**Name -Monika Gariya**

**Email-** [**monikagariya2023@gmail.com**](mailto:monikagariya2023@gmail.com)

**Data Engineering Batch 1**

**Date – 20-02-2024**

**TOPIC – Azure Data Factory, Copying Data from blob to blob storage**

**What is Azure Data Factory?**

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

**Azure Data Factory use cases**

ADF can be used for:

* Supporting data migrations
* Getting data from a client’s server or online data to an Azure Data Lake
* Carrying out various data integration processes
* Integrating data from different ERP systems and loading it into Azure Synapse for reporting

**How does Azure Data Factory work?**

Azure Data Factory pipelines (data-driven workflows) typically perform three steps.

### **Step 1: Connect and Collect**

Connect to all the required sources of data and processing such as SaaS services, file shares, FTP, and web services. Then,  move the data as needed to a centralized location for subsequent processing by using the Copy Activity in a data pipeline to move data from both on-premise and cloud source data stores to a centralization data store in the cloud for further analysis.

### **Step 2: Transform and Enrich**

Once data is present in a centralized data store in the cloud, it is transformed using compute services such as HDInsight Hadoop, Spark, Azure Data Lake Analytics, and Machine Learning.

### **Step 3: Publish**

Deliver transformed data from the cloud to on-premise sources like SQL Server or keep it in your cloud storage sources for consumption by BI and analytics tools and other applications.

**Data migration activities with Azure Data Factory**

By using Microsoft Azure Data Factory, data migration occurs between two cloud data stores and between an on-premise data store and a cloud data store.

Copy Activity in Azure Data Factory copies data from a source data store to a sink data store.

Azure Data Factory supported data stores for data movement activities, refer to Azure documentation for data movement activities.

Azure Data Factory supports transformation activities such as Hive, MapReduce, Spark, etc that can be added to pipelines either individually or chained with other activities.

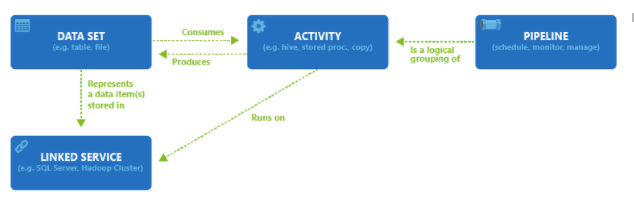
**Azure Data Factory key components**

Azure Data Factory has four key components that work together to define input and output data, processing events, and the schedule and resources required to execute the desired data flow:

* **Datasets represent data structures within the data stores.** An input dataset represents the input for an activity in the pipeline. An output dataset represents the output for the activity. For example, an Azure Blob dataset specifies the blob container and folder in the Azure Blob Storage from which the pipeline should read the data. Or, an Azure SQL Table dataset specifies the table to which the output data is written by the activity.
* **A pipeline is a group of activities.** They are used to group activities into a unit that together performs a task. A data factory may have one or more pipelines. For example, a pipeline could contain a group of activities that ingests data from an Azure blob and then runs a Hive query on an HDInsight cluster to partition the data.
* **Activities define the actions to perform on your data.** Currently, Azure Data Factory supports two types of activities: data movement and data transformation.
* **Linked services define the information needed for Azure Data Factory to connect to external resources.** For example, an Azure Storage linked service specifies a connection string to connect to the Azure Storage account.

**How the Azure Data Factory components work together**

The following schema shows us the relationships between the **Dataset**, **Activity**, **Pipeline**, and **Linked Services** components:



The following tools or APIs to create data pipelines in Azure Data Factory:

* Azure portal
* Visual Studio
* PowerShell
* .NET API
* REST API
* Azure Resource Manager template

