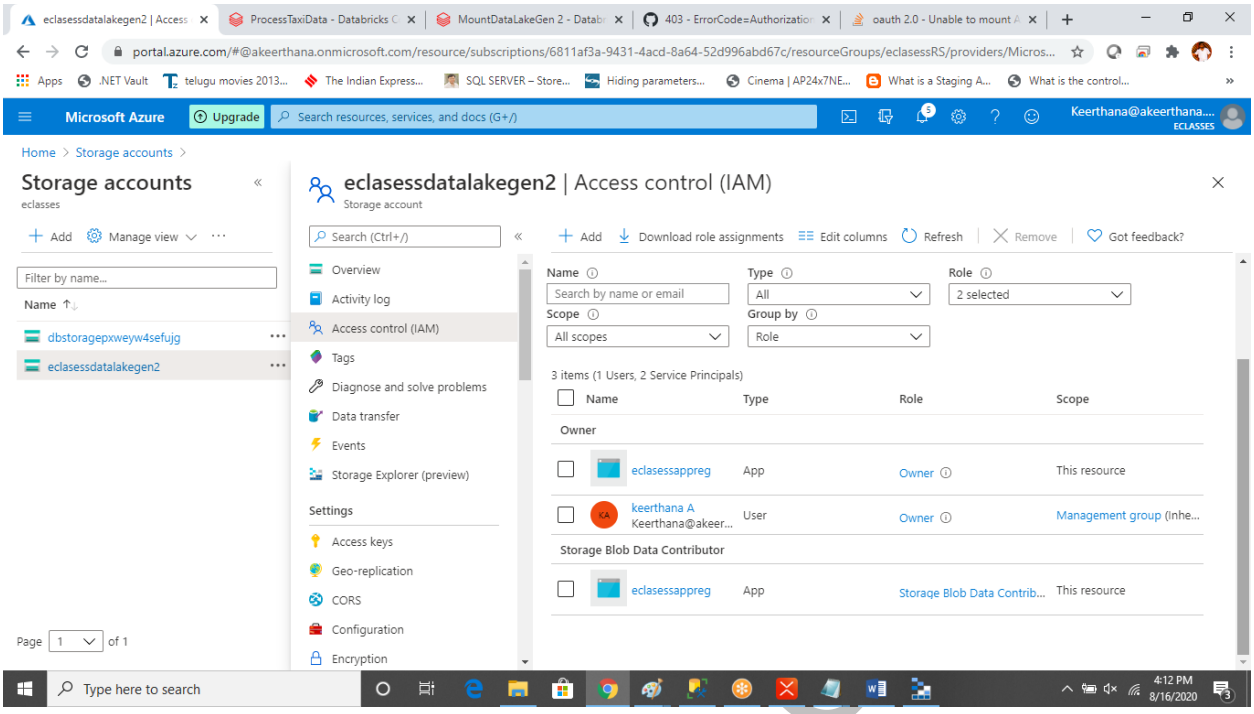

Home > eclasses | App registrations >
eclasessappreg | Certificates & secrets
Search (Ctrl+/)
Got feedback?
Overview
Quickstart
Integration assistant (preview)
Manage
Branding
Authentication
Certificates & secrets
Token configuration
API permissions
Expose an API
Owners
Roles and administrators (Pre...
Manifest
Copy the new client secret value. You won't be able to retrieve it after you perform another operation or leave this blade.
No certificates have been added for this application.
Client secrets
A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.
New client secret
Description
Expires
Value
eclasesssecrete
12/31/2299
bh2.OakFzG-h-Wmuk425o~Lv5fij7Q8GkC


Microsoft Azure
Search resources, services, and docs (G+/)
Home > Storage accounts
Storage accounts
eclasses
Add
Manage view
Filter by name...
Name
eclassesdatalakegen2
Page 1 of 1
eclassesdatalakegen2 | Access control (IAM)
Storage account
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Encryption
Add
Download role assignments
Edit columns
Refresh
Remove
Got feedback?
Check access
Role assignments
Roles
Deny assignments
Classic administrators
Check access
Review the level of access a user, group, service principal, or managed identity has to this resource.
Learn more
Find
Azure AD user, group, or service principal
Search by name or email address
Add a role assignment
Grant access to resources at this scope by assigning a role to a user, group, service principal, or managed identity.
Add
Learn more
View role assignments
View the users, groups, service principals and managed identities that have role assignments granting them access at this scope.
View
Learn more
Type here to search
3:02 PM
8/16/2020

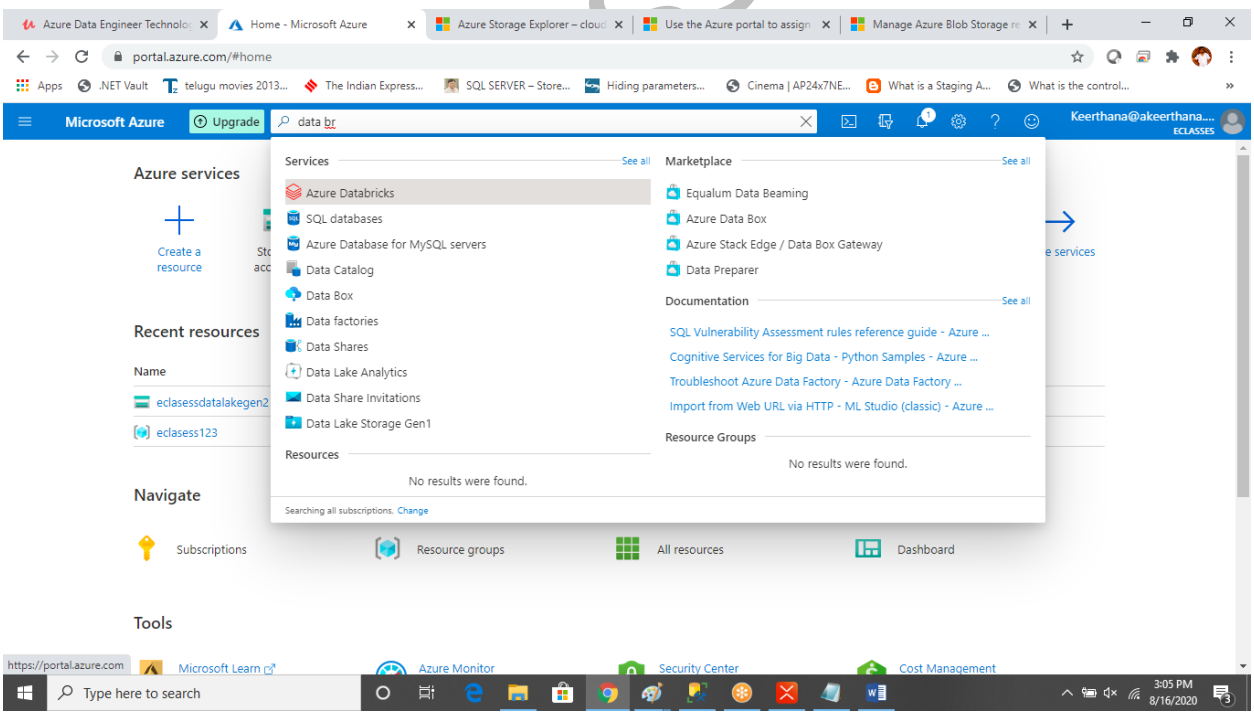

Microsoft Azure
Search resources, services, and docs (G+/)
Home > Storage accounts
Storage accounts
eclasses
Add
Manage view
Filter by name...
Name
eclassesdatalakegen2
Page 1 of 1
eclassesdatalakegen2 | Access control (IAM)
Storage account
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Encryption
Add
Download role assignments
Check access
Role assignments
Roles
Check access
Review the level of access a user, group, service principal, or managed identity has to this resource.
Learn more
Find
Azure AD user, group, or service principal
Search by name or email address
Add role assignment
Role
Select a role
Select a role
Owner
Contributor
Reader
Avere Contributor
Avere Operator
Backup Contributor
Backup Operator
DevTest Labs User
Log Analytics Contributor
Log Analytics Reader
Logic App Contributor
Save
Discard
Type here to search
3:03 PM
8/16/2020

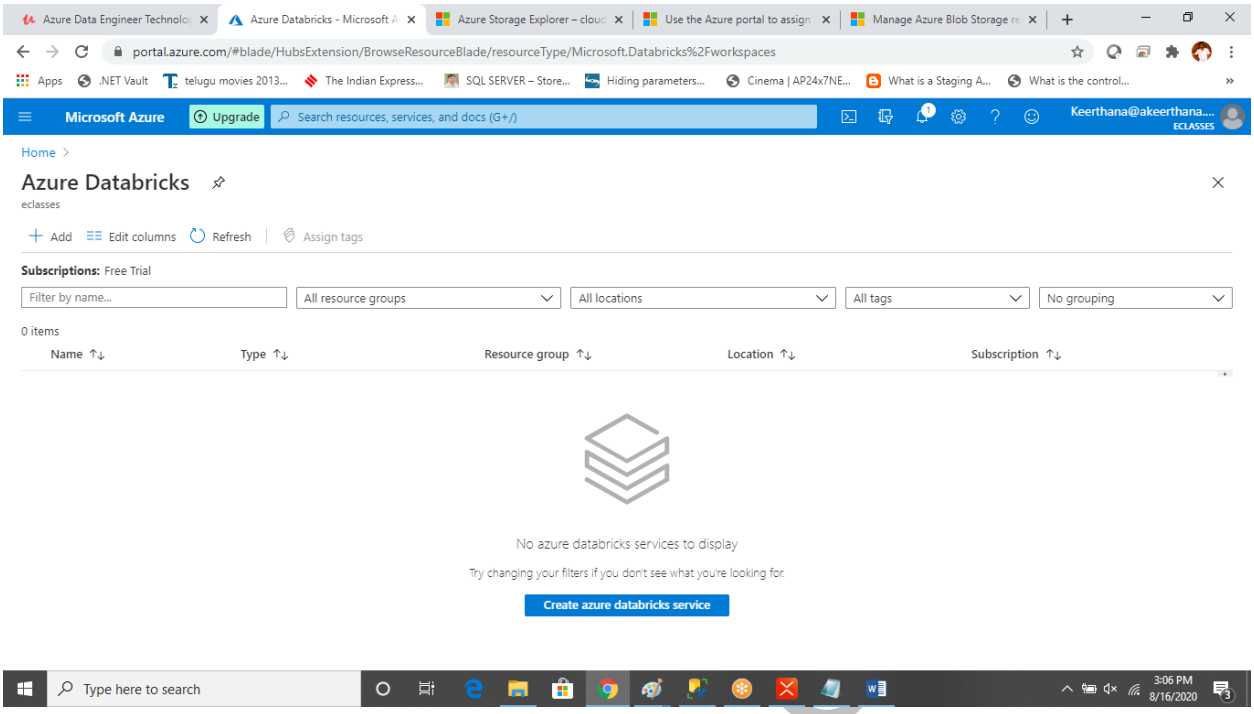

Microsoft Azure
Search resources, services, and docs (G+/)
Keerthana@akeerthana....
ECLASSES
Home > Storage accounts
Storage accounts
eclasses
Add
Manage view
Filter by name...
Name
eclasessdatalakegen2
eclasessdatalakegen2 | Access control (IAM)
Storage account
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Encryption
Add
Download role assignments
Check access
Role assignments
Roles
Check access
Review the level of access a user, group, service principal, or managed identity has to this resource.
Learn more
Find
Azure AD user, group, or service principal
Search by name or email address
Add role assignment
Role
Owner
Assign access to
Azure AD user, group, or service principal
Select
eclasessapp
eclasessappreg
Selected members:
No members selected. Search for and add one or more members you want to assign to the role for this resource.
Learn more about RBAC
Save
Discard
Page 1 of 1
Type here to search
3:03 PM
8/16/2020

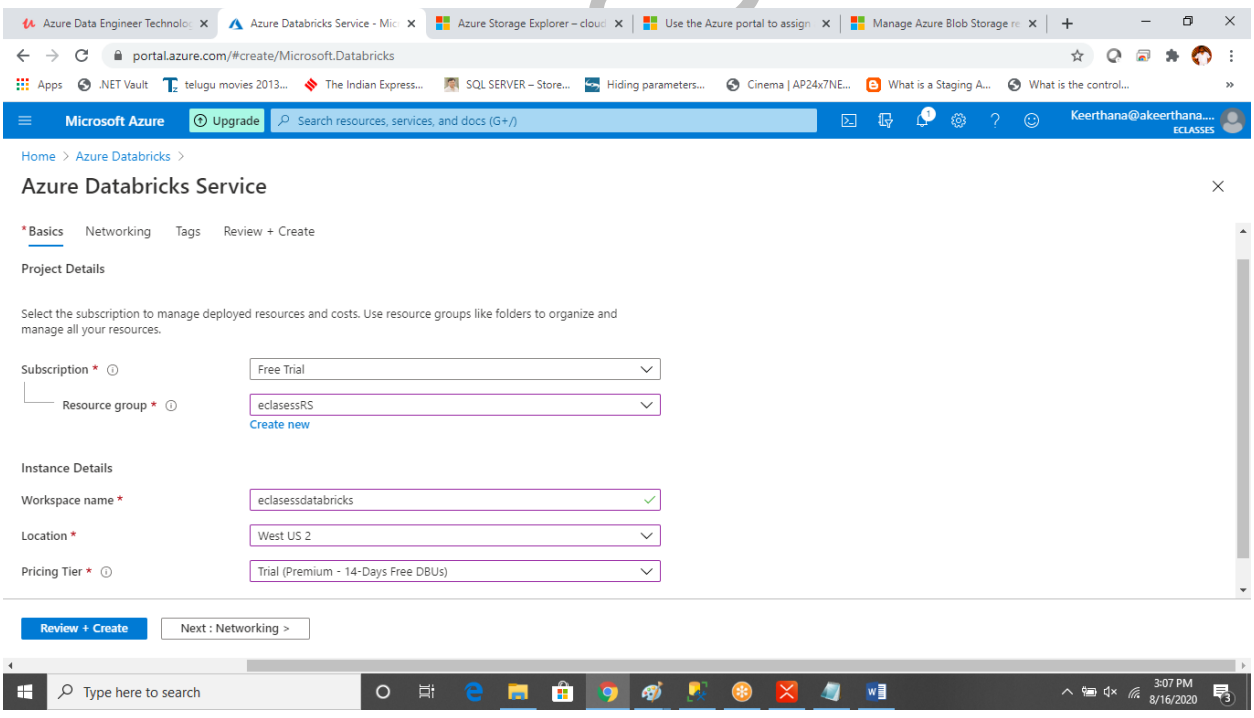

Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Keerthana@akeerthana....
ECLASSES
Home > Storage accounts
Storage accounts
eclasses
Add
Manage view
Filter by name...
Name
dbstoragepxweyw4sefujg
eclasessdatalakegen2
eclasessdatalakegen2 | Access control (IAM)
Storage account
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Encryption
Add
Download role assignments
Check access
Role assignments
Roles
Check access
Review the level of access a user, group, service principal, or managed identity has to this resource.
Learn more
Find
Azure AD user, group, or service principal
Search by name or email address
Add role assignment
Role
Storage Blob Data Contributor
Assign access to
Azure AD user, group, or service principal
Select
eclasessapp
eclasessappreg
Selected members:
No members selected. Search for and add one or more members you want to assign to the role for this resource.
Learn more about RBAC
Save
Discard
Page 1 of 1
Type here to search
4:11 PM
8/16/2020

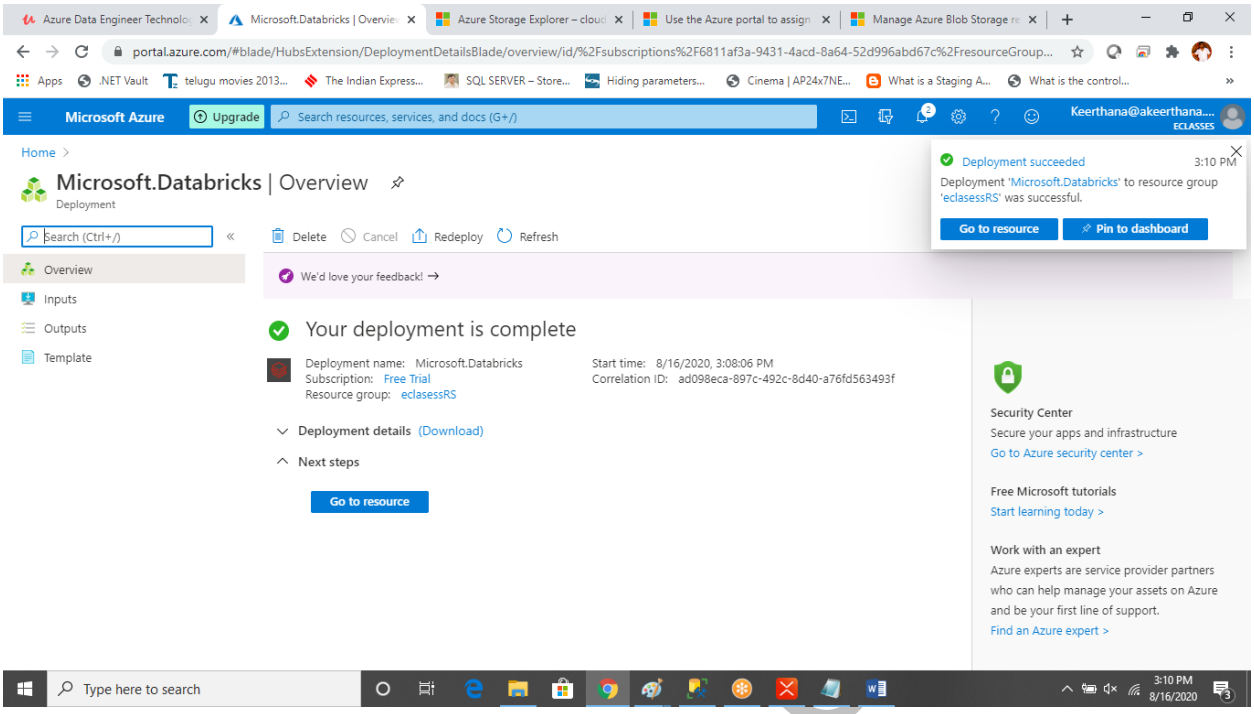


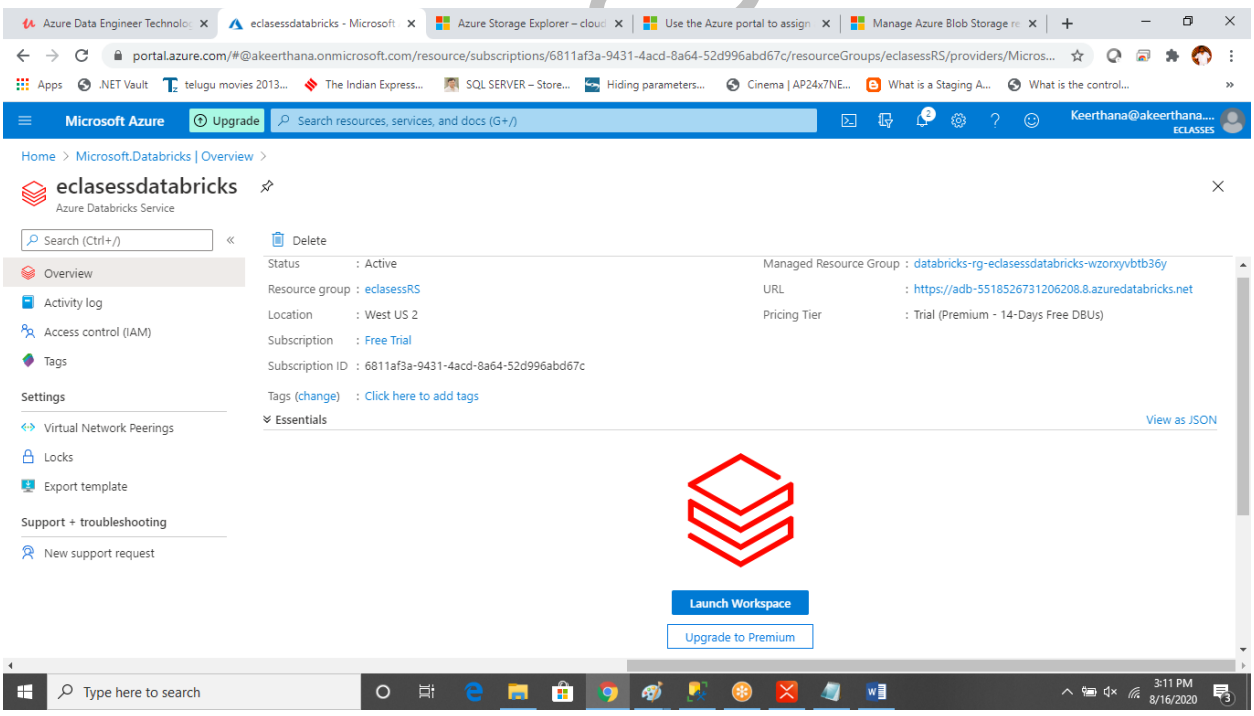
Step-4: Create Azure Bricks Work Space.

