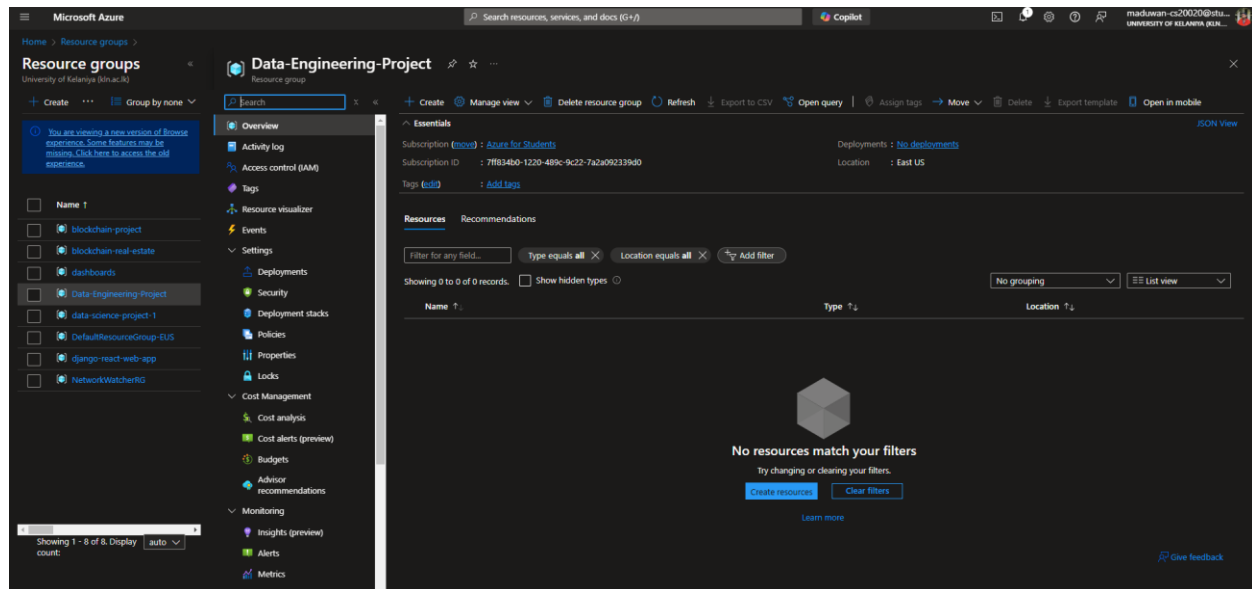


Phase 1

Why use this dataset?

Because we can work with several datasets and combine these datasets.

1. Create resource group



2. Create resources

2.1 Create Storage Account (Data Lake)

An Azure storage account contains all of your Azure Storage data objects: blobs, files, queues, and tables.

The storage account provides a unique namespace for your Azure Storage data that is accessible from anywhere in the world over HTTP or HTTPS.

Azure Blob Storage vs. Data Lake

Azure Blob Storage is one of the most common Azure storage types. It's an object storage service for workloads that need high-capacity storage. Azure Data Lake is a storage service intended primarily for big data analytics workloads.

Blob Storage (binary large object) -- is ideal for large amounts of **unstructured data**, such as text, videos, photos, application back-end data and backup data.

Azure Data Lake storage is currently separated into Gen1 and Gen2 options. Microsoft will retire Data Lake Gen1 storage in February 2024, and all customers using it must migrate to Gen2 before this date.

Azure Data Lake Gen1 is a storage service that's optimized for big data analytics workloads. Its **hierarchical file system** can store machine learning data, including log files, as well as interactive streaming analytics. It is performance-tuned to run large-scale analytics systems that require massive throughput and bandwidth to query and analyze large amounts of data.

Azure Data Lake Gen2 converges the features and capabilities of Data Lake Gen1 with Blob Storage. It inherits the file system semantics, file-level security and scaling features of Gen1 and builds them on Blob Storage. This results in a low-cost, tiered-access, high-security and high availability big data storage option.

The screenshot shows the 'Create a storage account' wizard in the Microsoft Azure portal. The 'Basics' tab is active. Under 'Project details', the subscription is 'Azure for Students' and the resource group is 'Data Engineering Project'. In the 'Instance details' section, the storage account name is 'awdatastoragedatalake', the region is 'US East US', and the primary service is 'Azure Blob Storage or Azure Data Lake Storage Gen 2'. The 'Performance' section has 'Standard' selected, and the 'Redundancy' section has 'Locally-redundant storage (LRS)' selected. At the bottom, there are 'Previous', 'Next', and 'Review + create' buttons.

Click Next button to create Data Lake. Other wise it will create a Blob Storage.

Microsoft Azure

Search resources, services, and docs (G17)

Copilot

maduwan-cs20020@stu... UNIVERSITY OF KELANIA (KLN...)

Home > Storage accounts >

Create a storage account

Basics **Advanced** Networking Data protection Encryption Tags Review + create

Security

Configure security settings that impact your storage account.

Require secure transfer for REST API operations ☒

Allow enabling anonymous access on individual containers ☐

Enable storage account key access ☒

Default to Microsoft Entra authorization in the Azure portal ☐

Minimum TLS version

Permitted scope for copy operations (preview)

Hierarchical Namespace

Hierarchical namespace, complemented by Data Lake Storage Gen2 endpoint, enables file and directory semantics, accelerates big data analytics workloads, and enables access control lists (ACLs) [Learn more](#)

Enable hierarchical namespace ☒

Access protocols

Blob and Data Lake Gen2 endpoints are provisioned by default [Learn more](#)

Enable SFTP ☐

Enable network file system v3 ☐

Blob storage

Put the tick on “Enable Hierarchical Namespace”

Microsoft Azure

Search resources, services, and docs (G17)

Copilot

maduwan-cs20020@stu... UNIVERSITY OF KELANIA (KLN...)

Home > Storage accounts >

Create a storage account

the Azure portal

Minimum TLS version

Permitted scope for copy operations (preview)

Hierarchical Namespace

Hierarchical namespace, complemented by Data Lake Storage Gen2 endpoint, enables file and directory semantics, accelerates big data analytics workloads, and enables access control lists (ACLs) [Learn more](#)

Enable hierarchical namespace ☒

Access protocols

Blob and Data Lake Gen2 endpoints are provisioned by default [Learn more](#)

Enable SFTP ☐

Enable network file system v3 ☐

Blob storage

Allow cross-tenant replication ☐

Cross-tenant replication and hierarchical namespace cannot be enabled simultaneously.

Access tier ☒ **Hot:** Optimized for frequently accessed data and everyday usage scenarios

☐ **Cool:** Optimized for infrequently accessed data and backup scenarios

☐ **Cold:** Optimized for rarely accessed data and backup scenarios

Azure Files

Enable large file shares ☒

[Previous](#) [Next](#) [Review + create](#) [Give feedback](#)

The screenshot shows the 'Create a storage account' wizard in the Microsoft Azure portal, specifically the 'Networking' tab. The 'Network access' section has three radio buttons: 'Enable public access from all networks' (selected), 'Enable public access from selected virtual networks and IP addresses', and 'Disable public access and use private access'. Below these is a warning icon and text: 'Enabling public access from all networks might make this resource available publicly. Unless public access is required, we recommend using a more restricted access type. Learn more'. The 'Private endpoint' section has a '+ Add private endpoint' button and a table with columns: Name, Subscription, Resource group, Region, Target sub..., Subnet, and Private DN... Below the table is a button 'Click on add to create a private endpoint'. The 'Network routing' section has a description: 'Determine how to route your traffic as it travels from the source to its Azure endpoint. Microsoft network routing is recommended for most customers.' At the bottom are 'Previous', 'Next', and 'Review + create' buttons, along with a 'Give feedback' link.

