# Azure Trailblazer Academy
Azure Data Factory (ADF) Lab

Overview

Azure Data Factory is a PaaS cloud-based ETL & data integration tool allows you to create data driven workflows in the cloud for orchestrating and automating data movement and data transformation.

It is a great tool to orchestrate big data and operationalize processes to refine these enormous stores of raw data into actionable business insights.

Lab Overview

This lab will help you gain the experience to ingest data from on-premises databases such as Oracle, SAP, Teradata, Hortonworks, DB2, SQL Server and Cloudera to Azure Data storage, databases and data warehouses services. It will showcase the steps to build a pipeline using ADF to ingest the data into ADLS GEN2 storage and secure the PII data using data transformation functions using the Data Flow activity and finally store the data in Synapse SQL Pool (Data warehouse) for building BI dashboards.

Azure Data Factory Lab
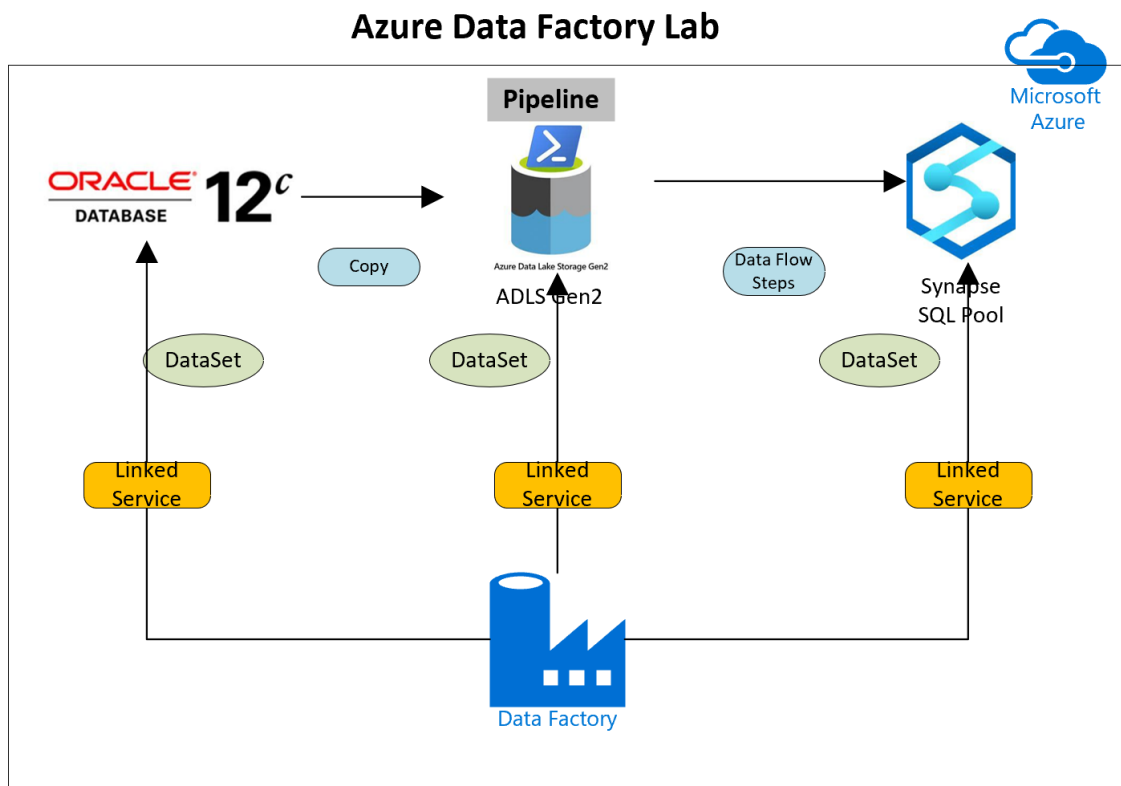
Pre-requisites

- Write Access to Azure Data Lake Storage Account (ADLS Gen2)
- Read Access to Sample HR schema in Oracle Database
- Write Access to Synapse SQL Pool Data warehouse

