

Project -1
Parallel Processing Data Pipeline

By – Mitushi Vishwakarma
email id : mitushi.vish@gmail.com
Data Engineering Batch

Parallel Processing Data Pipeline

Household Power Consumption

Project Overview:

The objective of this project is to develop a robust and efficient data pipeline in Azure for processing and analyzing household power consumption data. The pipeline will leverage parallel processing capabilities to enhance performance and scalability, ultimately loading the processed data into Azure Data Lake Storage for further analysis and insights generation.

Project Requirements:

1. Data Integration and Ingestion:

- Implement data ingestion mechanisms to raw household power consumption dataset from <https://data.world/databeats/household-power-consumption> and load it into Azure Data Lake Gen2 Storage Account.
- Ensure data ingestion processes are reliable, scalable, and capable of handling large volumes of data.

2. Data Transformation and Preprocessing:

- Develop data transformation logic to preprocess and cleanse raw power consumption data.
- Implement data quality checks and validation processes to ensure data accuracy and integrity.
- Perform necessary data transformations such as aggregation, and enrichment to prepare the data for analysis.

3. Parallel Processing with Azure Databricks:

- Utilize Azure Databricks for parallel processing of power consumption data to improve performance and scalability.
- Design and implement parallel processing workflows using Spark clusters in Azure Databricks.

- Optimize cluster configurations and resource allocation for efficient parallel processing.

4. Integration with Azure Data Lake Gen2:

- Configure Azure Data Lake Gen2 Storage Account as the destination for storing processed power consumption data.
- Establish seamless integration between Azure Databricks and Azure Data Lake Gen2 for data transfer and storage.

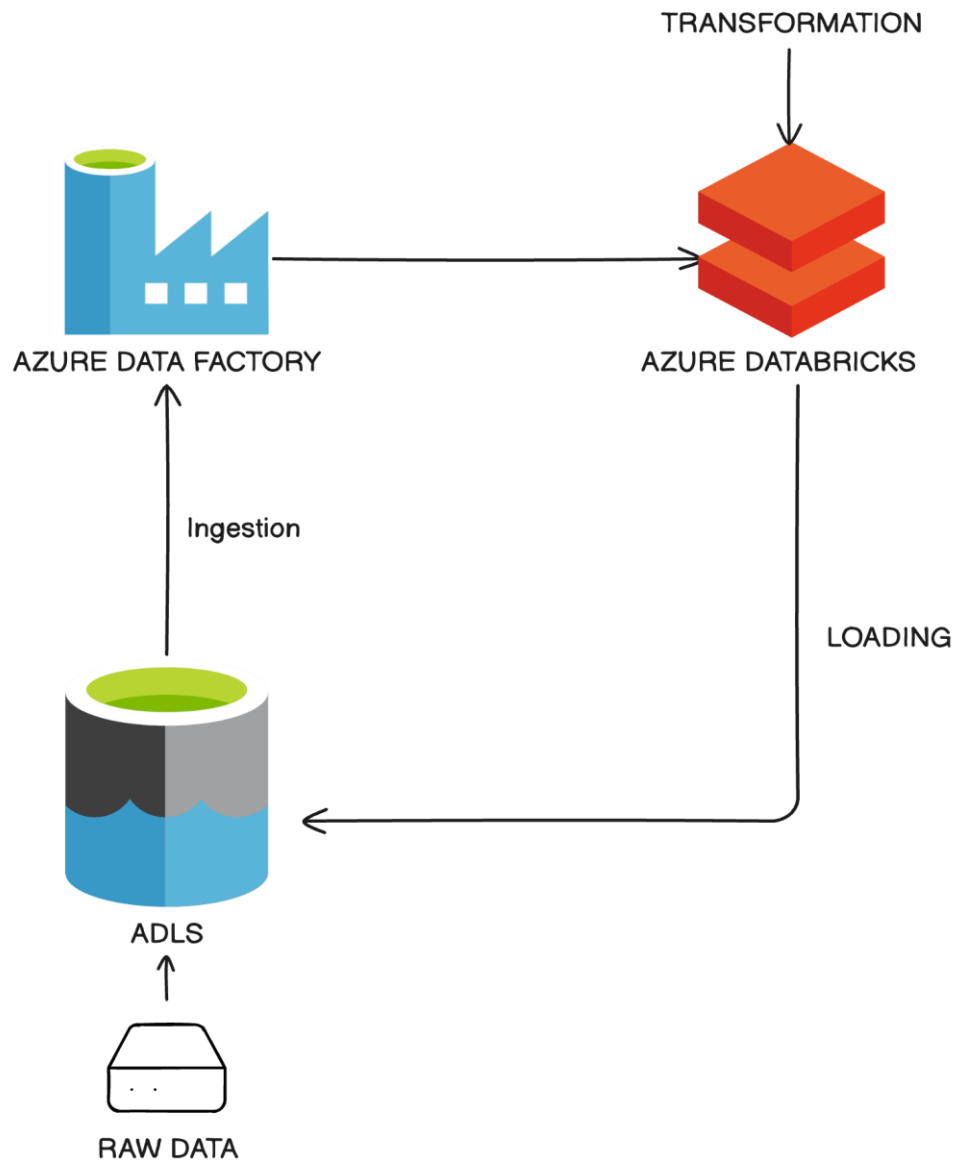
5. Orchestration with Azure Data Factory:

- Develop data pipelines within Azure Data Factory to orchestrate data movement and transformation tasks.
- Define data ingestion activities to extract data from source systems and load it into Azure Databricks for processing.
- Implement data transformation activities to preprocess and cleanse the data within Databricks, then store the processed data in Azure Data Lake Gen2.

6. Monitoring and Logging:

- Implement monitoring and logging mechanisms to track the performance and health of the data pipeline.
- Monitor key metrics such as data ingestion rates, processing times, and error rates.
- Configure alerts and notifications for proactive monitoring and issue resolution.

Architecture Diagram:



Azure Resources used for the Project:

1. Azure Databricks

Azure Databricks is a fast, easy, and collaborative Apache Spark-based analytics platform optimized for Azure. It provides a fully managed, cloud-based environment that integrates seamlessly with other Azure services, allowing data engineers, data scientists, and analysts to collaborate on big data and machine learning projects.

2. Azure Data Lake Gen 2 Storage Account

Azure Data Lake Storage Gen2 is a scalable and secure data lake solution built on top of Azure Blob Storage and the Azure Blob Storage file system (ABFS). It combines the capabilities of Azure Blob Storage with hierarchical namespace and file system semantics, providing a unified data lake storage solution for big data analytics and data warehousing workloads.

3. Azure Data Factory

Azure Data Factory is a cloud-based data integration service that allows you to create, schedule, and manage data pipelines for moving and transforming data across various sources and destinations. It enables you to orchestrate and automate data workflows, facilitating data integration, transformation, and loading tasks.

About Dataset:

Individual household electric power consumption dataset collected via submeters placed in 3 distinct areas of a home

Data Set Information

This household electricity consumption dataset contains 260,640 measurements gathered between January 2007 and June 2007 (6 months). It is a subset of a larger, original archive that contains 2,075,259 measurements gathered between December 2006 and November 2010 (47 months).

Attribute Information

1. **date:** Date in format dd/mm/yyyy
2. **time:** time in format hh:mm:ss
3. **global_active_power:** household global minute-averaged active power (in kilowatt)
4. **global_reactive_power:** household global minute-averaged reactive power (in kilowatt)
5. **voltage:** minute-averaged voltage (in volt)
6. **global_intensity:** household global minute-averaged current intensity (in ampere)
7. **sub_metering_1:** energy sub-metering No. 1 (in watt-hour of active energy). It corresponds to the kitchen, containing mainly a dishwasher, an oven and a microwave (hot plates are not electric but gas powered).
8. **sub_metering_2:** energy sub-metering No. 2 (in watt-hour of active energy). It corresponds to the laundry room, containing a washing-machine, a tumble-drier, a refrigerator and a light.
9. **sub_metering_3:** energy sub-metering No. 3 (in watt-hour of active energy). It corresponds to an electric water-heater and an air-conditioner.