**COPYING DATA FROM BLOB TO BLOB STORAGE**

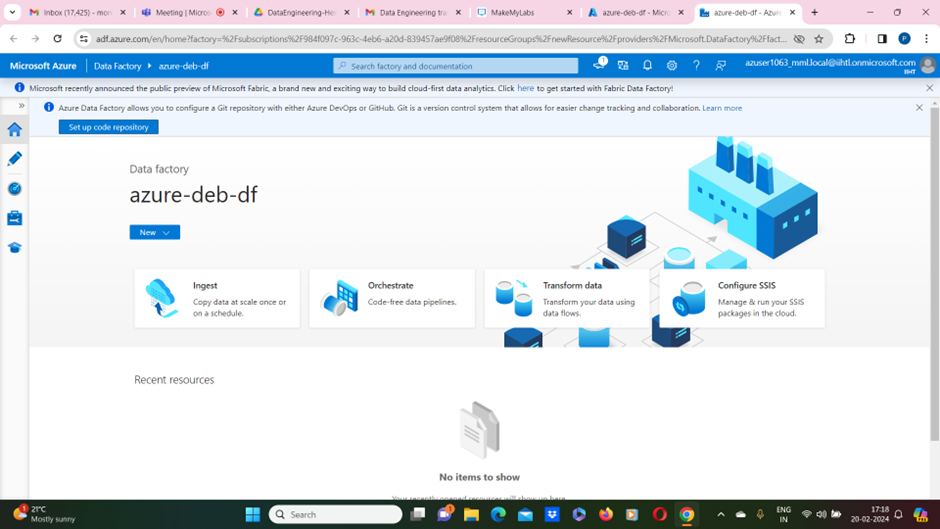
A new blob storage account will be created in the new resource group, and the moviesDB2.csv file will be stored in a folder called **input** in the blob storage.

**Use the copy data tool to copy data**

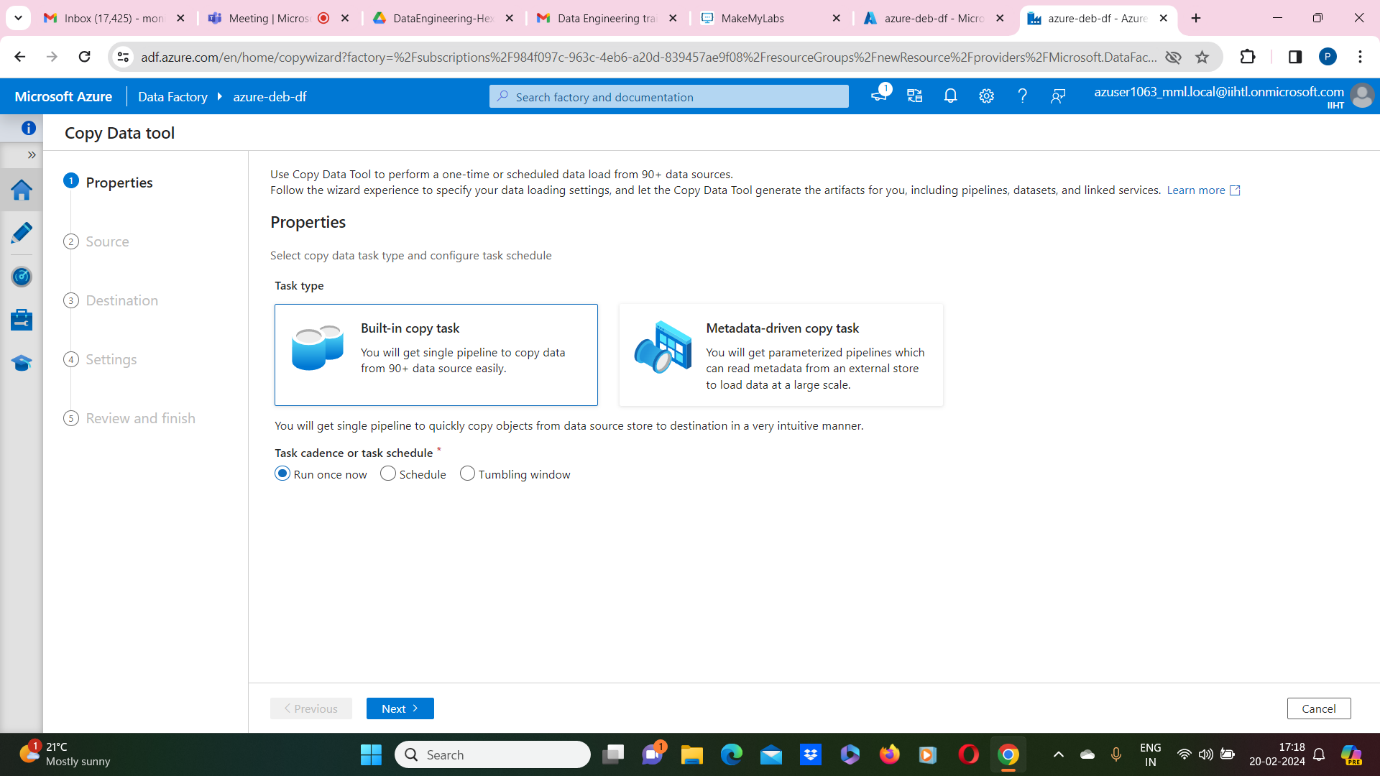
The steps below will walk you through how to easily copy data with the copy data tool in Azure Data Factory.

