Lab 4

Big Data Analytics on Cloud Computing Infrastructures

Azad Shokrollahi, Mahtab Jamali

1. What is Databricks?

Databricks is a cloud platform built for big-data processing and machine learning with Apache Spark. It provides:

- Spark clusters for distributed computing
- Notebooks (Python, SQL, Scala, R)
- Machine learning with MLlib and MLflow
- Streaming and ETL pipelines
- Lakehouse architecture

Important point:

- Databricks does **not** have its own cloud.
- It always runs on one of these clouds:
    - Microsoft Azure
    - Amazon AWS
    - Google GCP

Databricks is a platform that sits on top of these clouds and manages Spark for you.

---

2. Two Ways to Run Databricks (World 1 vs World 2)

Databricks can be used in two different ways, depending on where you start:

- Cloud provider users (Azure/AWS/GCP) → World 1
- Databricks website users → World 2

This creates two access paths.

WORLD 1 — Cloud-First Databricks

This is when you start from the cloud portal, not from databricks.com.

How you use it:

- Go to Azure Portal, AWS Console, or GCP Console
- Search for "Databricks"
- Create a Databricks workspace
- The cloud provider creates all resources:
    - Virtual machines
    - Networking
    - Storage
    - Spark clusters

What you get:

- Full Spark cluster support
- PySpark
- MLlib
- Streaming
- MLflow
- Workflows and jobs
- All enterprise features

Why it exists:

- Cloud providers want Databricks integrated into their ecosystem.

Best for:

- Students with Azure for Students credits
- University labs
- Companies already using Azure, AWS, or GCP

In World 1, you create Databricks inside the cloud, and Spark runs on your cloud account.

WORLD 2 — Databricks-First (Upgrade Free Edition)

This starts from the Databricks website, not from the cloud portal.

How you use it:

- Go to databricks.com and sign in
- You get the Free Edition
- Click “Upgrade”
- Databricks asks: “Choose your cloud — Azure, AWS, or GCP?”
- Databricks deploys your workspace into that cloud
- You attach your cloud account for billing

Why it exists:

- Many companies buy Databricks directly from Databricks
- They need to choose a cloud during setup
- This is Databricks' direct onboarding path

What you get after upgrading:

- Full Spark clusters
- Full compute power
- MLlib
- MLflow
- Jobs, workflows, and all features

In World 2, you start from the Databricks website, upgrade, choose a cloud, and Databricks creates your workspace in that cloud.

---

3. World 2 History: Community Edition → Free Edition

Before 2024, World 2 included Community Edition.

Old: Community Edition (free Spark)

- Free Spark cluster
- PySpark support
- MLlib machine learning
- Real distributed Spark jobs
- No cloud account needed
- Good for learning and teaching
- Completely free

Why Community Edition was removed:

- Too expensive to give free Spark to everyone
- Heavy workloads caused problems
- Databricks wanted a lightweight free tier

New: Free Edition (no Spark)

Free Edition provides:

- A small serverless SQL warehouse
- Ability to run SQL queries
- Ability to explore the Databricks interface

But it does NOT provide:

- Spark clusters
- PySpark
- MLlib
- Streaming
- Machine learning
- Workflows or jobs
- Any cluster creation

Big idea:
Free Edition = SQL only.
Community Edition = free Spark (now removed).

---

4. Comparison Table

Feature	Community Edition (OLD)	Free Edition (NEW)
Spark cluster	Yes	No
PySpark	Yes	No
MLlib	Yes	No
SQL support	Yes	Yes
Distributed compute	Yes	No
Real cluster attach	No (internal only)	No
Cloud usage	Databricks internal	Databricks internal
Purpose	Learning Spark	Exploring UI + SQL
Status	Removed	Active