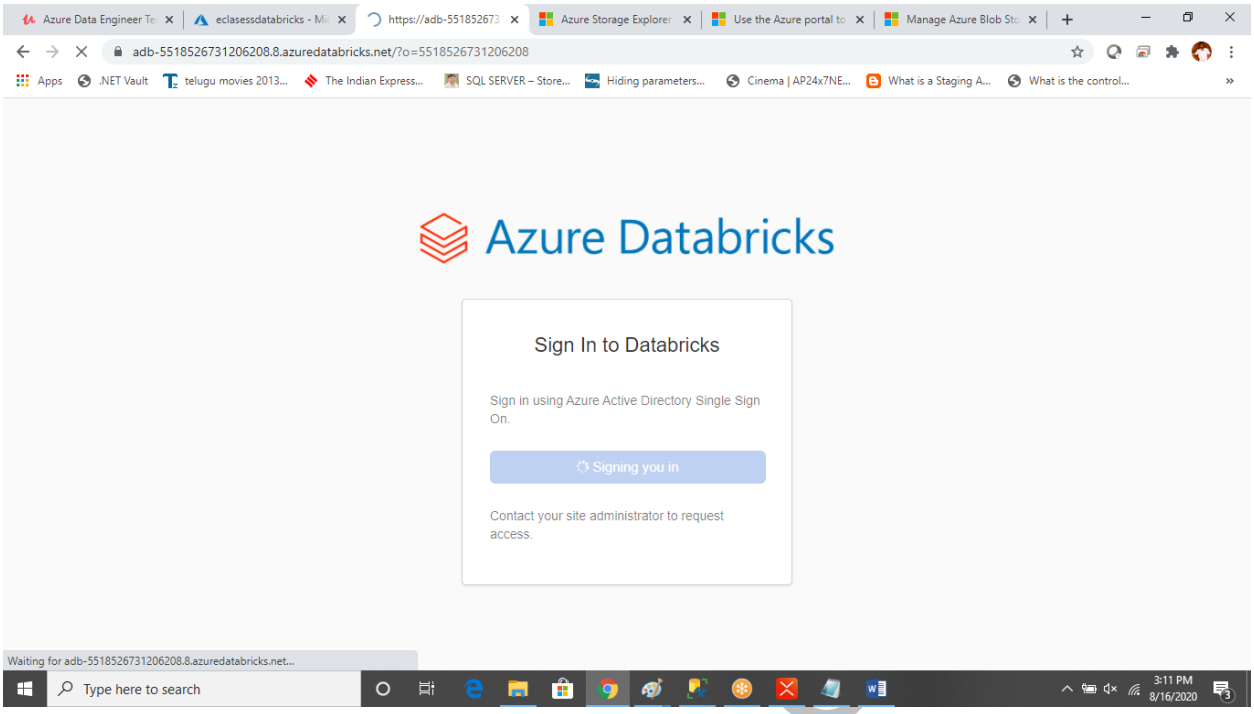


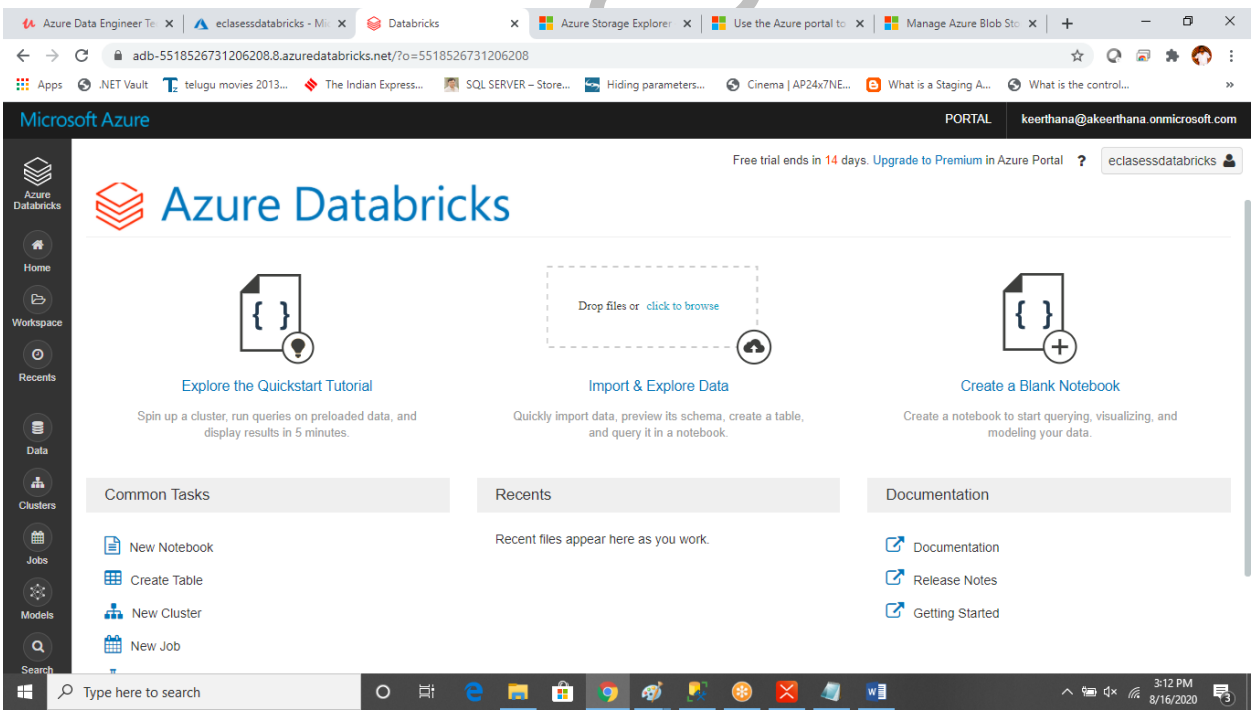

Azure Data Engineer Technolo
Azure Databricks - Microsoft A
Azure Storage Explorer – cloud
Use the Azure portal to assign
Manage Azure Blob Storage re
portal.azure.com/#blade/HubsExtension/BrowseResourceBlade/resourceType/Microsoft.Databricks%2Fworkspaces
Apps
.NET Vault
telugu movies 2013...
The Indian Express...
SQL SERVER – Store...
Hiding parameters...
Cinema | AP24x7NE...
What is a Staging A...
What is the control...
Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Keerthana@akeerthana....
ECLASSES
Home
Azure Databricks
eclasses
Add
Edit columns
Refresh
Assign tags
Subscriptions: Free Trial
Filter by name...
All resource groups
All locations
All tags
No grouping
0 items
Name
Type
Resource group
Location
Subscription
No azure databricks services to display
Try changing your filters if you don't see what you're looking for.
Create azure databricks service
Type here to search
3:06 PM
8/16/2020

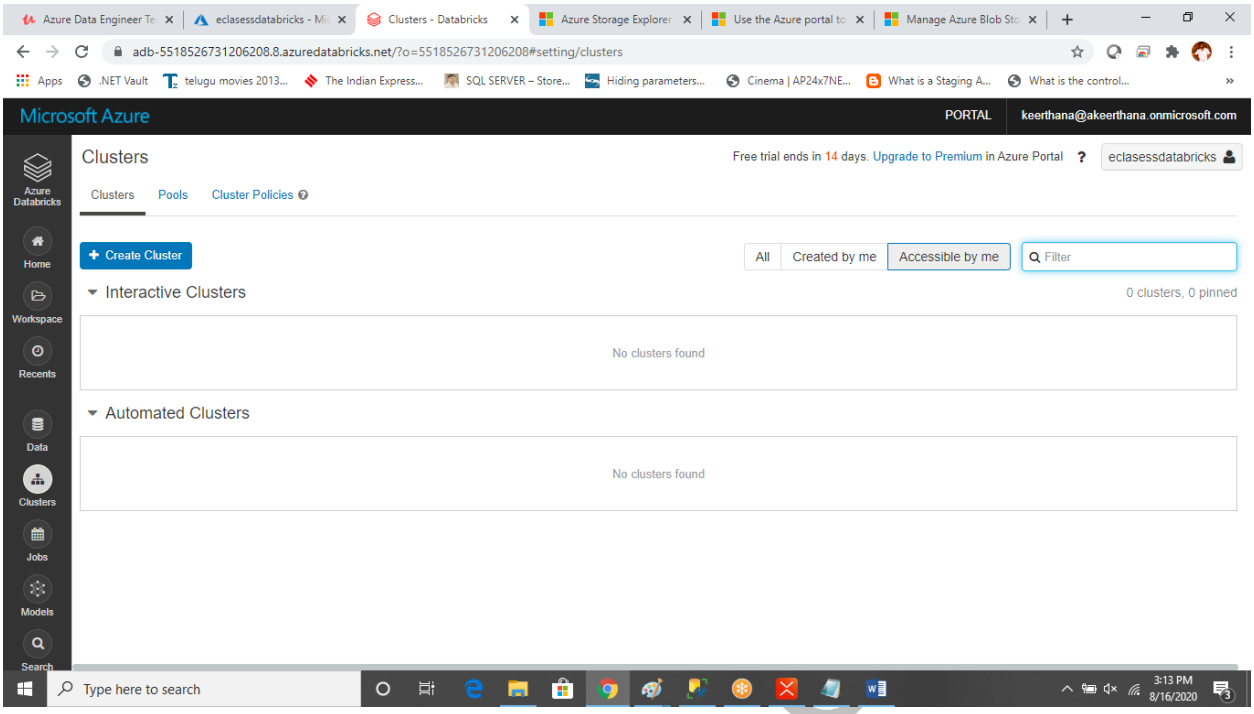

Azure Data Engineer Technolo
Azure Databricks Service - Mic
Azure Storage Explorer – cloud
Use the Azure portal to assign
Manage Azure Blob Storage re
portal.azure.com/#create/Microsoft.Databricks
Apps
.NET Vault
telugu movies 2013...
The Indian Express...
SQL SERVER – Store...
Hiding parameters...
Cinema | AP24x7NE...
What is a Staging A...
What is the control...
Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Keerthana@akeerthana....
ECLASSES
Home
Azure Databricks
Azure Databricks Service
*Basics
Networking
Tags
Review + Create
Project Details
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.
Subscription *
Free Trial
Resource group *
eclasessRS
Create new
Instance Details
Workspace name *
eclasessdatabricks
Location *
West US 2
Pricing Tier *
Trial (Premium - 14-Days Free DBUs)
Review + Create
Next : Networking >
Type here to search
3:07 PM
8/16/2020

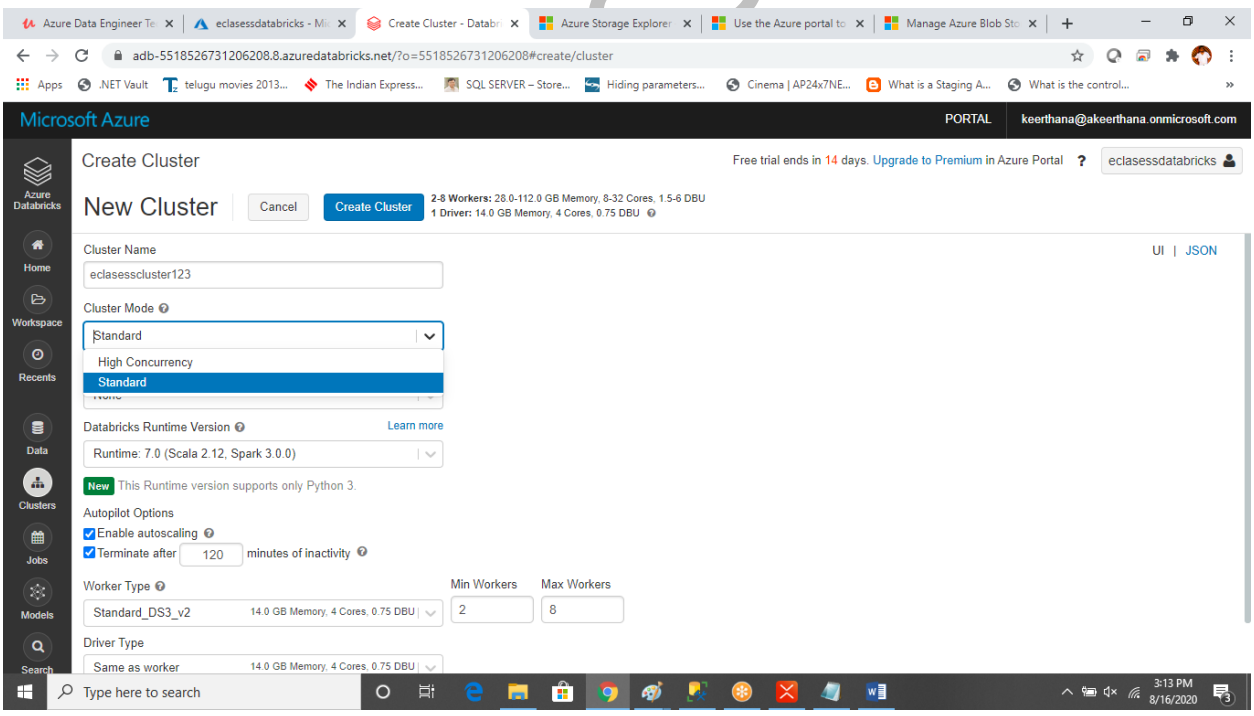

Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Home
Microsoft.Databricks | Overview
Deployment
Search (Ctrl+/)
Overview
Inputs
Outputs
Template
Delete
Cancel
Redeploy
Refresh
We'd love your feedback!
Your deployment is complete
Deployment name: Microsoft.Databricks
Subscription: Free Trial
Resource group: eclasessRS
Start time: 8/16/2020, 3:08:06 PM
Correlation ID: ad098eca-897c-492c-8d40-a76fd563493f
Deployment details (Download)
Next steps
Go to resource
Deployment succeeded
3:10 PM
Deployment 'Microsoft.Databricks' to resource group 'eclasessRS' was successful.
Go to resource
Pin to dashboard
Security Center
Secure your apps and infrastructure
Go to Azure security center >
Free Microsoft tutorials
Start learning today >
Work with an expert
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.
Find an Azure expert >
Type here to search
3:10 PM
8/16/2020

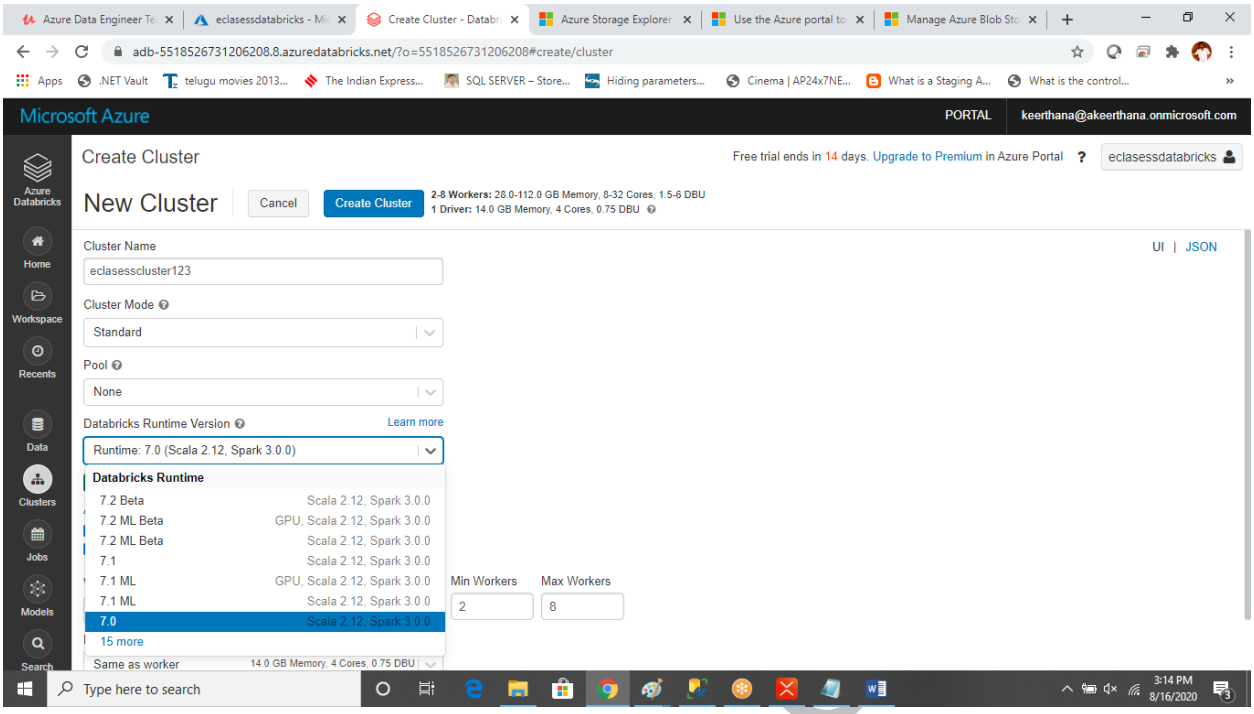

Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Home
Microsoft.Databricks | Overview
eclasessdatabricks
Azure Databricks Service
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Settings
Virtual Network Peerings
Locks
Export template
Support + troubleshooting
New support request
Delete
Status : Active
Resource group : eclasessRS
Location : West US 2
Subscription : Free Trial
Subscription ID : 6811af3a-9431-4acd-8a64-52d996abd67c
Tags (change) : Click here to add tags
Managed Resource Group : databricks-rg-eclasessdatabricks-wzorxyvbtb36y
URL : https://adb-5518526731206208.8.azuredatabricks.net
Pricing Tier : Trial (Premium - 14-Days Free DBUs)
Essentials
View as JSON
Launch Workspace
Upgrade to Premium
Type here to search
3:11 PM
8/16/2020

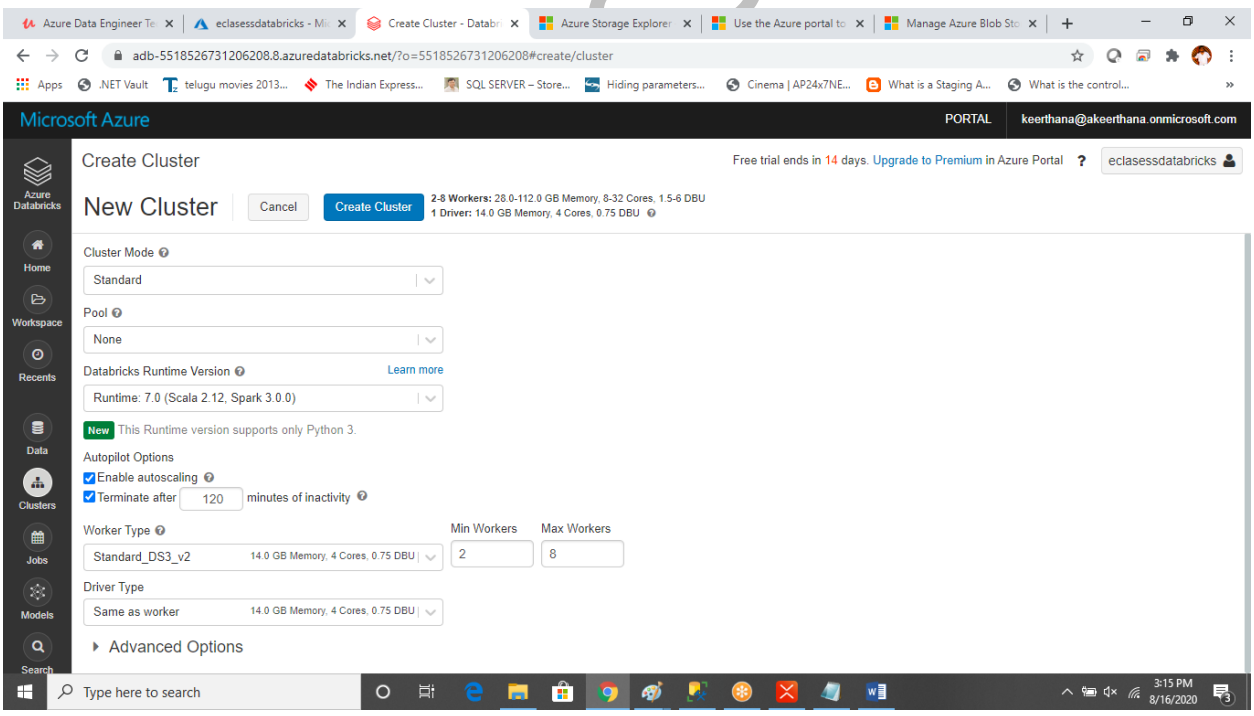

Azure Databricks
Sign In to Databricks
Sign in using Azure Active Directory Single Sign On.
Signing you in
Contact your site administrator to request access.
Waiting for adb-5518526731206208.8.azuredatabricks.net...
Type here to search
3:11 PM
8/16/2020

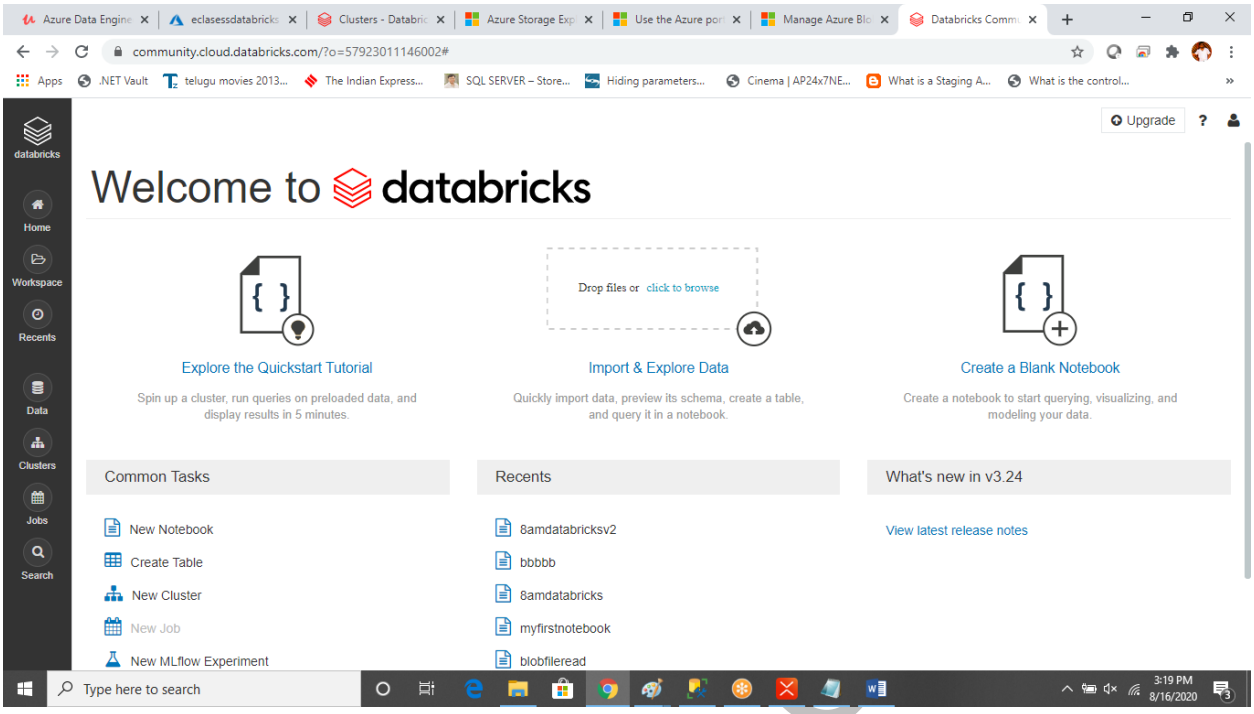

Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Free trial ends in 14 days. Upgrade to Premium in Azure Portal
eclasessdatabricks
Azure Databricks
Home
Workspace
Recents
Data
Clusters
Jobs
Models
Search
Drop files or click to browse
Explore the Quickstart Tutorial
Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.
Import & Explore Data
Quickly import data, preview its schema, create a table, and query it in a notebook.
Create a Blank Notebook
Create a notebook to start querying, visualizing, and modeling your data.
Common Tasks
New Notebook
Create Table
New Cluster
New Job
Recents
Recent files appear here as you work.
Documentation
Documentation
Release Notes
Getting Started
Type here to search
3:12 PM
8/16/2020