Automated Deployment

This automated deployment script will create thre required Azure srevices for this lab. Press the "*Deploy to Azure*" button below.

https://portal.azure.com/#create/Microsoft.Template/uri/https%3A%2F%2Fraw.githubusercontent.com%2Fmicrosoft%2FAzureTrailblazerAcademy%2Fmaster%2Fmonth2%2Flabs%2Flab_adf%2Fscripts%2Flab2_data_deployment.json

Enter the following information

- Subscription: Enter your subscription.
- Resource group: Select 'Create new' under Resource group. Enter 'ata-adf-lab-<YourName>-rg'
- Region: Select 'East US'.
- SQL_Server Name:Enter 'ata-adf-lab-sql-<YourName>'
- Server_location:Enter 'eastus'

Custom deployment

Deploy from a custom template

Subscription * ⓘ

[▽]

Resource group * ⓘ

[ata-adf-lab-srini-rg   ▽]

Create new

Parameters

Region * ⓘ

[East US   ▽]

SQL_Server Name ⓘ

[ata-adf-lab-sql-srini   ]

Server_Location ⓘ

[eastus   ]

Administrator Login * ⓘ

[azureadmin   ✓]

Administrator Login Password ⓘ

[•••••••••   ✓]

Collation ⓘ

[SQL_Latin1_General_CP1_CI_AS   ]

Database Name * ⓘ

[ataadflabsqldb   ✓]

Sku Name ⓘ

[DW100c   ]

Storage Accounts_Data Lake Gen2 ⓘ

[ataadflabstoragesrini   ✓]

Datalakestorage_location ⓘ

[eastus   ]

[Review + create]   [< Previous]   [Next : Review + create >]

- AdministratorLogin: 'azureadmin'
- Administrator Login Password: 'Ataadf123!'
- Database Name: 'ataadflabsqldb'
- Storage_Accounts_Data Lake Gen2:'ataadflabstorage<YourName>/
- Click on 'Review and Create'
- Click on 'Create'

Please check if it created the following services after successful deploypment.

☐	🔷 ataadflabsqldb (ata-adf-lab-sql-srini/ataadflabsqldb)	Synapse SQL pool (data warehouse)
☐	🗄 ata-adf-lab-sql-srini	SQL server
☐	▦ ataadflabstoragesrini	Storage account

Temporary access to an Oracle database for completing this lab will be provided in the class.

Task List

- Task-1: Create Azure Data Factory Service
- Task-2: Create linked services
- Task-3: Create Copy Activity to ingest Oracle HR Emplyee data to Azure Storage
- Task-4: Create Data Flows to Transofrm Employee data and store it in Synapse SQL Analytics

Task-1: Create Azure Data Factory Service

1. Provision an ADF service

- Type 'Data factories' in the search bar.
- Select 'Data factories' and select 'add' to create a new service

Provide the following info:

- Name: 'ata-adf-lab-<YourName>'
- Subscription: Make sure to select your subscription
- Select 'ata-adf-lab-<YourName>', the resource group you have created with custom deployment.
- Location: select 'East US'
- Enable GIT: uncheck the box