---

5. How to Get Real Spark Today

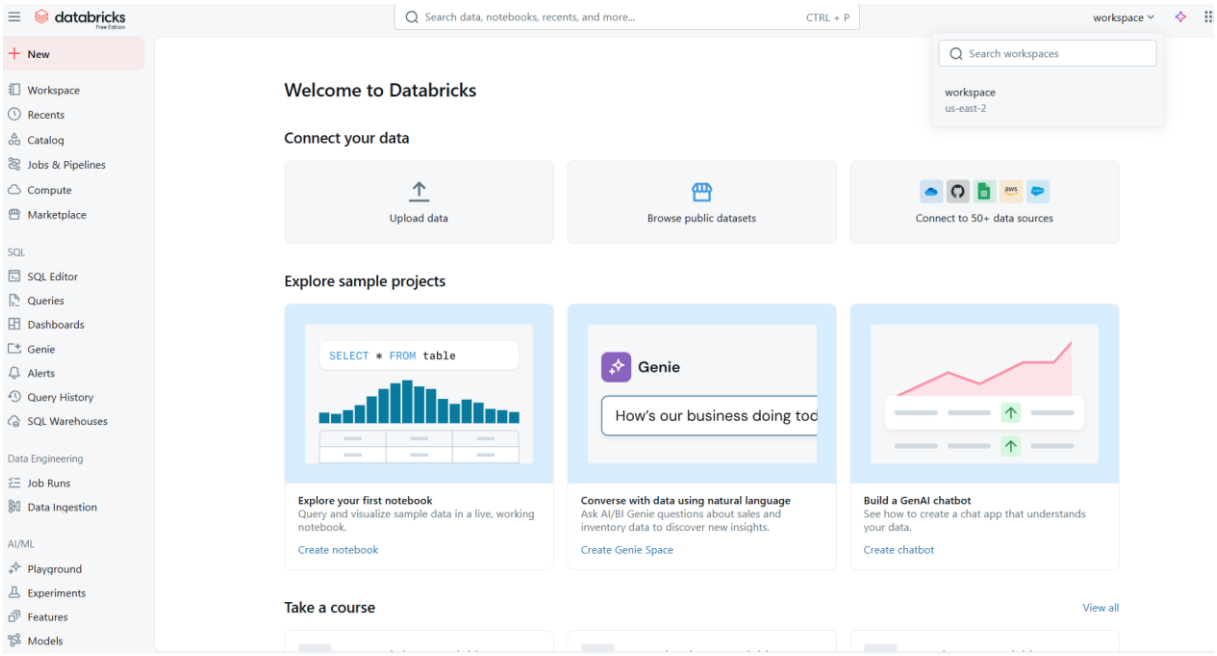
Because Free Edition has no Spark, you must use one of these options:

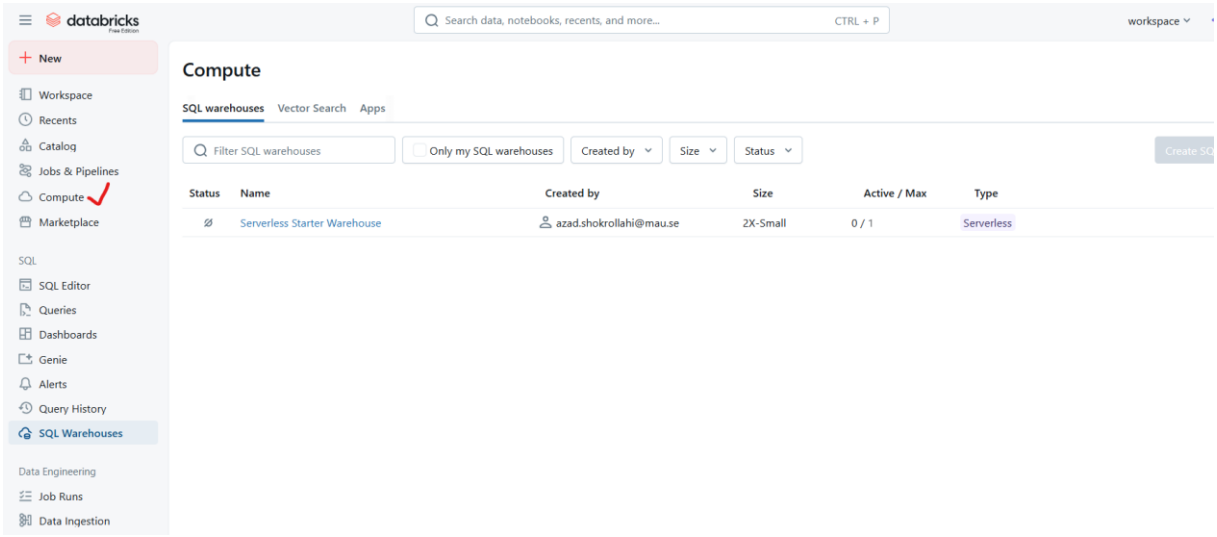
- Azure Databricks (recommended for students, free Azure credits available)
- AWS Databricks free trial (14 days)
- GCP Databricks free trial
- Paid Databricks workspace linked to your cloud account

All of these options provide real Spark clusters.