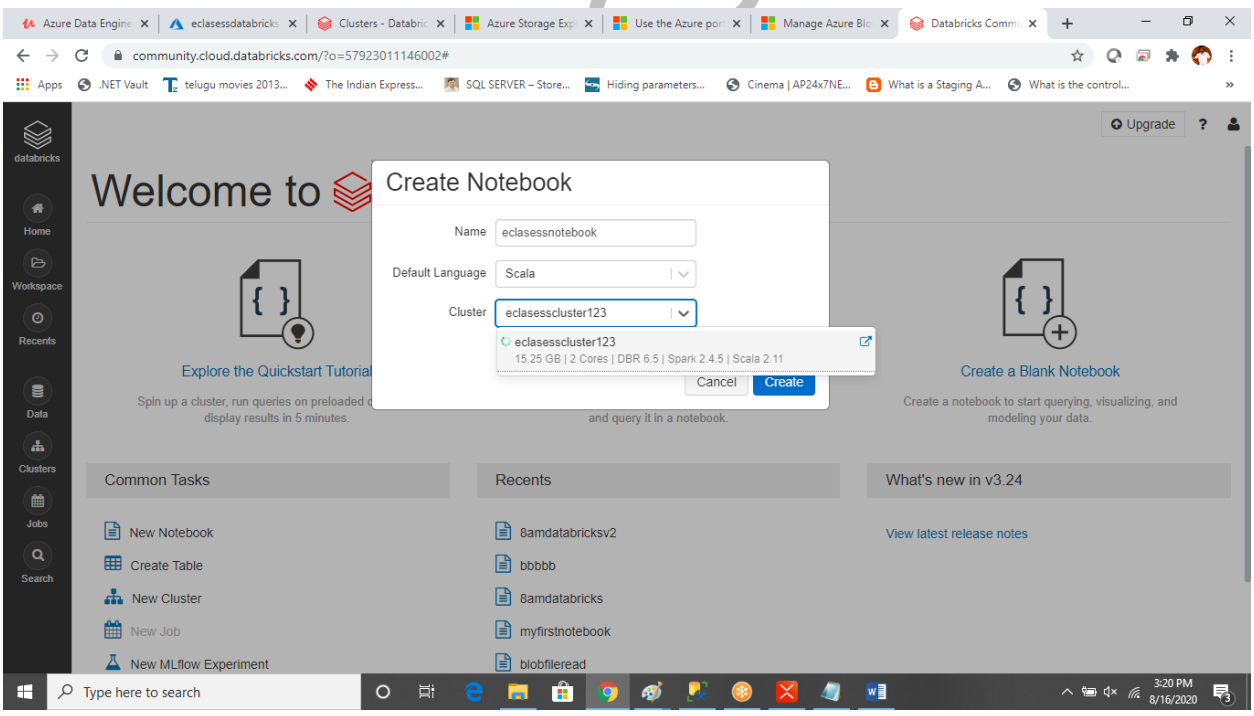

Azure Data Engineer Tr
eclasessdatabricks - Mi
Clusters - Databricks
Azure Storage Explorer
Use the Azure portal to
Manage Azure Blob Sto
adb-5518526731206208.8.azuredatabricks.net/?o=5518526731206208#setting/clusters
Apps
.NET Vault
telugu movies 2013...
The Indian Express...
SQL SERVER – Store...
Hiding parameters...
Cinema | AP24x7NE...
What is a Staging A...
What is the control...
Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Azure Databricks
Home
Workspace
Recents
Data
Clusters
Jobs
Models
Search
Clusters
Free trial ends in 14 days. Upgrade to Premium in Azure Portal
eclasessdatabricks
Clusters
Pools
Cluster Policies
Create Cluster
All
Created by me
Accessible by me
Filter
Interactive Clusters
0 clusters, 0 pinned
No clusters found
Automated Clusters
No clusters found
Type here to search
3:13 PM
8/16/2020

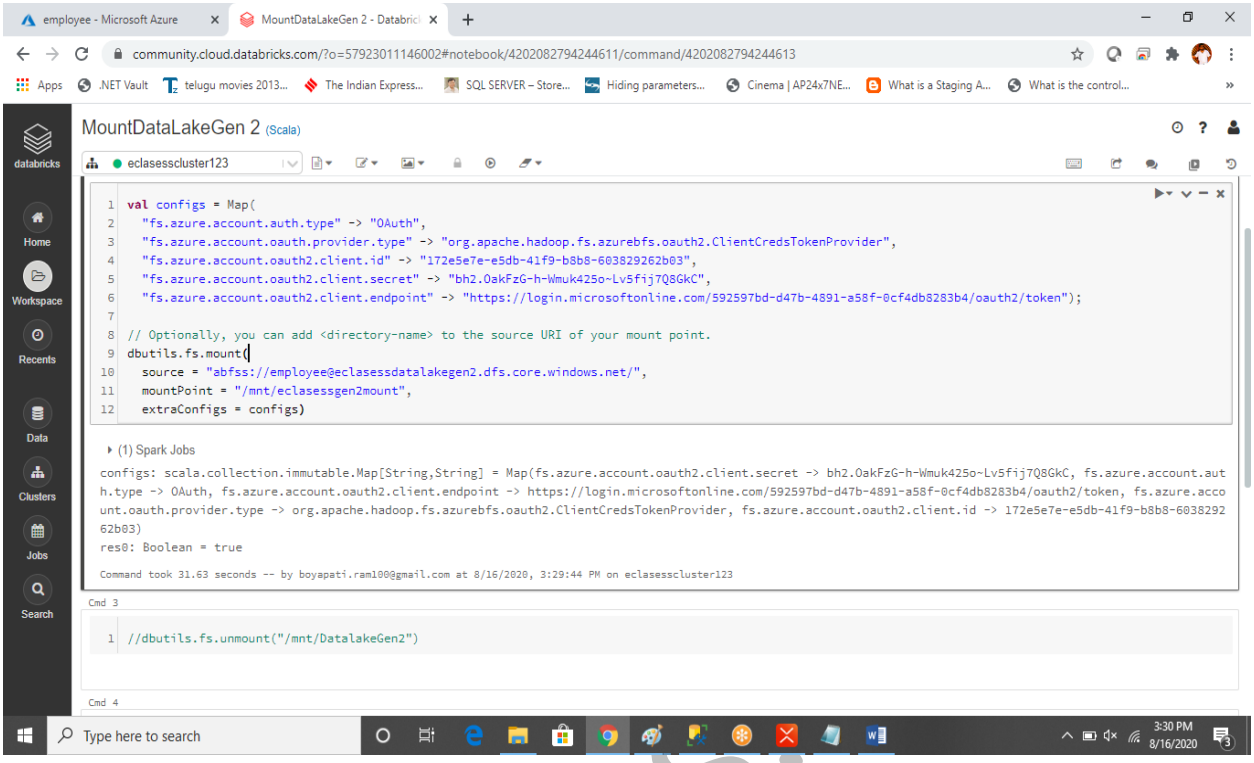

Azure Data Engineer Tr
eclasessdatabricks - Mi
Create Cluster - Databr
Azure Storage Explorer
Use the Azure portal to
Manage Azure Blob Sto
adb-5518526731206208.8.azuredatabricks.net/?o=5518526731206208#create/cluster
Apps
.NET Vault
telugu movies 2013...
The Indian Express...
SQL SERVER – Store...
Hiding parameters...
Cinema | AP24x7NE...
What is a Staging A...
What is the control...
Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Azure Databricks
Home
Workspace
Recents
Data
Clusters
Jobs
Models
Search
Create Cluster
Free trial ends in 14 days. Upgrade to Premium in Azure Portal
eclasessdatabricks
New Cluster
Cancel
Create Cluster
2-8 Workers: 28.0-112.0 GB Memory, 8-32 Cores, 1.5-6 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU
UI | JSON
Cluster Name
eclasesscluster123
Cluster Mode
Standard
High Concurrency
Standard
Databricks Runtime Version
Learn more
Runtime: 7.0 (Scala 2.12, Spark 3.0.0)
New
This Runtime version supports only Python 3.
Autopilot Options
Enable autoscaling
Terminate after
120
minutes of inactivity
Worker Type
Min Workers
Max Workers
Standard_DS3_v2
14.0 GB Memory, 4 Cores, 0.75 DBU
2
8
Driver Type
Same as worker
14.0 GB Memory, 4 Cores, 0.75 DBU
Type here to search
3:13 PM
8/16/2020

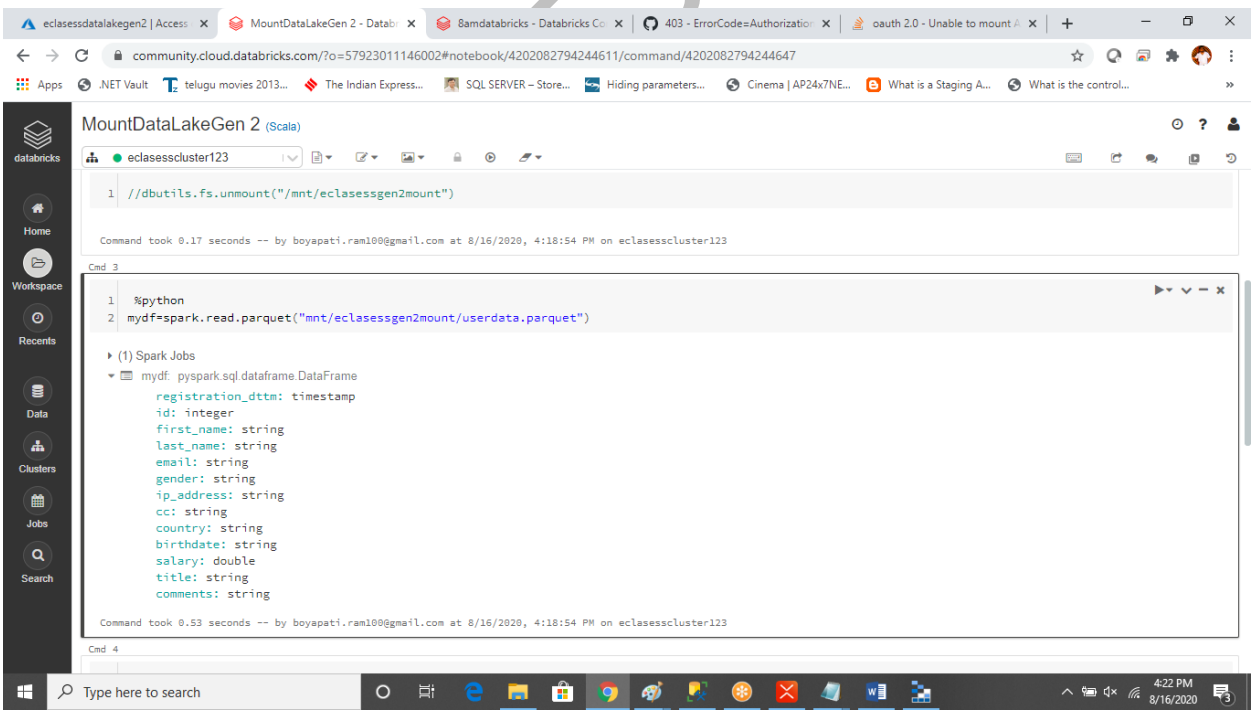

Azure Data Engineer Tr
eclasessdatabricks - Mi
Create Cluster - Databr
Azure Storage Explorer
Use the Azure portal to
Manage Azure Blob Sto
adb-5518526731206208.8.azuredatabricks.net/?o=5518526731206208#create/cluster
Apps
.NET Vault
telugu movies 2013...
The Indian Express...
SQL SERVER – Store...
Hiding parameters...
Cinema | AP24x7NE...
What is a Staging A...
What is the control...
Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Azure Databricks
Home
Workspace
Recents
Data
Clusters
Jobs
Models
Search
Create Cluster
Free trial ends in 14 days. Upgrade to Premium in Azure Portal
eclasessdatabricks
New Cluster
Cancel
Create Cluster
2-8 Workers: 28.0-112.0 GB Memory, 8-32 Cores, 1.5-6 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU
UI | JSON
Cluster Name
eclasesscluster123
Cluster Mode
Standard
Pool
None
Databricks Runtime Version
Learn more
Runtime: 7.0 (Scala 2.12, Spark 3.0.0)
Databricks Runtime
7.2 Beta Scala 2.12, Spark 3.0.0
7.2 ML Beta GPU, Scala 2.12, Spark 3.0.0
7.2 ML Beta Scala 2.12, Spark 3.0.0
7.1 Scala 2.12, Spark 3.0.0
7.1 ML GPU, Scala 2.12, Spark 3.0.0
7.1 ML Scala 2.12, Spark 3.0.0
7.0 Scala 2.12, Spark 3.0.0
15 more
Min Workers
Max Workers
2
8
Same as worker
14.0 GB Memory, 4 Cores, 0.75 DBU
Type here to search
3:14 PM
8/16/2020

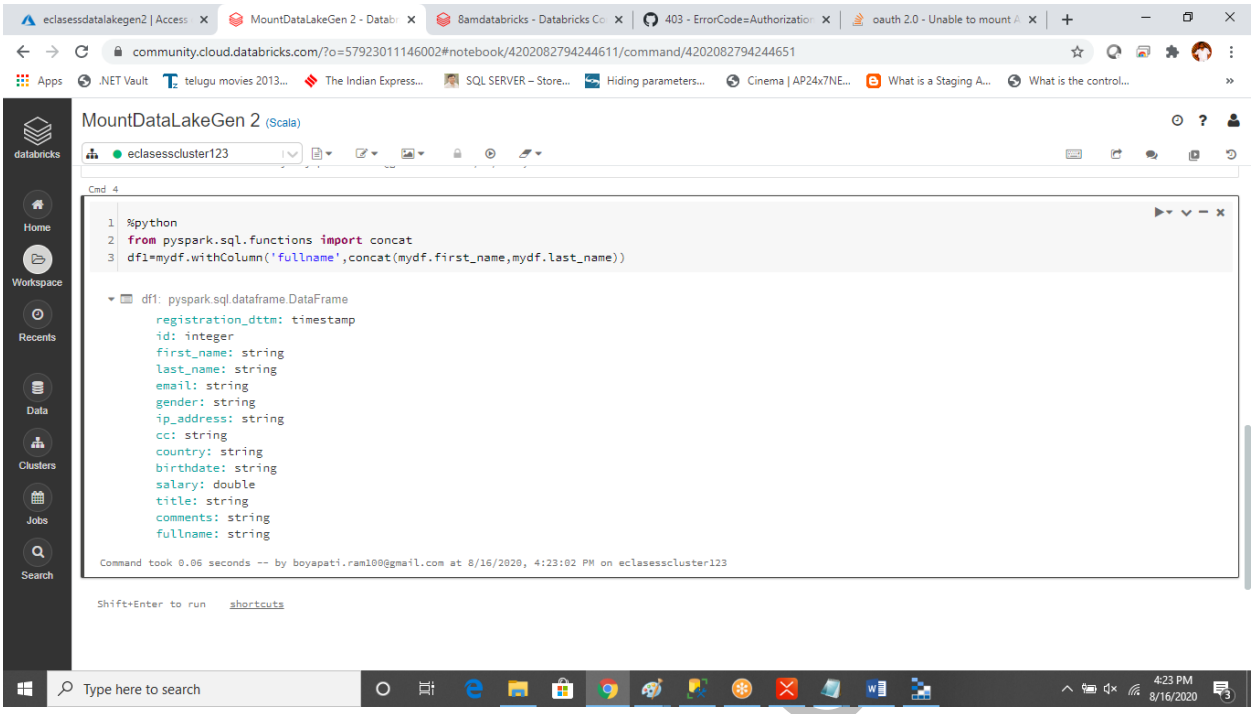

Azure Data Engineer Tr
eclasessdatabricks - Mi
Create Cluster - Databr
Azure Storage Explorer
Use the Azure portal to
Manage Azure Blob Sto
adb-5518526731206208.8.azuredatabricks.net/?o=5518526731206208#create/cluster
Apps
.NET Vault
telugu movies 2013...
The Indian Express...
SQL SERVER – Store...
Hiding parameters...
Cinema | AP24x7NE...
What is a Staging A...
What is the control...
Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Azure Databricks
Home
Workspace
Recents
Data
Clusters
Jobs
Models
Search
Create Cluster
Free trial ends in 14 days. Upgrade to Premium in Azure Portal
eclasessdatabricks
New Cluster
Cancel
Create Cluster
2-8 Workers: 28.0-112.0 GB Memory, 8-32 Cores, 1.5-6 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU
Cluster Mode
Standard
Pool
None
Databricks Runtime Version
Learn more
Runtime: 7.0 (Scala 2.12, Spark 3.0.0)
New
This Runtime version supports only Python 3.
Autopilot Options
Enable autoscaling
Terminate after
120
minutes of inactivity
Worker Type
Standard_DS3_v2
14.0 GB Memory, 4 Cores, 0.75 DBU
Min Workers
Max Workers
2
8
Driver Type
Same as worker
14.0 GB Memory, 4 Cores, 0.75 DBU
Advanced Options
Type here to search
3:15 PM
8/16/2020

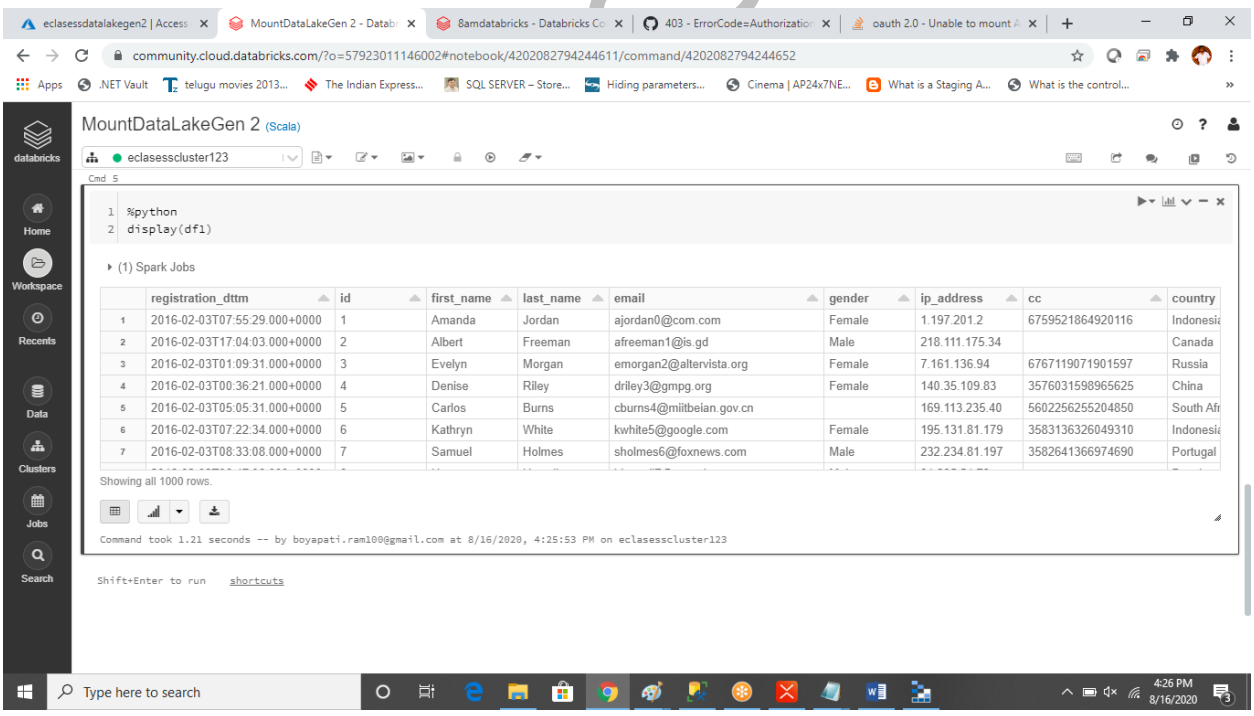

community.cloud.databricks.com/?o=5792301114600 2#
Upgrade
databricks
Home
Workspace
Recents
Data
Clusters
Jobs
Search
Welcome to databricks
Drop files or click to browse
Explore the Quickstart Tutorial
Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.
Import & Explore Data
Quickly import data, preview its schema, create a table, and query it in a notebook.
Create a Blank Notebook
Create a notebook to start querying, visualizing, and modeling your data.
Common Tasks
New Notebook
Create Table
New Cluster
New Job
New MLflow Experiment
Recents
8amdatabricksv2
bbbbb
8amdatabricks
myfirstnotebook
blobfileread
What's new in v3.24
View latest release notes
Type here to search
3:19 PM
8/16/2020


community.cloud.databricks.com/?o=5792301114600 2#
Upgrade
databricks
Welcome to
Create Notebook
Name
eclasessnotebook
Default Language
Scala
Cluster
eclasesscluster123
eclasesscluster123
15.25 GB | 2 Cores | DBR 6.5 | Spark 2.4.5 | Scala 2.11
Cancel
Create
Explore the Quickstart Tutorial
Create a Blank Notebook
Create a notebook to start querying, visualizing, and modeling your data.
Common Tasks
New Notebook
Create Table
New Cluster
New Job
New MLflow Experiment
Recents
8amdatabricksv2
bbbbb
8amdatabricks
myfirstnotebook
blobfileread
What's new in v3.24
View latest release notes
Type here to search
3:20 PM
8/16/2020