Dataset Snapshot

household_power_consumption.csv X									
household_power_consumption.csv > data									
1	Date	Time	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3
2	01-01-2007	00:00:00	2.58	0.136	241.97	10.6	0	0	0
3	01-01-2007	00:01:00	2.552	0.1	241.75	10.4	0	0	0
4	01-01-2007	00:02:00	2.55	0.1	241.64	10.4	0	0	0
5	01-01-2007	00:03:00	2.55	0.1	241.71	10.4	0	0	0
6	01-01-2007	00:04:00	2.554	0.1	241.98	10.4	0	0	0
7	01-01-2007	00:05:00	2.55	0.1	241.83	10.4	0	0	0
8	01-01-2007	00:06:00	2.534	0.096	241.07	10.4	0	0	0
9	01-01-2007	00:07:00	2.484	0	241.29	10.2	0	0	0
10	01-01-2007	00:08:00	2.468	0	241.23	10.2	0	0	0
11	01-01-2007	00:09:00	2.486	0	242.18	10.2	0	0	0
12	01-01-2007	00:10:00	2.492	0	242.46	10.2	0	0	0
13	01-01-2007	00:11:00	2.5	0	242.88	10.2	0	0	0
14	01-01-2007	00:12:00	2.494	0	242.57	10.2	0	0	0
15	01-01-2007	00:13:00	2.492	0	242.41	10.2	0	0	0
16	01-01-2007	00:14:00	2.48	0	241.81	10.2	0	0	0
17	01-01-2007	00:15:00	2.478	0	241.73	10.2	0	0	0
18	01-01-2007	00:16:00	2.47	0	241.29	10.2	0	0	0
19	01-01-2007	00:17:00	2.466	0	241.11	10.2	0	0	0
20	01-01-2007	00:18:00	2.456	0	240.59	10.2	0	0	0
21	01-01-2007	00:19:00	2.46	0	240.83	10.2	0	0	0
22	01-01-2007	00:20:00	2.544	0.092	240.9	10.6	0	0	0
23	01-01-2007	00:21:00	2.55	0.116	241.15	10.4	0	1	0
24	01-01-2007	00:22:00	2.554	0.118	241.55	10.6	0	1	0
25	01-01-2007	00:23:00	2.65	0.218	241.67	11	0	2	0
26	01-01-2007	00:24:00	2.682	0.258	242.45	11	0	1	0
27	01-01-2007	00:25:00	2.66	0.252	241.6	11	0	1	0

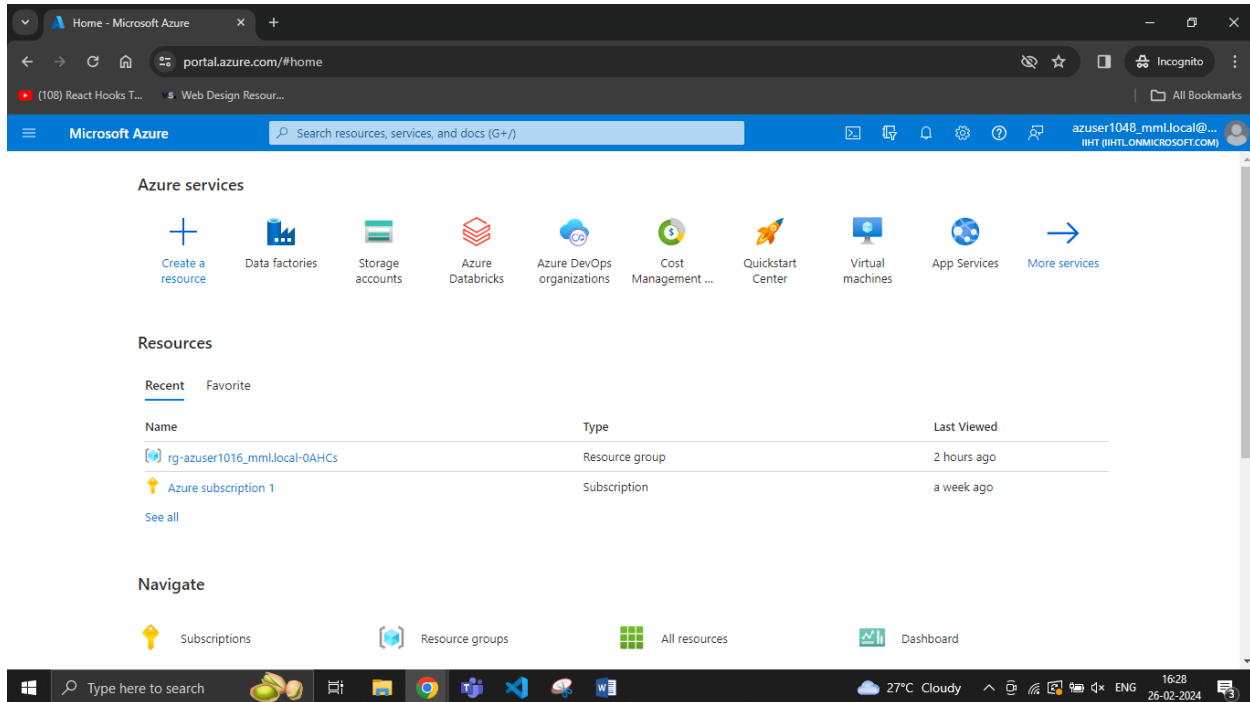
How it works:

1. **Set up Azure Data Lake Gen2:** Ensure that you have an Azure Data Lake Gen2 storage account created and properly configured.
2. **Create an Azure Databricks Workspace:** Set up an Azure Databricks workspace in your Azure portal. You can follow the documentation to create a Databricks workspace if you haven't already done so.
3. **Create an Azure Data Factory:** Create an Azure Data Factory instance in your Azure portal if you haven't already. ADF will orchestrate the data loading process and integrate with Databricks.
4. **Link Azure Databricks to Azure Data Factory:** In Azure Data Factory, create a linked service for Azure Databricks. This involves providing the necessary connection details and authentication credentials to access your Databricks workspace.
5. **Develop Data Processing Logic in Databricks:** In your Databricks workspace, develop the necessary data processing logic using Apache Spark. This can include data transformation, cleansing, enrichment, etc. Ensure that your Spark job is optimized for parallel processing to leverage Databricks' capabilities effectively.
6. **Create Data Factory Pipeline:** In Azure Data Factory, create a pipeline that orchestrates the data loading process. Add activities to trigger the Databricks notebook job, specifying any parameters or dependencies required for execution.
7. **Configure Parallelism:** Within your Data Factory pipeline, configure parallelism settings to optimize data loading performance. This may involve partitioning data, parallelizing processing tasks, and tuning resource allocation in Databricks to maximize throughput.
8. **Schedule and Monitor:** Schedule the Data Factory pipeline to run at specified intervals or trigger it based on event triggers. Monitor the execution of the pipeline and Databricks jobs to ensure they run smoothly and meet performance expectations.