Free Edition

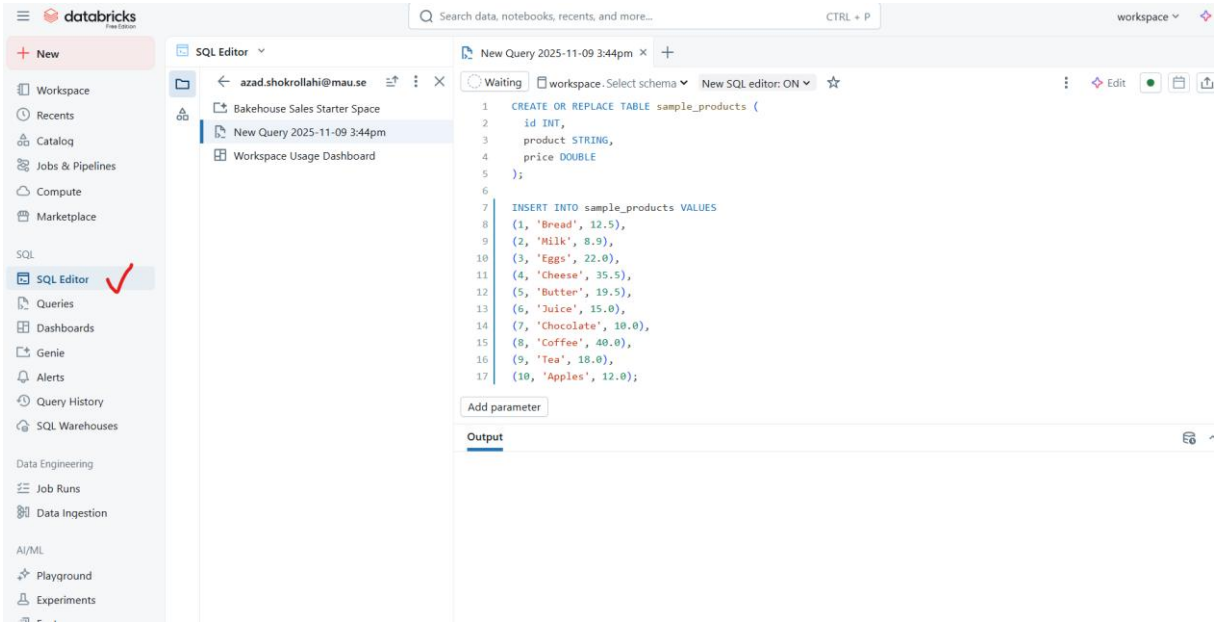
https://www.databricks.com/learn/free-edition