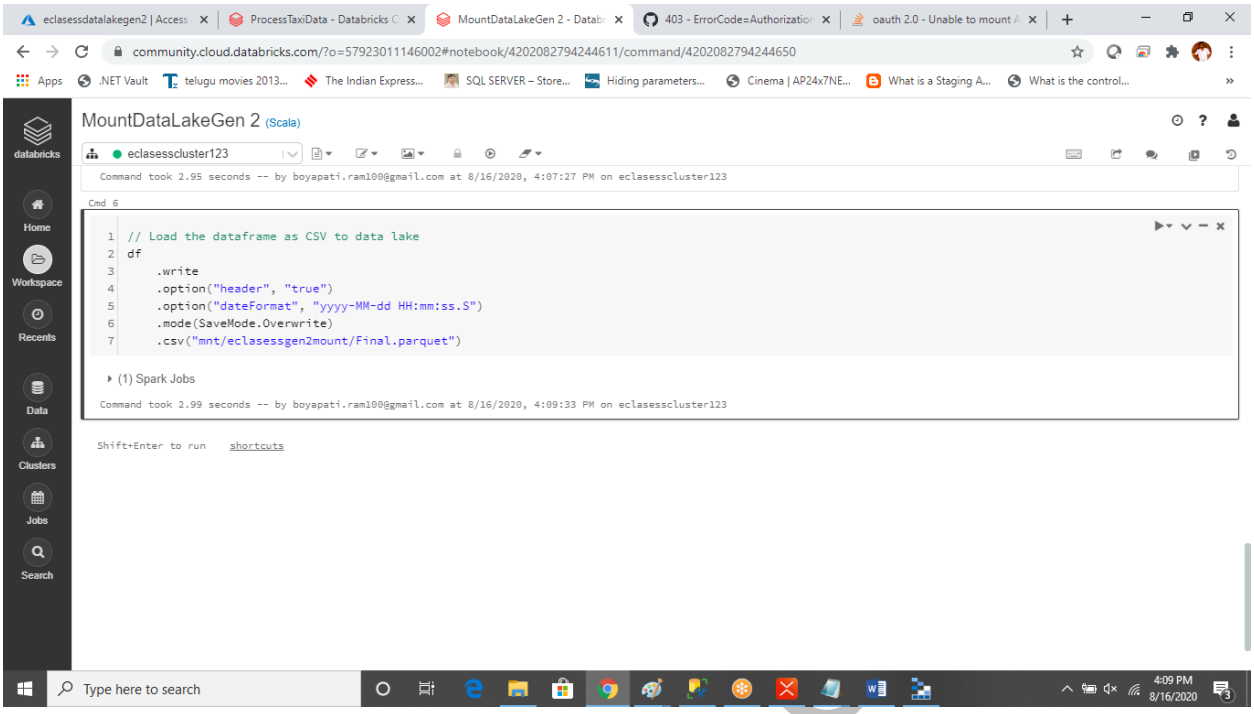

MountDataLakeGen 2 (Scala)
eclassescluster123
val configs = Map(
"fs.azure.account.auth.type" -> "OAuth",
"fs.azure.account.oauth.provider.type" -> "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider",
"fs.azure.account.oauth2.client.id" -> "172e5e7e-e5db-41f9-b8b8-603829262b03",
"fs.azure.account.oauth2.client.secret" -> "bh2.0akFzG-h-Wmuk425o-Lv5fij7Q8GkC",
"fs.azure.account.oauth2.client.endpoint" -> "https://login.microsoftonline.com/592597bd-d47b-4891-a58f-0cf4db8283b4/oauth2/token");
// Optionally, you can add <directory-name> to the source URI of your mount point.
dbutils.fs.mount(
source = "abfss://employee@eclasessdatalakegen2.dfs.core.windows.net/",
mountPoint = "/mnt/eclasessgen2mount",
extraConfigs = configs)
(1) Spark Jobs
Command took 31.63 seconds -- by boyapati.ram100@gmail.com at 8/16/2020, 3:29:44 PM on eclassescluster123
Cmd 3
//dbutils.fs.unmount("/mnt/DatalakeGen2")
Cmd 4

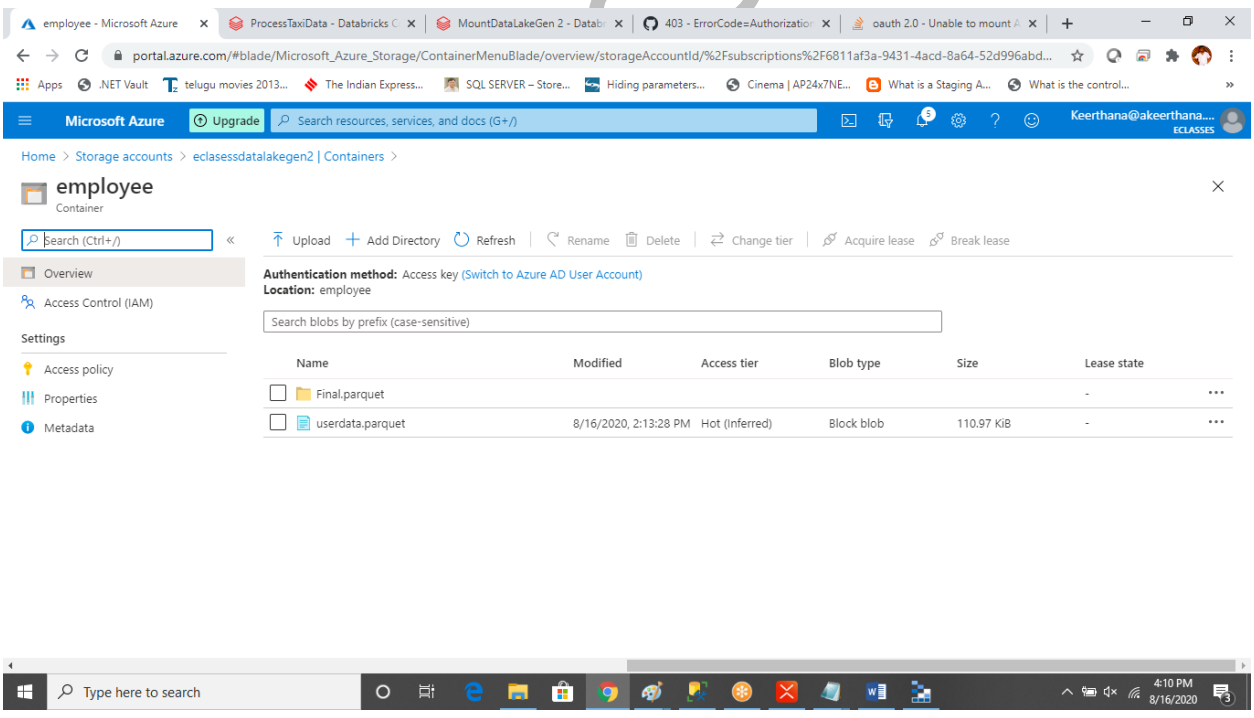

MountDataLakeGen 2 (Scala)
eclassescluster123
//dbutils.fs.unmount("/mnt/eclasessgen2mount")
Command took 0.17 seconds -- by boyapati.ram100@gmail.com at 8/16/2020, 4:18:54 PM on eclassescluster123
Cmd 3
%python
mydf=spark.read.parquet("mnt/eclasessgen2mount/userdata.parquet")
(1) Spark Jobs
mydf: pyspark.sql.dataframe.DataFrame
registration_dttm: timestamp
id: integer
first_name: string
last_name: string
email: string
gender: string
ip_address: string
cc: string
country: string
birthdate: string
salary: double
title: string
comments: string
Command took 0.53 seconds -- by boyapati.ram100@gmail.com at 8/16/2020, 4:18:54 PM on eclassescluster123
Cmd 4

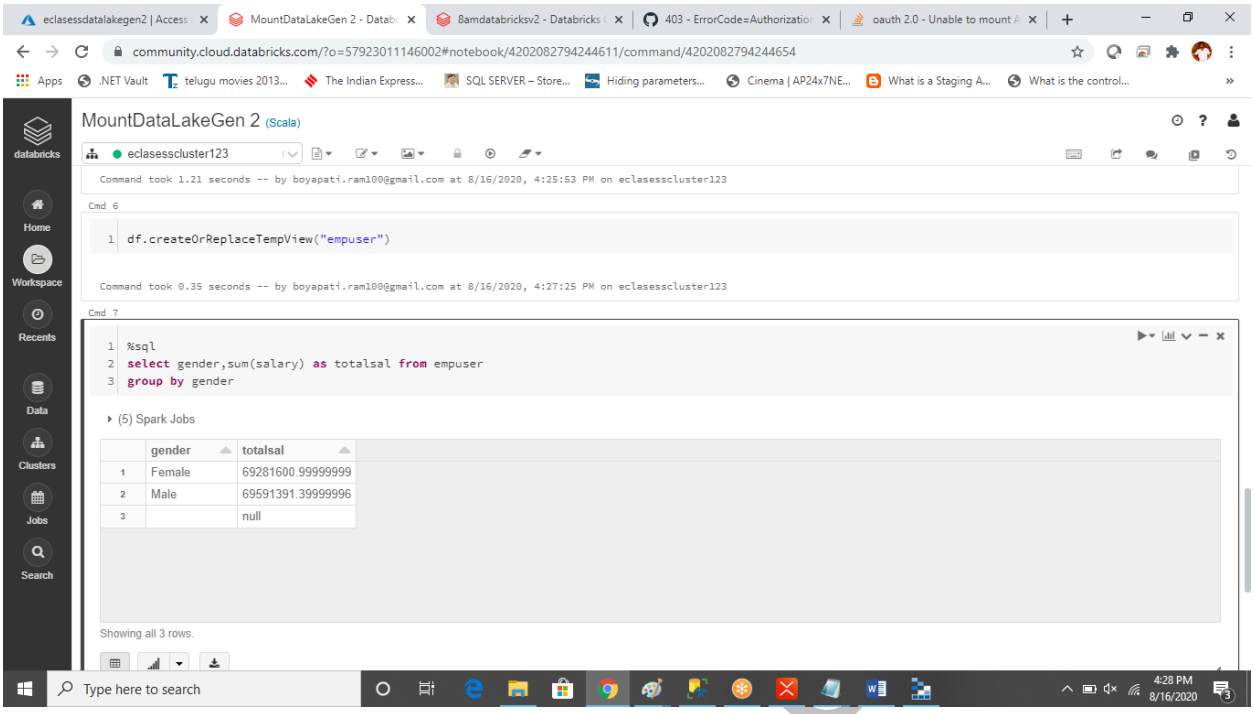


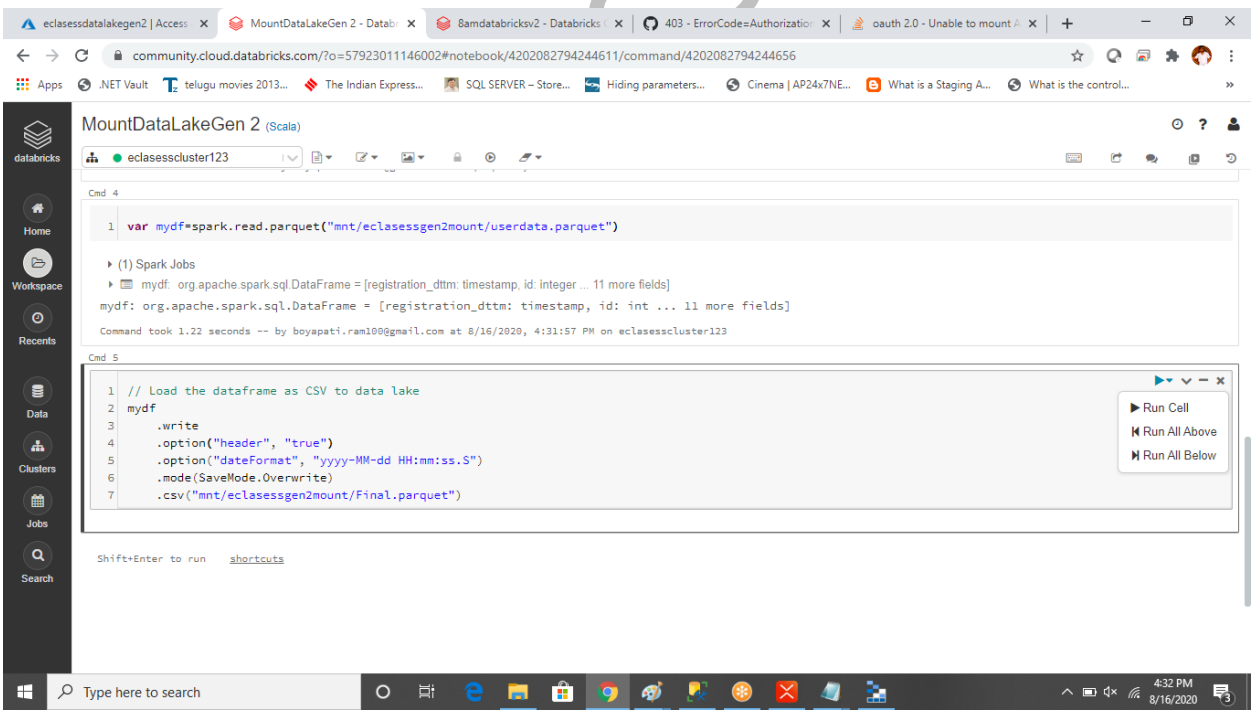


	registration_dttm	id	first_name	last_name	email	gender	ip_address	cc	country
1	2016-02-03T07:55:29.000+0000	1	Amanda	Jordan	ajordan0@com.com	Female	1.197.201.2	6759521864920116	Indonesia
2	2016-02-03T17:04:03.000+0000	2	Albert	Freeman	afreeman1@is.gd	Male	218.111.175.34		Canada
3	2016-02-03T01:09:31.000+0000	3	Evelyn	Morgan	emorgan2@altervista.org	Female	7.161.136.94	6767119071901597	Russia
4	2016-02-03T00:36:21.000+0000	4	Denise	Riley	driley3@gmpg.org	Female	140.35.109.83	3576031598965625	China
5	2016-02-03T05:05:31.000+0000	5	Carlos	Burns	cburns4@miitbeian.gov.cn		169.113.235.40	5602256255204850	South Afr
6	2016-02-03T07:22:34.000+0000	6	Kathryn	White	kwhite5@google.com	Female	195.131.81.179	3583136326049310	Indonesia
7	2016-02-03T08:33:08.000+0000	7	Samuel	Holmes	sholmes6@foxnews.com	Male	232.234.81.197	3582641366974690	Portugal


MountDataLakeGen 2 (Scala)
eclasesscluster123
Command took 2.95 seconds -- by boyapati.ram100@gmail.com at 8/16/2020, 4:07:27 PM on eclasesscluster123
Cmd 6
// Load the dataframe as CSV to data lake
df
.write
.option("header", "true")
.option("dateFormat", "yyyy-MM-dd HH:mm:ss.S")
.mode(SaveMode.Overwrite)
.csv("mnt/eclasesgen2mount/Final.parquet")
(1) Spark Jobs
Command took 2.99 seconds -- by boyapati.ram100@gmail.com at 8/16/2020, 4:09:33 PM on eclasesscluster123
Shift+Enter to run shortcuts
Type here to search
4:09 PM
8/16/2020


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Home > Storage accounts > eclasessdatalakegen2 | Containers >
employee
Container
Upload
Add Directory
Refresh
Rename
Delete
Change tier
Acquire lease
Break lease
Overview
Access Control (IAM)
Settings
Access policy
Properties
Metadata
Authentication method: Access key (Switch to Azure AD User Account)
Location: employee
Search blobs by prefix (case-sensitive)
Name
Modified
Access tier
Blob type
Size
Lease state
Final.parquet
userdata.parquet
8/16/2020, 2:13:28 PM
Hot (Inferred)
Block blob
110.97 KiB
Type here to search
4:10 PM
8/16/2020

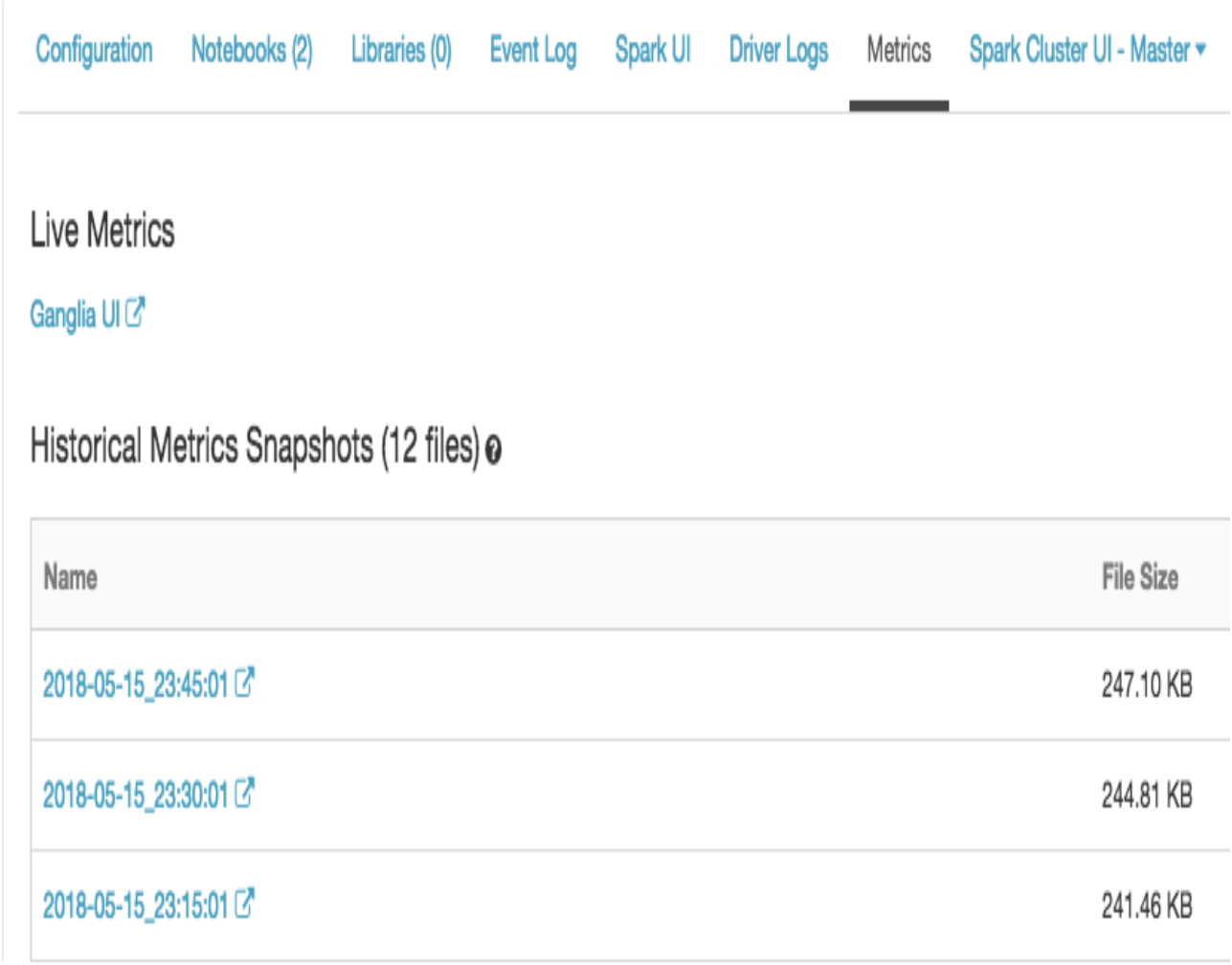

MountDataLakeGen 2 (Scala)
eclasesscluster123
Command took 1.21 seconds -- by boyapati.ram100@gmail.com at 8/16/2020, 4:25:53 PM on eclasesscluster123
Cmd 6
df.createOrReplaceTempView("empuser")
Command took 0.35 seconds -- by boyapati.ram100@gmail.com at 8/16/2020, 4:27:25 PM on eclasesscluster123
Cmd 7
%sql
select gender,sum(salary) as totalsal from empuser
group by gender
(5) Spark Jobs
gender totalsal
1 Female 69281600.99999999
2 Male 69591391.39999996
3 null
Showing all 3 rows.


MountDataLakeGen 2 (Scala)
eclasesscluster123
Cmd 4
var mydf=spark.read.parquet("mnt/eclasesgen2mount/userdata.parquet")
(1) Spark Jobs
mydf: org.apache.spark.sql.DataFrame = [registration_dttm: timestamp, id: integer ... 11 more fields]
Command took 1.22 seconds -- by boyapati.ram100@gmail.com at 8/16/2020, 4:31:57 PM on eclasesscluster123
Cmd 5
// Load the dataframe as CSV to data lake
mydf
.write
.option("header", "true")
.option("dateFormat", "yyyy-MM-dd HH:mm:ss.S")
.mode(SaveMode.Overwrite)
.csv("mnt/eclasesgen2mount/Final.parquet")
Run Cell
Run All Above
Run All Below
Shift+Enter to run shortcuts

Azure Data bricks Monitoring:

Ganglia metrics

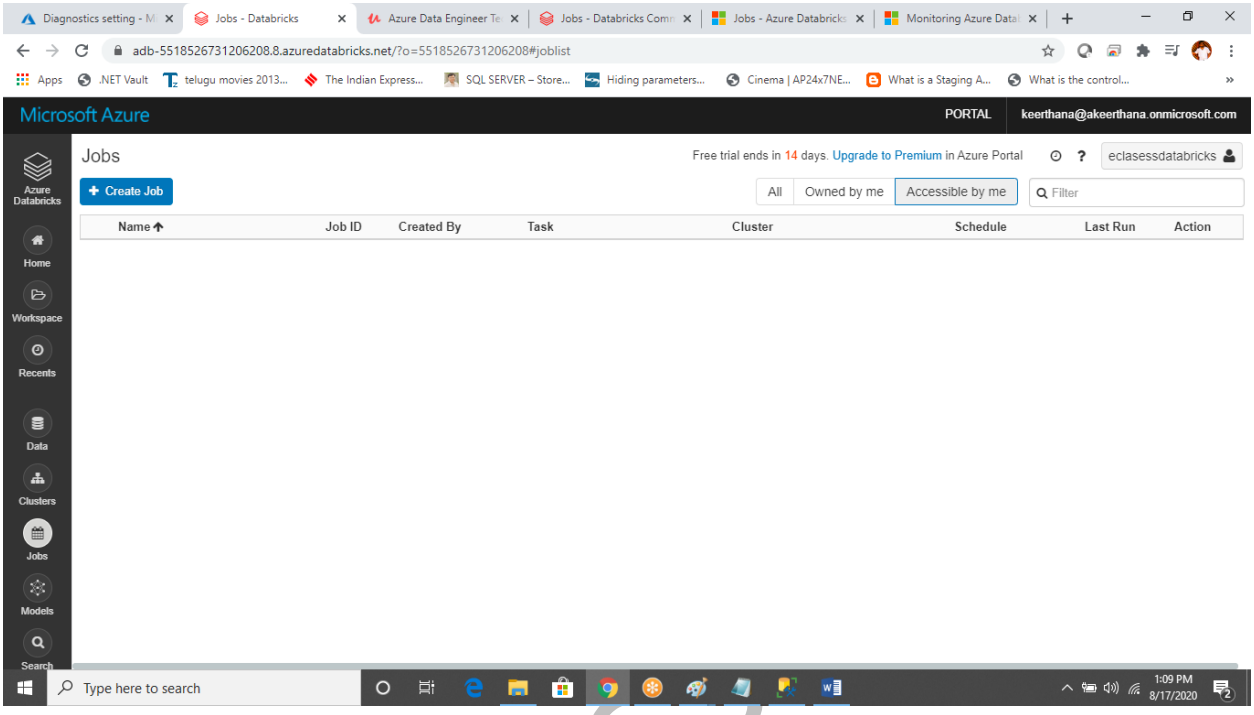
To access the Ganglia UI, navigate to the **Metrics** tab on the cluster details page. CPU metrics are available in the Ganglia UI for all Data bricks runtimes. GPU metrics are available for GPU-enabled clusters running Data bricks Runtime 4.1 and above.