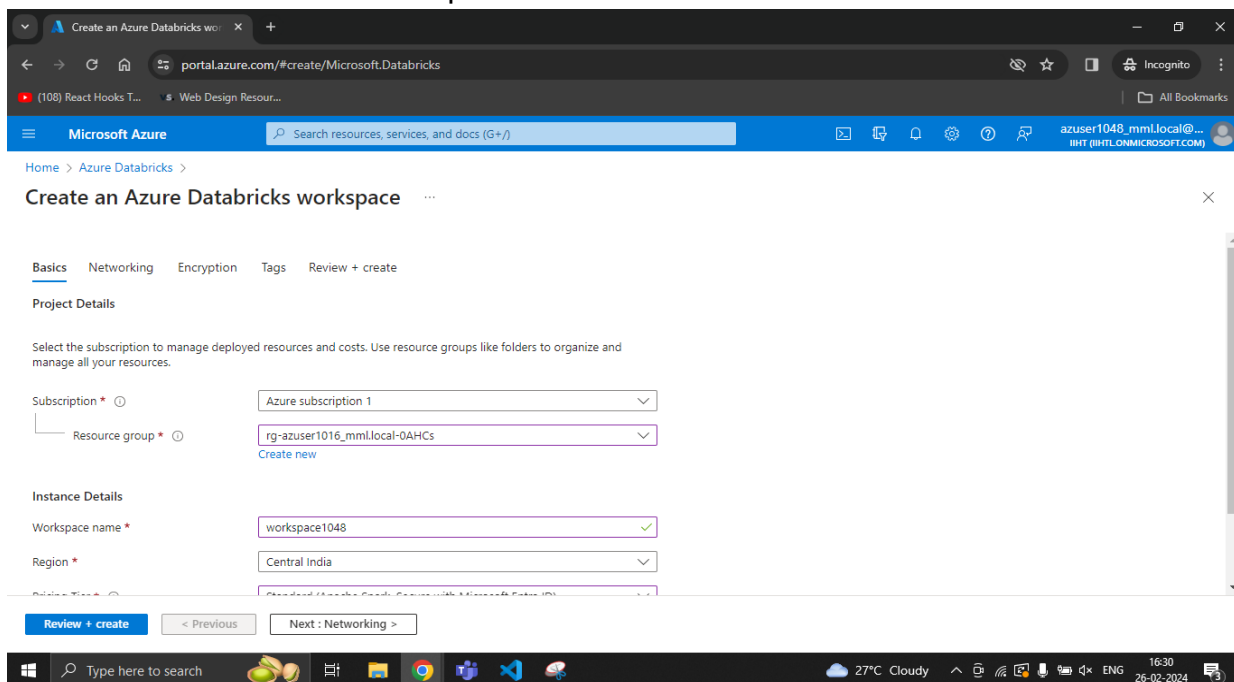
Tasks Performed:

1. Create Azure Databricks workspace, and create a compute and a notebook

Open Microsoft Azure



Create Azure Databricks workspace



Go to the resource

The screenshot shows the Microsoft Azure portal interface. The browser address bar displays the URL: `portal.azure.com/#view/HubsExtension/DeploymentDetailsBlade/~/overview/id/%2Fsubscriptions%2F984f097c-963c-4eb6-a20d-839457ae9f08%2FresourceGroups/rg-azuser1016_mml.local-0AHCs_workspace1048`. The page title is "rg-azuser1016_mml.local-0AHCs_workspace1048 | Overview". The left sidebar contains a search bar and a list of navigation items: Overview, Inputs, Outputs, and Template. The main content area shows a green checkmark and the text "Your deployment is complete". Below this, the deployment details are listed: Deployment name: rg-azuser1016_mml.local-0AHCs_wor..., Subscription: Azure subscription 1, Correlation ID: ff2e179a-42c0-47b3-93c7-79936afde..., and Resource group: rg-azuser1016_mml.local-0AHCs. There are buttons for "Go to resource" and "Give feedback". On the right, there are sections for "Cost management", "Microsoft Defender for Cloud", and "Free Microsoft tutorials".

Launch Workspace

The screenshot shows the Microsoft Azure portal interface for the workspace "workspace1048". The browser address bar displays the URL: `portal.azure.com/#@iitl.onmicrosoft.com/resource/subscriptions/984f097c-963c-4eb6-a20d-839457ae9f08/resourceGroups/rg-azuser1016_mml.local-0AHCs_workspace1048`. The page title is "workspace1048 | Overview". The left sidebar contains a search bar and a list of navigation items: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Virtual Network Peerings, Encryption, Networking, Properties, Locks, Automation, and CLI / PS. The main content area shows the workspace details: Status: Active, Resource group: rg-azuser1016_mml.local-0AHCs, Location: Central India, Subscription: Azure subscription 1, Subscription ID: 984f097c-963c-4eb6-a20d-839457ae9f08, and Tags: Add tags. There are buttons for "Launch Workspace" and "Upgrade to Premium".

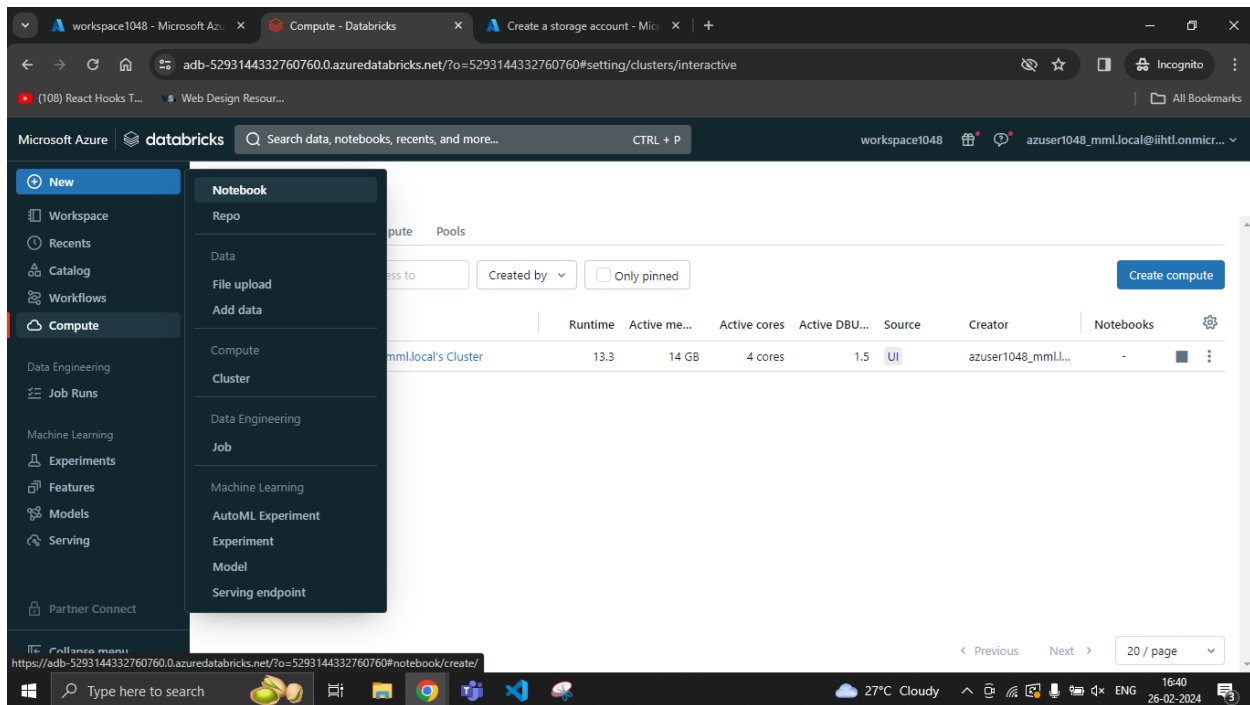
Create a cluster

The screenshot shows the 'Create Cluster' page in the Databricks workspace. The left sidebar contains navigation options like 'Workspace', 'Catalog', 'Workflows', 'Compute', 'Data Engineering', 'Job Runs', 'Machine Learning', 'Experiments', 'Features', 'Models', 'Serving', 'Partner Connect', and 'Collapse menu'. The main content area is titled 'azuser1048_mml.local's Cluster'. It includes options for 'Multi node' (selected) and 'Single node', 'Access mode' (Single user access), and 'Single user' (selected). The 'Performance' section shows 'Databricks runtime version' (Runtime: 13.3 LTS (Scala 2.12, Spark 3.4.1)), 'Node type' (Standard_DS3_v2, 14 GB Memory, 4 Cores), and 'Terminate after' (120 minutes of inactivity). A 'Summary' box on the right lists '1 Driver', '14 GB Memory, 4 Cores', 'Runtime 13.3.x-scala2.12', and 'Photon Standard_DS3_v2 1.5 DBU/h'. The bottom of the page shows a Windows taskbar with the date 25-02-2024 and time 16:35.