Click “review + create”

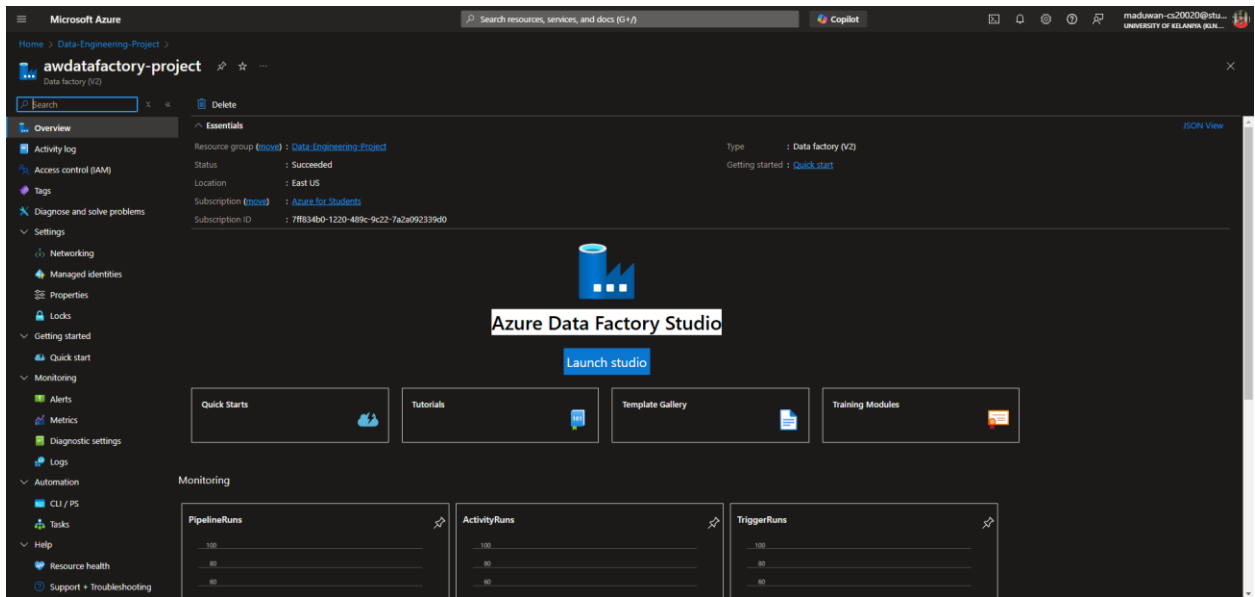
Next, click “create”

2.2 Create Data Factory

The aim of Azure Data Factory is to fetch data from one or more data sources and convert them into a format that we process. The data sources might contain noise that we need to filter out.

The screenshot shows the 'Create Data Factory' wizard in the Microsoft Azure portal, specifically the 'Basics' tab. The 'Project details' section has a description: 'Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.' Below this are two dropdown menus: 'Subscription' (set to 'Azure for Students') and 'Resource group' (set to 'Data Engineering Project' with a 'Create new' link below it). The 'Instance details' section has three dropdown menus: 'Name' (set to 'awdatafactory-project'), 'Region' (set to 'East US'), and 'Version' (set to 'V2'). At the bottom are 'Previous', 'Next', and 'Review + create' buttons, along with a 'Give feedback' link.

Next, go to the Azure Data Factory and click on “Launch Studio”

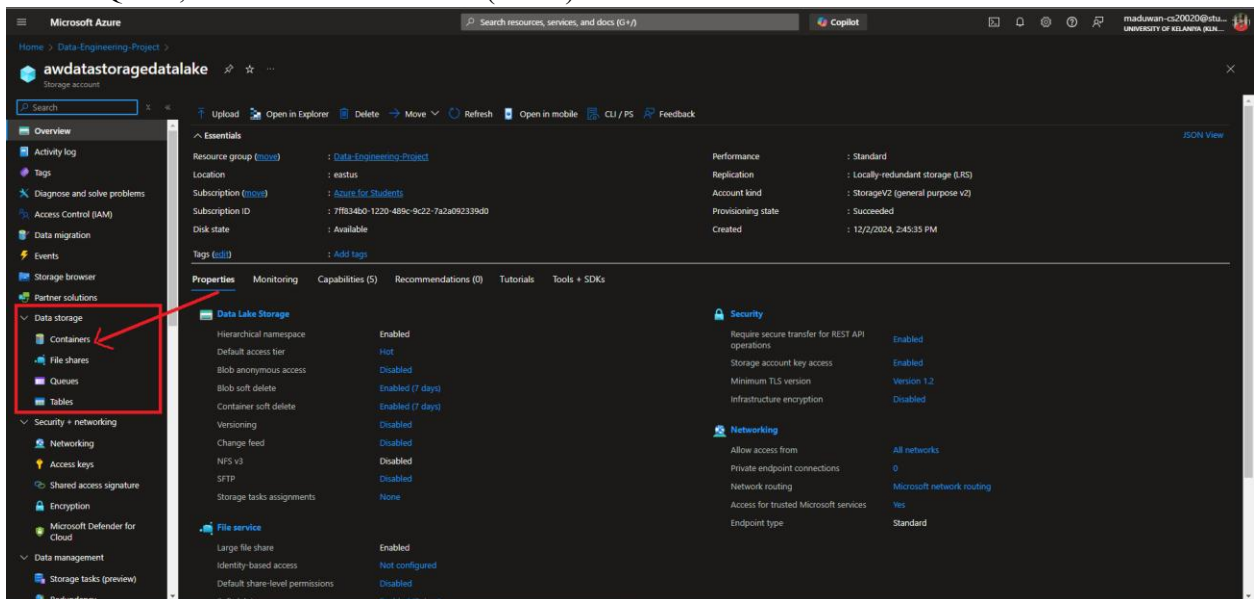


Let this tab open.

2.3

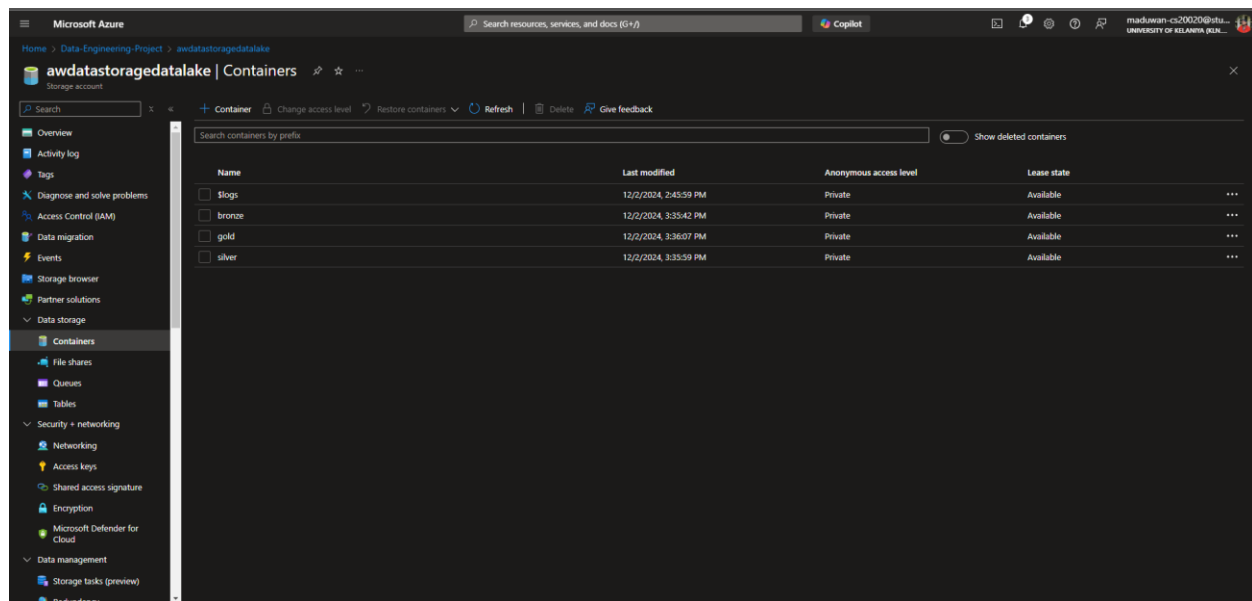
Now go to the resource group and click on the storage account resource. Go to the “Container” tab.

- Queue, stores Unstructured data (JSON).



Now we should create three Containers for Bronze, Silver & Gold.

- Bronze – Raw data Store
- Silver – Transformed Data
- Gold – Serving



2.3 Create Static pipeline

In here we are going to make a Link service in between the **GitHub repo (API service)** and the raw data store(Data Lake -Bronze).

We get the data from API service. That is our dataset is stored as a GitHub repo.

Now create API for data ingestion from GitHub repository and save it in bronze layer.

To create a pipeline, we need source and destination.

- Go to ADF and click create pipeline.
- Give a name for pipeline.
- Click the “Move and transform” and drag “Copy Data”.
- Give a name for activity. Then we need source and sink(Destination)
- First create **Two** Link service.
 - a. With Github API
 - b. With Storage Account (Data Lake)
- Go to Manage tab -> Linked Service
 - a. Click “+ new” -> select “HTTP” -> click continue
Give a name
Give base url. That is main part of the dataset of the github repo.
Select authentication type as “Anonymous”.

b. Click “+ new” -> select “Data Lake Gen 2” -> click continue

Give a name

Select subscription

Select storage account name which is created before.

