

Unity Catalog volumes

Volumes: Databricks Volumes are a feature in Unity Catalog that let you store and manage **non-table files** (like CSV, JSON, images, PDFs, ML models, etc.) in a secure, governed way.
They are used when you want to manage **files**, not just tables.

To Begin with the Lab

To create and manage a Volumes you need to follow these steps, to begin:

Open your **Databricks workspace** → Click on Catalog → select or create a Catalog → Select or create Schema → Click on create, create volume → Enter your Volume name → Select Managed volume (if you want to connect volume with cloud you can select External volume) → Click create.

There is another way to create volume:

Click on Workspace from the left sidebar → Select or create folder → Click on create, create notebook → you will see a cell where you need to give a path →

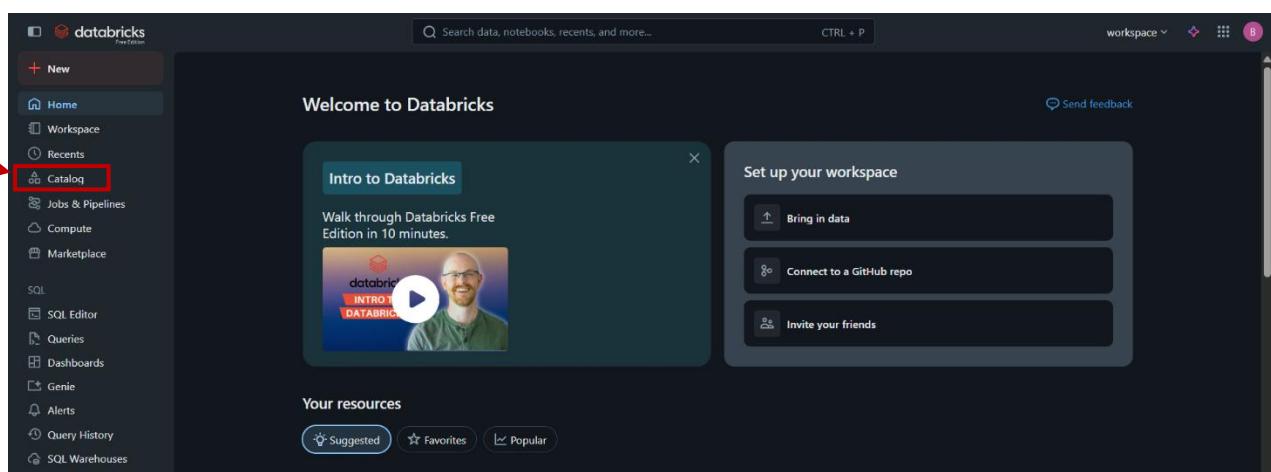
Managed Volume:

```
CREATE VOLUME catalog_name.schema_name.volume_name
```

External Volume:

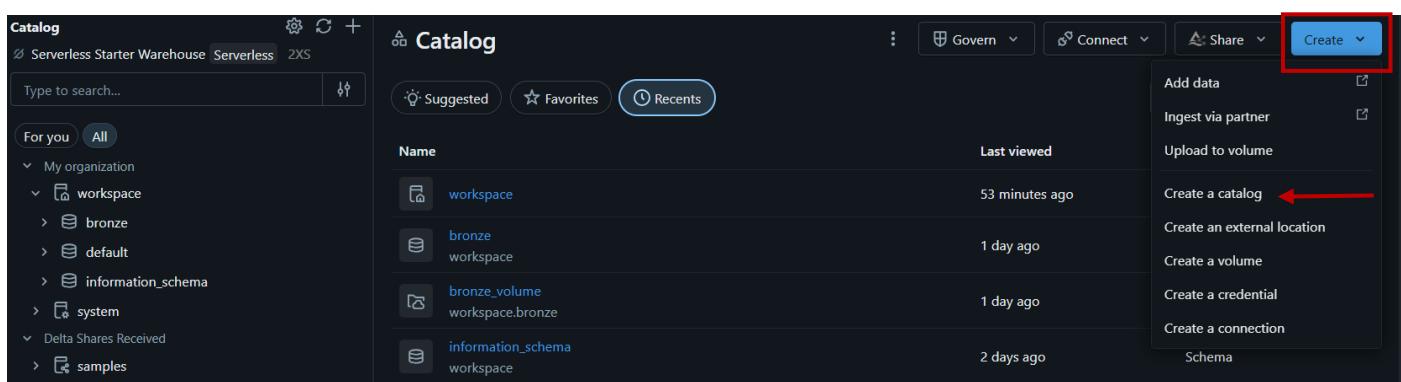
```
CREATE EXTERNAL VOLUME catalog_name.schema_name.volume_name  
LOCATION
```