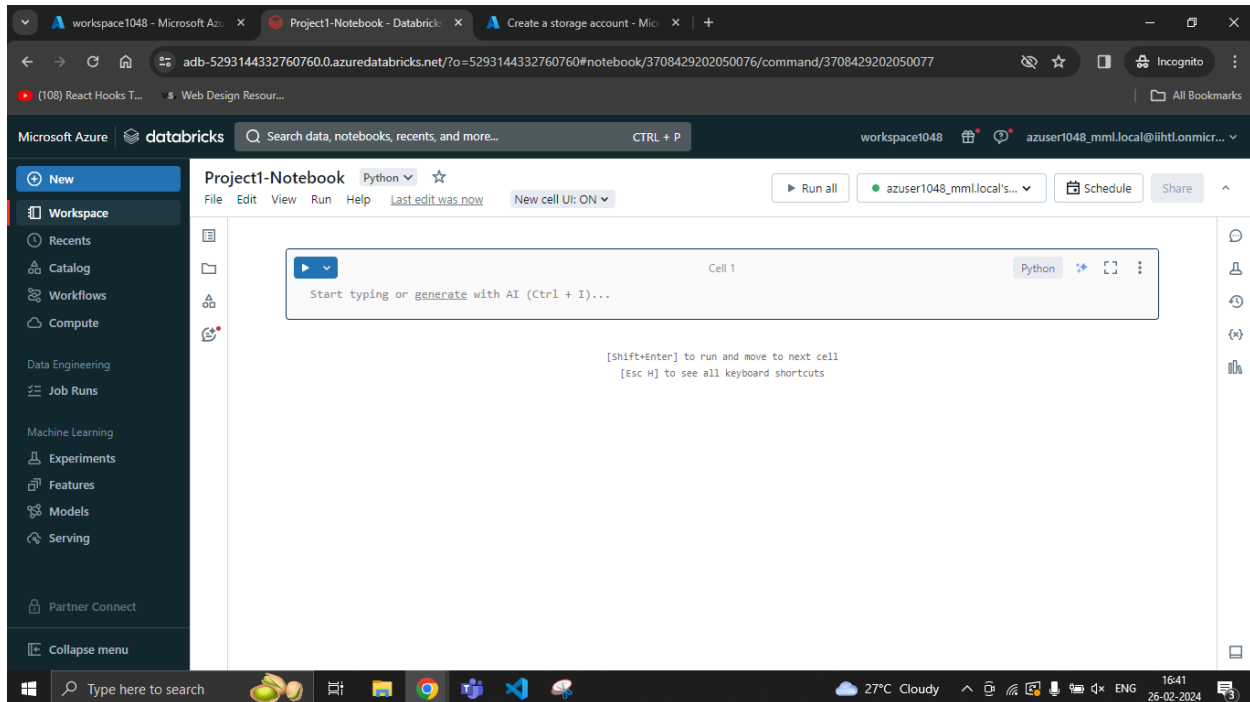
Cluster has been successfully created

The screenshot shows the 'Compute' page in the Databricks workspace. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Compute' and shows a table of clusters. The table has columns for 'State', 'Name', 'Runtime', 'Active me...', 'Active cores', 'Active DBU...', 'Source', 'Creator', and 'Notebooks'. The first row shows a cluster named 'azuser1048_mml.local's Cluster' with a state of 'Running', runtime of '13.3', active memory of '14 GB', active cores of '4 cores', active DBU of '1.5', source of 'UI', creator of 'azuser1048_mmlL...', and no notebooks. The bottom of the page shows a Windows taskbar with the date 25-02-2024 and time 16:39.

Create a new notebook



Rename the notebook



2. Create an Azure Data Lake Gen 2 Storage Account, create a container inside it and upload the raw data.

Create Azure Data Lake Storage Account

The screenshot shows the 'Create a storage account' page in the Azure portal. The browser address bar shows the URL: `portal.azure.com/#create/Microsoft.StorageAccount-ARM`. The page has a blue header with the Microsoft Azure logo and a search bar. Below the header, there's a breadcrumb trail: Home > Storage accounts >. The main heading is 'Create a storage account'. There are tabs for Basics, Advanced, Networking, Data protection, Encryption, Tags, and Review. The Basics tab is selected. The page content describes Azure Storage as a Microsoft-managed service. Under 'Project details', there are two dropdown menus: 'Subscription' (set to 'Azure subscription 1') and 'Resource group' (set to 'rg-azuser1016_mml.local-0AHCs'). At the bottom, there are buttons for 'Review', '< Previous', and 'Next : Advanced >'. A 'Give feedback' link is also present.

ADLS account has been successfully created. Go to the resource.

The screenshot shows the 'Deployment Details' page for the storage account 'adlsstorageaccount1048_1708945910703'. The browser address bar shows the URL: `portal.azure.com/#view/HubsExtension/DeploymentDetailsBlade/~/overview/id/%2Fsubscriptions%2F984f097c-963c-4eb6-a20d-839457ae9f08%2Fresourcegroups%2Frg-azuser1016_mml.local-0AHCs%2Fproviders%2FMicrosoft.Storage%2Fstorageaccounts%2Fadlsstorageaccount1048_1708945910703`. The page has a blue header with the Microsoft Azure logo and a search bar. Below the header, there's a breadcrumb trail: Home >. The main heading is 'adlsstorageaccount1048_1708945910703 | Overview'. There's a search bar and a list of actions: Delete, Cancel, Redeploy, Download, and Refresh. The 'Overview' tab is selected. The page content shows a green checkmark and the text 'Your deployment is complete'. Below this, there's a table with deployment details: Deployment name, Subscription, Resource group, Start time, and Correlation ID. There's a 'Go to resource' button. At the bottom, there's a 'Give feedback' link. A notification banner at the top right says 'Deployment succeeded' and 'Deployment 'adlsstorageaccount1048_1708945910703' to resource group 'rg-azuser1016_mml.local-0AHCs' was successful.' There are also links for 'Go to resource' and 'Pin to dashboard'. On the right side, there are recommendations for 'Cost Management', 'Microsoft Defender for Cloud', and 'Free Microsoft tutorials'.

Open ADLS and create a container

The screenshot shows the Microsoft Azure portal interface. The main navigation pane on the left lists various services under 'Data storage', with 'Containers' selected. The main content area displays the 'adlsstorageaccount1048' storage account overview, including a table of containers. A 'New container' dialog box is open on the right, showing the container name 'powerconsumptiondatafiles' and the anonymous access level 'Container (anonymous read access for containers and blobs)'. A warning message is displayed, stating: 'All container and blob data can be read by anonymous request. Clients can enumerate blobs within the container by anonymous request, but cannot enumerate containers within the storage account. Anonymous access bypasses Access Control List (ACL) settings.' The 'Create' button is visible at the bottom of the dialog.

Upload data file in the container

The screenshot shows the Microsoft Azure portal interface. The main navigation pane on the left lists various services under 'Data storage', with 'Containers' selected. The main content area displays the 'powerconsumptiondatafiles' container overview, including a table of blobs. A 'Upload blob' dialog box is open on the right, showing the container name 'powerconsumptiondatafiles' and the authentication method 'Access key'. The dialog shows a search bar for blobs and a table with 'No results'. The 'Upload' button is visible at the bottom of the dialog.

Raw data file has been uploaded successfully

Microsoft Azure portal view of the 'powerconsumptiondatafiles' container. The container is located under 'adlsstorageaccount1048_1708945910703 | Overview' and 'adlsstorageaccount1048 | Containers'. The container's authentication method is 'Access key' and its location is 'powerconsumptiondatafiles'. A table lists the contents of the container, showing a single file named 'household_power_co...' with a size of 12.04 MiB and a lease state of 'Available'.

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
household_power_co...	2/26/2024, 4:48:29 PM	Hot (Inferred)		Block blob	12.04 MiB	Available

Create another container in which the processed data will be loaded.