Click on 'Create' button

New data factory

Name *

ata-adf-lab-srin

Version ⓘ

V2

Subscription *

Resource Group *

ata-adf-lab-srini-rg

Create new

Location * ⓘ

East US

Enable GIT ⓘ

☐

Create

2. Open ADF Author Tool

- Select 'Go to resource' when it completes the deployment.
- Select 'Author & Monitor' in the middle of the screen.
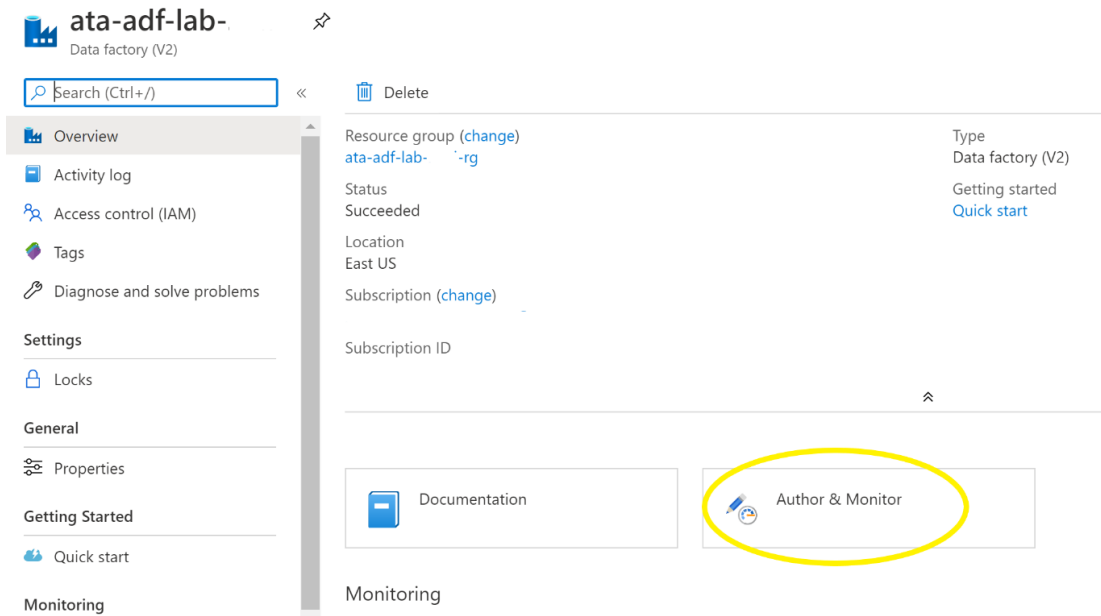
Opens up a new tab introducing the drag and drop interface to build pipelines.
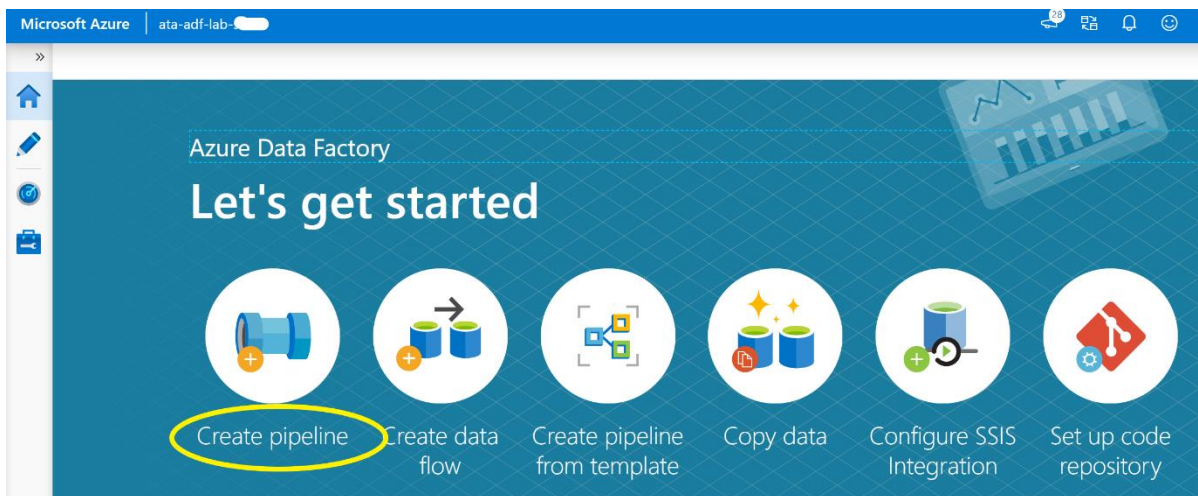
Turn on Dataflow Debug functionality to test the modifications.