- Next, create the dataset.

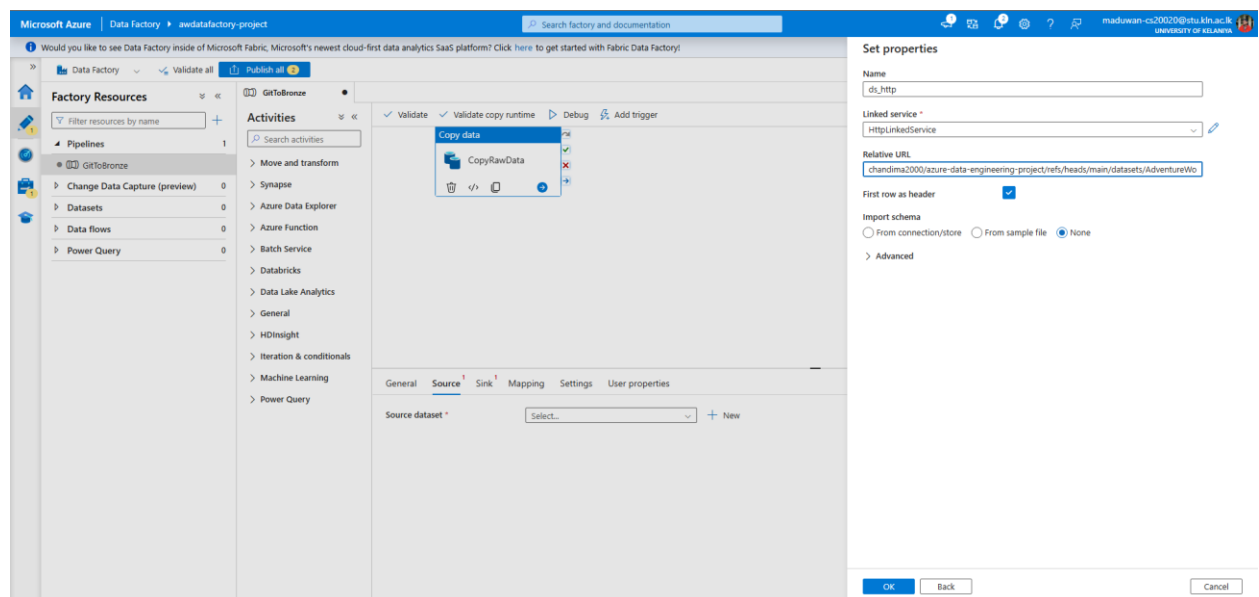
1. First create the source. That is github.

Go to Author tab -> click “CopyRawData” activity -> click source -> click “+new”

Select “HTTP” -> select the dataset format (Here, CSV)

Give a name -> select the Linked service which is created before.

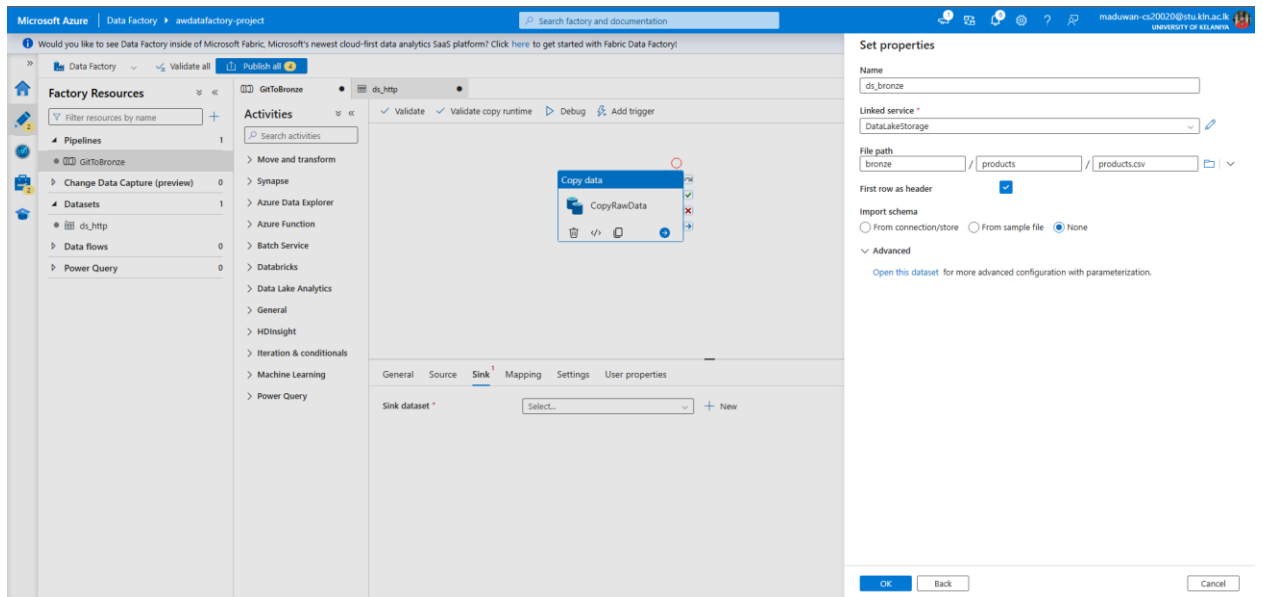
Give relative url.



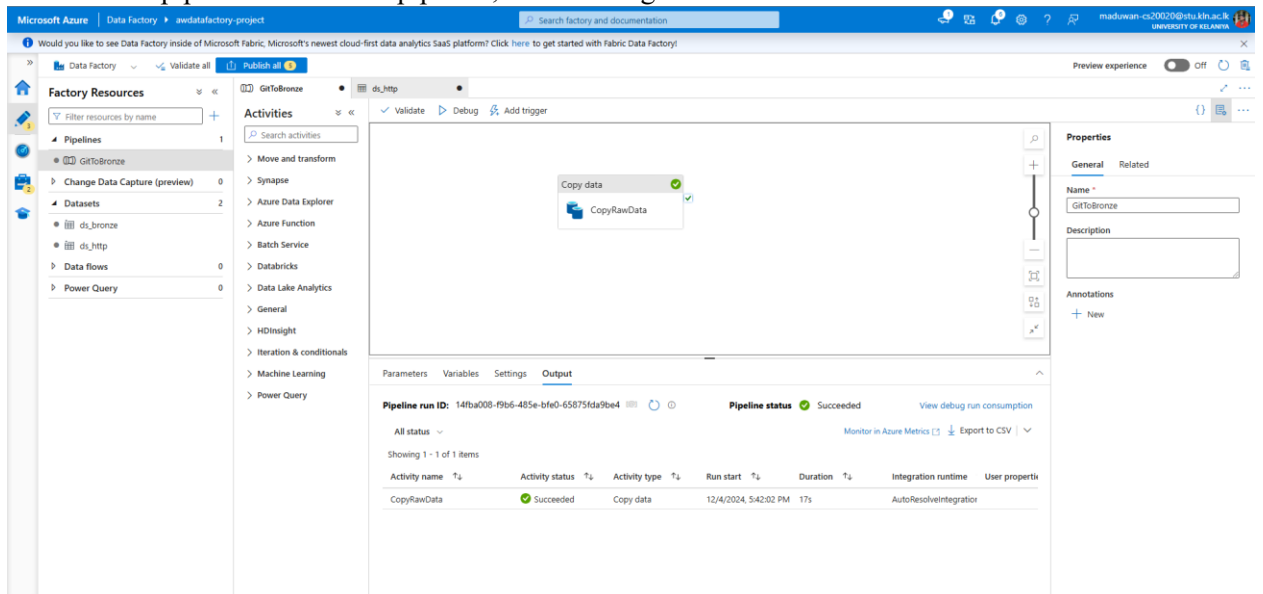
2. Next create the Sink (Destination).

click sink -> click “+new”

