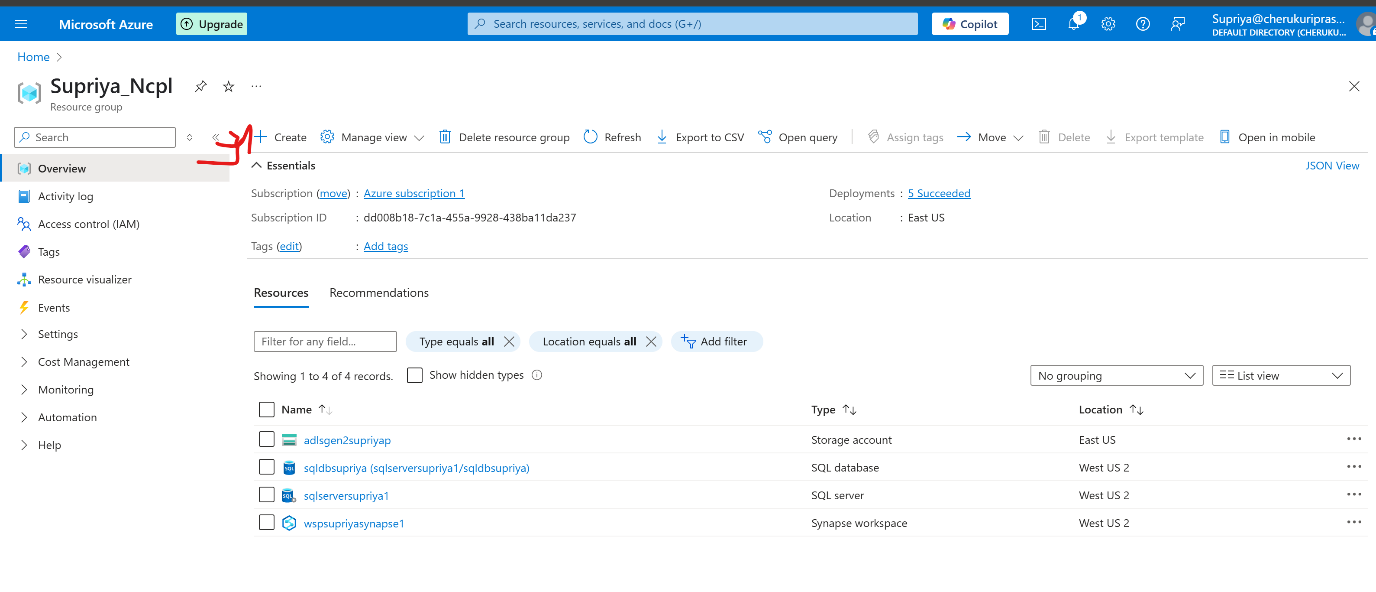
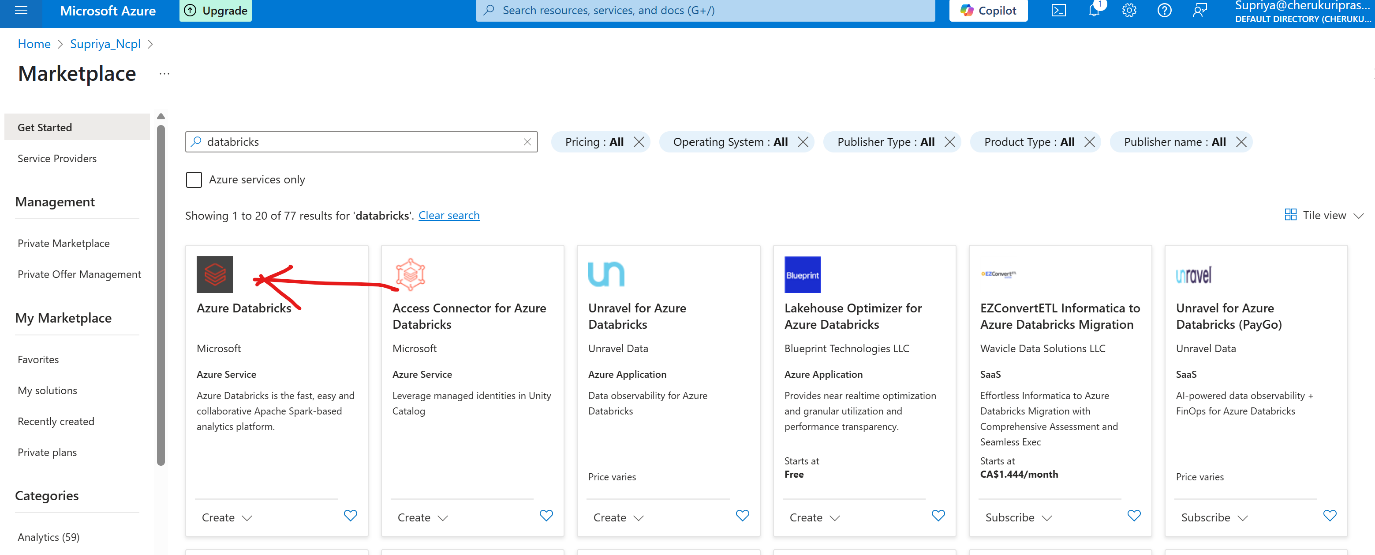
**How to create Data bricks**