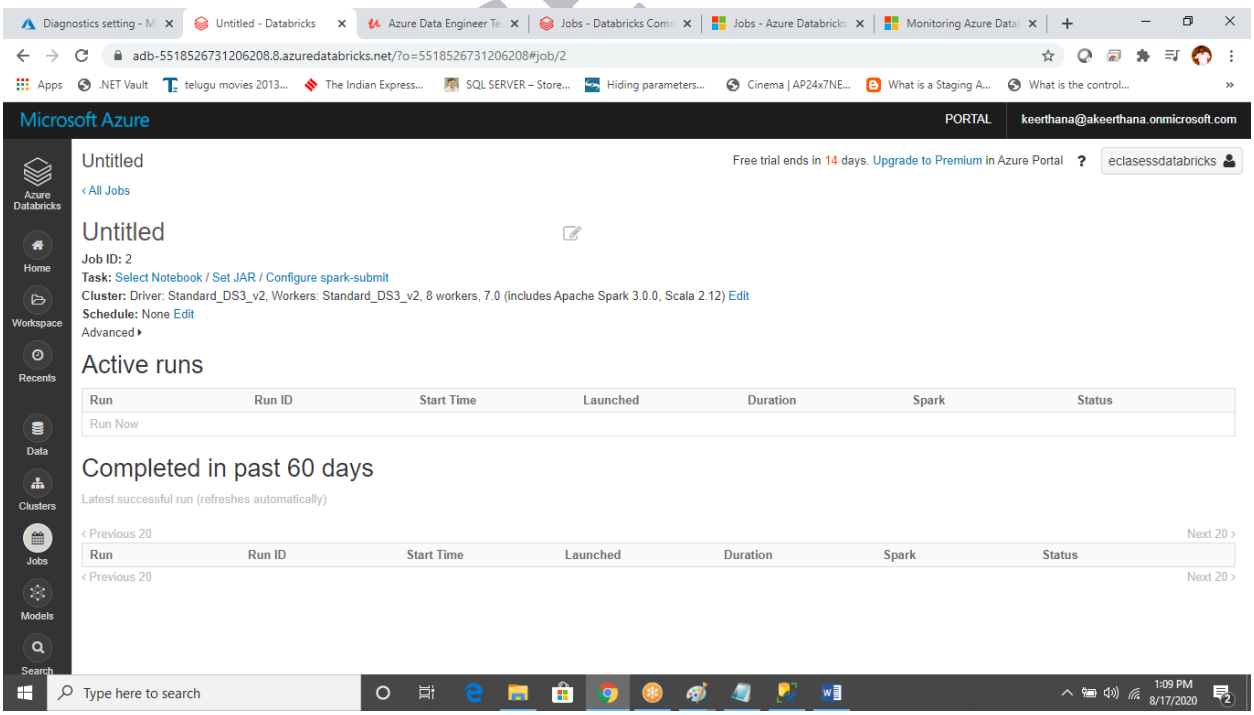


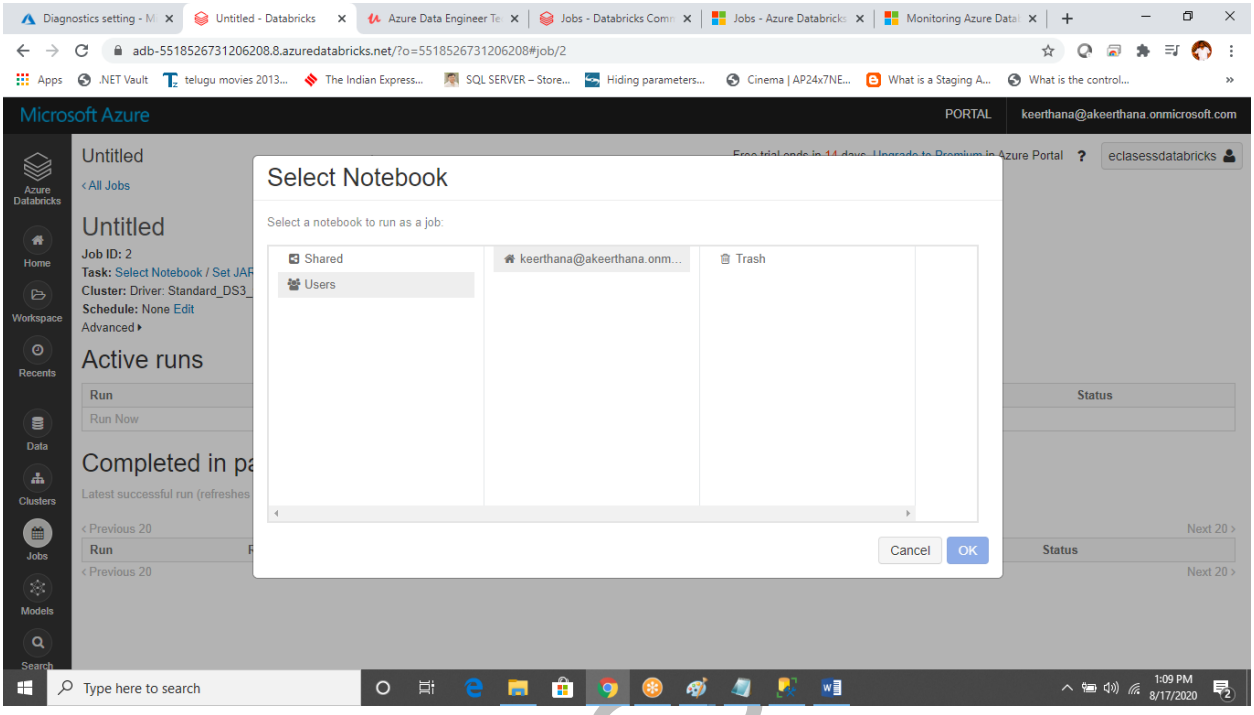
To view live metrics, click the **Ganglia UI** link.

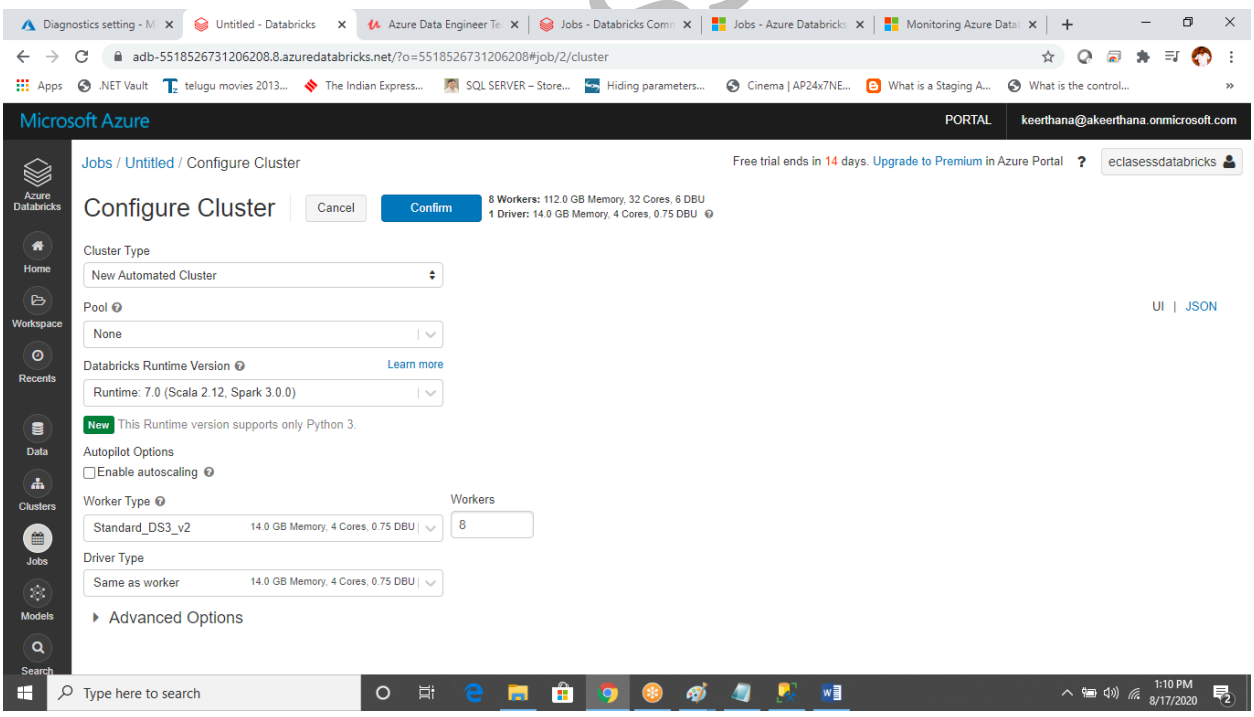
To view historical metrics, click a snapshot file. The snapshot contains aggregated metrics for the hour preceding the selected time.

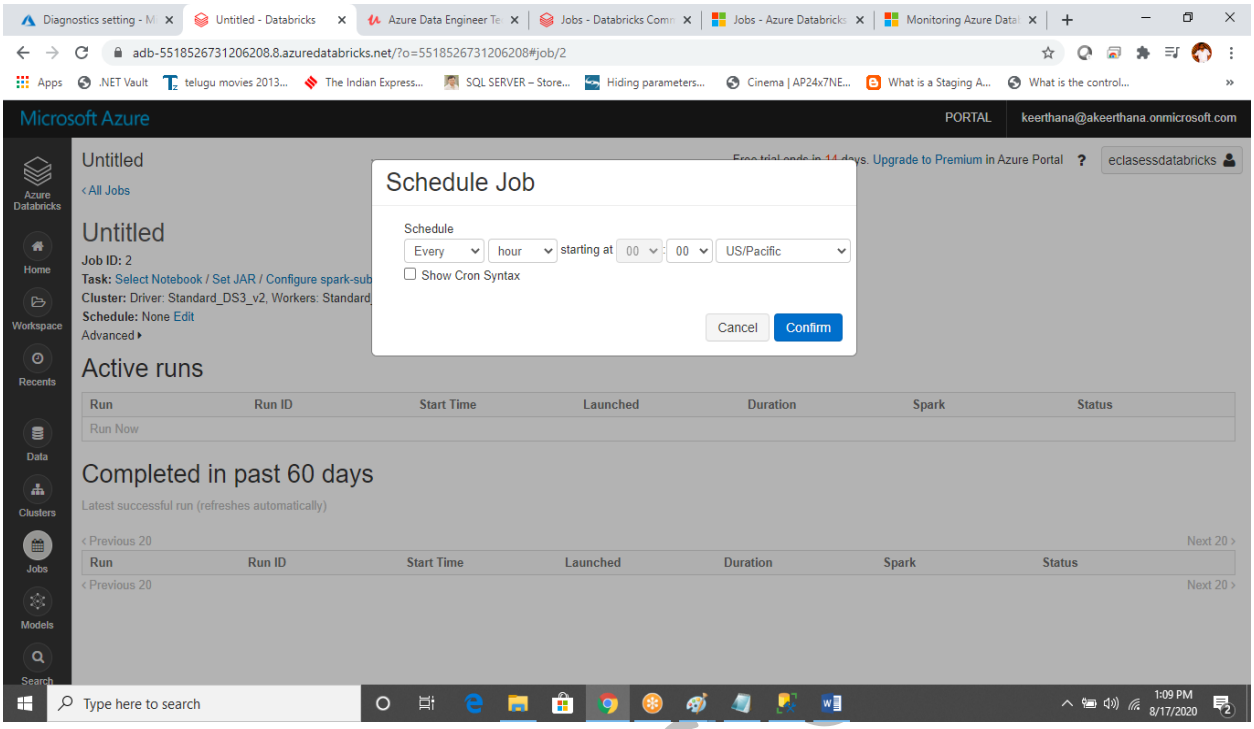
How to Create Jobs in Azure Data bricks:

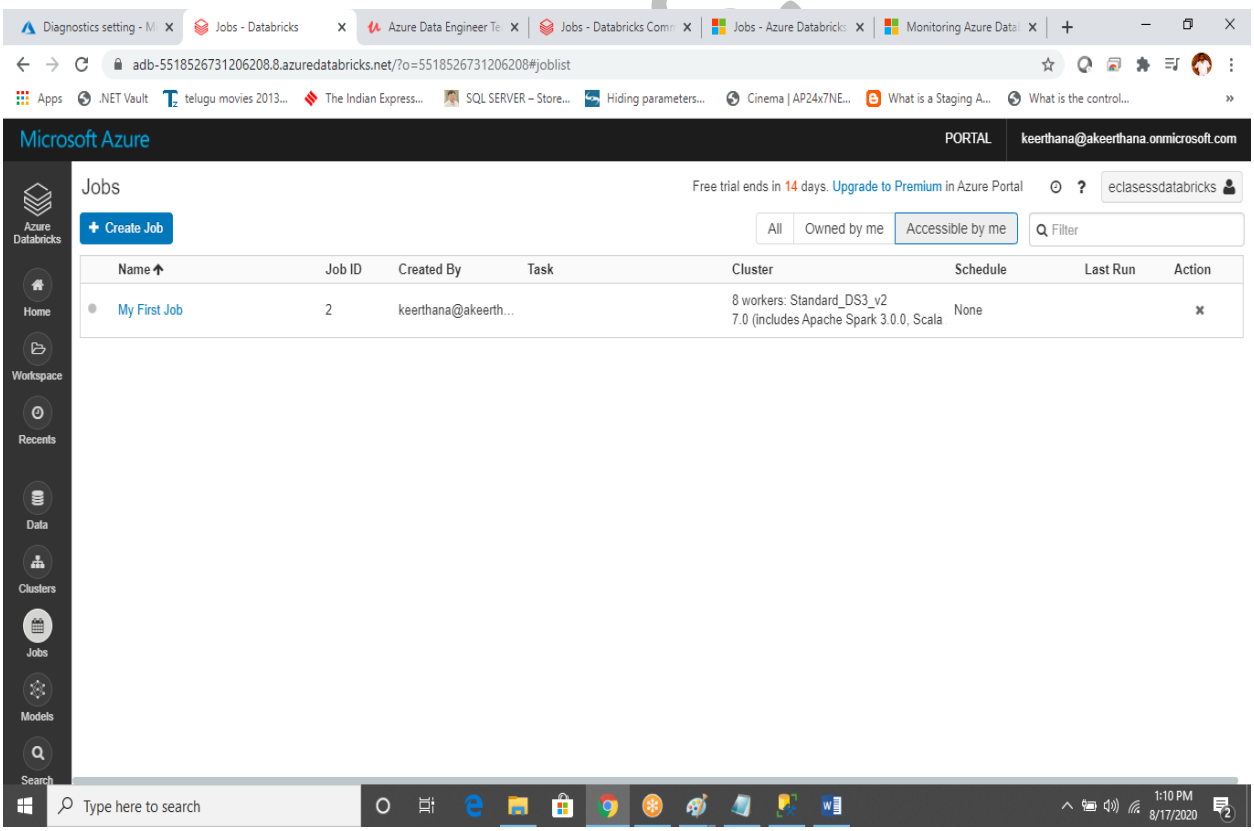





Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Untitled
All Jobs
Untitled
Job ID: 2
Task: Select Notebook / Set JAR
Cluster: Driver: Standard_DS3_
Schedule: None Edit
Advanced
Active runs
Run
Run Now
Status
Completed in pa
Latest successful run (refreshes
Previous 20
Next 20
Select Notebook
Select a notebook to run as a job:
Shared
Users
keerthana@akeerthana.onm...
Trash
Cancel
OK
Azure Databricks
Home
Workspace
Recents
Data
Clusters
Jobs
Models
Search
eclasessdatabricks
Type here to search
1:09 PM
8/17/2020


Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Jobs / Untitled / Configure Cluster
Free trial ends in 14 days. Upgrade to Premium in Azure Portal
eclasessdatabricks
Configure Cluster
Cancel
Confirm
8 Workers: 112.0 GB Memory, 32 Cores, 6 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU
Cluster Type
New Automated Cluster
Pool
None
Databricks Runtime Version
Learn more
Runtime: 7.0 (Scala 2.12, Spark 3.0.0)
New This Runtime version supports only Python 3.
Autopilot Options
Enable autoscaling
Worker Type
Standard_DS3_v2
14.0 GB Memory, 4 Cores, 0.75 DBU
Workers
8
Driver Type
Same as worker
14.0 GB Memory, 4 Cores, 0.75 DBU
Advanced Options
UI | JSON
Azure Databricks
Home
Workspace
Recents
Data
Clusters
Jobs
Models
Search
Type here to search
1:10 PM
8/17/2020


Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Untitled
All Jobs
Untitled
Job ID: 2
Schedule: None Edit
Advanced
Schedule Job
Schedule
Every
hour
starting at
US/Pacific
Show Cron Syntax
Cancel
Confirm
Active runs
Run
Run ID
Start Time
Launched
Duration
Spark
Status
Run Now
Completed in past 60 days
Latest successful run (refreshes automatically)
Previous 20
Next 20
Type here to search
1:09 PM
8/17/2020


Microsoft Azure
PORTAL
keerthana@akeerthana.onmicrosoft.com
Jobs
Free trial ends in 14 days. Upgrade to Premium in Azure Portal
eclasessdatabricks
Create Job
All
Owned by me
Accessible by me
Filter
Name
Job ID
Created By
Task
Cluster
Schedule
Last Run
Action
My First Job
2
keerthana@akeerth...
8 workers: Standard_DS3_v2
7.0 (includes Apache Spark 3.0.0, Scala
None
Type here to search
1:10 PM
8/17/2020

PySpark in data bricks

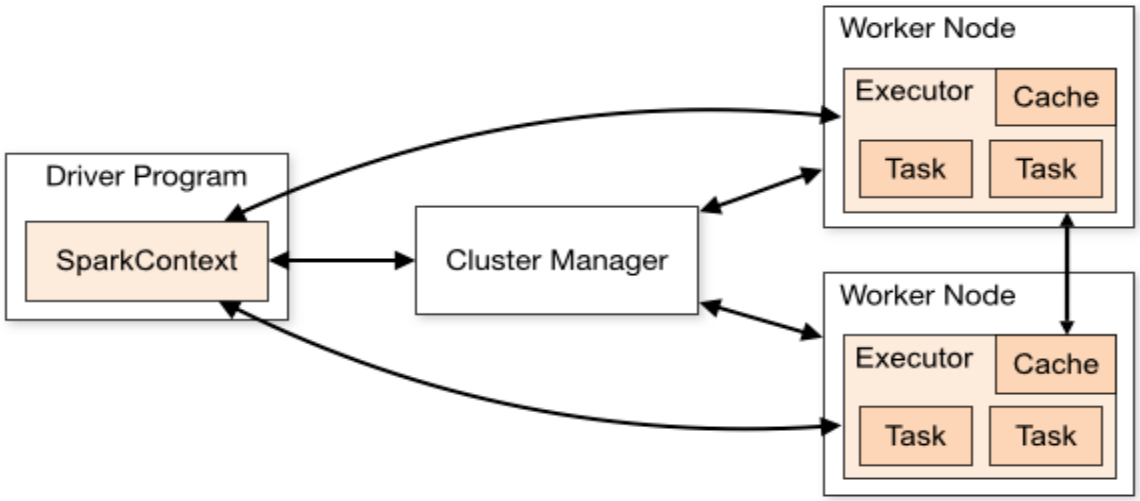
PySpark is a Spark library written in Python to run Python application using Apache Spark capabilities,
Using PySpark we can run applications parallel on the distributed cluster (multiple nodes).
Features:
In-memory computation
Distributed processing using parallelize
Can be used with many cluster managers (Spark, Yarn, Mesos etc.)
Fault-tolerant
Immutable

Advantages of PySpark:

PySpark is a general-purpose, in-memory, distributed processing engine that allows you to process data efficiently in a distributed fashion.
Applications running on PySpark are 100x faster than traditional systems.
You will get great benefits using PySpark for data ingestion pipelines.
Using PySpark we can process data from Hadoop HDFS, AWS S3, and many file systems.
PySpark also is used to process real-time data using Streaming and Kafka.
Using PySpark streaming you can also stream files from the file system and also stream from the socket.
PySpark natively has machine learning and graph libraries.

PySpark Architecture:

Apache Spark works in a master-slave architecture where the master is called “Driver”
And slaves are called “Workers”. When you run a Spark application,
Spark Driver creates a context that is an entry point to your application,
And all operations (transformations and actions) are executed on worker nodes,
And the resources are managed by Cluster Manager.



1. **What is Data Frame?**
   Data Frame is a distributed collection of rows under named columns. Simply represents a table of data with rows and columns.
2. **What is difference between RDD and Data Frame?**

a) **RDD** is Resilient Distributed Datasets.

Resilient means immutable (can't be modified once created)

Fault tolerance refers to the ability of a system (computer, network, cloud cluster, etc.)

Continue operating without interruption when one or more of its components fail

Distributed because it is distributed across cluster or multiple machines.

It doesn't contain schema and deals with structured and unstructured data.

You can move RDD into Data Frame by using toDF ().

Once moved cant revoke original RDD

No inbuilt optimization engine available.

Slower to perform simple grouping and aggregation operations

b) **A Data Frame** is a distributed collection of data organized into named columns.
It is conceptually equal to a table in a relational database.

It contain schema and deals with only structured and semi structured data.

Optimization takes place using catalyst optimizer.

It is faster for exploratory analysis, creating aggregated statistics on large data sets

3. **How to create data frame with schema.**
4.

```
Data = [
          ("ram","","boyapati","36636","M",3000),
          ("Vihaan","boyapati","","40288","M",4000),
        ]
Schema =

StructType ([
  StructField ("firstname",StringType (),True),
  StructField ("middlename",StringType(),True),
  StructField ("lastname",StringType(),True),
  StructField ("id", StringType(), True),
```

ECLASESS.COM

```
    StructField ("gender", StringType(), True),
    StructField ("salary", IntegerType(), True)
            ])

DF = spark.createDataFrame (data=data, schema=schema)
DF.printSchema ()
DF.show (truncate=False)
```

5. What is StructType and StructField in PySpark?

PySpark StructType & StructField classes are used to specify the schema to the Data Frame And creating complex columns like nested struck, array and map columns.
StructType is a collection of StructField's that defines column name, column data type, Boolean to specify if the field can be nullable or not and metadata.

6. **How to read data from file to create data frame**.

```
DF = spark.read.csv ("/src/resources/file.csv")
```

7. How to display schema of data frame.

```
DF.printschema ()
```

8. How to display limited rows and truncate characters in data frame.

```
# Display 2 rows & column values 25 characters
df.show (2, truncate=25)
```

9. How to display selected columns using select columns from data Frame.

```
Data =
    [
  ("Ram","boyapati","USA","CA"),
  ("Shyaam", boyapati","USA","NY")
    ]
Columns = ["first name", "last name","country","state"]
DF = spark.createDataFrame (data = data, schema = columns)
DF.select ("first name", "last name").show ()
```

10. What is collect () and diff between collect () vs select ().

Retrieve all the elements of the dataset (from all nodes) to the driver node.
We should use the collect() on smaller dataset usually after filter(), group() e.t.c.
Retrieving larger datasets results in OutOfMemory error.