Microsoft Azure portal view of the 'New container' dialog box. The dialog box is open over the 'adlsstorageaccount1048 | Containers' page. The 'Name' field is set to 'processeddatafiles'. The 'Anonymous access level' is set to 'Container (anonymous read access for containers and blobs)'. A warning message states: 'All container and blob data can be read by anonymous request. Clients can enumerate blobs within the container by anonymous request, but cannot enumerate containers within the storage account. Anonymous access bypasses Access Control List (ACL) settings.' The 'Create' button is visible at the bottom of the dialog box.

Container successfully created

The screenshot shows the Microsoft Azure portal interface. The main heading is 'adlsstorageaccount1048 | Containers'. A notification bubble in the top right corner states: 'Successfully created storage container' and 'Successfully created storage container 'processeddatafiles''. The left sidebar contains navigation links: Overview, Activity log, Tags, Diagnose and solve problems, Access Control (IAM), Data migration, Events, Storage browser, Data storage, Containers (selected), File shares, Queues, and Tables. The main content area shows a table of containers with columns: Name, Last modified, Anonymous access level, and Lease state. The table lists three containers: '\$logs', 'powerconsumptiondatafiles', and 'processeddatafiles'. The bottom of the screen shows a Windows taskbar with the search bar, task icons, and system tray information (27°C Mostly cloudy, 18:19, 27-02-2024).

Name	Last modified	Anonymous access level	Lease state
<input type="checkbox"/> \$logs	2/27/2024, 6:12:32 PM	Private	Available
<input type="checkbox"/> powerconsumptiondatafiles	2/27/2024, 6:17:47 PM	Container	Available
<input type="checkbox"/> processeddatafiles	2/27/2024, 6:19:03 PM	Container	Available

Upload an empty processed data.csv file in the container.

The screenshot shows the Microsoft Azure portal interface with the 'Upload blob' dialog box open. The main heading is 'processeddatafiles | Container'. The left sidebar contains navigation links: Overview (selected), Diagnose and solve problems, Access Control (IAM), Settings, Shared access tokens, Manage ACL, Access policy, Properties, and Metadata. The main content area shows the 'Upload blob' dialog box. The dialog has a search bar, a table of blobs (currently empty), and an 'Upload' button. The dialog also shows the 'Authentication method' as 'Access key' and the 'Location' as 'processeddatafiles'. The bottom of the screen shows a Windows taskbar with the search bar, task icons, and system tray information (Sunset, 18:20, 27-02-2024).

Upload blob

1 file(s) selected: processeddata.csv
Drag and drop files here or [Browse for files](#)

☐ Overwrite if files already exist

Advanced

Upload

File uploaded successfully

The screenshot shows the Microsoft Azure portal interface. The browser address bar displays `portal.azure.com/#view/Microsoft_Azure_Storage/ContainerMenuBlade/~/_/overview/storageAccountId/%2Fsubscriptions%2F984f097c-963c-4eb6-...`. The page title is "processeddatafiles" under the "Container" section. A notification bubble in the top right corner states: "Successfully uploaded blob(s). Successfully uploaded 1 blob(s)." The left sidebar contains navigation links: Overview, Diagnose and solve problems, Access Control (IAM), Settings, Shared access tokens, Manage ACL, Access policy, Properties, and Metadata. The main content area shows the "Authentication method" as "Access key" and the "Location" as "processeddatafiles". Below this is a search bar and a table of blobs.

	Name	Modified	Access tier	Archive status	Blob type	Size	Lease state	
<input type="checkbox"/>	processeddata.csv	2/27/2024, 6:20:44 PM	Hot (Inferred)		Block blob	0 B	Available	...

3. Create an Azure Data Factory and create linked services for databricks and ADLS inside it.

Create data factory

The screenshot shows the "Create Data Factory" wizard in the Microsoft Azure portal. The browser address bar displays `portal.azure.com/#create/Microsoft.DataFactory`. The page title is "Create Data Factory". The "Basics" tab is selected, showing the "Project details" section. The "Subscription" is set to "Azure subscription 1" and the "Resource group" is set to "rg-azuser1016_mml.local-OAHCs". The "Instance details" section shows the "Name" as "datafactory1048" and the "Region" as "East US". The "Review + create" button is visible at the bottom.

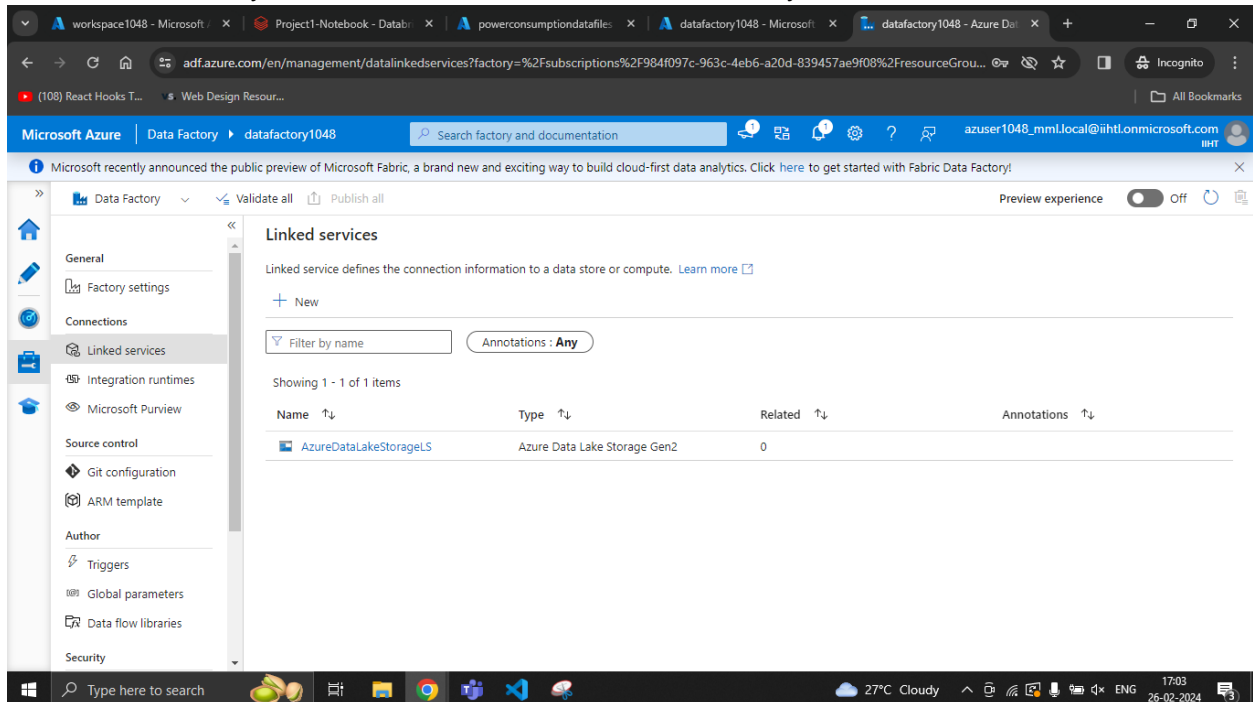
Azure Data Factory has been successfully created. Go to the resource.

The screenshot shows the Microsoft Azure portal interface. The main heading is "Microsoft.DataFactory-20240226165007 | Overview". Below the heading, there's a "Deployment" section with a search bar and buttons for "Delete", "Cancel", "Redeploy", "Download", and "Refresh". The "Overview" tab is selected, showing a green checkmark and the text "Your deployment is complete". Below this, deployment details are listed: "Deployment name : Microsoft.DataFactory-20240226165...", "Subscription : Azure subscription 1", and "Resource group : rg-azuser1016_mmlLocal-0AHcs". The "Start time" is "2/26/2024, 4:51:12 PM" and the "Correlation ID" is "94ece308-bac3-4634-aeb5-4252ec11...". A "Go to resource" button is prominently displayed. On the right, there are sections for "Cost management", "Microsoft Defender for Cloud", and "Free Microsoft tutorials".