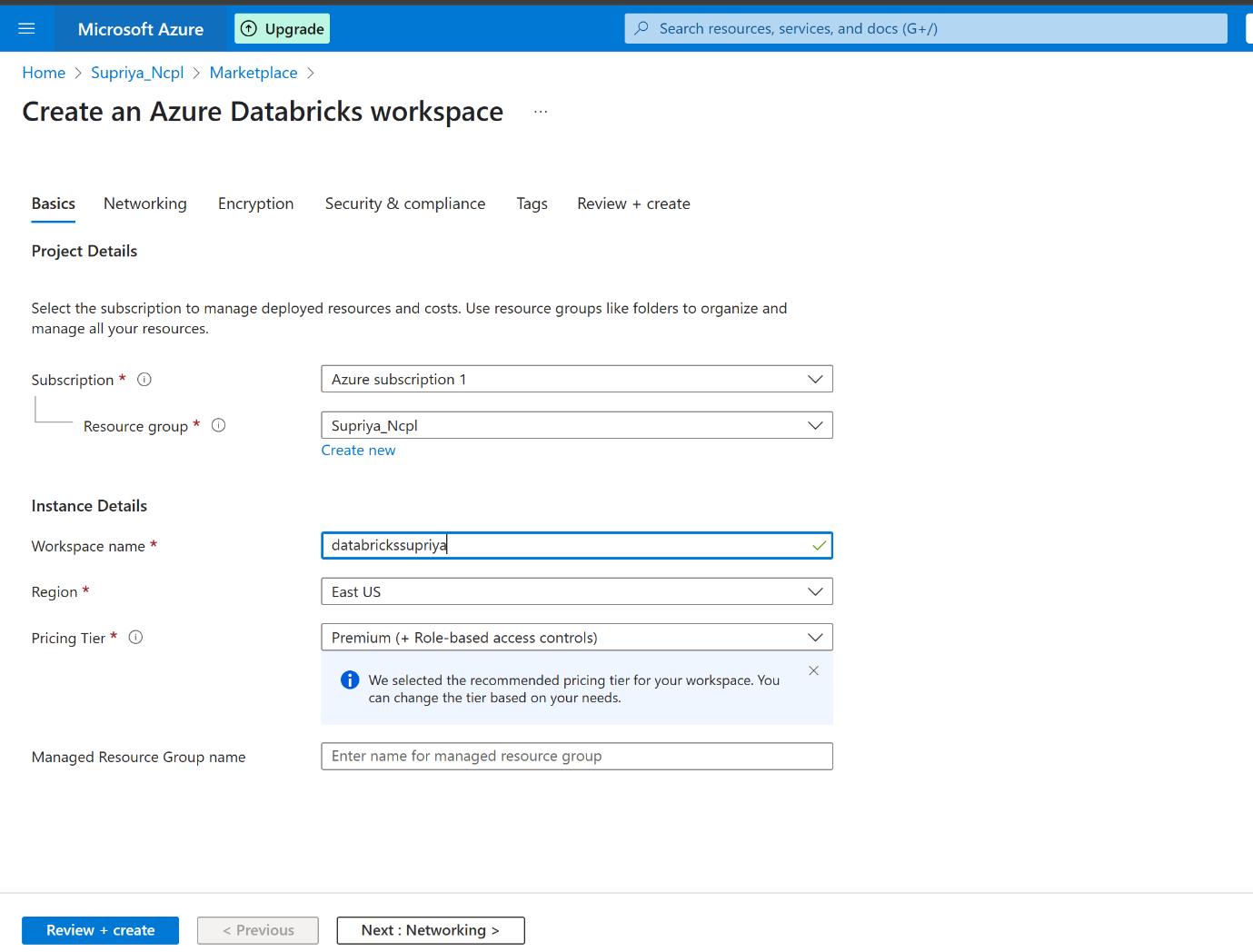
Create a Databricks workspace.