ECLASESS.COM

```
data Collect = deptDF.collect()
print(dataCollect)

#Returns value of First Row, First Column which is "Finance"
deptDF.collect()[0][0]
```

deptDF.collect() returns Array of Row type.
deptDF.collect()[0] returns the first element in an array (1st row).
deptDF.collect[0][0] returns the value of the first row & first column.

Difference between select and collect is:

select () is a transformation that returns a new Data Frame and holds the columns that are selected
whereas collect() is an action that returns the entire data set in an Array to the driver.

11. PySpark repartition () is used to increase or decrease the RDD/Data Frame partitions.

```
df2 = df.repartition (6)
Print (df2.rdd.getNumPartitions ())
```

12. PySpark WithColumn () function usage.

Which is used to change the value, convert the datatype of an existing column, and create a new column

Update the Value of an Existing Column

```
df.withColumn ("salary",col("salary")*100).show()
```

Create a Column from an Existing

```
df.withColumn ("CopiedColumn",col("salary")* -1).show()
```

Add a New Column using withColumn()

```
df.withColumn ("Country", lit("USA")).show()
```

Rename Column Name

```
df.withColumnRenamed("gender","sex")
  DF.show (truncate=False)
```

13. PySpark withColumnRenamed

```
df.withColumnRenamed ("dob", DateOfBirth").print Schema ()
```

ECLASESS.COM

14. PySpark Where Filter Function | Multiple Conditions

```
#Using SQL Expression
df.filter("gender == 'M'").show()
#For not equal
df.filter("gender != 'M'").show()
df.filter("gender <> 'M'").show()

//Filter multiple condition
df.filter( (df.state  == "OH") & (df.gender  == "M") ) \
   .show(truncate=False)

#Filter IS IN List values
li=["OH","CA","DE"]
df.filter(df.state.isin(li)).show()

PySpark Filter like and rlike

data2 = [(2,"Michael Rose"),(3,"Robert Williams"),
    (4,"Rames Rose"),(5,"Rames rose")
 ]
df2 = spark.createDataFrame(data = data2, schema = ["id","name"])

df2.filter(df2.name.like("%rose%")).show()
```

15. PySpark group by and aggregate on Data Frame columns.

```
df.groupBy ("department").sum ("salary").show (truncate=False)

df.groupBy ("department").count ()
Calculate the minimum salary of each department using min ()


df.groupBy ("department").min ("salary")
Calculate the maximin salary of each department using max ()


df.groupBy ("department").max ("salary")
Calculate the average salary of each department using avg ()


df.groupBy ("department").avg (“salary")
Calculate the mean salary of each department using mean ()
```

ECLASESS.COM

```
df.groupBy ("department").mean (“salary")

df.groupBy ("department")
   .agg(sum("salary").alias("sum_salary"),
      avg("salary").alias("avg_salary"),
      sum("bonus").alias("sum_bonus"),
      max("bonus").alias("max_bonus")
    )
   .show (truncate=False)
```

16. What are different kind of joins in PySpark

Inner Join DataFrame

Full Outer Join DataFrame

Left Outer Join DataFrame

Right Outer Join DataFrame

Left Anti Join DataFrame

Left Semi Join DataFrame

Self-Join Data Frame

Create two data Frames to perform several kind of joins.

```
emp = [(1,"Smith",-1,"2018","10","M",3000), \
   (2,"Rose",1,"2010","20","M",4000), \
   (3,"Williams",1,"2010","10","M",1000), \
   (4,"Jones",2,"2005","10","F",2000), \
   (5,"Brown",2,"2010","40","",-1), \
    (6,"Brown",2,"2010","50","",-1) \
 ]
empColumns = ["emp_id","name","superior_emp_id","year_joined", \
     "emp_dept_id","gender","salary"]
```


ECLASESS.COM

```
empDF = spark.createDataFrame(data=emp, schema = empColumns)
empDF.printSchema()
empDF.show(truncate=False)


dept = [("Finance",10), \
   ("Marketing",20), \
   ("Sales",30), \
   ("IT",40) \
 ]
deptColumns = ["dept_name","dept_id"]
deptDF = spark.createDataFrame(data=dept, schema = deptColumns)
deptDF.printSchema()
deptDF.show(truncate=False)
```

Inner Join: Giving only matched records from both sides.

```
empDF.join(deptDF,empDF.emp_dept_id ==  deptDF.dept_id,"inner") \
     .show(truncate=False)
```

Left Join: Giving all records from left side and only matching records from right side,

If any record not matching then null value will return from right side value.

```
empDF.join(deptDF,empDF("emp_dept_id") ==  deptDF("dept_id"),"left")
     .show(false)
empDF.join(deptDF,empDF("emp_dept_id") ==  deptDF("dept_id"),"leftouter")
     .show (false)
```

Right outer Join: Giving all records from Right side and only matching records from left side,

If any record not matching then null value will return from left side value

```
empDF.join (deptDF,empDF.emp_dept_id ==  deptDF.dept_id,"right") \
     .show (truncate=False)
empDF.join (deptDF, empDF.emp_dept_id ==  deptDF.dept_id,"rightouter") \
```

ECLASESS.COM

```
.show (truncate=False)
```

Full outer Join: Giving all records from both sides,

If any values not matched corresponding records will return nulls

```
empDF.join (deptDF, empDF.emp_dept_id == deptDF.dept_id,"outer") \
  .show (truncate=False)
empDF.join (deptDF, empDF.emp_dept_id ==  deptDF.dept_id,"full") \
  .show (truncate=False)
empDF.join (deptDF,empDF.emp_dept_id ==  deptDF.dept_id,"fullouter") \
  .show (truncate=False)
```

Left Semi Join: leftsemi join is similar to inner join difference being leftsemi join returns

All columns from the left side and ignores all columns from the right side.

```
empDF.join (deptDF, empDF.emp_dept_id == deptDF.dept_id,"leftsemi") \
    .show (truncate=False)
```

Left Anti join: leftanti join returns only columns from the left side for non-matched records.

```
empDF.join(deptDF,empDF.emp_dept_id ==  deptDF.dept_id,"leftanti") \
      .show (truncate=False)
```

Self-Join: There is no self-join type available, we can use any of the above-explained join types

To join DataFrame to itself. below example use inner self join.

```
empDF.alias("emp1").join(empDF.alias("emp2"), \
   Col("emp1.superior_emp_id") == col("emp2.emp_id"),"inner") \
   .select (col("emp1.emp_id"),col("emp1.name"), \
     col("emp2.emp_id").alias("superior_emp_id"), \
     col("emp2.name").alias("superior_emp_name")) \
   .show(truncate=False)
```

How to join Multiple Data Frames.

```
df1.join(df2,df1.id1 == df2.id2,"inner") \
  .join(df3,df1.id1 == df3.id3,"inner")
```

ECLASESS.COM

How to join two data frames by converting to SQL temp tables.

```
empDF.createOrReplaceTempView("EMP")
deptDF.createOrReplaceTempView("DEPT")

joinDF = spark.sql("select * from EMP e, DEPT d where e.emp_dept_id == d.dept_id") \
  .show(truncate=False)

joinDF2 = spark.sql("select * from EMP e INNER JOIN DEPT d ON e.emp_dept_id == d.dept_id") \
  .show(truncate=False)
```

17. PySpark union() and union All() transformations are used to merge two or more

DataFrame's of the same schema or structure.

unionAll() is deprecated since Spark "2.0.0" version and replaced with union()

```
simpleData = [("James","Sales","NY",90000,34,10000), \
   ("Michael","Sales","NY",86000,56,20000), \
   ("Robert","Sales","CA",81000,30,23000), \
   ("Maria","Finance","CA",90000,24,23000) \
 ]

Columns= ["employee_name","department","state","salary","age","bonus"]
df = spark.createDataFrame(data = simpleData, schema = columns)
df.printSchema()
df.show(truncate=False)

simpleData2 = [("James","Sales","NY",90000,34,10000), \
   ("Maria","Finance","CA",90000,24,23000), \
   ("Jen","Finance","NY",79000,53,15000), \
```

ECLASESS.COM

```
  ("Jeff","Marketing","CA",80000,25,18000), \
  ("Kumar","Marketing","NY",91000,50,21000) \
 ]
columns2= ["employee_name","department","state","salary","age","bonus"]

df2 = spark.createDataFrame(data = simpleData2, schema = columns2)

df2.printSchema()
df2.show(truncate=False)

unionDF = df.union(df2)
unionDF.show(truncate=False)
disDF = df.union(df2).distinct()
disDF.show(truncate=False)

UnionAllDF = df.unionAll (df2)
unionAllDF.show (truncate=False)
```

18. PySpark – Distinct to Drop Duplicate Rows.

PySpark distinct () function is used to drop/remove the duplicate rows (all columns) from Data Frame and drop Duplicates () is used to drop rows based on selected (one or multiple) columns.

```
DistinctDF = df.distinct ()

df2 = df.dropDuplicates ()

DropDisDF = df.dropDuplicates (["department", "salary"])
```

19. PySpark coalesce () is used to only decrease the number of partitions in an efficient way.

```
df3 = df.coalesce (2)
Print (df3.rdd.getNumPartitions ())
```

ECLASESS.COM

20. Broadcast variables.

Broadcast variables are read-only shared variables that are cached and available on all nodes in a cluster in-order to access or use by the tasks.

Ex: you are getting a two-letter country state code in a file and you wanted to transform it to full state name, (for example CA to California, NY to New York etc.) by doing a lookup to reference mapping.

Instead of distributing this information along with each task over the network (overhead and time consuming), we can use the broadcast variable to cache this lookup info on each machine and tasks use this cached info while executing the transformations.

States = {"NY": "New York", "CA": "California", "FL": "Florida"}
Broadcast States = spark.sparkContext.broadcast (states)

21. What is Map and MapPartiton function?

Map () transformation is used to apply any complex operations like adding a column, Updating a column, transforming the data etc., the output of map transformations Would always have the same number of records as input.

Note1: Data Frame doesn't have map () transformation to use with Data Frame
Hence you need to Data Frame to RDD first.

Note2: If you have a heavy initialization use PySpark map Partitions () transformation
Instead of map (), as with map Partitions () heavy initialization executes
Only once for each partition instead of every record.

Lambda Function:

A lambda function is a small anonymous function.
When we require a nameless function for a short period of time. In Python
Normal functions are defined using the def keyword, anonymous functions are defined using the lambda keyword.

```
Data = ["Project","Gutenberg's","Alice's","Adventures",
"in","Wonderland","Project","Gutenberg's","Adventures",
"in","Wonderland","Project","Gutenberg's"]

Rdd=spark.sparkContext.parallelize (data)

rdd2=rdd. Map (lambda x: (x, 1))
```

ECLASESS.COM

```
For element in rdd2.collect ():
Print (element)
```

22. What are sample and sample by and takesample?

Working as a Data Scientist or Data analyst you often required to analyze a large dataset/file with billions or trillions of records, processing these large datasets takes some time hence during the analysis phase it is recommended to use a random subset sample from the large files.

23. How to fill null or empty values in data frame using fill function.

Used to replace NULL/None values on all or selected multiple Data Frame columns with

Either zero (0), empty string, space, or any constant literal values

```
#Replace 0 for null for all integer columns

df.fillna (value=0).show ()

#Replace 0 for null on only population column

df.fillna (value=0, subset= ["population"]).show ()

df.fillna ("").show (false)
```

24. How to pivot Data using pivot function.

Pivot() function is used to rotate/transpose the data from one column into multiple Data frame columns.

```
data = [("Banana",1000,"USA"), ("Carrots",1500,"USA"), ("Beans",1600,"USA"), \
    ("Orange",2000,"USA"),("Orange",2000,"USA"),("Banana",400,"China"), \
    ("Carrots",1200,"China"),("Beans",1500,"China"),("Orange",4000,"China"), \
    ("Banana",2000,"Canada"),("Carrots",2000,"Canada"),("Beans",2000,"Mexico")]

columns= ["Product","Amount","Country"]
df = spark.createDataFrame(data = data, schema = columns)
df.printSchema()
df.show(truncate=False)
```

Pivot:

```
df.groupBy ("Product").pivot ("Country").sum ("Amount")
pivotDF.printSchema ()
pivotDF.show (truncate=False)
```

ECLASESS.COM

Unpivot:

```
unpivotExpr = "stack(3, 'Canada', Canada, 'China', China, 'Mexico', Mexico) as (Country,Total)"
unPivotDF = pivotDF.select("Product", expr(unpivotExpr)) \ .where("Total is not null")
unPivotDF.show(truncate=False)
```

25. Pyspark partitions in data frames.

PySpark partition is a way to split a large dataset into smaller datasets based on one or more partition keys
PySpark creates the Data Frame with a certain number of partitions in memory
This is one of the main advantages of PySpark Data Frame over Pandas Data Frame.
Transformations on partitioned data run faster as they execute transformations parallelly for each partition

PySpark supports partition in two ways;

Partition in memory (Data Frame)
Partition on the disk (File system).

```
#partitionBy ()
df.write.option ("header", True) \
      .partitionBy ("state") \
      .mode ("overwrite") \
      .csv ("/tmp/zipcodes-state")

#partitionBy() multiple columns
df.write.option("header",True) \
      .partitionBy("state","city") \
      .mode("overwrite") \
      .csv("/tmp/zipcodes-state")
```

26. Maptype Data Type.

PySpark Map Type (also called map type) is a data type to represent
Python Dictionary (dict) to store key-value pair.

```
DataDictionary = [
('James',{'hair':'black','eye':'brown'}),
('Michael',{'hair':'brown','eye':None}),
('Robert',{'hair':'red','eye':'black'}),
('Washington',{'hair':'grey','eye':'grey'}),
('Jefferson',{'hair':'brown','eye':''})
```


ECLASESS.COM

```
    ]
df = spark.createDataFrame(data=dataDictionary, schema = schema)
df.printSchema()
df.show(truncate=False)


df3=df.rdd.map(lambda x: \
   (x.name,x.properties["hair"],x.properties["eye"])) \
   .toDF(["name","hair","eye"])
df3.printSchema()
df3.show()
```

27. Windows Functions Row Number (), Rank() , Dense Rank().

Row Number: Generating sequence number based on partition specified.

```
simpleData = (("James", "Sales", 3000), \
   ("Michael", "Sales", 4600),  \
   ("Robert", "Sales", 4100),   \
   ("Maria", "Finance", 3000),  \
   ("James", "Sales", 3000),    \
   ("Scott", "Finance", 3300),  \
   ("Jen", "Finance", 3900),    \
   ("Jeff", "Marketing", 3000), \
   ("Kumar", "Marketing", 2000),\
   ("Saif", "Sales", 4100) \
 )

columns= ["employee_name", "department", "salary"]

df = spark.createDataFrame(data = simpleData, schema = columns)

df.printSchema()
df.show(truncate=False)

from pyspark.sql.window import Window
from pyspark.sql.functions import row_number
windowSpec  = Window.partitionBy("department").orderBy("salary")

df.withColumn("row_number",row_number().over(windowSpec)) \
   .show(truncate=False)

from pyspark.sql.functions import rank
```

ECLASESS.COM

```
df.withColumn("rank",rank().over(windowSpec)) \
   .show()

from pyspark.sql.functions import dense_rank
df.withColumn("dense_rank",dense_rank().over(windowSpec)) \
   .show()

from pyspark.sql.functions import percent_rank
df.withColumn("percent_rank",percent_rank().over(windowSpec)) \
   .show()

from pyspark.sql.functions import ntile
df.withColumn("ntile",ntile(2).over(windowSpec)) \
   .show()

from pyspark.sql.functions import cume_dist
df.withColumn("cume_dist",cume_dist().over(windowSpec)) \
  .show()

from pyspark.sql.functions import lag
df.withColumn("lag",lag("salary",2).over(windowSpec)) \
     .show()

from pyspark.sql.functions import lead
df.withColumn("lead",lead("salary",2).over(windowSpec)) \
   .show()

windowSpecAgg  = Window.partitionBy("department")
from pyspark.sql.functions import col,avg,sum,min,max,row_number
df.withColumn("row",row_number().over(windowSpec)) \
  .withColumn("avg", avg(col("salary")).over(windowSpecAgg)) \
  .withColumn("sum", sum(col("salary")).over(windowSpecAgg)) \
  .withColumn("min", min(col("salary")).over(windowSpecAgg)) \
  .withColumn("max", max(col("salary")).over(windowSpecAgg)) \
  .where(col("row")==1).select("department","avg","sum","min","max") \
  .show ()
```

ECLASESS.COM

28. What is difference between PySpark Data Frame and Pandas Data Frame?

PySpark run faster than Pandas due to its distributed nature and parallel execution on multiple cores and machines.

Pandas run operations on a single node.
After processing data in PySpark we would need to convert it back to Pandas Data Frame for a further procession with Machine Learning application or any Python applications.

Panda's data frames are in-memory and single-server, so their size is limited by your server memory and you will process them with the power of a single server.

29. How to run one notebook from another notebook in PySaprk

%run <path of the notebook that has to be run>

30. PySpark performance improvements.

- Prefer using Dataset/Data Frame over RDD as Dataset and Data Frame's includes Several optimization modules to improve the performance of the Spark workloads.

  Use repartition() when you wanted to increase the number of partitions.

  Using cache() and persist() methods, Spark provides an optimization mechanism to store the intermediate computation of a Spark DataFrame so they can be reused in subsequent action

ECLASESS.COM

Azure Cosmos DB

Why Cosmos DB: To overcome problems like setting up own globally distributed centers with traditional relational database system like SQL Server.

Need More Efforts

More expensive

Time consumption

Need Many Teams to manage

Need Many Data centers

In 2015 Slowly Microsoft Started to overcome this issue by introducing Document Db with json Document. We can able to write SQL queries on top of this document db.

In 2017 Microsoft rebranded Document DB as Cosmos DB for Globally distribution and auto scalability.

What is Cosmos DB?

Cosmos DB is globally distributed multi model database for mission critical applications.

Features of Cosmos DB

Microsoft Azure providing this as PAAS.

This is No SQL Database

Globally Distributed –Automatic replication across multiple data centers

Fully managed

Unlimited storage and throughput

Multi Model- key pairs, Graph, Column, Document

Multi languages – java, .net, Python, NodeJS

No schema and Index are defined

ECLASESS.COM

Always on

99.999 SLA

Automatic Indexing

There are four different kind of No SQL database involved inside Cosmos db.

7. Key-value
8. Column
9. Graph
10. Document

There are 4 API supporting cosmos db based on type of db.

1. Table API Supports- Key Value Type DB
2. Casandra API Supports- Column Type DB
3. Gremlin API Supports - Graph Type DB
4. Mango Db or SQL API Supports- Document Type DB

Choose the appropriate API for Azure Cosmos DB

https://docs.microsoft.com/en-us/learn/modules/choose-api-for-cosmos-db/

Data Consistency in Cosmos DB:

Consistency

Replication comes with a choice of consistency. So, when one instance of your app writes data to a write-region, Azure needs to replicate this data to other regions.

Azure Cosmos DB offers 5 type of consistency levels. It means, you need to select how Azure should replicate your data between your Azure Cosmos DB regions. Let's see what those consistency levels are:

Strong

In this model, there are no dirty reads. It means, when a data is updated, everybody will read the old value until the data is replicated to all regions. This is the slowest option.

Bounded Staleness

In this option, you can define period of time or update count for the staleness of your data. You can say that, no dirty reads for 1 minute or no dirty reads for data updated more than 5 times. When you set the time option to 0, it will be exactly same as Strong consistency option.

Session

In this option, no dirty reads are possible for writers but dirty reads are possible for readers. This is the default option. So, if you are the one writing the data, you can read that data. But for others, they can read stale data for a while.

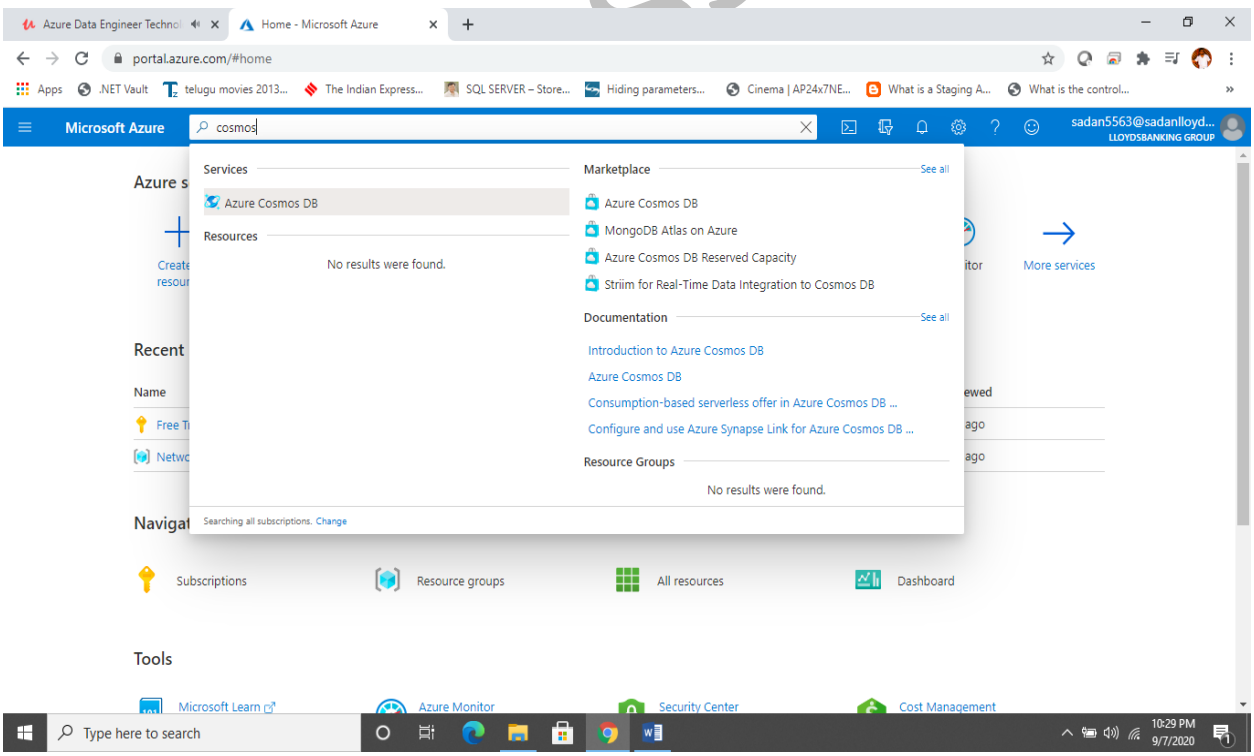
Consistent Prefix

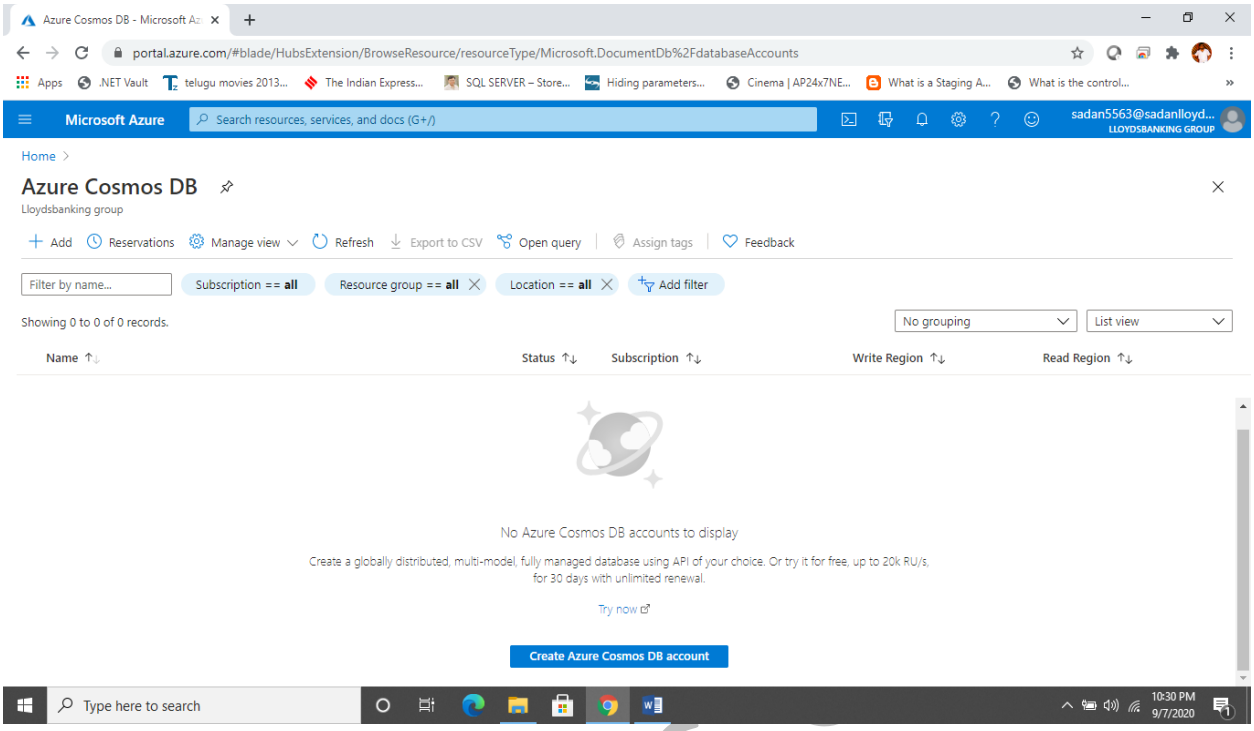
In this option dirt-reads are possible but they are always on order. So, if a data is updated with the values 1, 2, 3 in order, readers always see the updated data in this order. No one will see the value 3 before 2.