- To preview data, we need to turn on the dataflow debug option
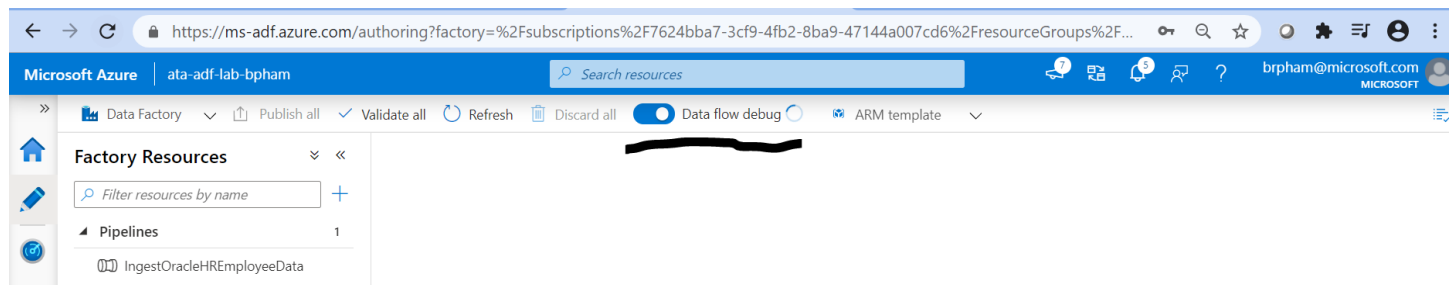
Task-2: Create linked services

You will be creating connection linked services to sync and source systems such as Oracle, ADLS Gen2 and Synapse in this task.

1. Create Oracle Linked Service

- Select Management hub (ToolBox) icon on the left and select 'Linked Services' under Connections section.
- Select 'New' under Linked Services
- Search for 'Oracle' under 'Data Store' and select 'Oracle Database' and click on 'Continue'
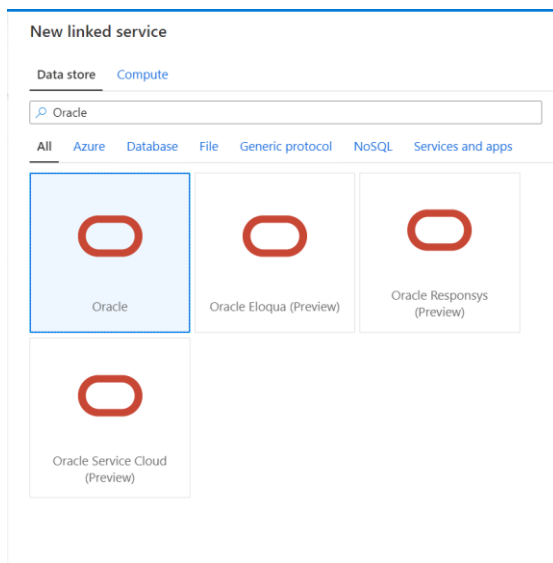
Enter the following Oracle Connect Info:

- Name: Enter 'OracleDB12cHR'
- Leave the default 'Connection string' option
- Host: <mark>Enter Instructor provided server IP address</mark>
- Port: 1521 (Default Oracle Port)
- Connection type: Select 'Oracle Service Name'
- Service name: '**nonpdb**'
- Username: hr
- Password: hr

New linked service (Oracle)

Name *

OracleDB12cHR

Description

Connect via integration runtime *   ⓘ

AutoResolveIntegrationRuntime   ⌄

[**Connection string**] [Azure Key Vault]

Host *

Port

1521

Connection type   ⓘ

Oracle Service Name   ⌄

Service name *

pdb1

User name *

hr

[**Password**] [Azure Key Vault]

Password *

••

✓ Connection successful

[**Create**]   [Back]   🔗 Test connection   [Cancel]

- Select 'test connection' to verify the successful connection

Click on 'Create' button to create the Oracle linked service.