**Step 1: Start the copy data Tool**

1. On the home page of Azure Data Factory, select the **Ingest** tile to start the Copy Data tool.

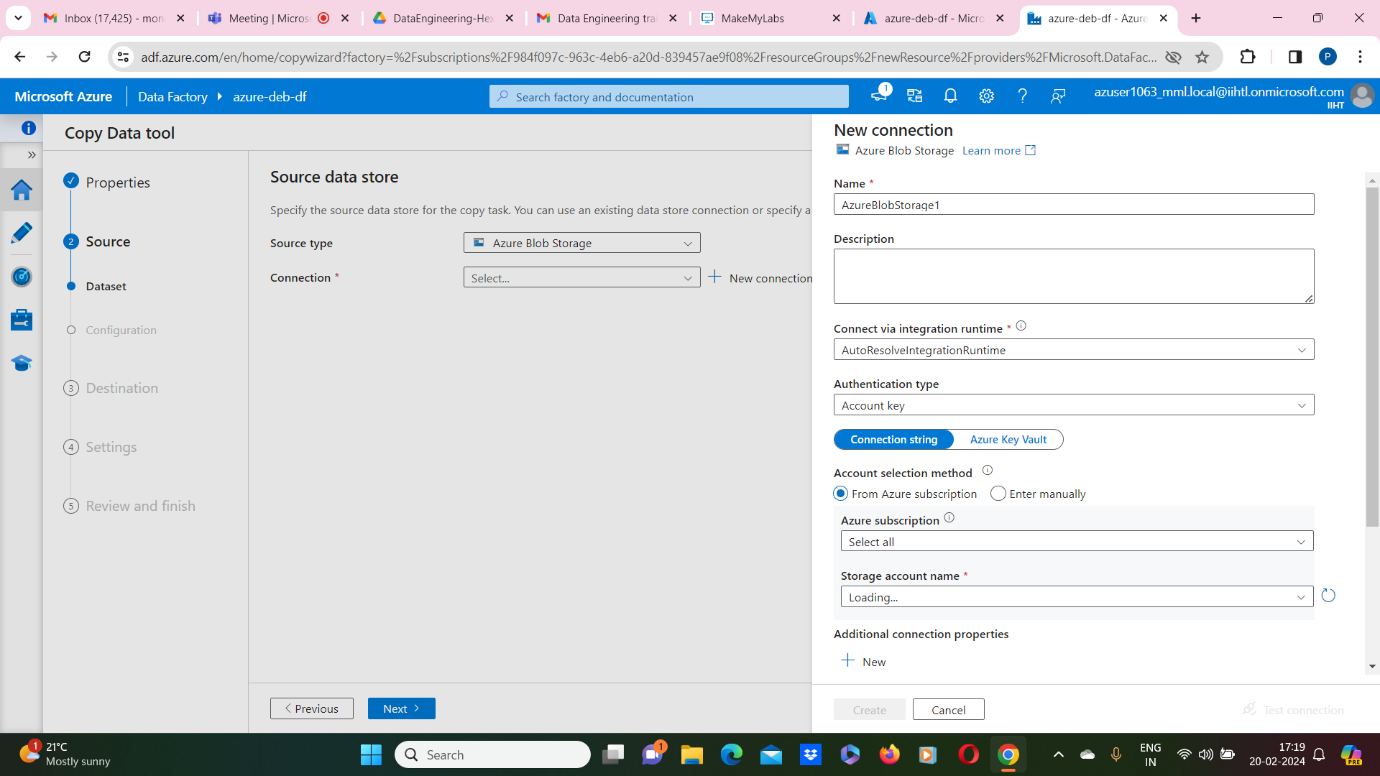
****

2)On the **Properties** page of the Copy Data tool, choose **Built-in copy task** under **Task type**, then select **Next**.