Step 1: Open your **Databricks workspace** → Click on Catalog



The screenshot shows the Databricks workspace interface. On the left, there is a sidebar with various options: New, Home, Workspace (selected), Recents, Catalog (highlighted with a red arrow), Jobs & Pipelines, Compute, Marketplace, SQL, SQL Editor, Queries, Dashboards, Genie, Alerts, Query History, and SQL Warehouses. The main area is titled "Welcome to Databricks" and contains a "Set up your workspace" section with three buttons: "Bring in data", "Connect to a GitHub repo", and "Invite your friends". Below this is a "Your resources" section with "Suggested", "Favorites", and "Popular" buttons. A modal window titled "Intro to Databricks" is open, showing a video thumbnail of a man speaking.

Step 2: select or create a Catalog



The screenshot shows the Catalog interface. At the top, there is a search bar and a "Create" button highlighted with a red box. Below the search bar is a "Type to search..." input field. The main area shows a list of catalogs: "Serverless Starter Warehouse" (Serverless 2XS), "workspace" (bronze), "bronze.workspace" (bronze_volume), "bronze.workspace.bronze", "information_schema", "information_schema.workspace", and "samples". To the right of the catalog list is a "Govern" button, a "Connect" button, a "Share" button, and a "Create" dropdown menu. The "Create" menu is open, showing options: "Ingest via partner", "Upload to volume", "Create a catalog" (highlighted with a red arrow), "Create an external location", "Create a volume", "Create a credential", "Create a connection", and "Schema".

Step 3: Select or create Schema

The screenshot shows the Databricks interface with the Catalog sidebar selected. In the Catalog Explorer, the 'workspace' catalog is selected. A red box highlights the 'Create schema' button in the top right corner of the main content area.

Step 4: enter schema name (e.g. new_one) and click on create:

The screenshot shows the 'Create a new schema' dialog. The 'Schema name*' field is filled with 'new_one'. The 'Create' button at the bottom right is highlighted with a red box.

Step 5: your schema is created, now click on create:

The screenshot shows the Databricks Catalog Explorer with the 'new_one' schema listed under the workspace catalog. The 'Create' button in the top right corner is highlighted with a red box.

Step 6: click on create Volume:

Catalog Explorer > workspace > new_one

Overview Details Permissions Policies

Description

AI generate Add

Filter tables Tables 0 Volumes 0 Models 0 Functions 0

Name Owner Created at Popularity

⋮ Use with BI tools Share Create

Volume Table Model Metric view Sort

Step 7: Now Enter your volume name, External Volume lets you store files in **your own cloud storage location** (like S3, ADLS, GCS) instead of Databricks-managed storage
If you want to store your files in your own cloud storage select External location and provide Location of your cloud.

Create a new volume

Volumes are locations for storing files in Unity Catalog.

Volume name*

alpha_volume

Volume type Learn more ↗

Managed volume
Files are stored in a location managed by Unity Catalog.

External volume
Files are stored in an external location that you configure.

External location*

The volume will use this external location for storage.

Select external location

Path*

Path prefix inside the external location where the volume will be created.

s3://<bucket_name>/<external_location>/<path>

Choose catalog and schema*

Volumes are organized under a catalog and a schema. Select existing or create new.