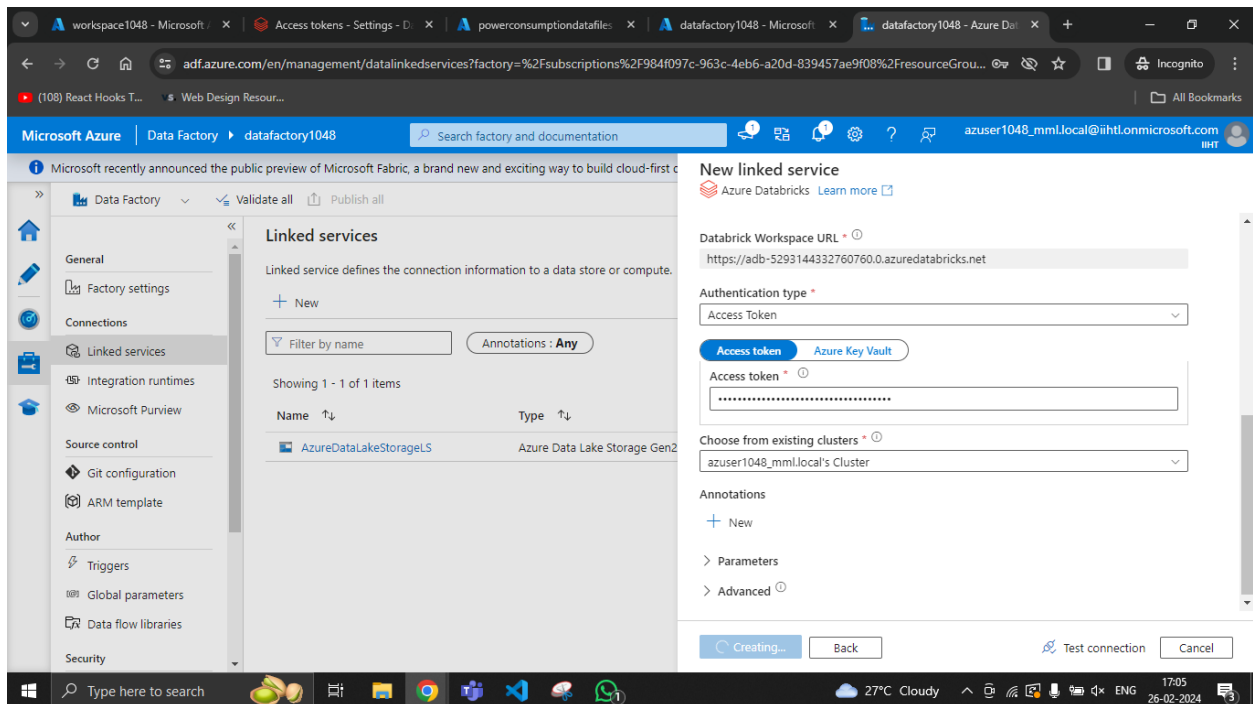
Go to Azure Data Factory and create linked service for ADLS account

The screenshot shows the Azure Data Factory portal interface. The main heading is "Microsoft recently announced the public preview of Microsoft Fabric, a brand new and exciting way to build cloud-first...". Below the heading, there's a "Data Factory" section with a search bar and buttons for "Validate all" and "Publish all". The "Linked services" tab is selected, showing a "New" button and a "Filter by name" dropdown. The "New linked service" dialog is open, showing the "Name" field as "AzureDataLakeStorageLS" and the "Description" field. The "Connect via integration runtime" dropdown is set to "AutoResolveIntegrationRuntime". The "Authentication type" dropdown is set to "Account key". The "Account selection method" is set to "From Azure subscription", and the "Azure subscription" dropdown is set to "Azure subscription 1 (984f097c-963c-4eb6-a20d-839457ae9f08)". A "Create" button is visible at the bottom of the dialog. On the right, there's a "Connection successful" message with a green checkmark and a "Test connection" button.

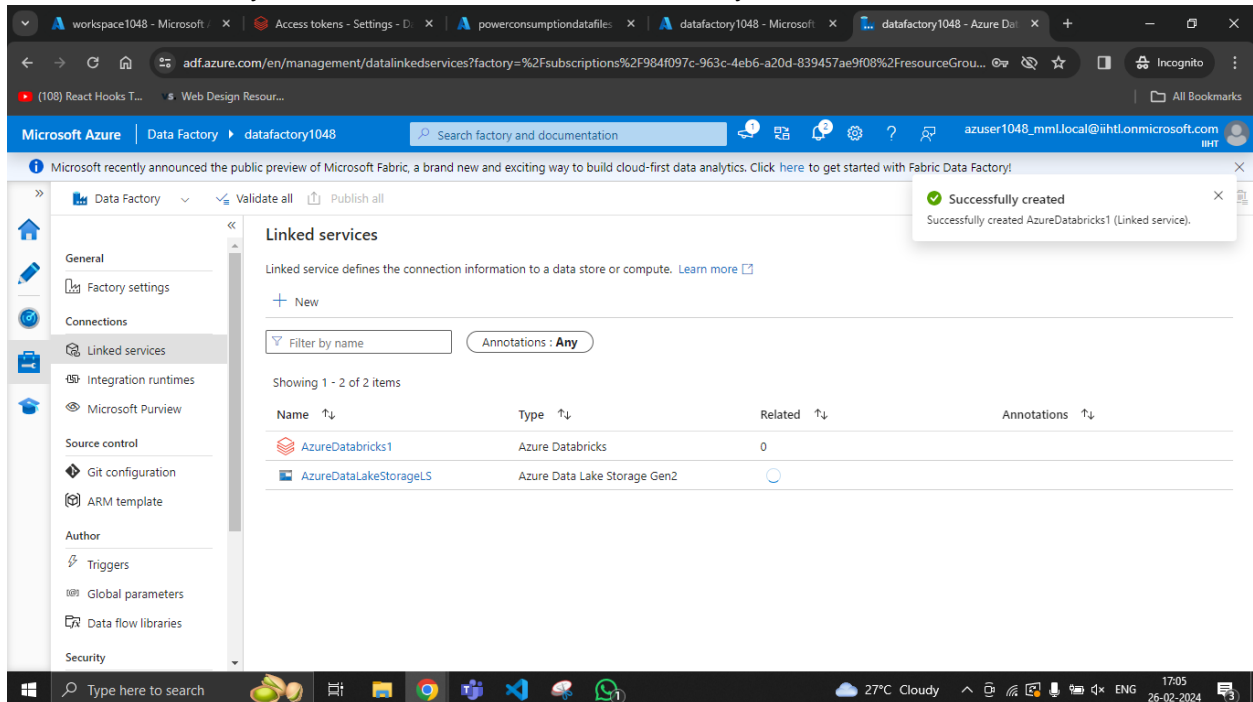
Linked Service for your ADLS Account has been successfully created.



Now create Linked Service for Azure Databricks



Linked Service for your Databricks has been successfully created.



Microsoft recently announced the public preview of Microsoft Fabric, a brand new and exciting way to build cloud-first data analytics. Click [here](#) to get started with Fabric Data Factory!