**Step 2: Complete source configuration**

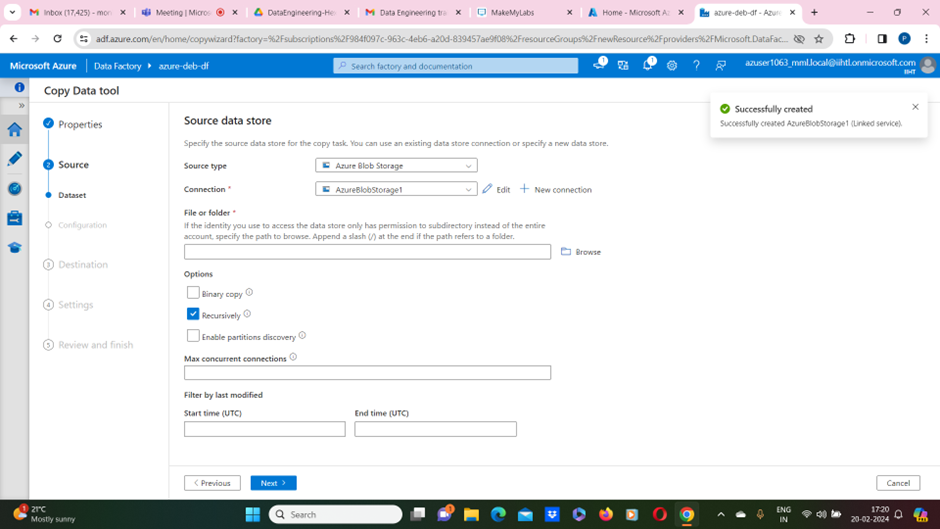
1. Click **+ Create new connection** to add a connection.
2. Select the linked service type that you want to create for the source connection. In this tutorial, we use **Azure Blob Storage**. Select it from the gallery, and then select **Continue**.



1. Select the newly created connection in the **Connection** block.

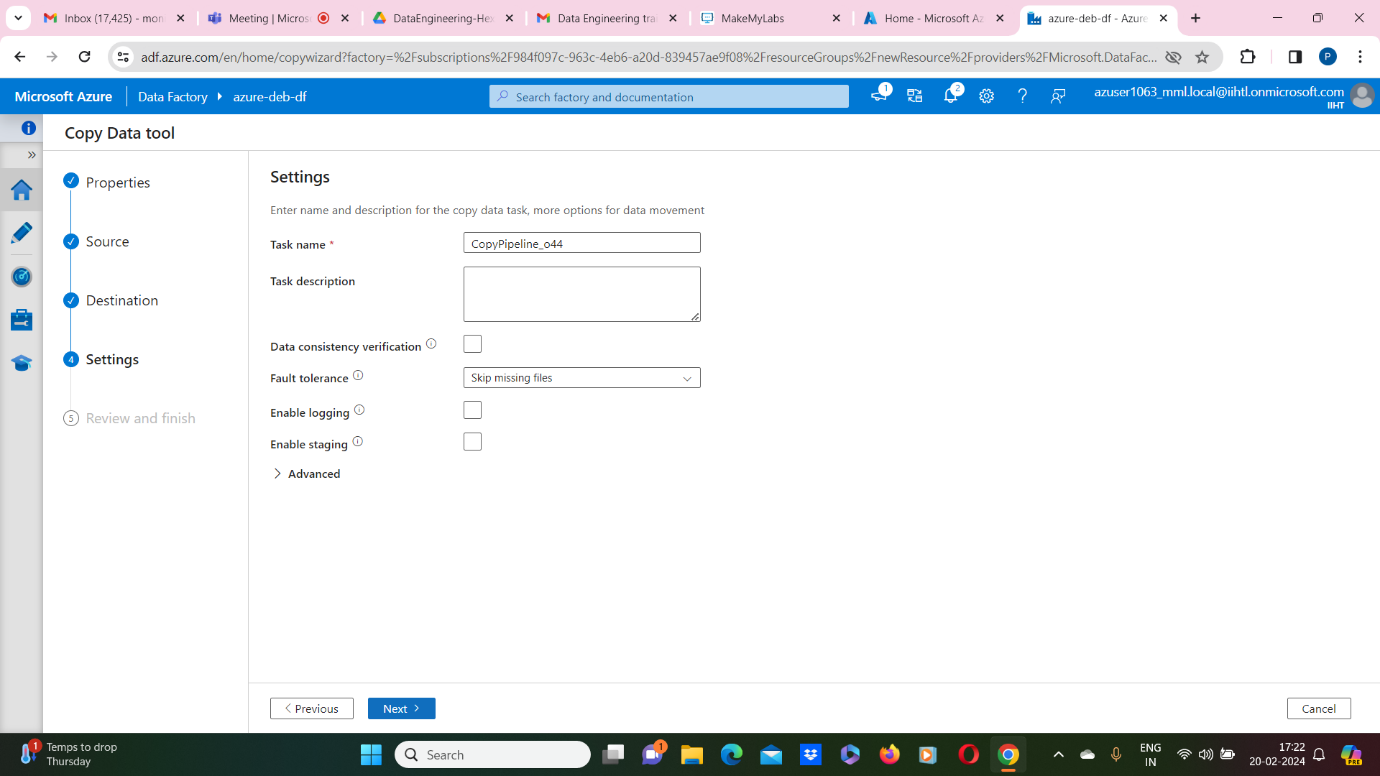
2.In the **File or folder** section, select **Browse** to navigate to the **adftutorial/input** folder, select the **emp.txt** file, and then click **OK**.