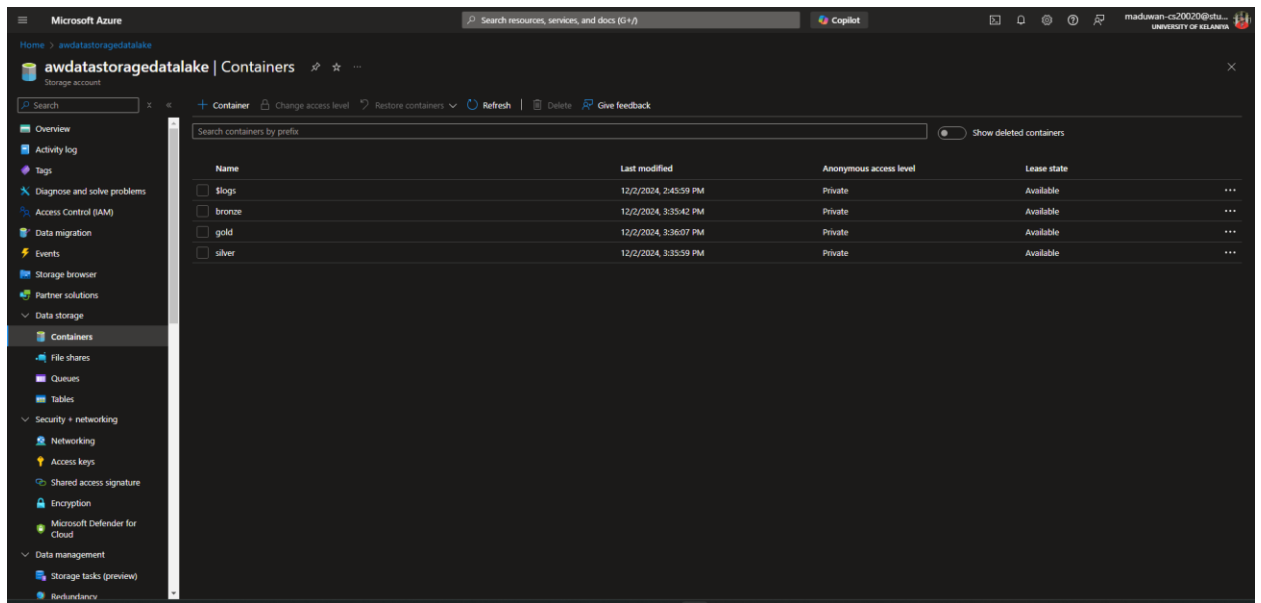
Select “Data Lake Gen 2” -> select the dataset format (Here, CSV)



- Next run the pipeline. To run the pipeline, click “debug”.

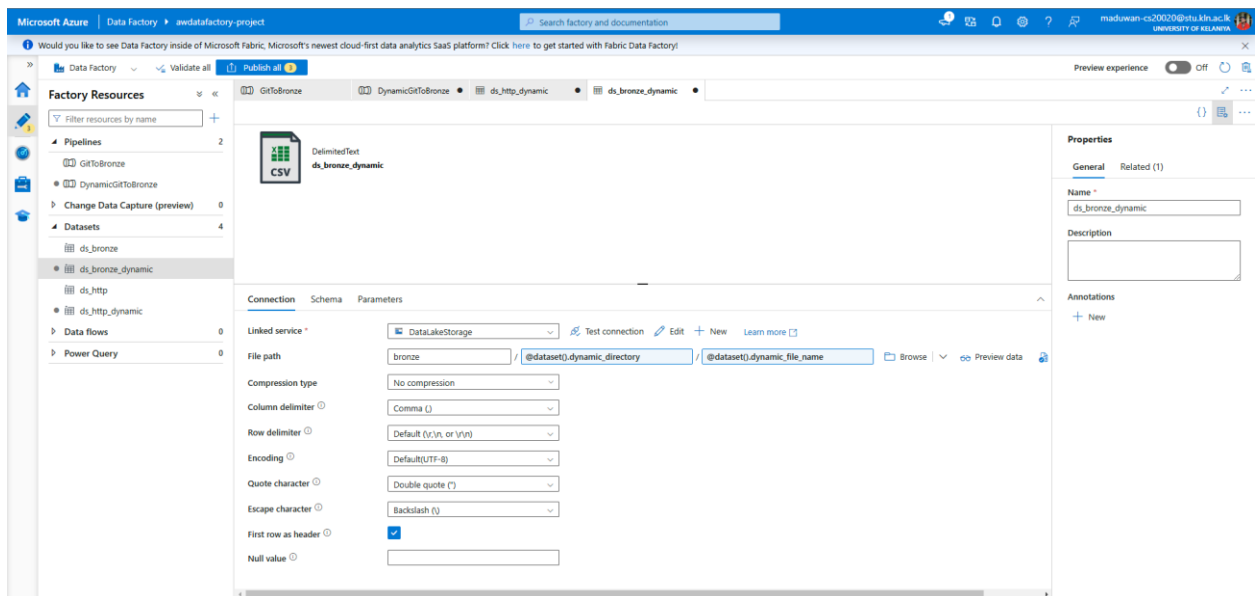


- Now we can see the raw data in bronze layer. Click bronze and then see the data.



2.4 Create Dynamic pipeline

Create dynamic parameters



Microsoft Azure | Data Factory | awdatafactory-project

Would you like to see Data Factory inside of Microsoft Fabric, Microsoft's newest cloud-first data analytics SaaS platform? Click [here](#) to get started with Fabric Data Factory!

Factory Resources

- Pipelines: 2
 - GitToBronze
 - DynamicGitToBronze
- Change Data Capture (preview): 0
- Datasets: 5
 - ds_bronze
 - ds_bronze_dynamic
 - ds_http
 - ds_http_dynamic
 - ds_param_json
- Data flows: 0
- Power Query: 0

Activities

- Move and transform
 - Copy data
 - Data flow
- Synapse
 - Azure Data Explorer
 - Azure Function
 - Batch Service
 - Databricks
 - Data Lake Analytics
- General
 - Append variable
 - Delete
 - Execute Pipeline
 - Execute SSIS package
 - Fail
 - Get Metadata
 - Lookup
 - Stored procedure

Diagram: Lookup -> ForEachGitToRaw -> Copy data (DynamicallyCopyRawData)

Pipeline run ID: 7e712b05-1f0d-4afb-848e-cf779b614f37

Pipeline status: Succeeded

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime	User properties
LookupGit	Succeeded	Lookup	12/4/2024, 8:45:10 PM	22s	AutoResolveIntegration	
DynamicallyCopyRawData	Inactive	Copy data	12/4/2024, 8:45:10 PM	Less than 1s	Unknown	
ForEachGitToRaw	Inactive	ForEach	12/4/2024, 8:45:10 PM	Less than 1s		

Properties

General

Name: DynamicGitToBronze

Description:

Annotations

+ New

Microsoft Azure | Data Factory | awdatafactory-project

Would you like to see Data Factory inside of Microsoft Fabric, Microsoft's newest cloud-first data analytics SaaS platform? Click [here](#) to get started with Fabric Data Factory!

Factory Resources

- Pipelines: 2
 - GitToBronze
 - DynamicGitToBronze
- Change Data Capture (preview): 0
- Datasets: 5
 - ds_bronze
 - ds_bronze_dynamic
 - ds_http
 - ds_http_dynamic
 - ds_param_json
- Data flows: 0
- Power Query: 0

Activities

- Move and transform
 - Copy data
 - Data flow
- Synapse
 - Azure Data Explorer
 - Azure Function
 - Batch Service
 - Databricks
 - Data Lake Analytics
- General
 - Append variable
 - Delete
 - Execute Pipeline
 - Execute SSIS package
 - Fail
 - Get Metadata
 - Lookup
 - Stored procedure

Diagram: Copy data (DynamicallyCopyRawData)

DynamicGitToBronze > ForEachGitToRaw

General Source Sink Mapping Settings User properties

Source dataset: ds_http_dynamic

Dataset properties

Name	Value	Type
dyn_rela_git_url	@item().dyn_rela_git_url	String

Request method: GET

Additional headers:

Request body:

Request timeout:

Properties

General

Name: DynamicGitToBronze

Description:

Annotations

+ New

Phase-2

Azure Data Bricks

It is an Apache spark based analytics platform.

Apache Spark is a powerful open-source data processing engine written in Scala, designed for large-scale data processing.

Apache Spark is written in Scala programming language. To support Python with Spark, Apache Spark Community released a tool, PySpark.

PySpark is the Python API for Apache Spark.

Spark itself is not a programming language but provides APIs in several programming languages to interact with its engine: Python (PySpark), Scala (native language of Spark), Java, R, SQL (for Spark SQL)

Features:-

1. High performance computing
2. Notebooks
3. Switch between programming languages (Spar sql, Scala, PySpark)
4. ETL
5. Model Training

- Create a Cluster

Click “compute” -> fill like below.