2. Create ADLS Gen2 Storage Linked Service

- Select 'New' under Linked Services

- Search for 'Gen2' under data store and select ADLS Gen2 Click on 'Continue' and enter the following information:

- Name: Enter 'Gen2HRStorage'

- Authentication Method: Leave the default 'Account Key' selection

- Account Selection method: Leave the default 'From Azure subscription'

- Storage account name: select 'ataadflabstorage<YourName>'

- Click on 'Test connection' Click on 'Create' after successful connection to create the ADLS Gen2 storage linked service

New linked service (Azure Data Lake Storage Gen2)

hosted integration runtime is higher than version 4.0 if connecting via self-hosted integration runtime.

Name *

Gen2HRStorage

Description

Connect via integration runtime * ⓘ

AutoResolveIntegrationRuntime ⌄

Authentication method

Account key ⌄

Account selection method ⓘ

⦿ From Azure subscription   ◯ Enter manually

Azure subscription ⓘ

⌄

Storage account name *

ataadflabstoragesrini ⌄

Test connection

⦿ To linked service   ◯ To file path

Annotations

＋ New

✅ Connection successful

🔗 Test connection   Cancel

Create   **Back**

3. Create Synapse Analytics Linked Service

- Select 'New' under Linked Services
- Search for 'Synapse' under data store and select 'Azure Synapse Analytics'
- Click on 'Continue'

Enter the following info:

- Name: Enter 'SynapseDBHR'
- Server name: select 'ata-adf-lab-sql-<yourname>'
- Database name: select 'ataadflabsqldb'
- Username: enter 'azureadmin'
- Password: enter the default 'ataadf123!'
- Click on 'test connect' to test the connection.

Click on 'Create' after the successful connection to create the Synapse linked service

New linked service (Azure Synapse Analytics (formerly SQL DW))

Name *

SynapseDBHR

Description

Connect via integration runtime *   ⓘ

AutoResolveIntegrationRuntime   ⌄

(Connection string   Azure Key Vault)

Account selection method   ⓘ
  ⦿ From Azure subscription   ◯ Enter manually

Azure subscription

[   ] ⌄

Server name

ata-adf-lab-sql-srini   ⌄

Database name

ataadflabsqldb   ⌄ ↻

Authentication type *

SQL authentication   ⌄

User name *

azureadmin

(Password   Azure Key Vault)

Password *
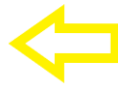
[Create]   [Back]   ⇦   🔗 Test connection   [Cancel]

You have successfully created connection linked services to Oracle, ADLS Gen2 and Synapse SQL Pool.

Task-3: Create Copy Activity to ingest Oracle HR Employee data to Azure Storage

We have established the connection services to the source Oracle DB and the sink Azure storage, you will create a copy activity to ingest the data from Oracle to Azure Storage in this task.