Linked services

Linked service defines the connection information to a data store or compute. [Learn more](#)

+ New

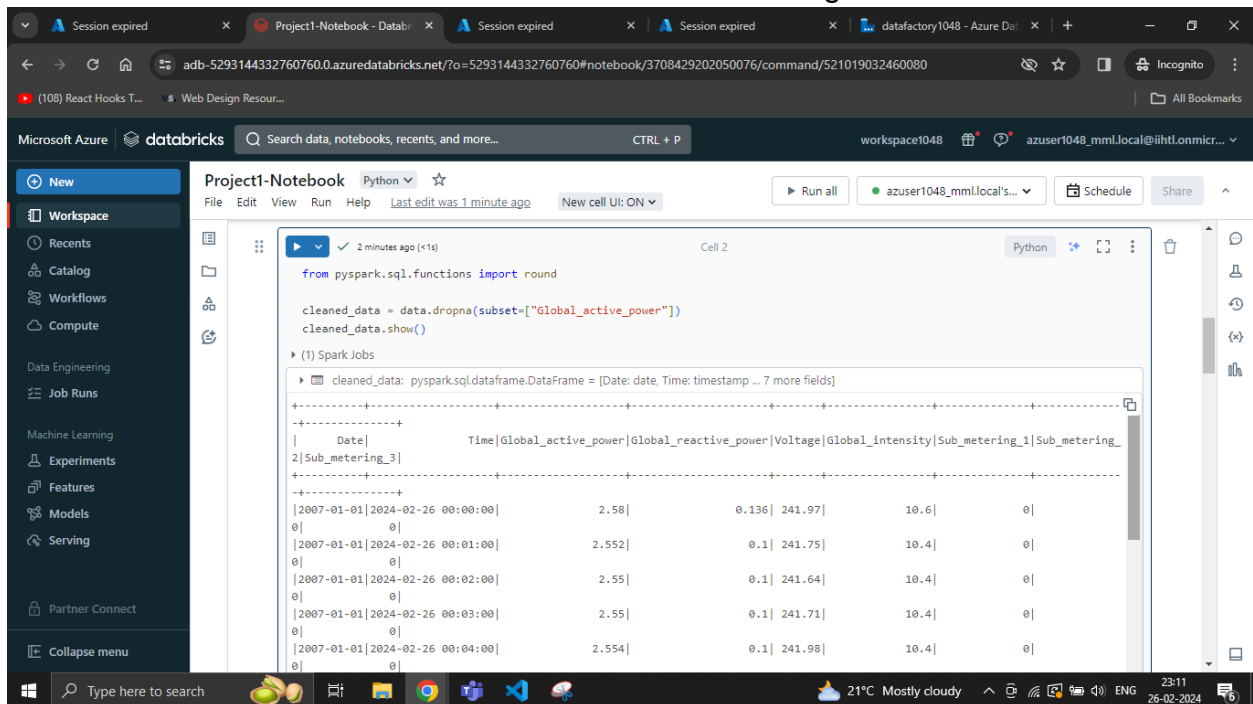
Filter by name Annotations: Any

Showing 1 - 2 of 2 items

Name	Type	Related	Annotations
AzureDatabricks1	Azure Databricks	0	
AzureDataLakeStorageLS	Azure Data Lake Storage Gen2	0	

4. Configure the Databricks notebook with the required logic for data cleaning and transformation

In the Databricks notebook write the code for data cleaning and transformation



Project1-Notebook Python

```
from pyspark.sql.functions import round

cleaned_data = data.dropna(subset=["Global_active_power"])
cleaned_data.show()
```

(1) Spark Jobs

cleaned_data: pyspark.sql.dataframe.DataFrame = [Date: date, Time: timestamp ... 7 more fields]

Date	Time	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2
2007-01-01	2024-02-26 00:00:00	2.58	0.136	241.97	10.6	0	0
2007-01-01	2024-02-26 00:01:00	2.552	0.1	241.75	10.4	0	0
2007-01-01	2024-02-26 00:02:00	2.55	0.1	241.64	10.4	0	0
2007-01-01	2024-02-26 00:03:00	2.55	0.1	241.71	10.4	0	0
2007-01-01	2024-02-26 00:04:00	2.554	0.1	241.98	10.4	0	0

5. Create data pipeline with databricks as activity and trigger the pipeline.

Add copy activity in the pipeline and add databricks notebook as the activity in it

The screenshot shows the Microsoft Azure Data Factory portal interface. The top navigation bar includes the Microsoft Azure logo, the Data Factory service, and the specific factory name 'datafactory1048'. A search bar is present for finding factory resources and documentation. A notification banner at the top states: 'Microsoft recently announced the public preview of Microsoft Fabric, a brand new and exciting way to build cloud-first data analytics. Click here to get started with Fabric Data Factory!'. The main content area is divided into three panes. The left pane, 'Factory Resources', shows a tree view with 'Pipelines' (1 item: 'pipeline1'), 'Datasets' (2 items: 'processeddata', 'Rawdata'), 'Data flows' (0 items), and 'Power Query' (0 items). The middle pane, 'Activities', shows a list of activity types under 'Move and transform', including 'Copy data', 'Data flow', 'Synapse', 'Azure Data Explorer', 'Azure Function', 'Batch Service', 'Databricks', 'Data Lake Analytics', 'General', and 'HDInsight'. The right pane, 'Properties', shows the configuration for the selected 'Copy data' activity. The 'Sink' tab is active, displaying 'Copy behavior' settings: 'Select...' for the sink, 'Max concurrent connections' (1), 'Block size (MB)' (1), 'Metadata' (checked), 'Quote all text' (checked), 'File extension' (.txt), and 'Max rows per file' (1). A 'Publishing completed Successfully published' notification is visible in the top right corner of the main content area.

Trigger pipeline run

The screenshot shows the Microsoft Azure Data Factory portal interface, similar to the previous one, but with a 'Running' notification. The 'Factory Resources' pane on the left shows 'pipeline1' under 'Pipelines'. The 'Activities' pane on the right shows the 'Copy data' activity selected. The 'Properties' pane on the right shows the 'Sink' tab. A 'Running' notification is displayed in the top right corner, stating: 'Running Successfully running pipeline1 (Pipeline). View pipeline run a few seconds ago'. The notification includes a 'Dismiss all' button and a 'Close' button.

Pipeline ran successfully

The screenshot displays the Microsoft Azure Data Factory portal interface. The top navigation bar shows the user is logged in as 'azuser1048_mml.local@ihtl.onmicrosoft.com'. The main content area is titled 'All pipeline runs > pipeline1 - Activity runs'. Below this, there are buttons for 'Rerun', 'Cancel', 'Refresh', and 'Update pipeline', along with 'List' and 'Gantt' view options. A visual representation of the pipeline shows two activities: 'Copy data' (labeled 'datapipeline') and 'Notebook' (labeled 'datacleaning'), both marked with green checkmarks indicating success. Below the pipeline diagram, the 'Activity runs' section shows a table of runs for the selected pipeline run ID 'b6a028b0-bab8-45c3-9100-1ca3d49a38c6'. The table lists two activities: 'datacleaning' and 'datapipeline', both with a status of 'Succeeded'. The 'Activity runs' section also includes a 'Monitor in Azure Metrics' link and an 'Export to CSV' option. The bottom of the screen shows the Windows taskbar with the date '26-02-2024' and time '23:29'.

Microsoft Azure | Data Factory | datafactory1048

Microsoft recently announced the public preview of Microsoft Fabric, a brand new and exciting way to build cloud-first data analytics. Click [here](#) to get started with Fabric Data Factory!