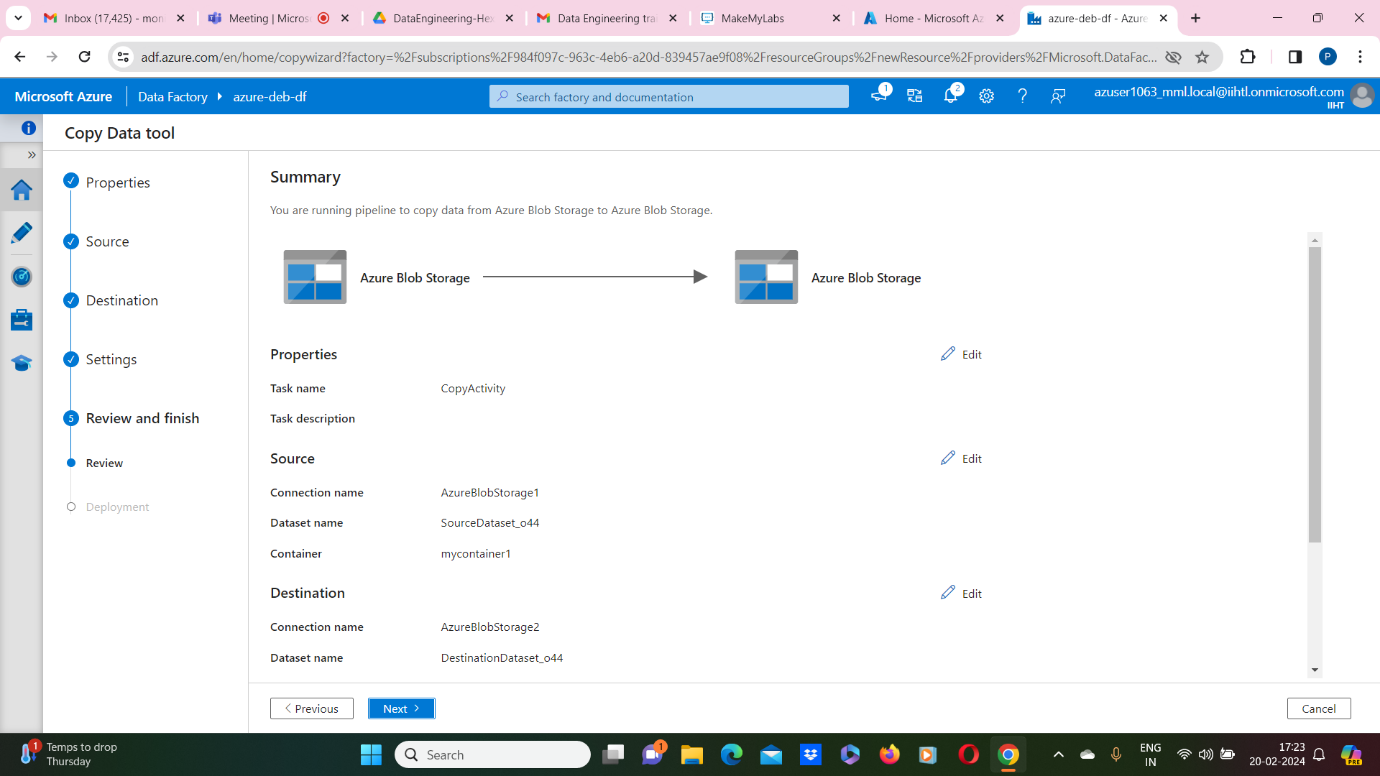
3.Select the **Binary copy** checkbox to copy file as-is, and then select **Next**.