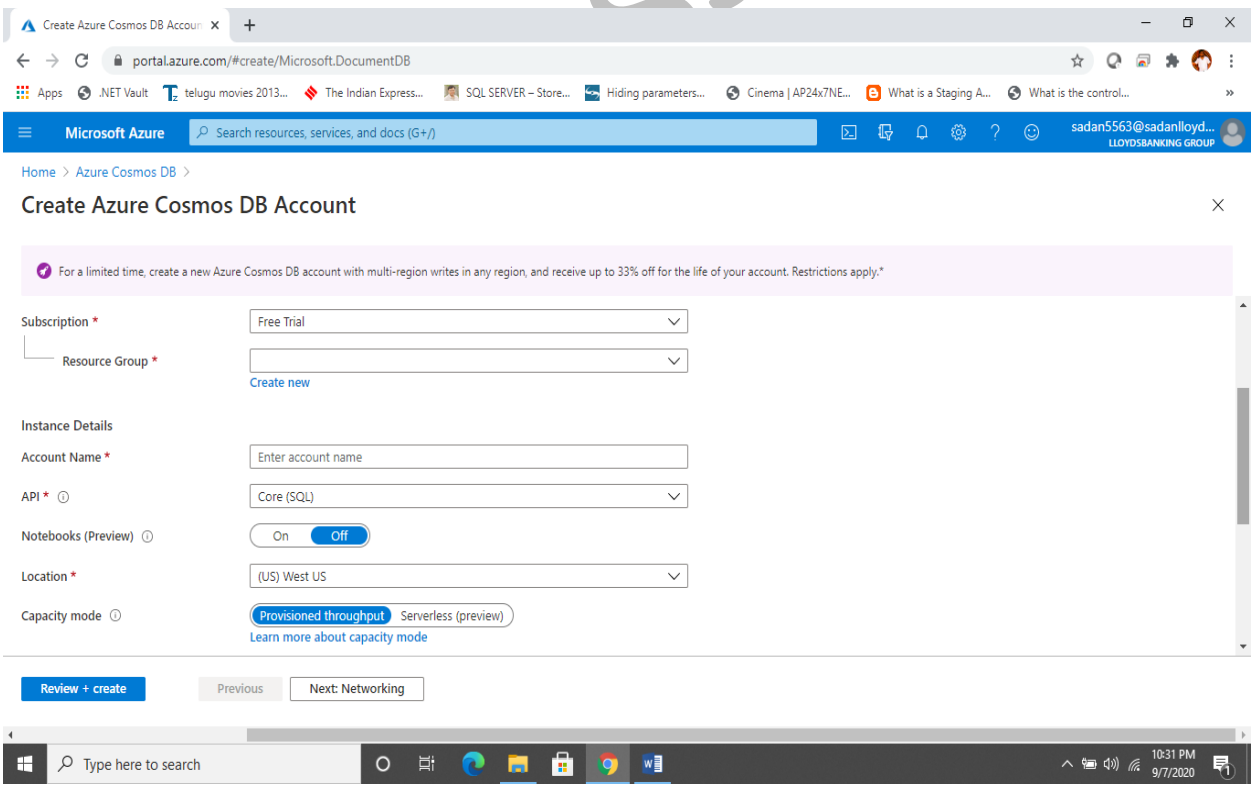
Eventual

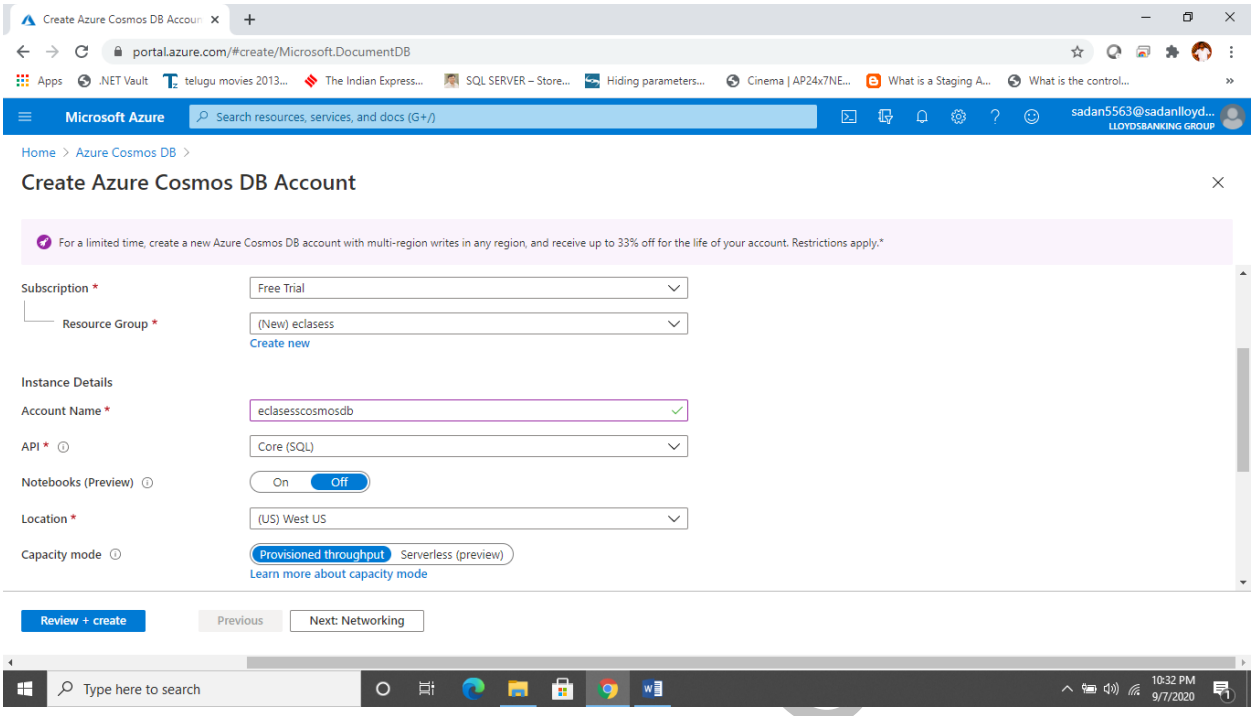
In this option, dirty reads are possible and there is no guarantee of order. So, if a data is updated with the values 1, 2, 3 in order, a reader can see value 3 before seeing value 2. But, this is the fastest option

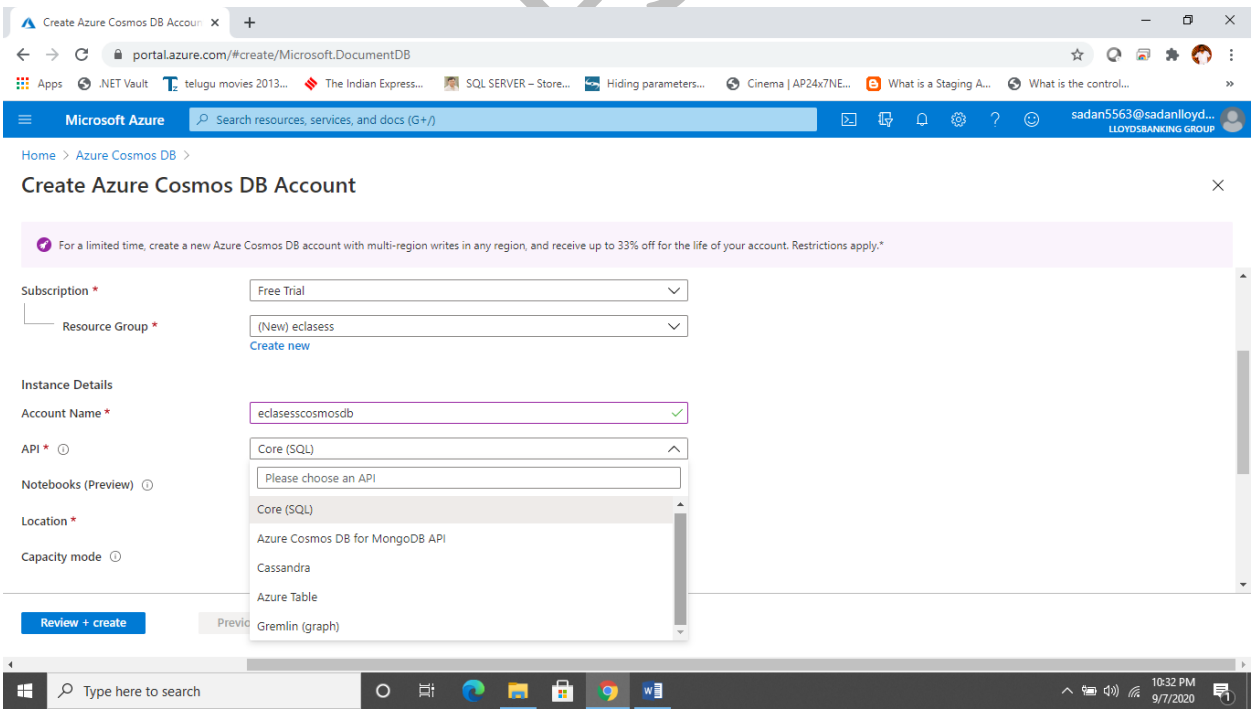
How to Create Cosmos DB:

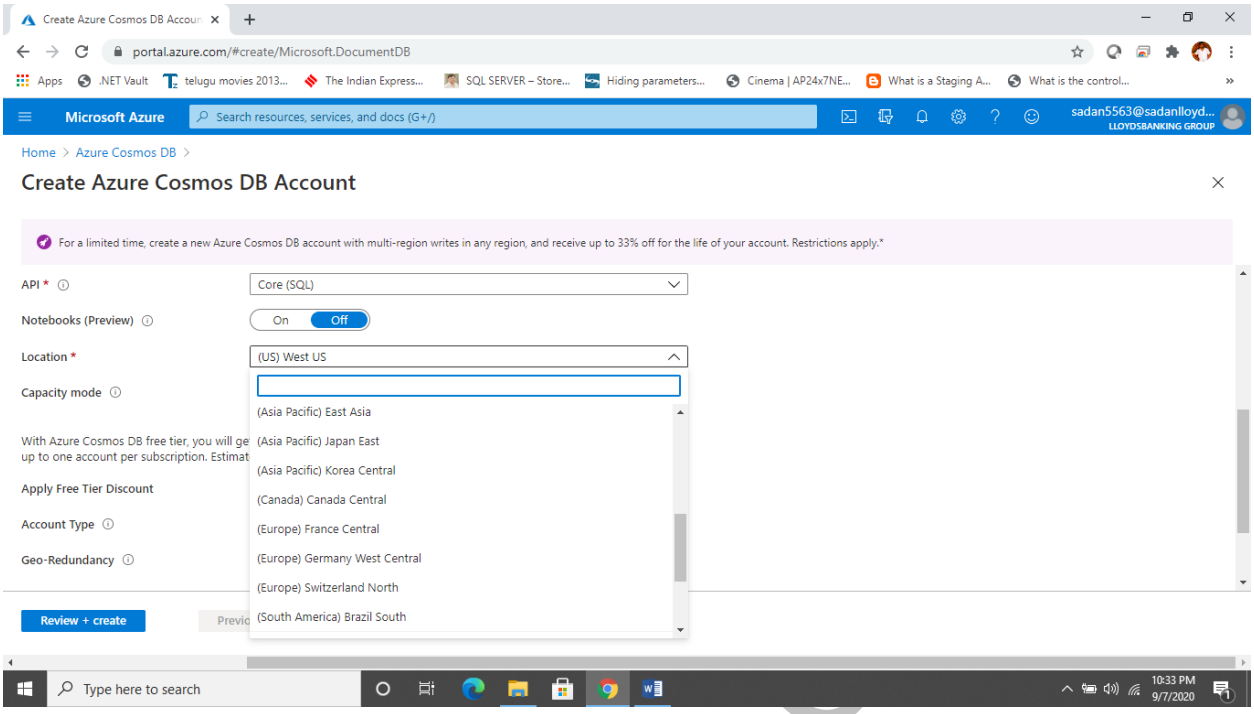


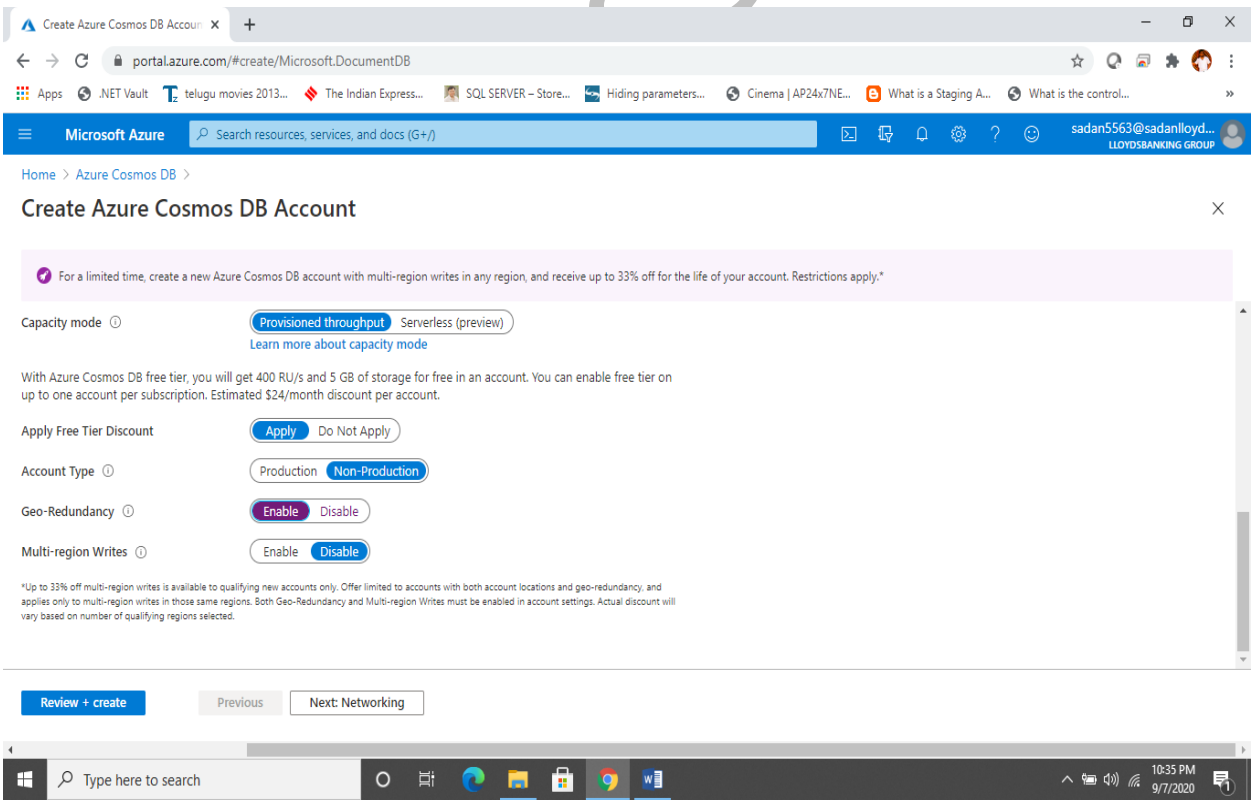

Azure Cosmos DB - Microsoft Az
portal.azure.com/#blade/HubsExtension/BrowseResource/resourceType/Microsoft.DocumentDb%2FdatabaseAccounts
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home
Azure Cosmos DB
Lloydsbanking group
Add
Reservations
Manage view
Refresh
Export to CSV
Open query
Assign tags
Feedback
Filter by name...
Subscription == all
Resource group == all
Location == all
Add filter
Showing 0 to 0 of 0 records.
No grouping
List view
Name
Status
Subscription
Write Region
Read Region
No Azure Cosmos DB accounts to display
Create a globally distributed, multi-model, fully managed database using API of your choice. Or try it for free, up to 20k RU/s, for 30 days with unlimited renewal.
Try now
Create Azure Cosmos DB account
Type here to search
10:30 PM
9/7/2020

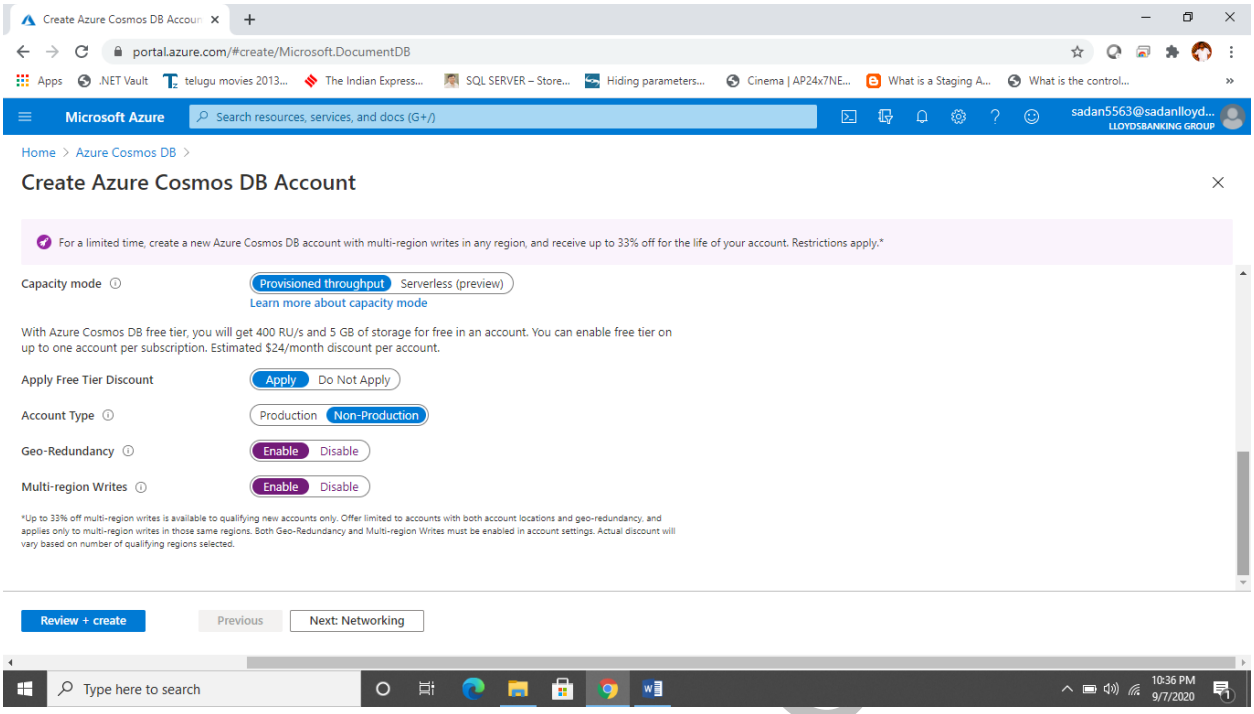

Create Azure Cosmos DB Accour
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home
Azure Cosmos DB
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
Subscription *
Free Trial
Resource Group *
Create new
Instance Details
Account Name *
Enter account name
API *
Core (SQL)
Notebooks (Preview)
On
Off
Location *
(US) West US
Capacity mode
Provisioned throughput
Serverless (preview)
Learn more about capacity mode
Review + create
Previous
Next: Networking
Type here to search
10:31 PM
9/7/2020