Microsoft Azure | databricks | Search data, notebooks, recent, and more... | CTRL + P | adb-adventure-works

Compute > New compute

adventure_project_cluster

Policy

Unrestricted

Access mode

Multi node | Single node

No isolation shared

Performance

Databricks runtime version

Runtime: 14.3 LTS (Scala 2.12, Spark 3.5.0)

Use Photon Acceleration

Node type

Standard_DS3_v2 | 14 GB Memory, 4 Cores

Terminate after 120 minutes of inactivity

Tags

Add tags

Key | Value | Add

> Automatically added tags

> Advanced options

Summary

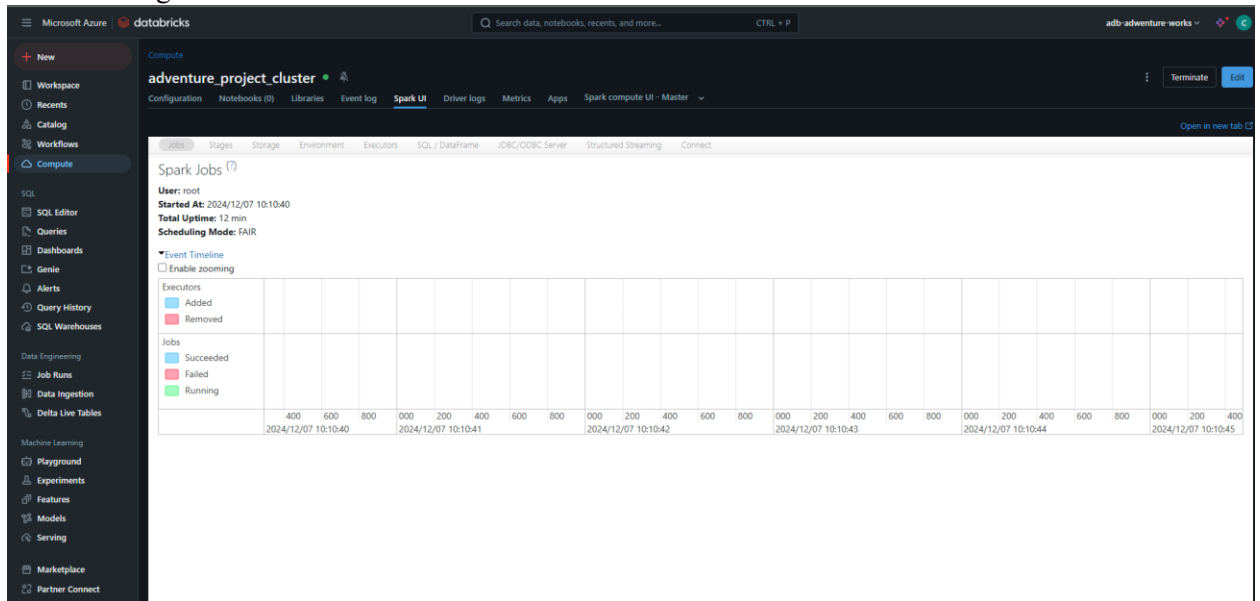
1 Driver | 14 GB Memory, 4 Cores

Runtime: 14.3 x scala2.12

Standard_DS3_v2 | 0.75 DBU/h

Create compute | Cancel

After creating the cluster:-



- Create Storage Access

Step 1:- Create an Application

search “Microsoft Entra ID” → App registration -> + new registration -> complete the registration

Delete	Endpoints	Preview features
^ Essentials		
Display name	: adventure_project	Client credentials : Add a certificate or secret
Application (client) ID	: 979a0794-0963-4b3c-b078-566c59d89823	Redirect URIs : Add a Redirect URI
Object ID	: 0a199945-a778-4f56-94b8-8f2ec0075756	Application ID URI : Add an Application ID URI
Directory (tenant) ID	: aa232db2-7a78-4414-a529-33db9124cba7	Managed application in I... : adventure_project
Supported account types	: My organization only	

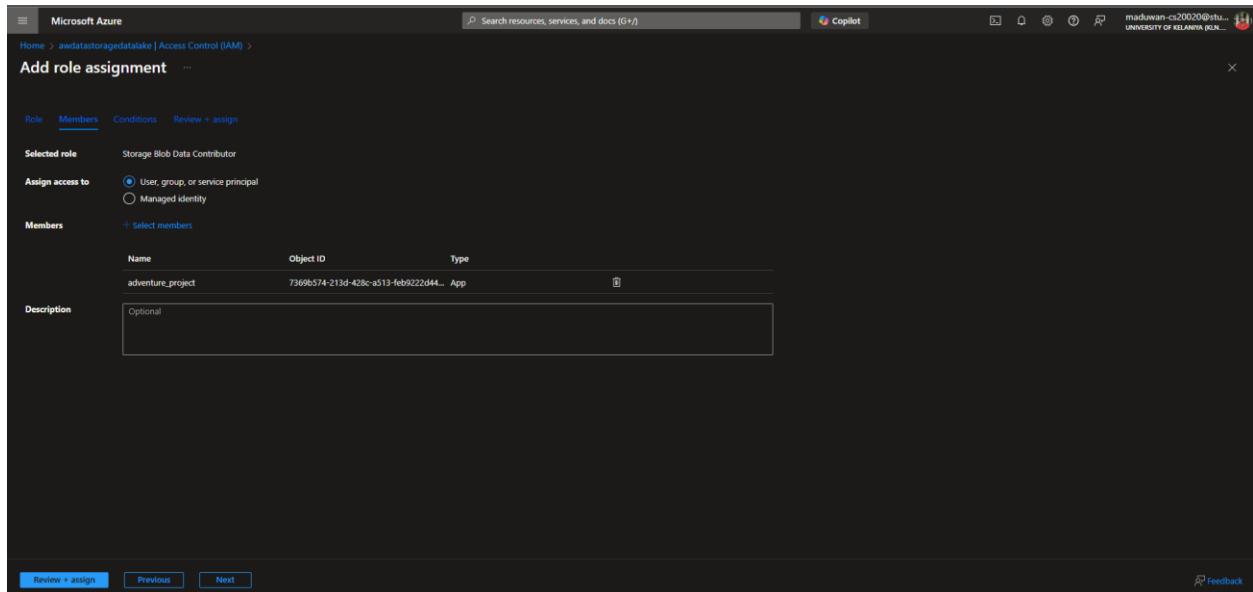
Simply copy these informations.

Next, create secrets.

Click “Certificate & Secrets” -> + New client Secrets -> copy the secret-value and secret-id

Step 2 :- Assigning roles to this application

Goto home and select “Storage Account” -> Access Control -> + Add -> search “Storage Blob Data Contributor” -> click “select members” -> select the application name which was created earlier -> click “Review + Assign”



Step 3 :-

Go to “data bricks” -> click “workspace” (because need to create a folder) -> click "folder" -> create a folder -> click “notebook” (because In this folder need to create a notebook) -> connect to the cluster

Do the transformation tasks in notebook and import data into silver layer. With this end our Phase 2.

What is Parquet?

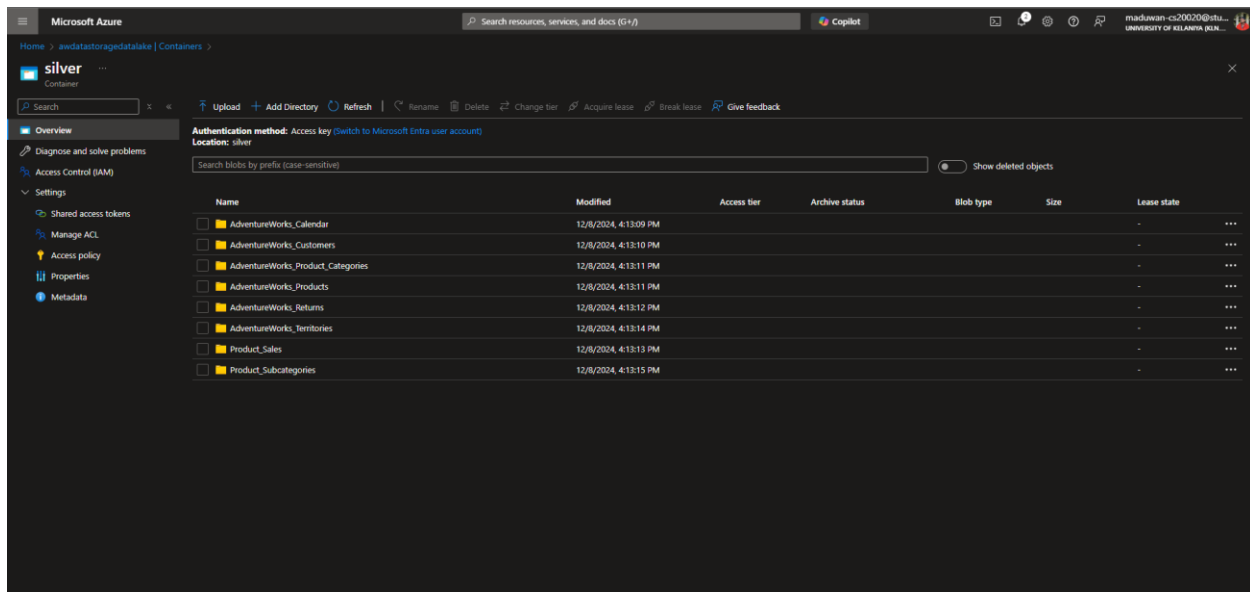
Apache Parquet is an open source, column-oriented data file format designed for efficient data storage and retrieval. It provides efficient data compression and encoding schemes with enhanced performance to handle complex data in bulk.