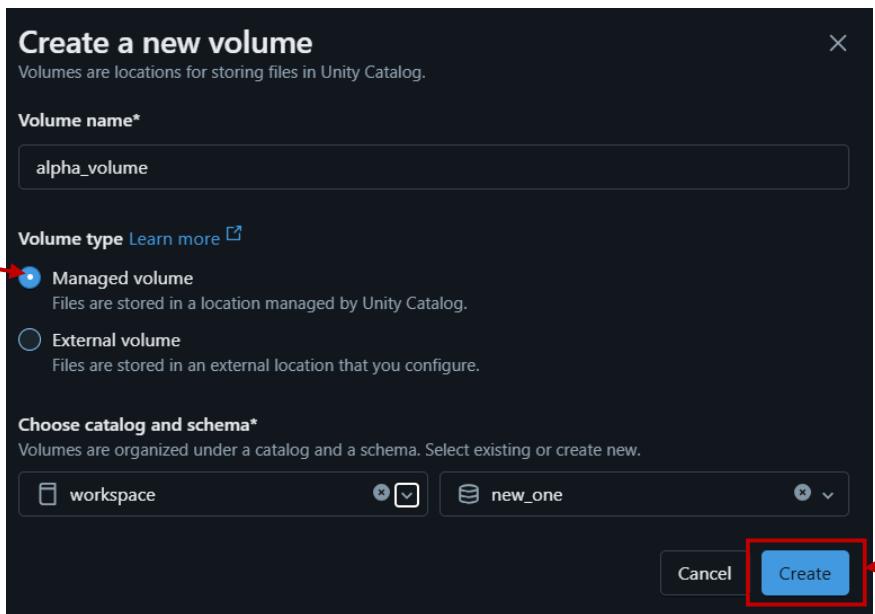
workspace new_one

Cancel Create

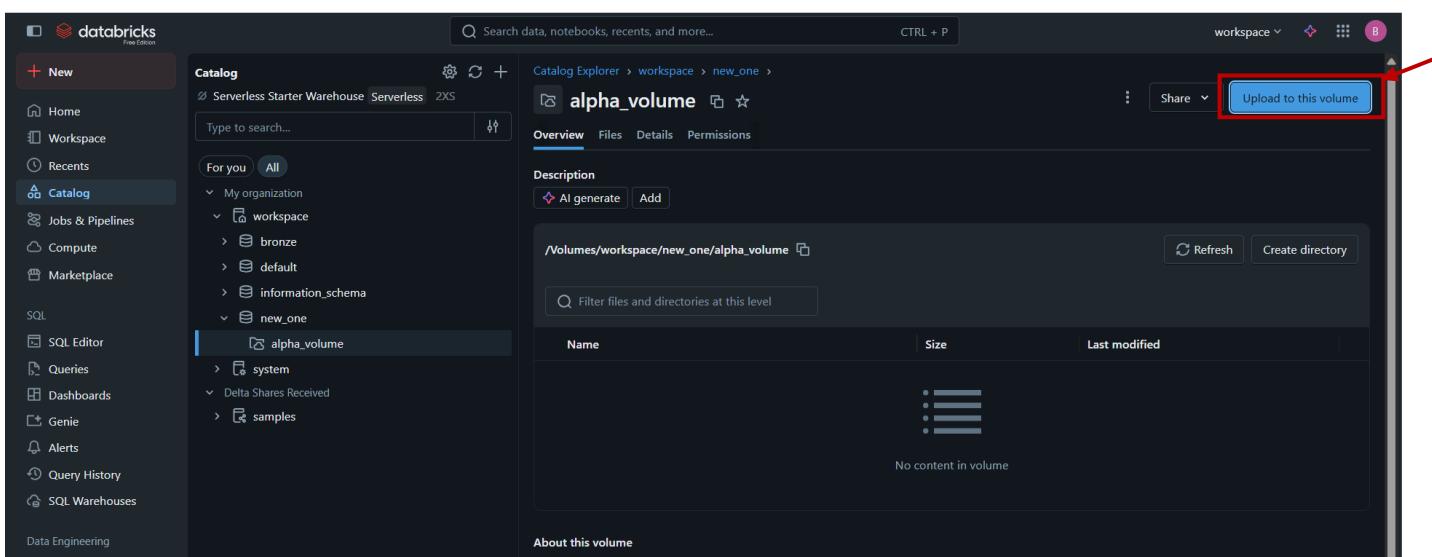
Name of volume

Location and path of your External location

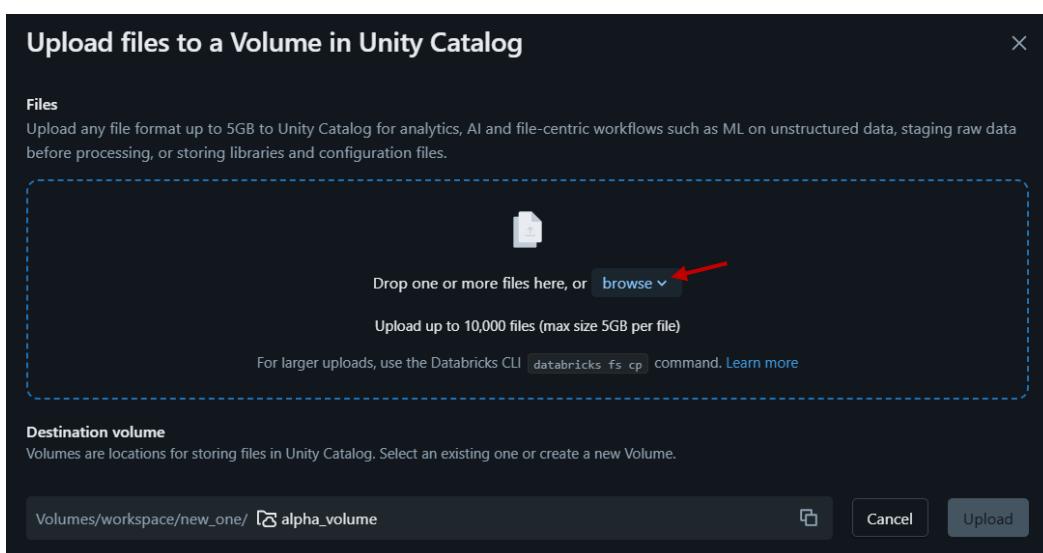
Otherwise: select Managed location and click on create:



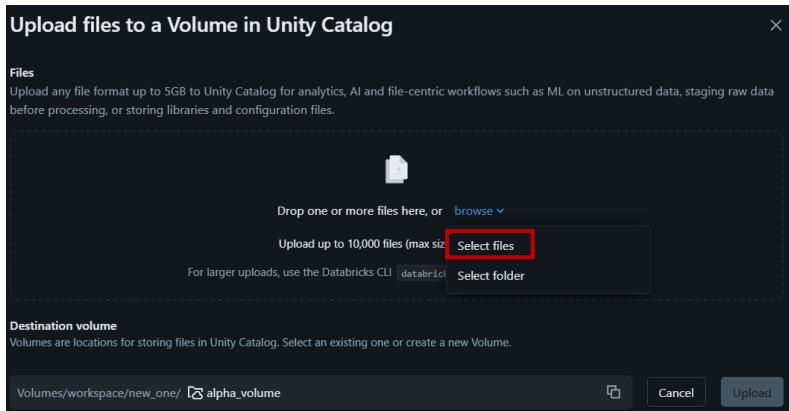
Step 8: Now the volume is created so you need to upload some file in the volume → click on ‘Upload to this volume’:



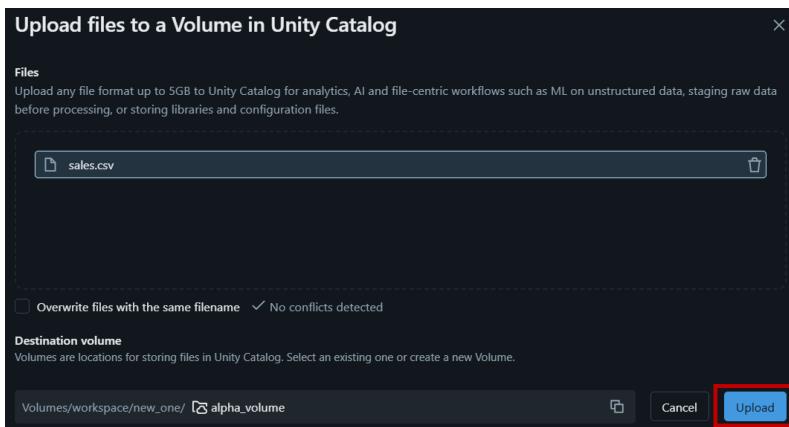
Step 9: click on browse:



Step 10: Select the file:



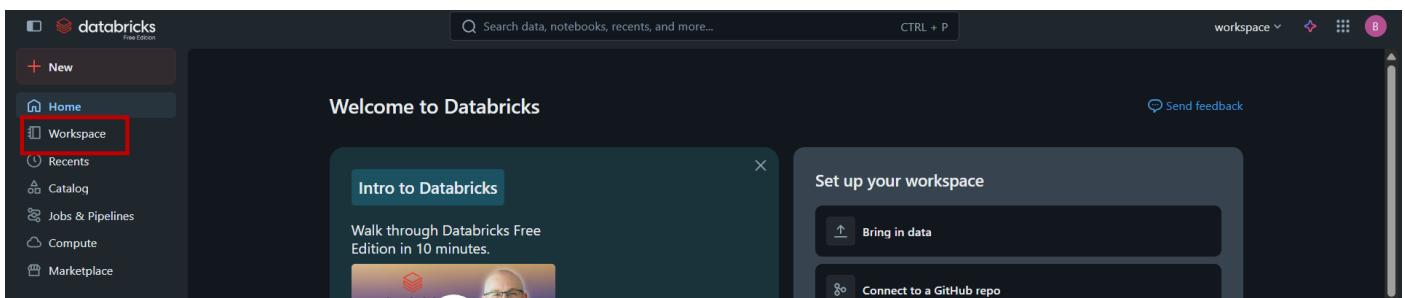
Step 11: click on upload and your file will be uploaded in your volume:



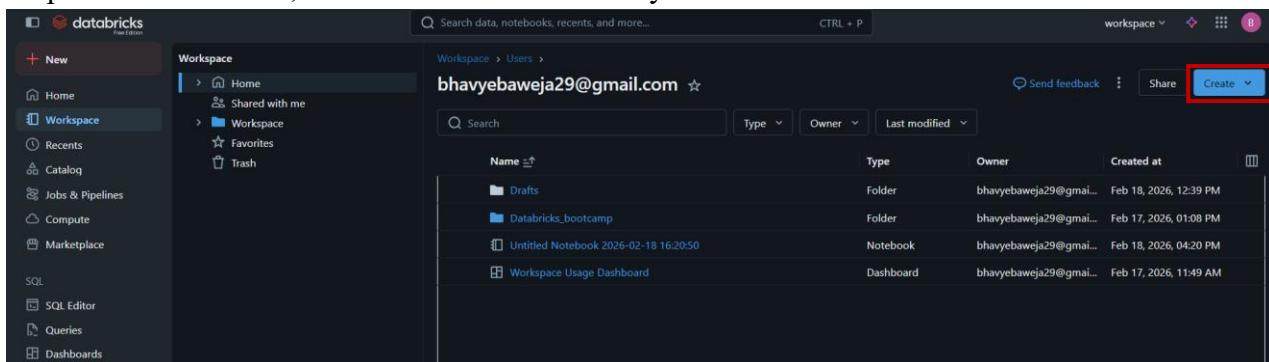
Now I will demonstrate how to execute this using SQL and also explain the steps to create a volume using SQL:

To create volume using SQL you need to follow these steps:

Step 1: Click on Workspace:



Step 2: click on create, to create a folder in which your notebook will run:



Step 3: click on folder and give a name to your folder:

The screenshot shows the Databricks workspace interface. On the left, there's a sidebar with options like 'Send feedback', 'Share', and 'Create'. Under 'Owner', several email addresses are listed. The 'Create' dropdown menu is open, with 'Folder' highlighted and a red box around it. To the right, a modal window titled 'New folder' is displayed, containing a text input field with the value 'Alpha'. Below the input field are 'Cancel' and 'Create' buttons, with a red arrow pointing to the 'Create' button.