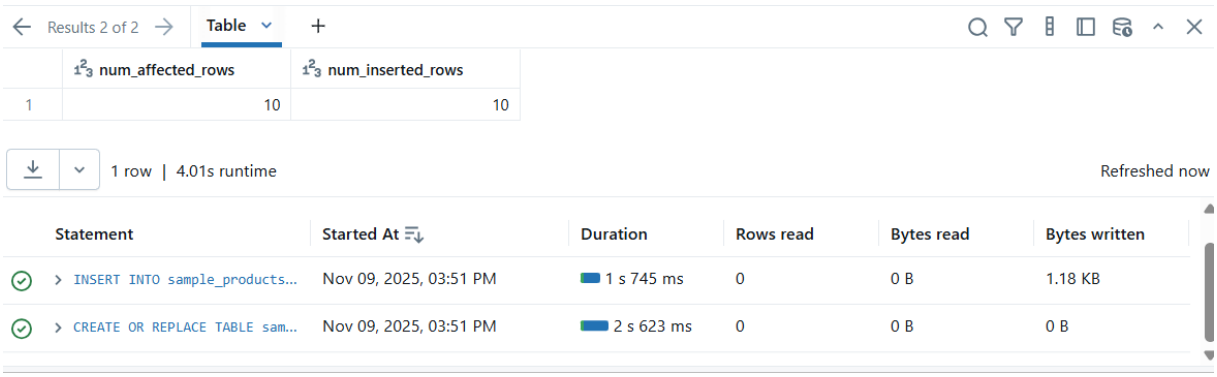


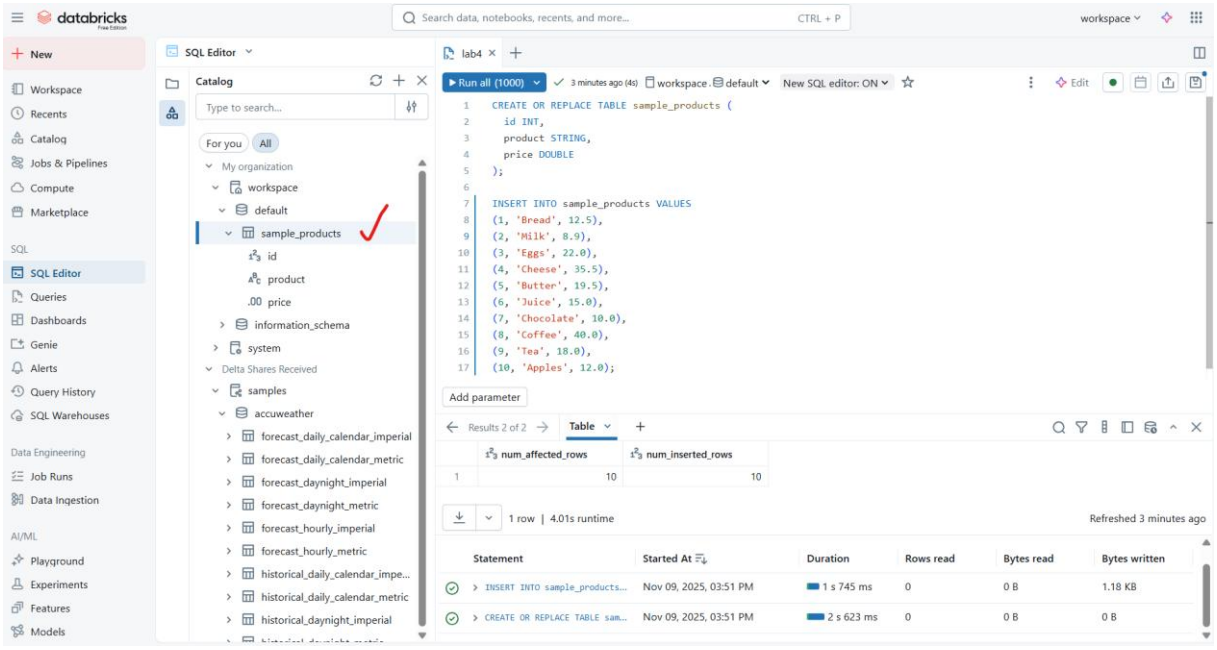


This screen shows the only compute available in Databricks Free Edition—a small serverless SQL warehouse that runs SQL queries but cannot run Spark or machine-learning code.

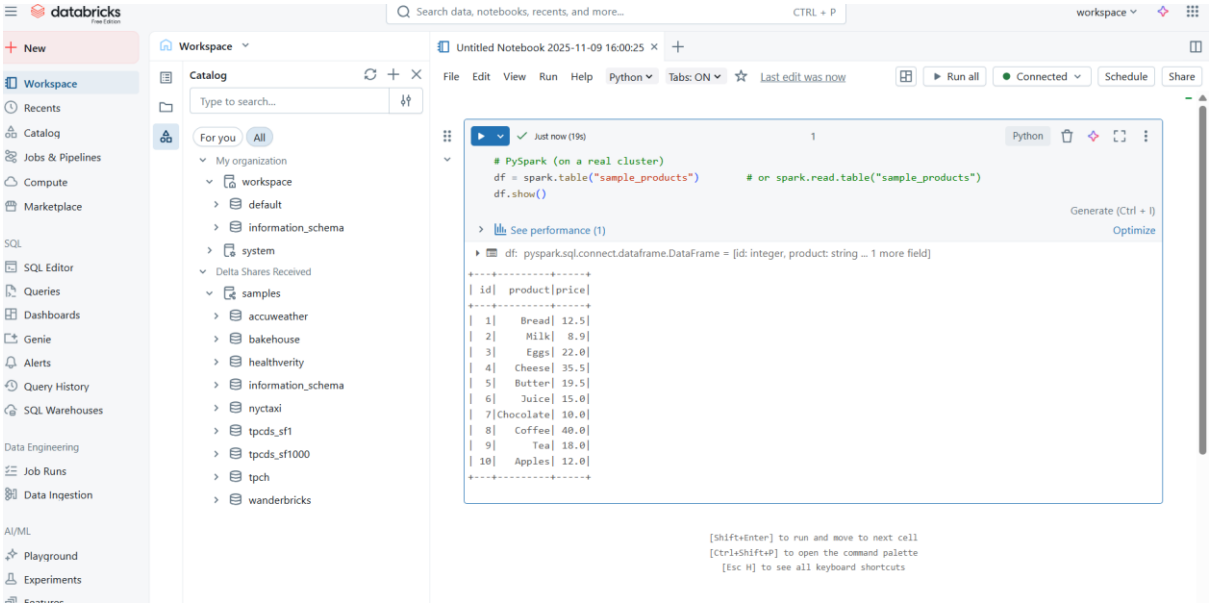
Databricks replaced Community Edition with Free Edition to limit free compute costs by removing Spark clusters and allowing only a small serverless SQL warehouse for basic SQL use.