1. Select 'Pencil' icon on the left and select three dots next to pipelines to select 'new pipeline' action.

- Name the pipeline as 'IngestOracleHREmployeeData' under the properties section on the right side.
- Drag the 'Copy data' from 'Move & transform' section under 'Activities' list to the canvas in the middle of the screen.
- Name the copy activity as 'CopyOracleHREmpToGen2' below the canvas under the 'General' tab
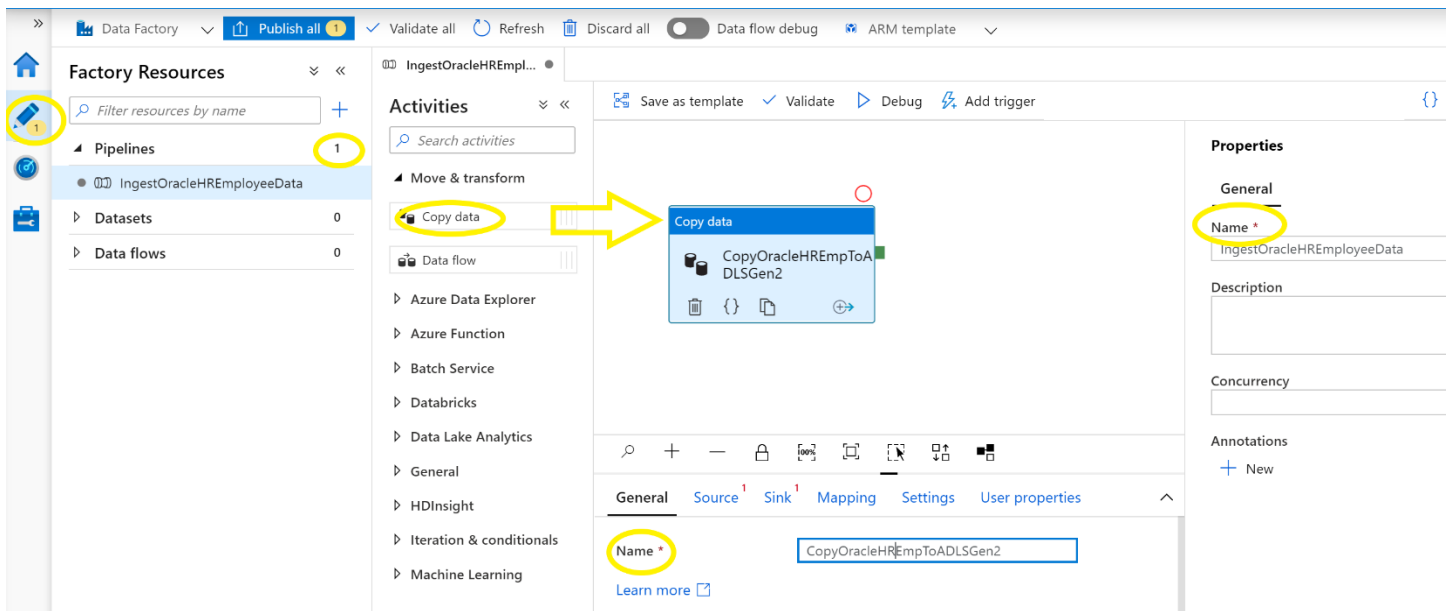
2. Select 'Source' tab next to 'General' to define the source system.

- Select 'New' to create a new source dataset
- Select 'Oracle database' as the data store after filtering with 'Oracle' and click on 'Continue'
- Select 'Open' to define the source dataset
- Name the dataset as 'Oracle_HR_Employee_table' under the properties section on the right side.
- Select the 'OracleDB12cHR' linked service and click on 'Test connection' to test the connectivity.
- Filter table list by typing 'hr.emp' and select 'HR.EMPLOYEES' table and select 'Preview data'.
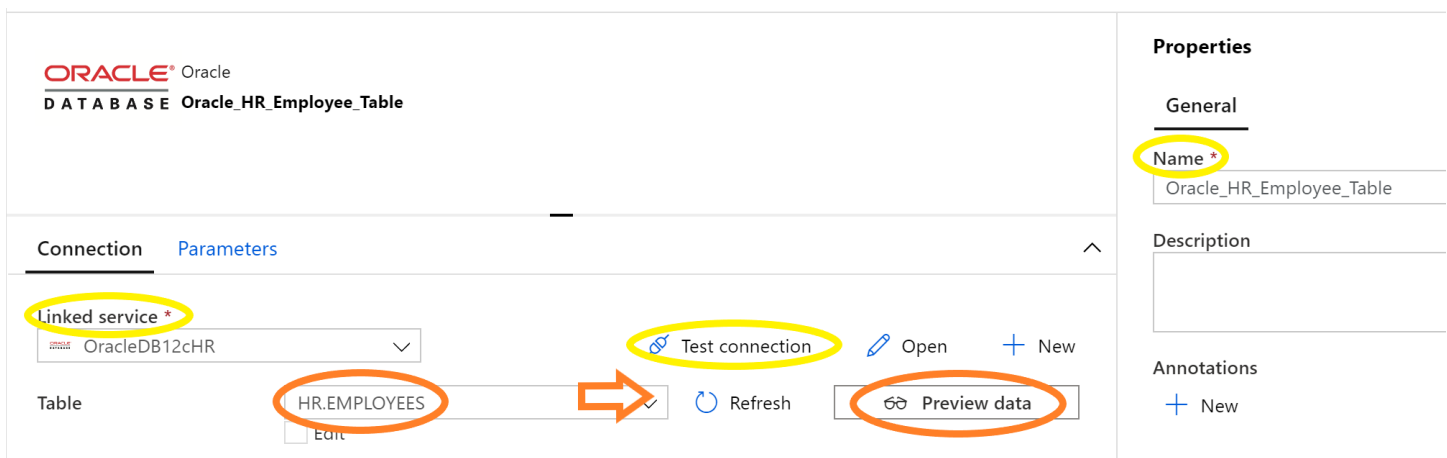
- First, we need to create the directory path in the storage account. Leave this browser tab as is and switch to the Azure Services tab in the browser.
- **Access storage account and open up the 'storage explorer' to create 'hr' under 'data' file system and create 'employee' as a subfolder under 'hr' folder.**
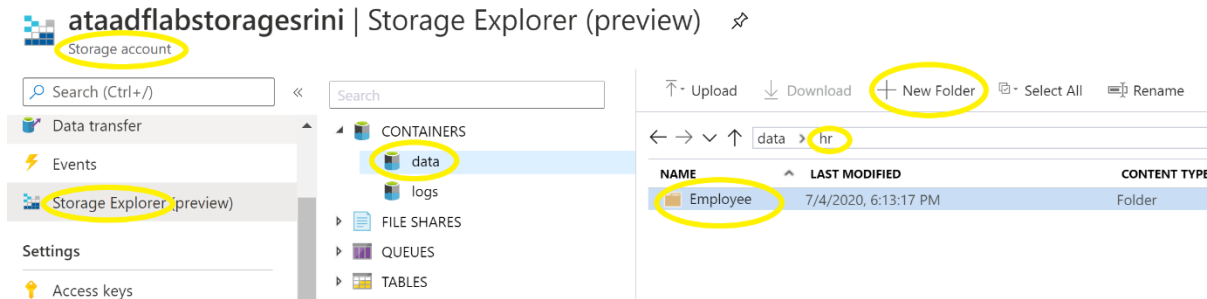
3. Select 'Sink' tab next to 'Source' to define the sink system