Characteristics of Parquet

- Free and open-source file format.
- Language agnostic.
- Column-based format - **files are organized by column, rather than by row(CSV files), which saves storage space and speeds up analytics queries.**
- Used for analytics (OLAP) use cases, typically in conjunction with traditional OLTP databases.
- Highly efficient data compression and decompression.
- Supports complex data types and advanced nested data structures.

The open-source Delta Lake project builds upon and extends the Parquet format.

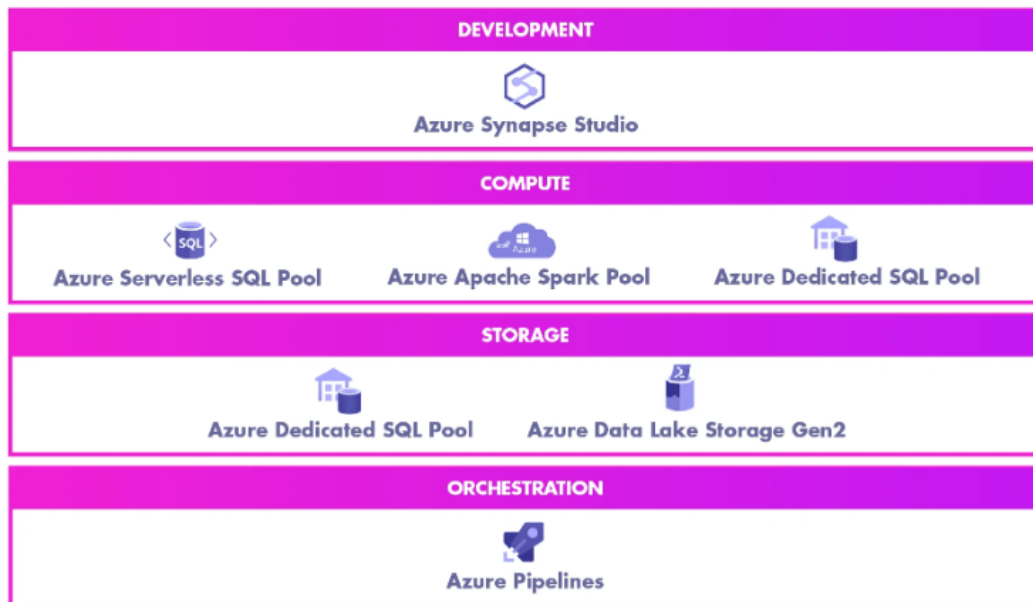
Silver Layer file system



Phase – 3

In here, we are going to serving the data. That is Gold Layer.

Azure Synapse Analytics is a comprehensive data integration, analytics, and data warehousing platform provided by Microsoft Azure. It combines enterprise data warehousing with big data analytics to offer a unified environment for processing, managing, and analyzing large volumes of structured and unstructured data.



Key Features of Azure Synapse Analytics

Data Integration:

- Azure Synapse Pipelines: Data integration service that allows you to create ETL (Extract, Transform, Load) workflows to move and transform data from various sources.
- Connects to on-premises, cloud-based, and hybrid data sources.

Data Warehousing:

- Offers **dedicated SQL pools** (provisioned resources) for massive parallel processing (MPP) of data.
- Serverless SQL pools provide on-demand querying capabilities for ad hoc analysis without requiring dedicated resources.

Big Data Analytics:

- Integrated with Apache Spark pools for distributed big data processing.
- Can process and analyze data stored in Azure Data Lake or other storage systems.

Data Exploration and Analysis:

- Allows querying across structured, semi-structured, and unstructured data.
- Supports T-SQL, making it accessible for users familiar with SQL Server.

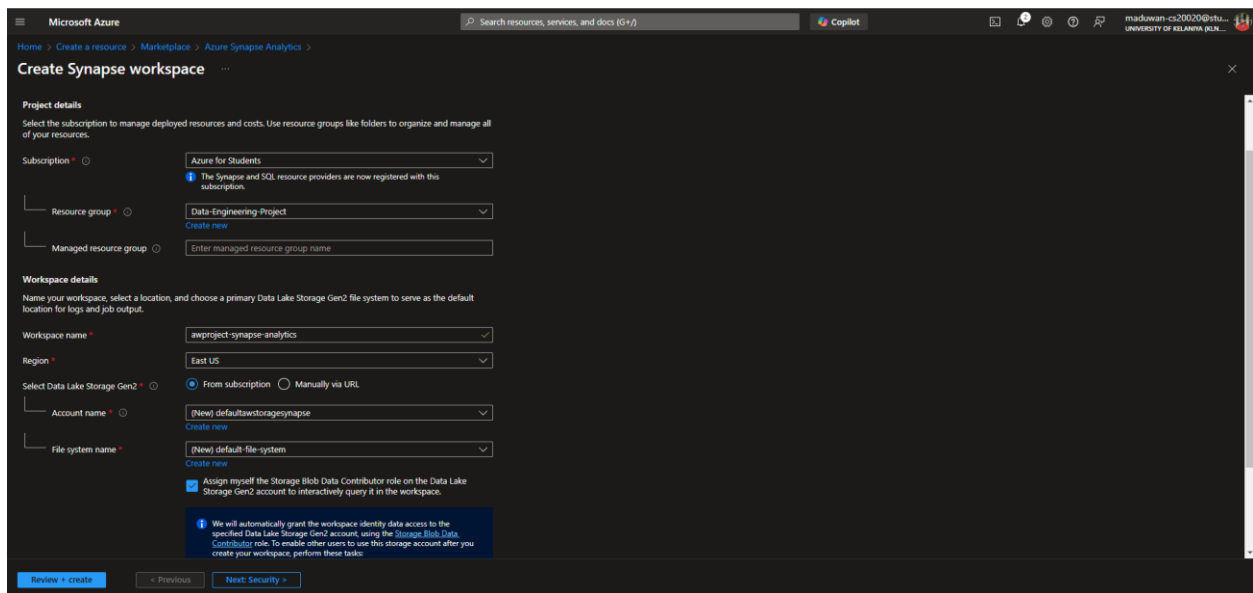
Data Visualization:

- Seamlessly integrates with Power BI for creating interactive dashboards and reports.
- Enables real-time business intelligence.