Step 4: open the folder and click on create:

The screenshot shows the 'Alpha' folder in the Databricks workspace. The left sidebar shows the 'Workspace' section with 'Alpha' selected. The main area displays the contents of the 'Alpha' folder, which is currently empty. A red box highlights the 'Create' button in the top right corner of the workspace header.

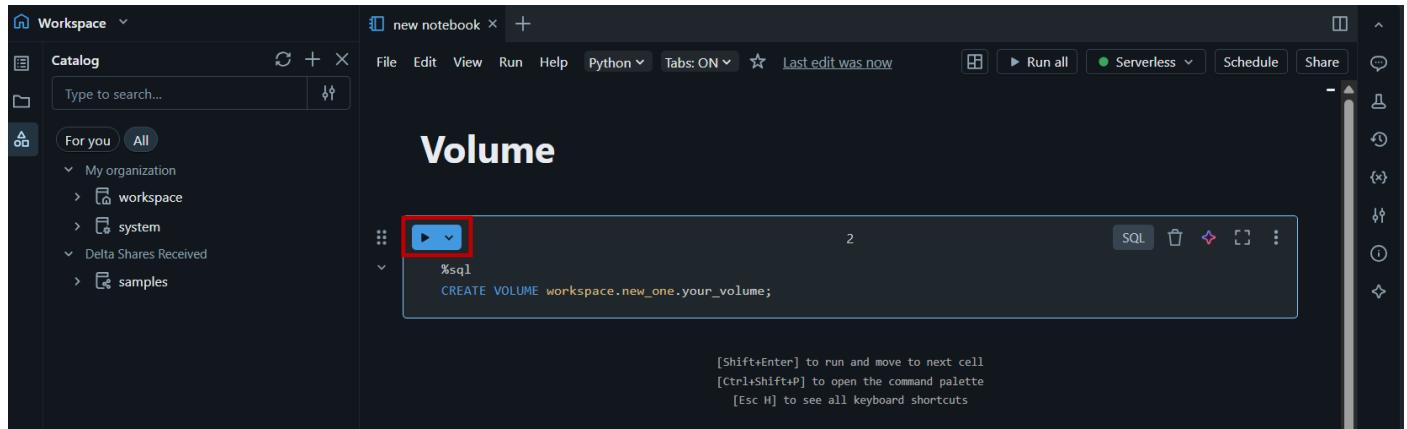
Step 5: click on Notebook (We use **notebooks** in Databricks because they allow us to write, execute, and document code in one place.):

The screenshot shows the 'Alpha' folder in the Databricks workspace. The left sidebar shows the 'Workspace' section with 'Alpha' selected. The main area displays the contents of the 'Alpha' folder, which is currently empty. A red box highlights the 'Notebook' option in the 'Create' dropdown menu on the right side of the screen.

Step 6: In the cell, you need to enter the command to create your volume:

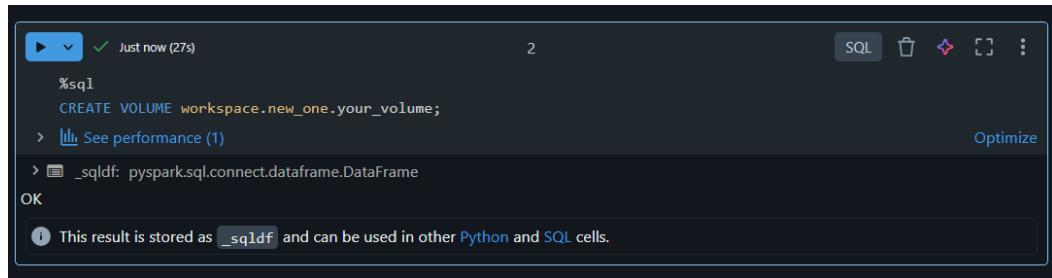
The screenshot shows a Databricks notebook titled 'New Notebook 2026-02-20 15:26:50'. The left sidebar shows the 'Catalog' section. The main area has a title 'Volume' and a code cell. The code cell contains the placeholder text 'Start typing or generate with AI (Ctrl + I)...'. A red box highlights this placeholder text. At the bottom of the notebook interface, there are keyboard shortcut instructions: '[Shift+Enter] to run and move to next cell', '[Ctrl+Shift+P] to open the command palette', and '[Esc H] to see all keyboard shortcuts'.