- Select 'New' to create a new sink data set
- Select 'ADLS Gen2' as the data store after filtering with 'Gen2' and click on 'Continue'
- Select 'Open' to define the sink dataset
- Select 'Delimited Text' as the format and click on 'Continue'
- Name the dataset as 'Gen2HREmpData'
- Select the 'Gen2HRStorage' linked service.

Set properties

Name

Gen2HREmpData

Linked service *

Gen2HRStorage

File path

data / hr/employee / File

First row as header ☐

Import schema

◉ From connection/store ○ From sample file ○ None

▷ Advanced

[OK]   [Back]   [Cancel]

- Select 'Gen2HREmpData' under Datasets section on the left to set the first row as the header.
- Check the box

5. Test the copy activity by clicking on 'Debug' option just above the canvas.

- It will start the process and put it in the queue. Wait till it finishes.
- Check for the status and make sure it is successful

6. Verify the data ingestion in ADLS Gen2 storage.

Switch to Azure services tab and access the storage account.

- Open up the 'Storage Explorer' and access the data file syste and drill down to 'hr' and 'Employee' folder.
- Confirm the 'HR.EMPLOYEES.txt' file. Double click on the file to download and view the data.

You can see how the phone numbers are ingested as text. This is PII data and we should protect this data.

Task-4: Create Data flow to transform Employee data and store it in Synapse SQL Analytics

We have noticed the PII data we just ingested into ADLS Gen2 storage. You will secure the PII data using the Data flow functionality in this task.

1. Create Source data

- Drag and drop the data flow activity into the canvas from the 'Move &Transform' section.
- Select 'create new data flow' and select 'Data Flow' option and click on 'OK'.

Adding data flow

○ Use existing data flow   ● Create new data flow

Mapping Data Flow
Code free data transformation at scale

Wrangling Data Flow (Preview)
Code free data preparation at scale

OK   Cancel

- Select "Source" data
- Name the data flow as 'SecurePIIdata'.