Security and Governance:

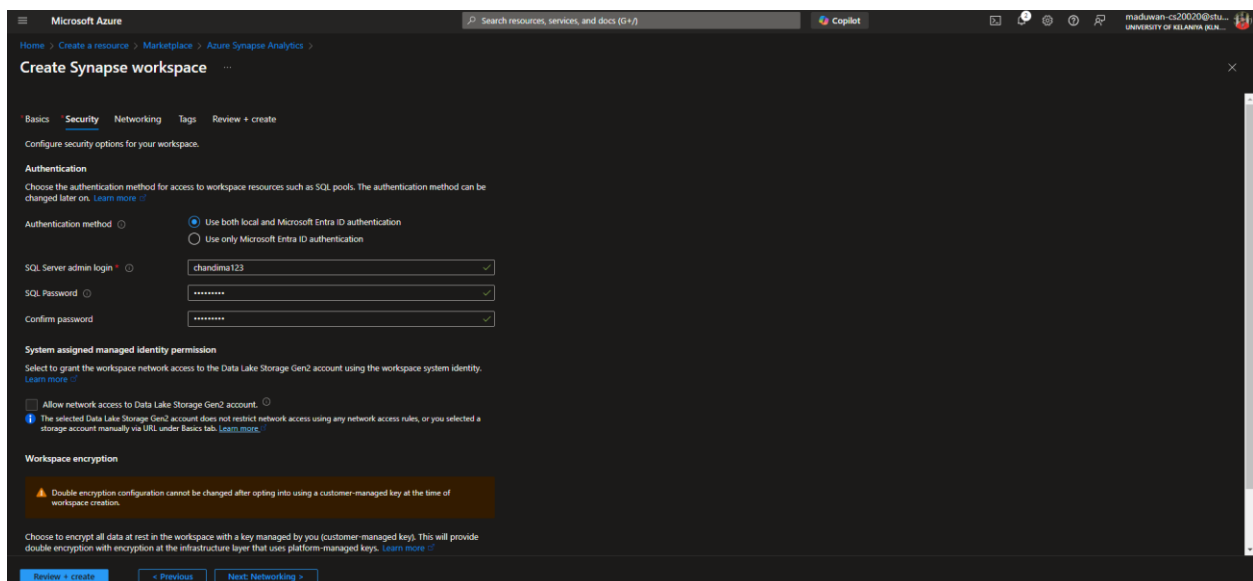
- Built-in security features like role-based access control (RBAC), managed identities, and data encryption.
- Integration with Azure Purview for data governance and lineage tracking.

Click ‘Create resource’ -> search “Synapse Analytics” -> fill the form.(Here we should create a new Storage account for synapse analytics)



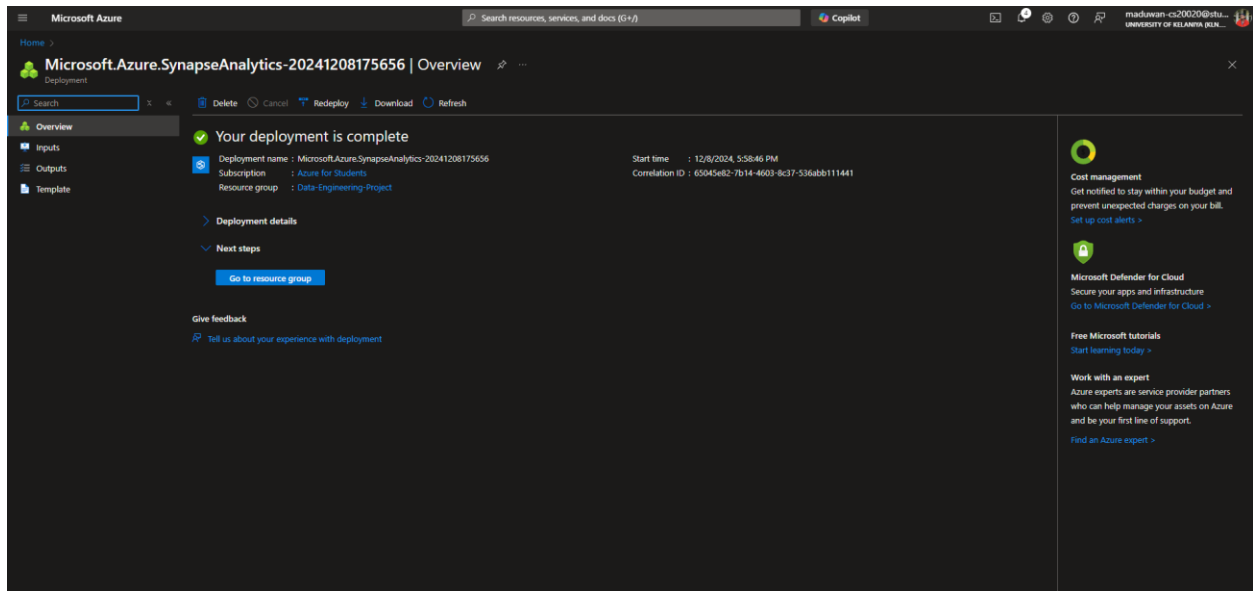
The screenshot shows the 'Create Synapse workspace' form in the Microsoft Azure portal, specifically the 'Project details' tab. The form is for creating a new Synapse workspace. The 'Subscription' is set to 'Azure for Students'. The 'Resource group' is 'Data-Engineering-Project'. The 'Workspace name' is 'awproject-synapse-analytics'. The 'Region' is 'East US'. The 'Data Lake Storage Gen2' account is '(New) defaultstoragesynapse'. The 'File system name' is '(New) default-file-system'. There are checkboxes for 'Assign myself the Storage Blob Data Contributor role' and 'We will automatically grant the workspace identity data access to the specified Data Lake Storage Gen2 account'. The 'Review + create' button is at the bottom.