All pipeline runs > **pipeline1 - Activity runs**

Rerun Cancel Refresh Update pipeline List Gantt

Copy data Notebook

datapipeline datacleaning

Activity runs

Pipeline run ID b6a028b0-bab8-45c3-9100-1ca3d49a38c6

All status Monitor in Azure Metrics Export to CSV

Showing 1 - 2 items

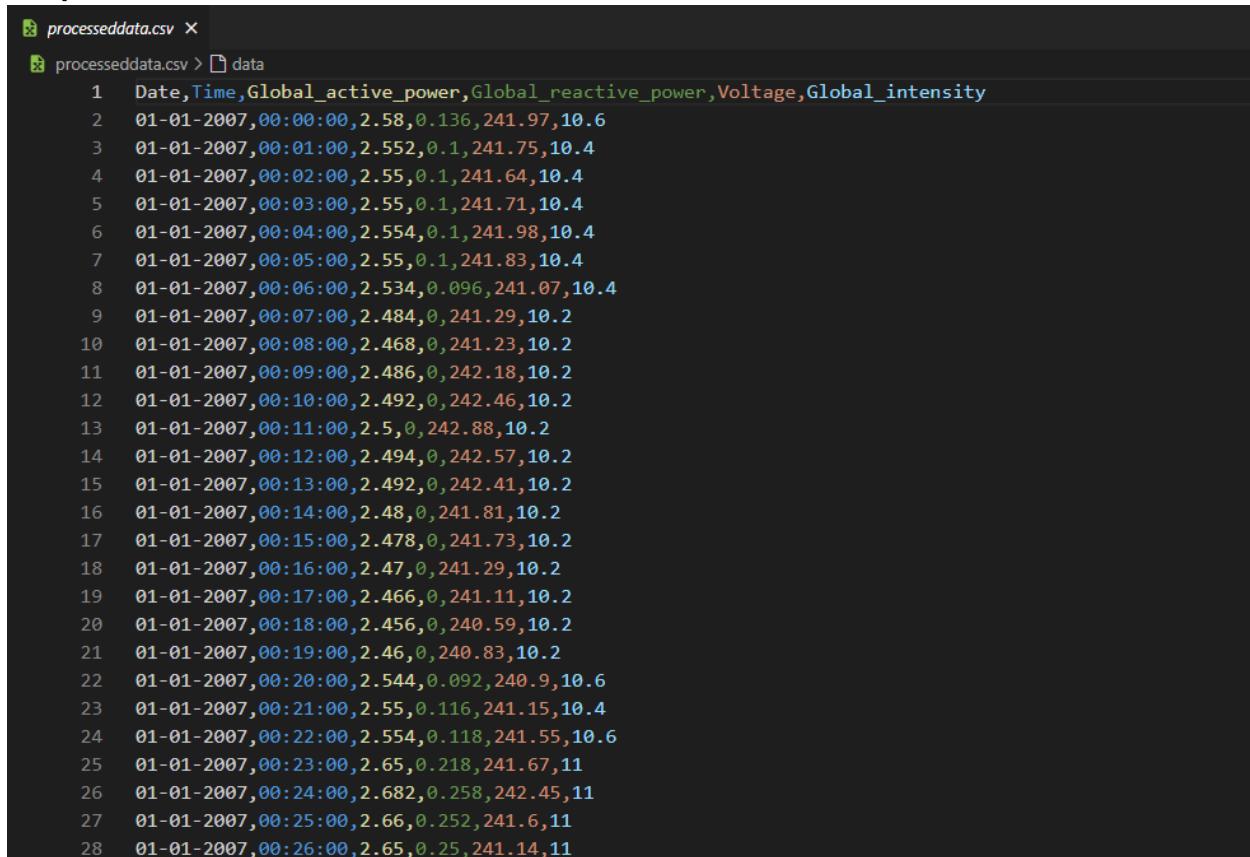
Activity name	Activity status	Activity type	Run start	Duration	Integration runtime	User properties
datacleaning	Succeeded	Notebook	2/26/2024, 11:28:12 PM	38s	AutoResolveIntegration	
datapipeline	Succeeded	Copy data	2/26/2024, 11:27:55 PM	16s	AutoResolveIntegration	

21°C Mostly cloudy 23:29 26-02-2024

Snapshot of Raw Data from Source:

household_power_consumption.csv									
household_power_consumption.csv > data									
1	Date	Time	Global_active_power	Global_reactive_power	Voltage	Global_intensity	Sub_metering_1	Sub_metering_2	Sub_metering_3
2	01-01-2007	00:00:00	2.58	0.136	241.97	10.6	0	0	0
3	01-01-2007	00:01:00	2.552	0.1	241.75	10.4	0	0	0
4	01-01-2007	00:02:00	2.55	0.1	241.64	10.4	0	0	0
5	01-01-2007	00:03:00	2.55	0.1	241.71	10.4	0	0	0
6	01-01-2007	00:04:00	2.554	0.1	241.98	10.4	0	0	0
7	01-01-2007	00:05:00	2.55	0.1	241.83	10.4	0	0	0
8	01-01-2007	00:06:00	2.534	0.096	241.07	10.4	0	0	0
9	01-01-2007	00:07:00	2.484	0	241.29	10.2	0	0	0
10	01-01-2007	00:08:00	2.468	0	241.23	10.2	0	0	0
11	01-01-2007	00:09:00	2.486	0	242.18	10.2	0	0	0
12	01-01-2007	00:10:00	2.492	0	242.46	10.2	0	0	0
13	01-01-2007	00:11:00	2.5	0	242.88	10.2	0	0	0
14	01-01-2007	00:12:00	2.494	0	242.57	10.2	0	0	0
15	01-01-2007	00:13:00	2.492	0	242.41	10.2	0	0	0
16	01-01-2007	00:14:00	2.48	0	241.81	10.2	0	0	0
17	01-01-2007	00:15:00	2.478	0	241.73	10.2	0	0	0
18	01-01-2007	00:16:00	2.47	0	241.29	10.2	0	0	0
19	01-01-2007	00:17:00	2.466	0	241.11	10.2	0	0	0
20	01-01-2007	00:18:00	2.456	0	240.59	10.2	0	0	0
21	01-01-2007	00:19:00	2.46	0	240.83	10.2	0	0	0
22	01-01-2007	00:20:00	2.544	0.092	240.9	10.6	0	0	0
23	01-01-2007	00:21:00	2.55	0.116	241.15	10.4	0	1	0
24	01-01-2007	00:22:00	2.554	0.118	241.55	10.6	0	1	0
25	01-01-2007	00:23:00	2.65	0.218	241.67	11	0	2	0
26	01-01-2007	00:24:00	2.682	0.258	242.45	11	0	1	0
27	01-01-2007	00:25:00	2.66	0.252	241.6	11	0	1	0

Snapshot of Processed Data from Sink:



	Date	Time	Global_active_power	Global_reactive_power	Voltage	Global_intensity
1	01-01-2007	00:00:00	2.58	0.136	241.97	10.6
2	01-01-2007	00:01:00	2.552	0.1	241.75	10.4
3	01-01-2007	00:02:00	2.55	0.1	241.64	10.4
4	01-01-2007	00:03:00	2.55	0.1	241.71	10.4
5	01-01-2007	00:04:00	2.554	0.1	241.98	10.4
6	01-01-2007	00:05:00	2.55	0.1	241.83	10.4
7	01-01-2007	00:06:00	2.534	0.096	241.07	10.4
8	01-01-2007	00:07:00	2.484	0	241.29	10.2
9	01-01-2007	00:08:00	2.468	0	241.23	10.2
10	01-01-2007	00:09:00	2.486	0	242.18	10.2
11	01-01-2007	00:10:00	2.492	0	242.46	10.2
12	01-01-2007	00:11:00	2.5	0	242.88	10.2
13	01-01-2007	00:12:00	2.494	0	242.57	10.2
14	01-01-2007	00:13:00	2.492	0	242.41	10.2
15	01-01-2007	00:14:00	2.48	0	241.81	10.2
16	01-01-2007	00:15:00	2.478	0	241.73	10.2
17	01-01-2007	00:16:00	2.47	0	241.29	10.2
18	01-01-2007	00:17:00	2.466	0	241.11	10.2
19	01-01-2007	00:18:00	2.456	0	240.59	10.2
20	01-01-2007	00:19:00	2.46	0	240.83	10.2
21	01-01-2007	00:20:00	2.544	0.092	240.9	10.6
22	01-01-2007	00:21:00	2.55	0.116	241.15	10.4
23	01-01-2007	00:22:00	2.554	0.118	241.55	10.6
24	01-01-2007	00:23:00	2.65	0.218	241.67	11
25	01-01-2007	00:24:00	2.682	0.258	242.45	11
26	01-01-2007	00:25:00	2.66	0.252	241.6	11
27	01-01-2007	00:26:00	2.65	0.25	241.14	11
28						

Conclusion:

In conclusion, this project successfully demonstrated parallel processing using Azure Data Factory's Data Pipeline for processing, analysis and movement of household power consumption data. This project has successfully developed a robust and efficient data pipeline in Azure for processing and analyzing household power consumption data. The data was ingested from the source by Azure Data Factory, cleaned and transformed by the Azure Databricks Notebook and the loaded to ADLS with the copy activity in the data pipeline.