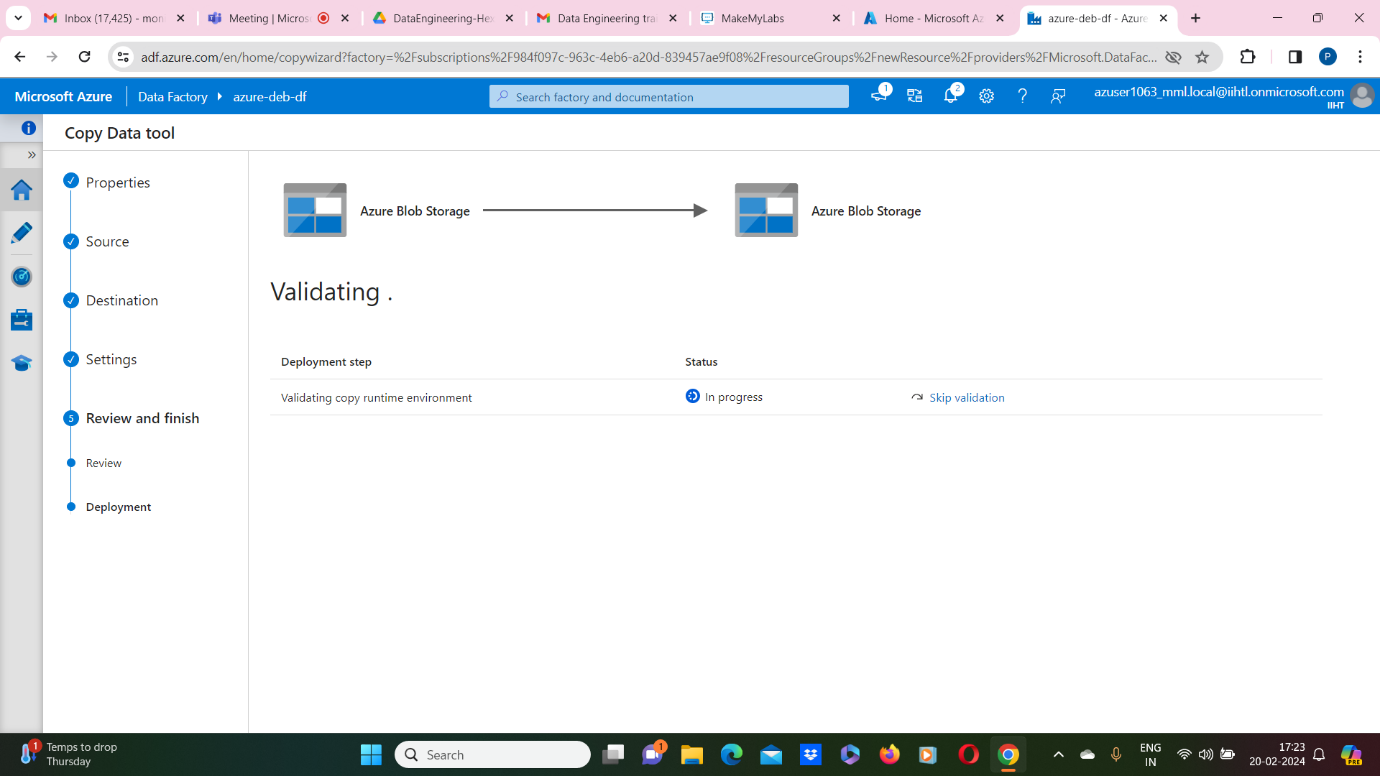
****

**Step 4: Review all settings and deployment**

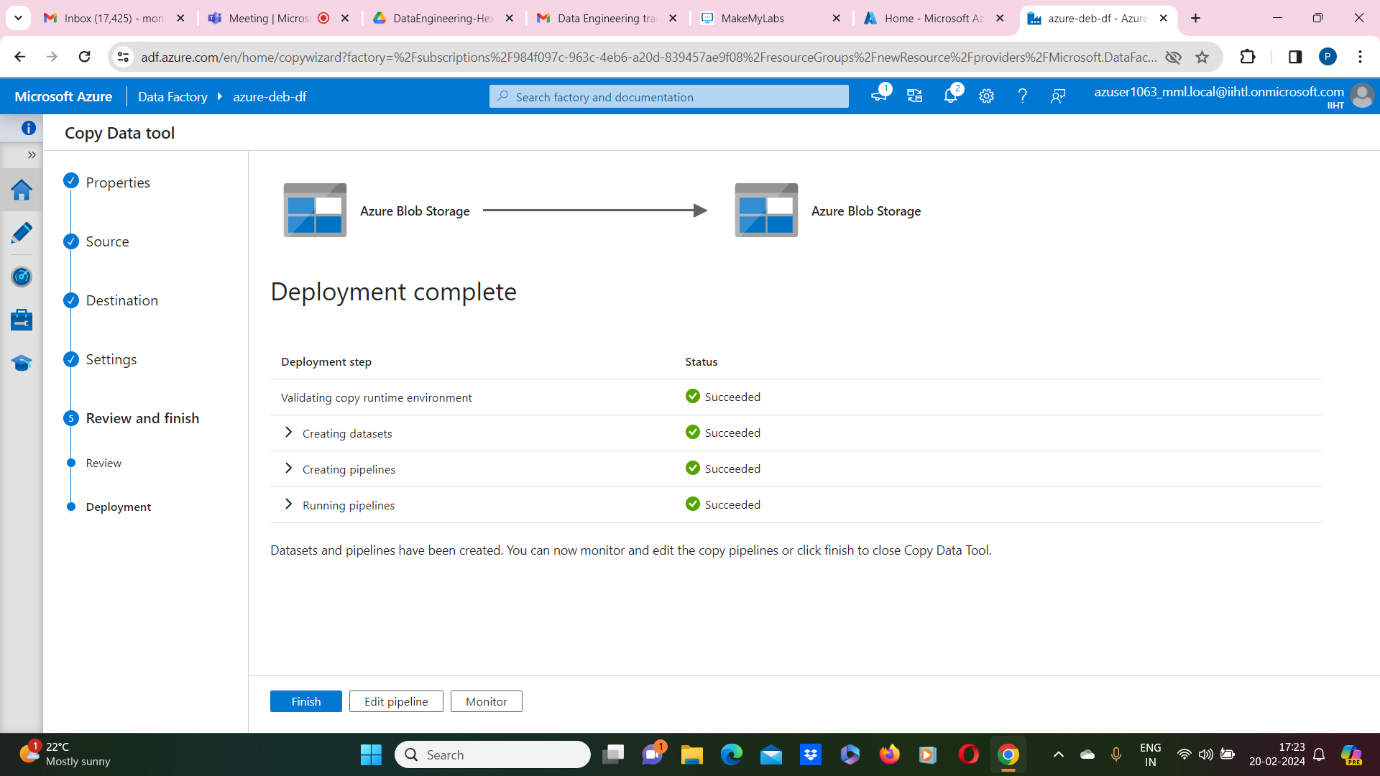
On the **Settings** page, specify a name for the pipeline and its description, then select **Next** to use other default configurations







1. On the **Summary** page, review all settings, and select **Next**.
2. On the **Deployment complete** page, select **Monitor** to monitor the pipeline that you created.



**Step 5: Monitor the running results**

1. The application switches to the **Monitor** tab. You see the status of the pipeline on this tab. Select **Refresh** to refresh the list. Click the link under **Pipeline name** to view activity run details or rerun the pipeline.