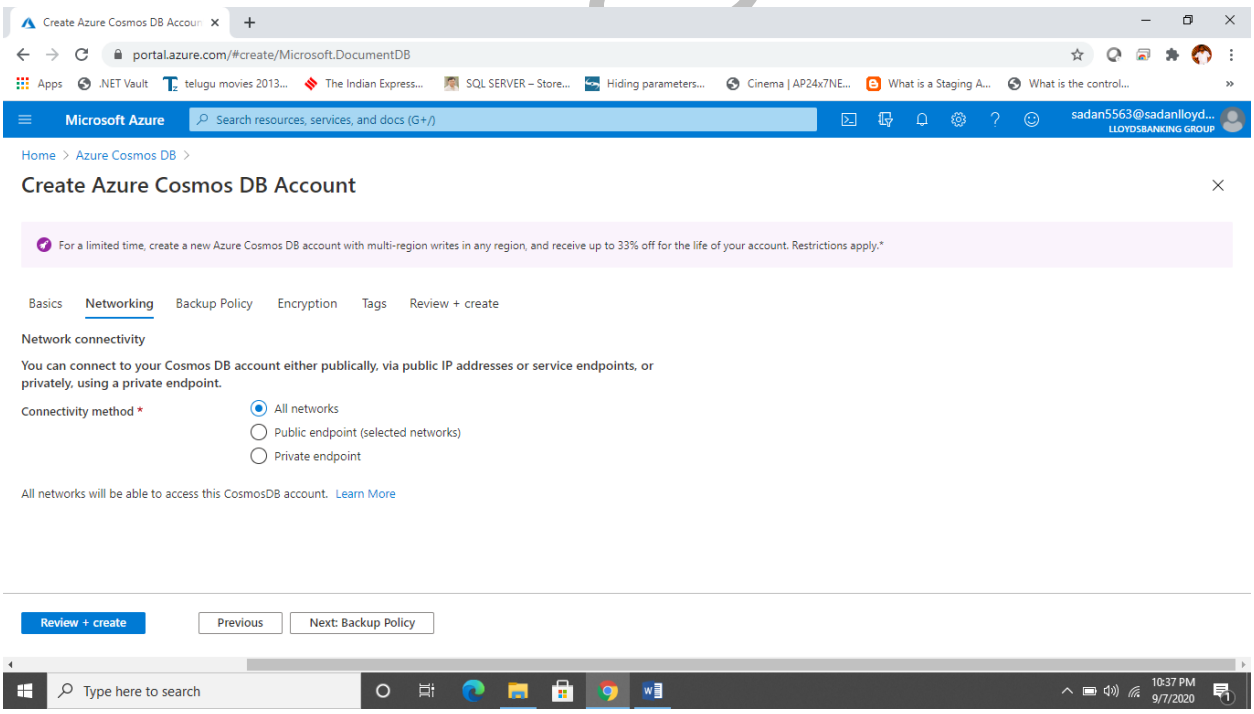

Create Azure Cosmos DB Accoun
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
Subscription *
Free Trial
Resource Group *
(New) eclasess
Create new
Instance Details
Account Name *
eclasesscosmosdb
API *
Core (SQL)
Notebooks (Preview)
On
Off
Location *
(US) West US
Capacity mode
Provisioned throughput
Serverless (preview)
Learn more about capacity mode
Review + create
Previous
Next: Networking
Type here to search
10:32 PM
9/7/2020

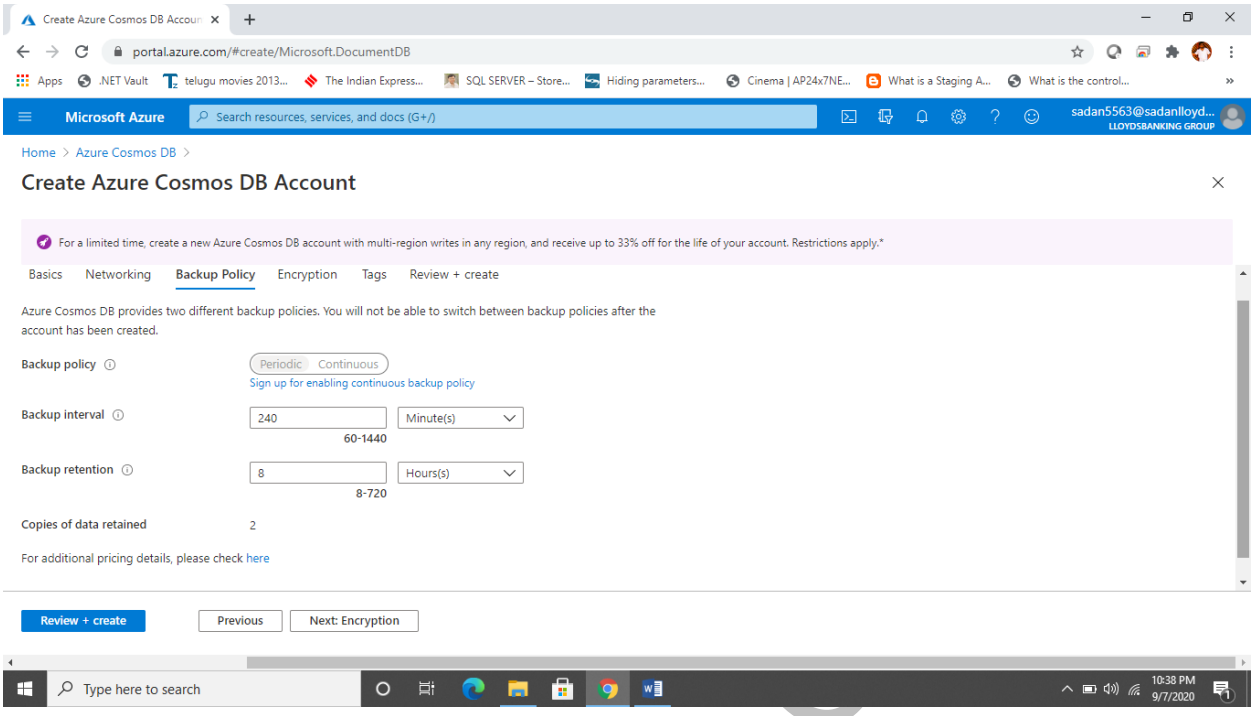

Create Azure Cosmos DB Accoun
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
Subscription *
Free Trial
Resource Group *
(New) eclasess
Create new
Instance Details
Account Name *
eclasesscosmosdb
API *
Core (SQL)
Please choose an API
Core (SQL)
Azure Cosmos DB for MongoDB API
Cassandra
Azure Table
Gremlin (graph)
Notebooks (Preview)
Location *
Capacity mode
Review + create
Type here to search
10:32 PM
9/7/2020

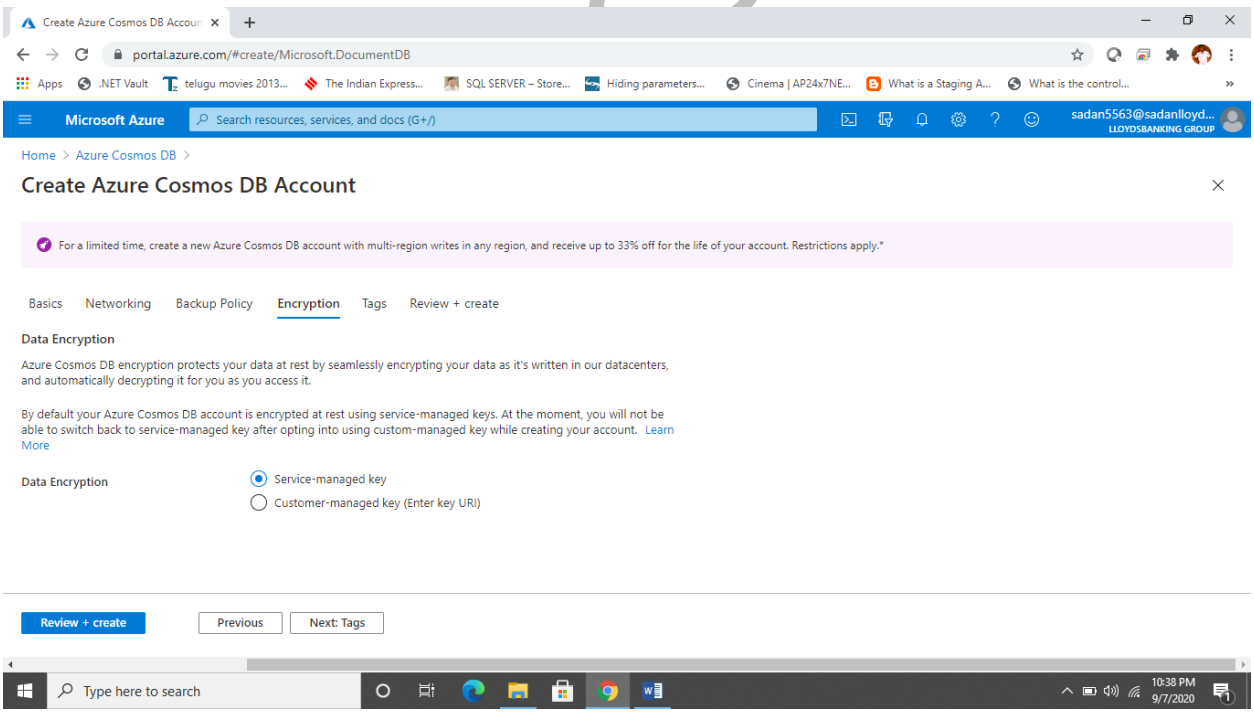

Create Azure Cosmos DB Accoun
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
API *
Core (SQL)
Notebooks (Preview)
On
Off
Location *
(US) West US
Capacity mode
(Asia Pacific) East Asia
(Asia Pacific) Japan East
(Asia Pacific) Korea Central
(Canada) Canada Central
(Europe) France Central
(Europe) Germany West Central
(Europe) Switzerland North
(South America) Brazil South
Apply Free Tier Discount
Account Type
Geo-Redundancy
Review + create
Type here to search
10:33 PM
9/7/2020

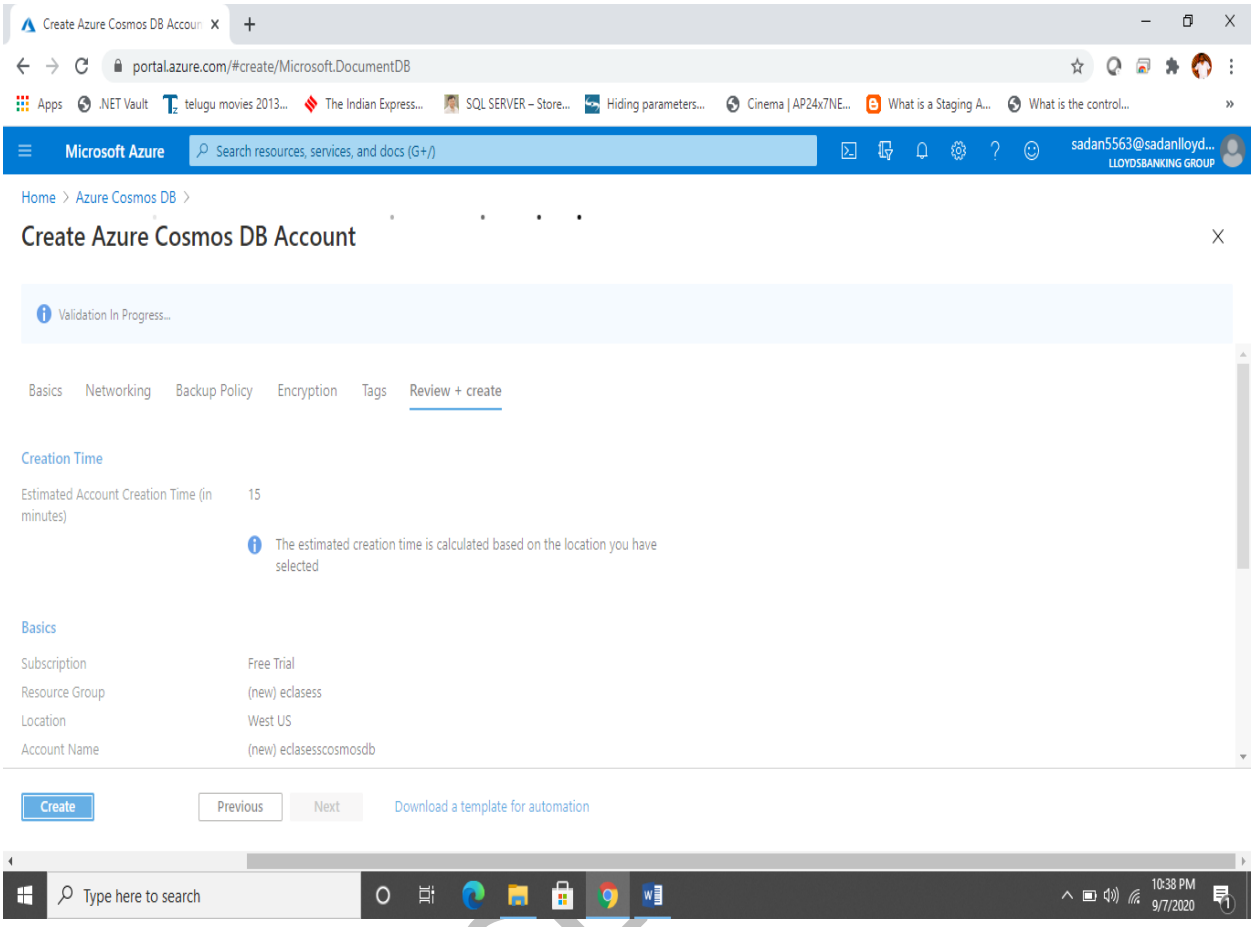

Create Azure Cosmos DB Accoun
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
Capacity mode
Provisioned throughput
Serverless (preview)
Learn more about capacity mode
With Azure Cosmos DB free tier, you will get 400 RU/s and 5 GB of storage for free in an account. You can enable free tier on up to one account per subscription. Estimated $24/month discount per account.
Apply Free Tier Discount
Apply
Do Not Apply
Account Type
Production
Non-Production
Geo-Redundancy
Enable
Disable
Multi-region Writes
Enable
Disable
*Up to 33% off multi-region writes is available to qualifying new accounts only. Offer limited to accounts with both account locations and geo-redundancy, and applies only to multi-region writes in those same regions. Both Geo-Redundancy and Multi-region Writes must be enabled in account settings. Actual discount will vary based on number of qualifying regions selected.
Review + create
Previous
Next: Networking
Type here to search
10:35 PM
9/7/2020


Create Azure Cosmos DB Account
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
Capacity mode
Provisioned throughput
Serverless (preview)
Learn more about capacity mode
With Azure Cosmos DB free tier, you will get 400 RU/s and 5 GB of storage for free in an account. You can enable free tier on up to one account per subscription. Estimated $24/month discount per account.
Apply Free Tier Discount
Apply
Do Not Apply
Account Type
Production
Non-Production
Geo-Redundancy
Enable
Disable
Multi-region Writes
Enable
Disable
*Up to 33% off multi-region writes is available to qualifying new accounts only. Offer limited to accounts with both account locations and geo-redundancy, and applies only to multi-region writes in those same regions. Both Geo-Redundancy and Multi-region Writes must be enabled in account settings. Actual discount will vary based on number of qualifying regions selected.
Review + create
Previous
Next: Networking
Type here to search
10:36 PM
9/7/2020


Create Azure Cosmos DB Account
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
Basics
Networking
Backup Policy
Encryption
Tags
Review + create
Network connectivity
You can connect to your Cosmos DB account either publically, via public IP addresses or service endpoints, or privately, using a private endpoint.
Connectivity method *
All networks
Public endpoint (selected networks)
Private endpoint
All networks will be able to access this CosmosDB account. Learn More
Review + create
Previous
Next: Backup Policy
Type here to search
10:37 PM
9/7/2020


Create Azure Cosmos DB Account
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
Basics Networking Backup Policy Encryption Tags Review + create
Azure Cosmos DB provides two different backup policies. You will not be able to switch between backup policies after the account has been created.
Backup policy
Periodic Continuous
Sign up for enabling continuous backup policy
Backup interval
240
Minute(s)
60-1440
Backup retention
8
Hours(s)
8-720
Copies of data retained
2
For additional pricing details, please check here
Review + create
Previous
Next: Encryption
Type here to search
10:38 PM
9/7/2020


Create Azure Cosmos DB Account
portal.azure.com/#create/Microsoft.DocumentDB
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
For a limited time, create a new Azure Cosmos DB account with multi-region writes in any region, and receive up to 33% off for the life of your account. Restrictions apply.*
Basics Networking Backup Policy Encryption Tags Review + create
Data Encryption
Azure Cosmos DB encryption protects your data at rest by seamlessly encrypting your data as it's written in our datacenters, and automatically decrypting it for you as you access it.
By default your Azure Cosmos DB account is encrypted at rest using service-managed keys. At the moment, you will not be able to switch back to service-managed key after opting into using custom-managed key while creating your account. Learn More
Data Encryption
Service-managed key
Customer-managed key (Enter key URI)
Review + create
Previous
Next: Tags
Type here to search
10:38 PM
9/7/2020


Create Azure Cosmos DB Accour
portal.azure.com/#create/Microsoft.DocumentDB
Apps .NET Vault telugu movies 2013... The Indian Express... SQL SERVER – Store... Hiding parameters... Cinema | AP24x7NE... What is a Staging A... What is the control...
Microsoft Azure
Search resources, services, and docs (G+/)
sadan5563@sadanlloyd...
LLOYDSBANKING GROUP
Home > Azure Cosmos DB >
Create Azure Cosmos DB Account
Validation In Progress...
Basics Networking Backup Policy Encryption Tags Review + create
Creation Time
Estimated Account Creation Time (in minutes) 15
The estimated creation time is calculated based on the location you have selected
Basics
Subscription Free Trial
Resource Group (new) eclasess
Location West US
Account Name (new) eclasesscosmosdb
Create
Previous
Next
Download a template for automation
Type here to search
10:38 PM
9/7/2020

ECLASESS

Interview questions

- What is cloud computing
- What are different kind of deployment modes available?
- private
- public
- hybrid
- what are different category of services available in cloud
- iaas---Infra structer as
- paas
- saas
- what is storage account,different kind of storage account
- blob
- Table
- Quese or messeges
- FileShare
- what are different kind of blob types
- Block blob
- Page blob
- Append blob
- security of Blob or storage account
- access keys
- connection strings
- SAS(Shared Access Signatures)
- what is advantage of Table Stoarge
- what is polybase
- what are steps needs to follow to implement polybase
- What is advantage of polybase
- What are different services available to deploye sql
- Azure SQL Database
- what is elastic pool
- what are Pricing Tiers
    1. DTU

ECLASESS.COM

2. vcore processers

- How to implement security
  1. window firewall-IP Address Needs to be added
  2. SQL Authentication---username and password

- What is difference between Azure SQL and Synapse DWH.

Azure SQL Database vs Azure Synapse DWH

1.	Diff Between Azure SQL Database	vs Azure SQL DW(Synapse Analytics)
2.	Not supporting MPP	MPP--massive parlele processing
3.	4 TB(Tera Bytes)	1 PB
4.	Can't Pause Service.	Can pause Service
5.	OLTP(Insert,Update,Delete)	OLAP(Analytics)
6.	polybase not supporting	supporting
7.	Dynamic data masking	Not supporting
8.	30000 connurent connection	1024
9.	concurent quries 6400	32
10.	not supporting cross query	
11.	(3 part) not supporting	supporting (3 part naming)

ECLASESS.COM

- **Data warehousing interview questions:**
- ----------
- https://youtu.be/FpYqPC4F5ww

- what is data mart
- whata is OLTP
- what is DW
- Whata is OLAP
- What is ER Modeling
- What is Master and transaction tables
- what is Primary foreign keys
- what is Dimensional Modleing
- What is Star Schema and Snowflake schema
- what is Surrogatekey
- what are different kind of dimensions
- what are different kind of Facts
- what is conformed dimesnion
- what is role playing dimension
- what is junk dimension
- what is degenerated dimension
- what is Additive measure
- what is semi additive measure
- what is non additive measure
- what is Fatless Fact table

Interview Questions of Data Lake:

- what is Data lake
- What is Diff between data Lake and Blob storage
- What are diff between data lake Gen1 and Gen2
- What is Data Lake analytics
- What is U- SQL Programing
- How to run U-SQL Scrpts by using Data Fcatory
- What are security Methods following While connecting to DataLake...
- What is App registration and principal identity

ECLASESS.COM

- what are different methods following while uploading files into datalake
- what is Datalake catelog
- which activity are using to run datalake analytics U SQL Scripts

Interview Questions of Data Factory:

- What is difference between ADF V1 and V2.

1. v1 doesnt conatin azure devops git repository integration
2. V1 doesnt conatin many of activites
3. v1 doesnt conatins CICD methodoly of deployment

- what are Main Components of Azure Data Factory or Buliding blocks of ADF

1. Pipeline--Designing of ETL process by using different activates and defining execution flow.
2. Activity---to perform a specific operation like copy data ex: Copy Data Activity
    a. to delete file existing in location Ex: Delete Activity
3. Dataset---Preperation of Data to move and transform and load
4. Linked Service---Connection Information to any data source
5. Integration RunTimes---compute infrastructer to execute activites and performing data movements from different networks
6. Triggers---schedule of pielines on specific time automatically, we need triggers
7. Parameters---if u want to pass any input value to pipeline while executing from outside.
8. Variables---if u wnat to store or assign value temporarly in pipeline execution.

- What is integration runtime?

Integration runtimes providing compute infrstruer to execute activites to perform certain operations.

1. Auto resolve integration runtime---to connect cloud services.
2. Self-Hosted Integration runtime---to connect to private and virtual networks.
3. SSIS integration runtimes---lift and shifting of ssis packages into azure data factory

- How to load multiple Files from Blob to SQL Server

ECLASESS.COM

- How to load multiple Files from Data Lake to SQL Server
- How to load Data From Azure SQL Database to SQL Server
- How to load Data From SQL Server On premises to SQL Server

- How to perform incremental Loading
- How to perform Slowly changing Dimension mechanism
- How to perform Looping of Files

- How to secure data connection information using Azure keyvaluts
- How to send email notification using logic apps
- How to lift and shifting of SSIS Packages into adf
- How to integrate with Git Reposotiory
- How to Schedule Pipelines and different kind of Triggers.
- To Run Azure Data Factory pipelines, we can use any of one triggers.

  There are 3 different kind of triggers in ADF

  1. Normal scheduled triggers----This is normal schedule trigger as per scheduled timing recursively executing pipelines.
  2. Event Based Triggers---This type of trigger will executing pipelines based on event specified like file uploaded or deleted in blob.
  3. Tumbling Window----executing pipelines in periodic time interval from a specified start time,
     Tumbling windows are a series of fixed-sized, non-overlapping, and contiguous time intervals.
     Whenever executing pipelines in historical manner, we can use
- How to Monitor
- How to deploye pipelines using CICD
- What are different kind of integration runtimes , how to setup.
- What are different kind of activites.

1. Getmetadata activity
2. Copy Activity
3. Lookup Activity

ECLASESS.COM

4. For each Loop
5. Stored Procedure Activity
6. Web Activity
7. Dataflows

Azure Synapse Dwh Interview Questions:

- What is Architecture of Synapse Data ware house
- What is server less and dedicated pool
- What are different kind of Distributions?
  Round Robin
  Hash
  Replication

ECLASESS.COM