Configure the dedicated SQL pool

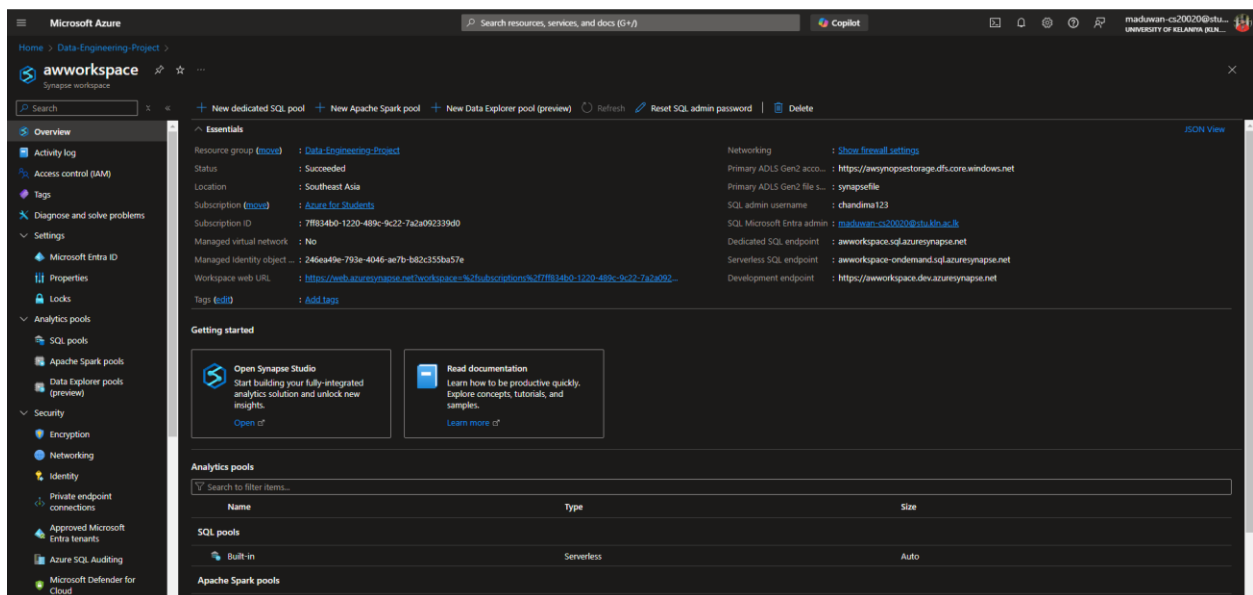


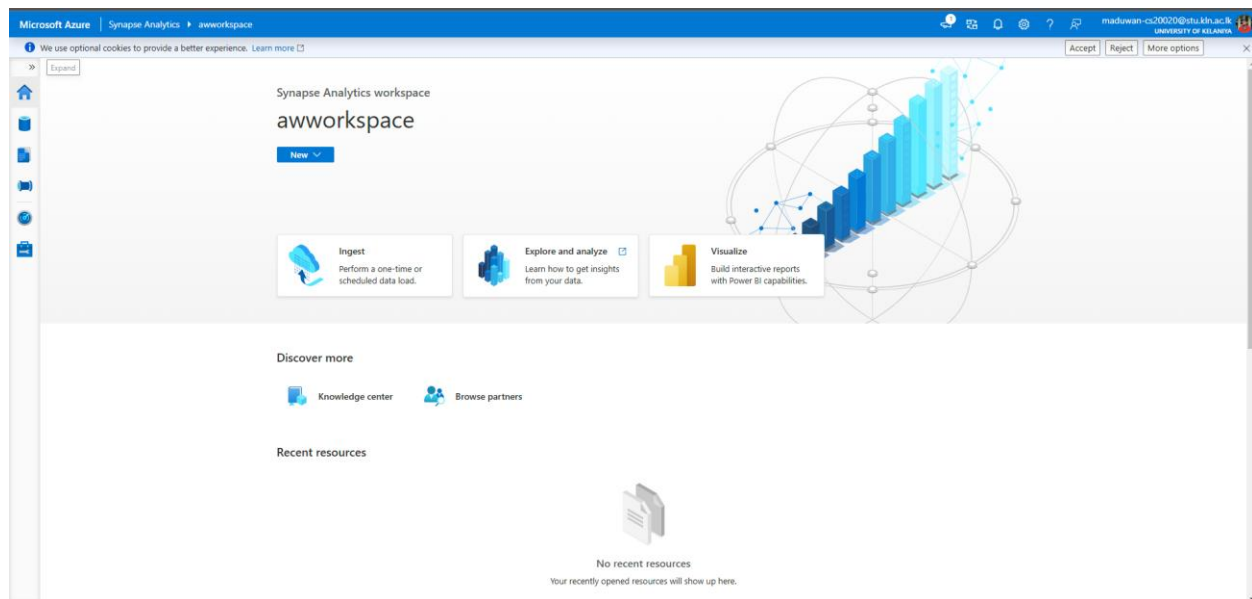
The screenshot shows the 'Create Synapse workspace' form in the Microsoft Azure portal, specifically the 'Security' tab. The form is for configuring security options for the workspace. The 'Authentication method' is 'Use both local and Microsoft Entra ID authentication'. The 'SQL Server admin login' is 'chandima123'. The 'SQL Password' and 'Confirm password' fields are filled with asterisks. The 'System assigned managed identity permission' section has a checkbox for 'Allow network access to Data Lake Storage Gen2 account'. The 'Workspace encryption' section has a warning icon and text about double encryption configuration. The 'Review + create' button is at the bottom.

Then click “review + create”.

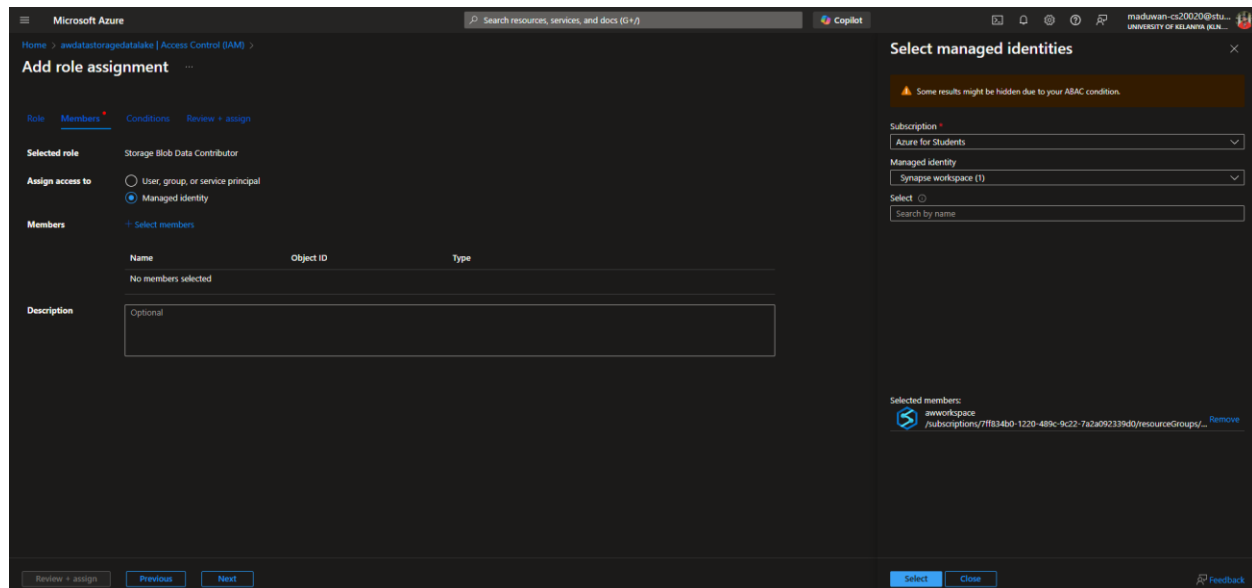


Synapses Workspace Homepage



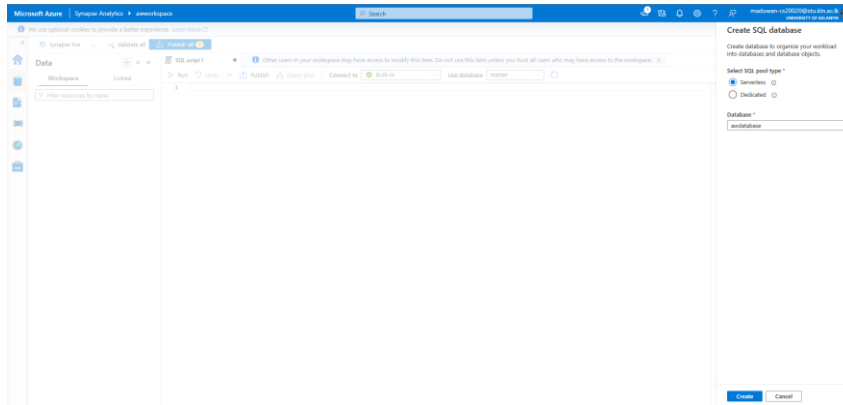


1. Goto “storage account” -> click “Access Control” -> click “+ add” -> search “Storage blob data contributor” -> do following

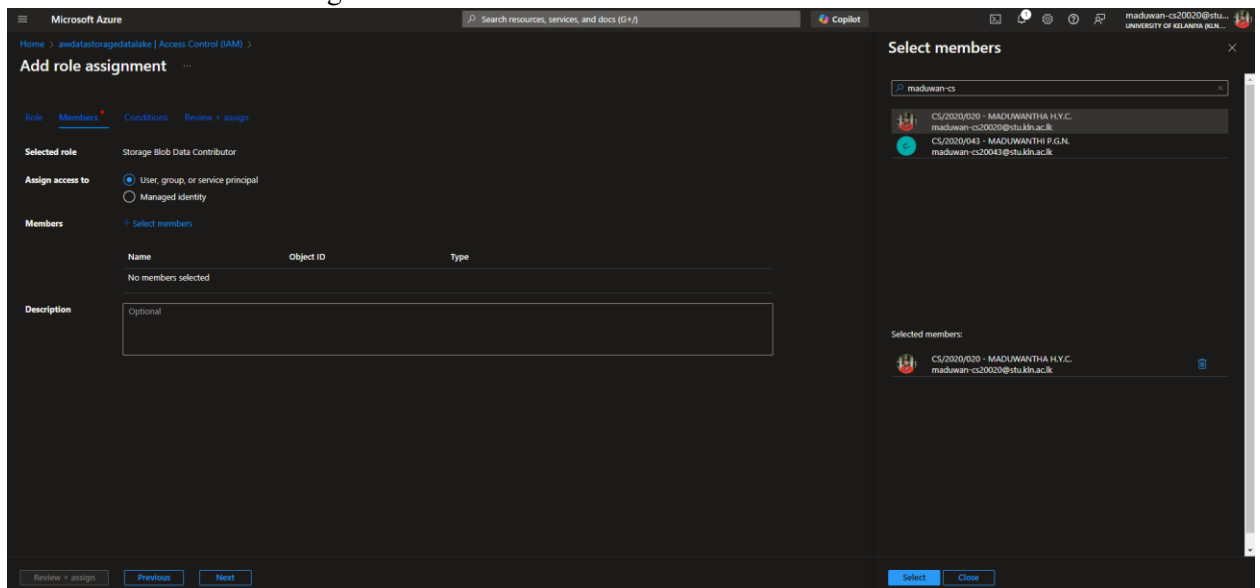


Lastly click “ Review + Assign”

2. Goto Azure synapse Analysis window -> select “develop” tab -> select “+” -> select “SQL Script”
3. Now we want to create a Database. Goto “Data” tab -> select “SQL database”
- 4.



5. Again, we need to create a Access control -> click “+ add” -> search “Storage blob data contributor” -> do following



- How to show data using power BI?
 - All data should be displayed using Views. We create Views on top of the Query. For that we need Schemas. These views stores in gold layer. Then power BI use gold layer.

Lastly show the Data using Power BI.

Goto synapse workspace and copy the serverless SQL endpoint