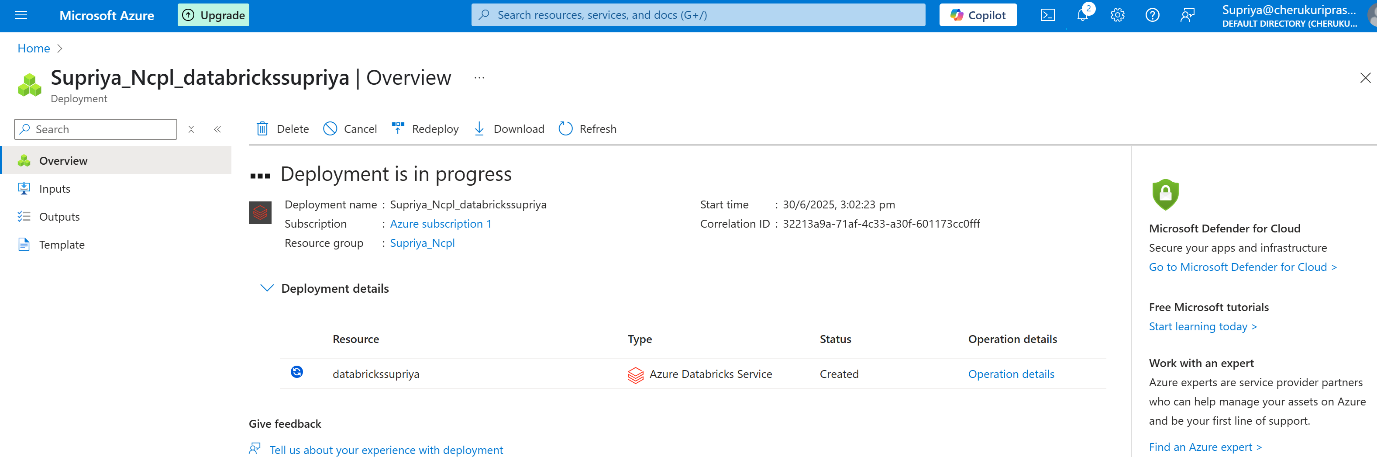
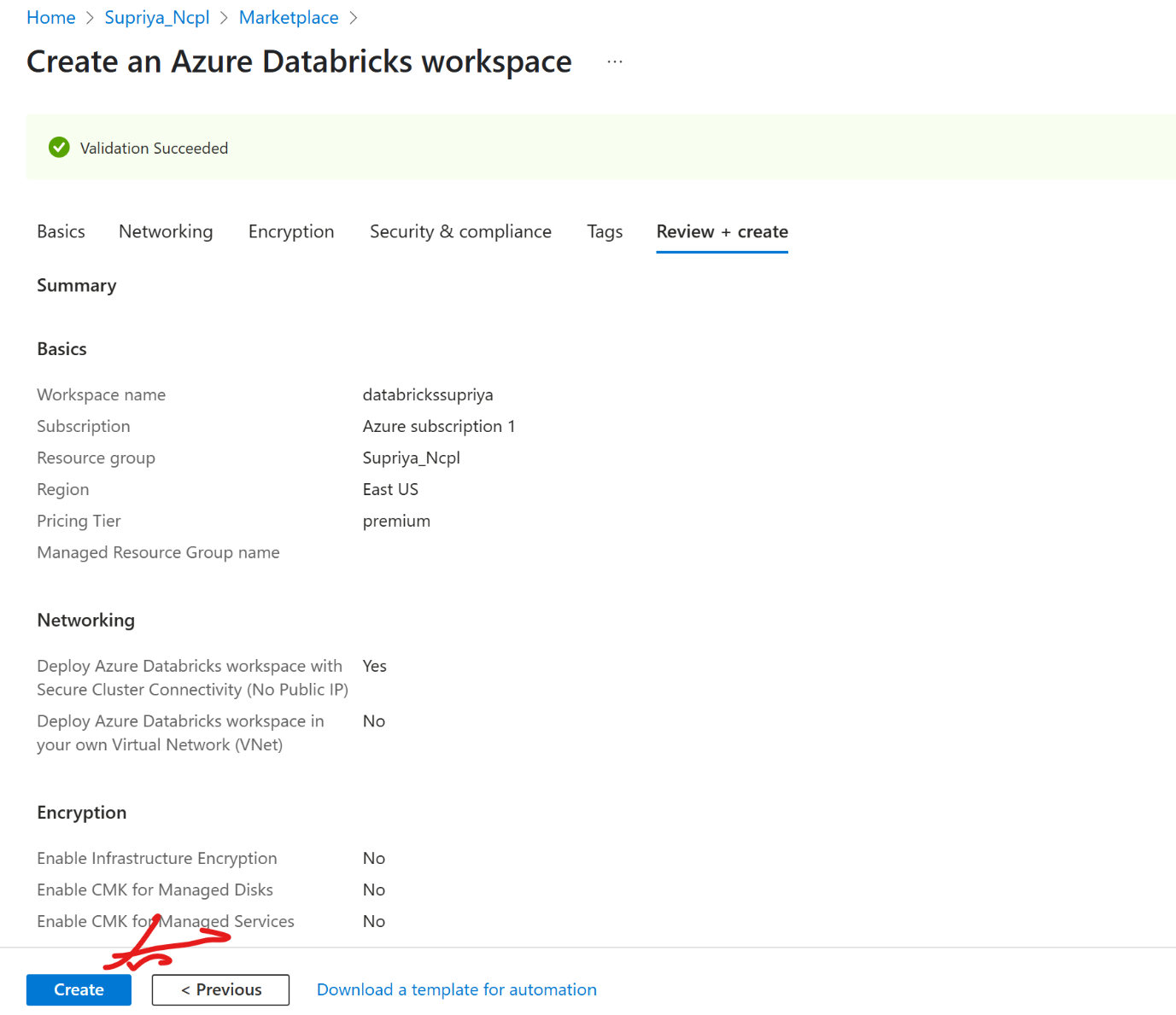
Explore the UI without creating a compute (to avoid unnecessary charges).