Step 7: run the SQL command :



The screenshot shows a Jupyter Notebook interface with a dark theme. On the left, there's a sidebar titled 'Catalog' with sections like 'For you', 'All', 'My organization', 'workspace', 'system', 'Delta Shares Received', and 'samples'. The main area has a title 'Volume' and a cell numbered '2'. The cell contains the SQL command: 'CREATE VOLUME workspace.new_one.your_volume;'. A red box highlights the play button in the toolbar above the cell. Below the cell, keyboard shortcuts are listed: [Shift+Enter] to run and move to next cell, [Ctrl+Shift+P] to open the command palette, and [Esc H] to see all keyboard shortcuts.

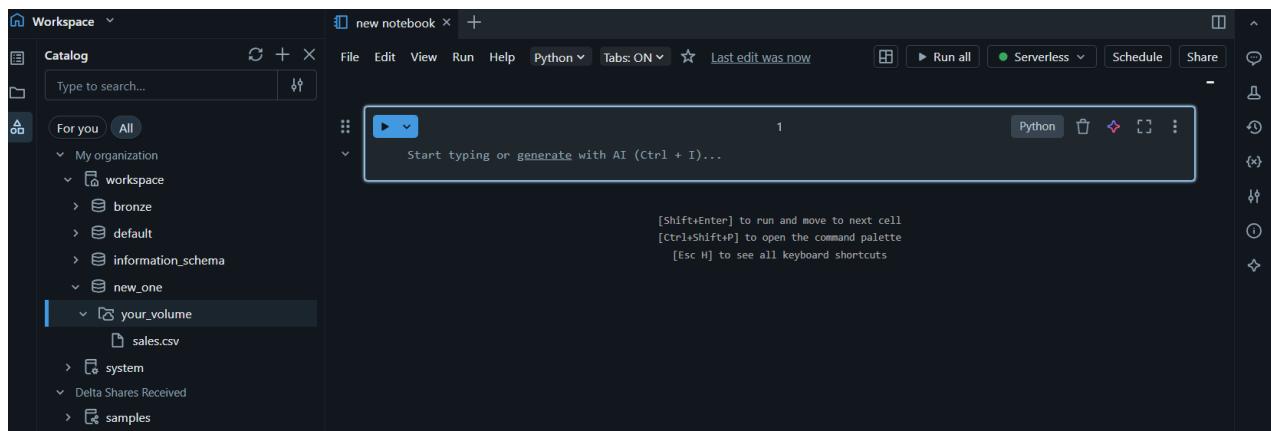
The Volume is created successfully :



The screenshot shows a Jupyter Notebook cell with a green checkmark icon and the text 'Just now (27s)'. The cell contains the SQL command: '%sql CREATE VOLUME workspace.new_one.your_volume;'. Below the cell, it says 'OK' and 'This result is stored as _sqlpdf and can be used in other Python and SQL cells.'