Provide the following information under 'Source settings' tab:

- Output stream name: Gen2HrEmpDataOut
- Source type – Select 'Dataset'
- Dataset – Select 'Gen2HREmpData'

- To preview data, we need to turn on the dataflow debug option.
- Click on 'Import projections' in the 'Projection' tab to view all the columns.

2. Create a data flow step to hash the first 6 digits of the phone number.

- Add another data flow step to secure the phone number by selecting '+' sign and select 'Derived Column' under 'Schema modifier' section.

- Name the data flow as 'HREmpSecurePhoneNumber'
- Select 'PHONE_NUMBER' column from the list
- Click on 'Enter Expression' box

Write RegExpression to replace the digits before the '.' with '#' as shown in the picture:

```
regexReplace({PHONE_NUMBER}, '\\d+\\.', '###.')
```

- Click on 'Refresh' to verify the output of the function.
- Click 'Save and finish' button after you satisfy the result

3. Create a data flow step to store the transformed data into Synapse SQL Pool.

We are now ready to output the transformed data into Synapse SQL Pool

- Select '+' and select 'Sink' in the 'Destination' section. Last one in the list.
- Enter 'sinkToSynapseDb' for Output stream name
- Click the '+' next to Dataset to create new Azure Synapse Analytics (formerly SQL DW) dataset

New dataset

In pipeline activities and data flows, reference a dataset to specify the location and structure of your data within a data store. Learn more ⬚

Select a data store

🔍 Search

All   Azure   Database   File   Generic protocol   NoSQL   Services and apps

Azure Blob Storage	Azure Cosmos DB (SQL API)	Azure Data Lake Storage Gen1
Azure Data Lake Storage Gen2	Azure SQL Database	Azure SQL Database Managed Instance
Azure Synapse Analytics (formerly SQL DW)	Snowflake	Amazon Marketplace Web Service

- For dataset name, enter 'Synape_HR_Employee_table'
- Check the 'Edit' and enter schema: dbo and table name: employee
- Click 'OK'

Set properties

Name

Synapse_HR_Employee_table

Linked service *

SynapseDBHR   ✎

🔘 Select from existing table   ⚪ Create new table

Table name

dbo   .   employee

☑ Edit

Import schema

🔘 From connection/store   ⚪ None

▷ Advanced

- Click the 'Settings' tab
- Check the 'Recreate table' radio button
- Uncheck the 'Enable staging' checkbox

Task-5: Build a pipeline to connect Copy and Data flow activities and Test the Execution

We have a create copy activity to ingest the data from Oracle to Azure Storage and Dataflow activity to secure PII data. You will connect both activities to build a pipeline to execute the end to end functionality.

1. Connect Copy activity with Data Flow activity

2. Trigger the pipeline execution.

- Click on 'Debug' to test the data transformation to secure the PII data and storing it to Synapse SQL Analytics database.

- Execution will be queued and wait till it finishes.

4. Verify the transformed data was inserted into Azure Synapse Analytics SQL Pool.

- Select 'Firewalls and virtual networks' from 'Security' section
- Click the '+ Add client IP'
- Click on 'Save' button after you added the rule.

- Switch to Azure Services and access the Synapse SQL Analytics to verify the employee data with secured phone numbers
- Select the Overview and scroll down to select the database
- Select the 'Query editor' and enter the login info again
- Login with 'azureadmin' and password 'Ataadf123!' if needed.
- With successful login, you will see the tables and views.
- Access synapse SQL Analytics 'ataadflabsqldb' and select 'Query editor' from the 'Common Tasks' section.

- Access synapse SQL Analytics 'ataadflabsqldb' and select 'Query editor' from the 'Common Tasks' section.
- Enter 'select * from dbo.employee' to view the ingested data from the Oracle database.
- You should see all the employee data with the secured phone number.

Congratulations!! You have successfully ingested the data from Oracle to Azure Datalake, transformed the data as per your needs such as securing the data and stored the data into Azure Synapse Analytics for your business analytics needs!!!