Prepare for DBUTIL commands in the upcoming session.

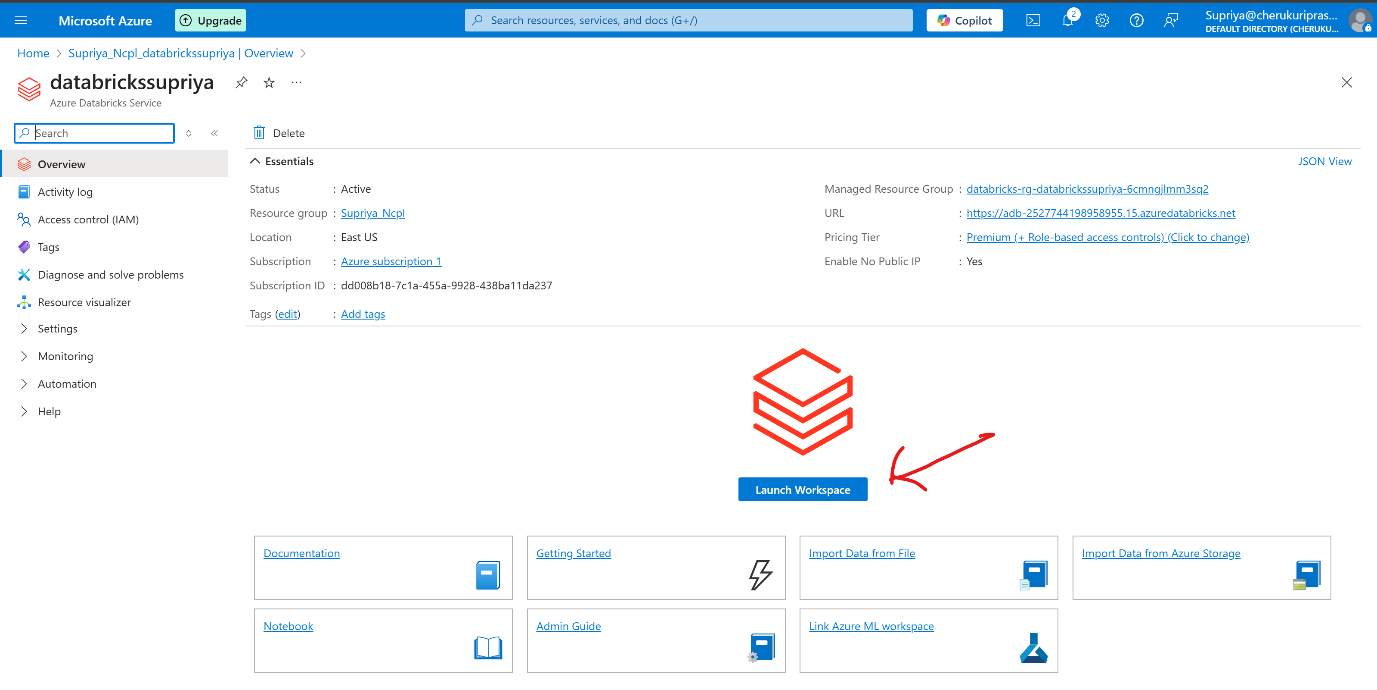








Click on launch to open in new page.



Page layout looks like this.

A screenshot of a computer

AI-generated content may be incorrect.

Workspace layout looks like this.

A close-up of a computer screen

AI-generated content may be incorrect.

Recent shows what file are opened, what action is done.

A screenshot of a computer

AI-generated content may be incorrect.

Catalag is like storage,where data is stored in tables.

A screenshot of a computer

AI-generated content may be incorrect.

Job&pipelines: here we can run pipelines.

A screenshot of a computer

AI-generated content may be incorrect.

Compute: it provide virtual machines in backend to run the code.

A screenshot of a computer

AI-generated content may be incorrect.

Marketplace: It is for external toolswhich we can inegrate.

A screenshot of a computer

AI-generated content may be incorrect.

SQl editor: catalog tables can be queried in SQL editor.

A screenshot of a computer

AI-generated content may be incorrect.

Queries : we can save our queries here.

A screenshot of a computer

AI-generated content may be incorrect.

Dashboards: we can create dashboards like Power BI here.

A screenshot of a computer

AI-generated content may be incorrect.

Genie it is used for asking questions.

A screenshot of a computer

AI-generated content may be incorrect.

Alerts:it will send alerts

A screenshot of a computer

AI-generated content may be incorrect.

Question history: It has all the queries.

A screenshot of a computer

AI-generated content may be incorrect.

SQL Warehouse :

A screenshot of a computer

AI-generated content may be incorrect.

JobRuns:

A screenshot of a computer

AI-generated content may be incorrect.