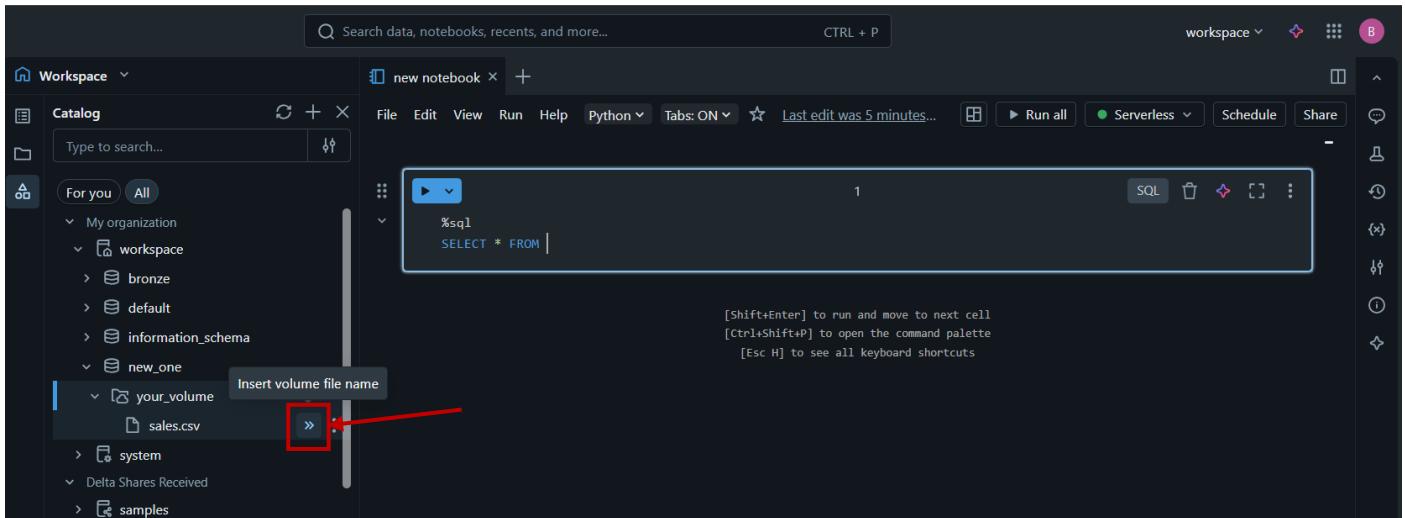
Now I will demonstrate how to execute this using:

Step 1: Open notebooks again:

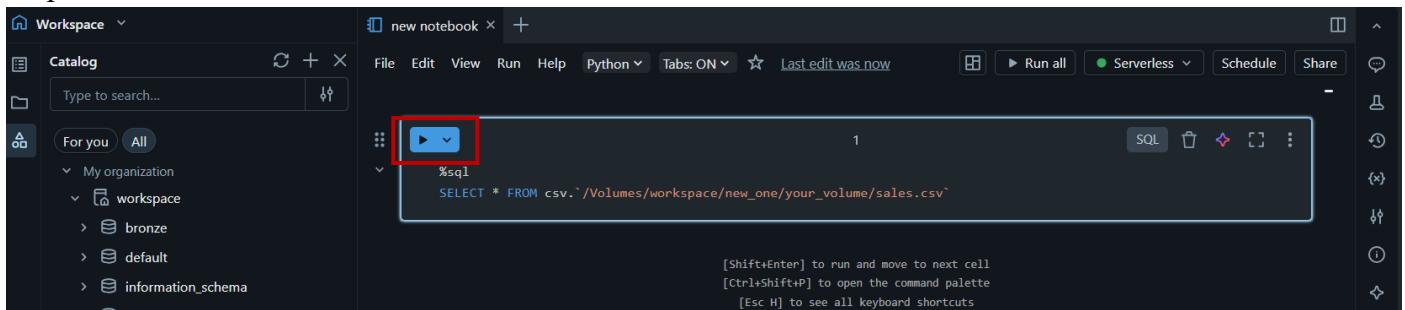


The screenshot shows a Jupyter Notebook interface with a dark theme. On the left, the 'Catalog' sidebar shows the 'new_one' schema expanded, with 'your_volume' selected. This schema contains 'sales.csv'. Other schemas like 'bronze', 'default', 'information_schema', and 'system' are also listed. The main area has a cell numbered '1' with the placeholder text 'Start typing or generate with AI (Ctrl + I)...'. Below the cell, keyboard shortcuts are listed: [Shift+Enter] to run and move to next cell, [Ctrl+Shift+P] to open the command palette, and [Esc H] to see all keyboard shortcuts.

Step 2: Execute the SQL code efficiently by using this icon (it will automatically give the path for the particular file):



Step 3: Run the code



We use **notebooks** in Databricks to access files because they provide an **interactive, easy, and integrated environment** for working with data.

What we did in this lab?

In this lab, we worked in Databricks to understand data organization and storage management, with a strong focus on **volumes**. We created a catalog and schema to structure the workspace, then created a volume to store and manage files securely. I uploaded a file into the volume and accessed it through the notebook environment. We used SQL within the notebook to create and manage the volume, demonstrating how volumes can be controlled programmatically for secure, governed, and efficient file handling in Databricks.