Data Ingestion:

A screenshot of a computer

AI-generated content may be incorrect.

Click on new to open note book.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

To write code click on code , to add notes click on text , you can also make headings.

The languages that it supports are as below.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Explore dbutils

**What is dbutils?**

dbutils is a utility command set in Databricks that helps manage:

* File systems (dbutils.fs)
* Notebooks (dbutils.notebook)
* Widgets (dbutils.widgets)
* Secrets (dbutils.secrets)
* Jobs and runtime information (dbutils.jobs, dbutils.entry\_point, etc.)

| **Command** | **Use Case** |
| --- | --- |
| dbutils.fs | Interact with the DBFS (Databricks File System): list files, copy, move, delete. |
| dbutils.widgets | Create input widgets in notebooks (dropdown, textbox). |
| dbutils.notebook | Call or chain notebooks, pass parameters. |
| dbutils.secrets | Securely access secrets from a key vault. |
| dbutils.library *(legacy)* | Used to install libraries (now replaced by %pip or cluster libraries). |

Here’s a complete list of **dbutils** modules and subcommands available in **Databricks**, along with a brief description of each:

**🔹 Top-Level dbutils Modules**

| **Module** | **Description** |
| --- | --- |
| dbutils.fs | File system utilities for interacting with DBFS (Databricks File System). |
| dbutils.widgets | Create input widgets (text, dropdown, etc.) to parameterize notebooks. |
| dbutils.notebook | Run and chain notebooks, pass and retrieve parameters. |
| dbutils.secrets | Access secrets stored in a secret scope (like credentials). |
| dbutils.library | Legacy method to install libraries (replaced by %pip). |
| dbutils.jobs | Information about the job context (e.g., job run ID). |
| dbutils.entry\_point | Access the current notebook's entry point for advanced use. |
| dbutils.credentials | Manage credential passthrough (used in secure access). |
| dbutils.data | Access datasets used in notebooks (less commonly used). |
| dbutils.help() | Lists available dbutils modules and commands. |

**Details of Common Modules**

**1. dbutils.fs — File System Commands**

| **Command** | **Description** |
| --- | --- |
| ls(path) | Lists files in a directory. |
| cp(from, to, recurse=False) | Copies a file or directory. |
| mv(from, to, recurse=False) | Moves a file or directory. |
| rm(path, recurse=False) | Deletes a file or directory. |
| head(path, maxBytes) | Shows the first N bytes of a file. |
| put(path, contents, overwrite=False) | Writes text to a file. |
| mount(source, mount\_point, extra\_configs) | Mount external storage. |
| unmount(mount\_point) | Unmount a storage location. |
| mounts() | List all mount points. |

**2. dbutils.widgets — User Inputs**

| **Command** | **Description** |
| --- | --- |
| text(name, defaultValue, label) | Create a text box. |
| dropdown(name, defaultValue, choices, label) | Create a dropdown. |
| combobox(name, defaultValue, choices, label) | Editable dropdown. |
| multiselect(name, defaultValue, choices, label) | Select multiple values. |
| get(name) | Get user input value. |
| remove(name) | Remove a widget. |
| removeAll() | Remove all widgets. |

**3. dbutils.notebook — Notebook Control**

| **Command** | **Description** |
| --- | --- |
| run(path, timeout\_seconds, arguments) | Run another notebook. |
| exit(value) | Exit the current notebook and return a value. |

**4. dbutils.secrets — Secure Access**

| **Command** | **Description** |
| --- | --- |
| get(scope, key) | Get a secret value. |
| getBytes(scope, key) | Get a secret as bytes. |
| list(scope) | List all keys in a scope. |
| listScopes() | List all available secret scopes. |

**5. dbutils.jobs — Job Info**

| **Command** | **Description** |
| --- | --- |
| taskContext() | Returns info like job ID, run ID, task key. Useful in job debugging. |

**🛠️ To View Available Commands in Your Environment:**

If you're in a Databricks notebook, run:

dbutils.help()

To get help for a specific module:

dbutils.fs.help()

dbutils.widgets.help()