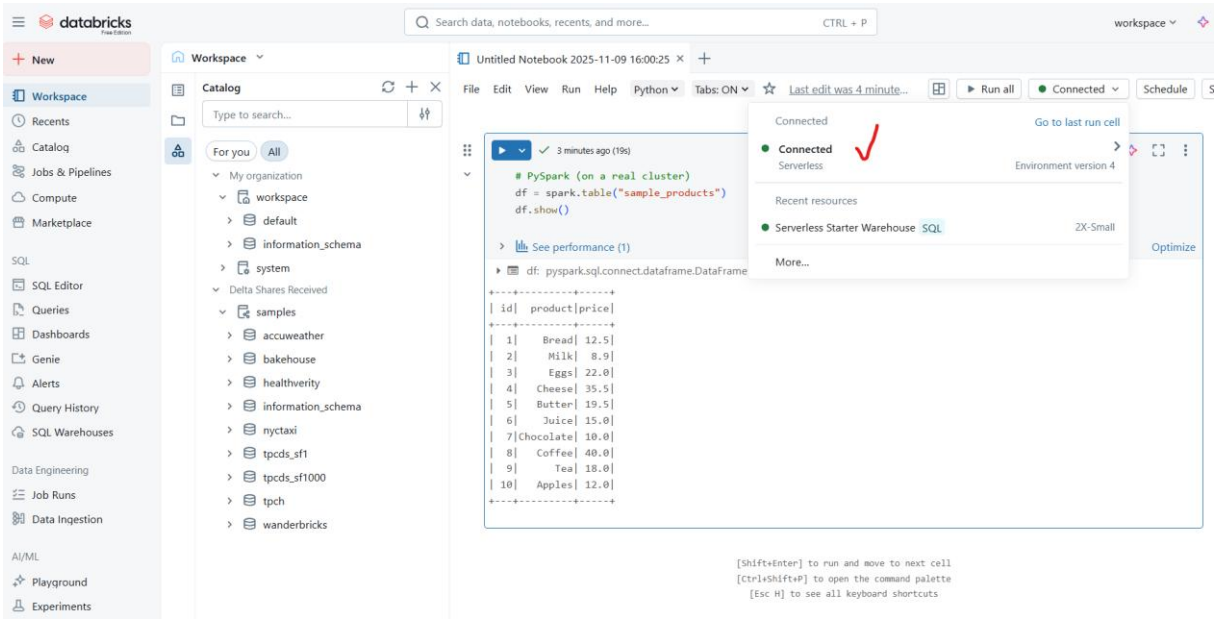






Create notebook





Databricks Lab

Student Tasks

Public Dataset

download the dataset from this link:

Tips Dataset (CSV)

https://raw.githubusercontent.com/mwaskom/seaborn-data/master/tips.csv

Save it as: **tips.csv**

This dataset contains the following columns:
total_bill, tip, sex, smoker, day, time, size

Tasks

Task 1 — Start the SQL Warehouse

- Go to **Compute → SQL Warehouses**
- Start your **Serverless Starter Warehouse**
- Take a screenshot showing it is *Connected*

Task 2 — Upload the Public Dataset

- Go to **Catalog → default**
- Click **Create → Add data → Upload file**
- Upload **tips.csv**
- Name the table **tips_data**
- Select your SQL warehouse
- Create the table
- Provide a screenshot showing the table under the **default** catalog

---

Task 3 — Explore the Dataset (SQL)

Write and run SQL queries to answer the following.
Provide a screenshot for each result.

1. Display the first **20 rows** of the dataset
2. Count the **total number of rows**
3. Find the **average total bill**
4. Show the **top 5 highest total bills**
5. Calculate the **total tip amount for each day**
6. Compare the **average bill of smokers vs non-smokers**
7. Compare the **average tip amount for lunch vs dinner**

Task 4 — Create a New Calculated Column

Create a column named **tip_percentage**
(tip divided by total_bill times 100)

Then display the following for all rows:

- total_bill
- tip
- tip_percentage

Screenshot required.

Task 5 — Create a New Table

Create a new table named **high_spenders** containing only rows where:

total_bill > 30

Then:

- Display the contents of **high_spenders**
- Show how many rows this table contains
- Provide screenshots

Task 6 — Read the Tables in a Python Notebook

Open a Python notebook and:

- Read the **tips_data** table using spark.table()
- Read the **high_spenders** table
- Display both DataFrames using .show()

Provide screenshots of the outputs.

Task 7 — Export Results

Download a CSV export of the **high_spenders** table using the Databricks download option.

Provide a screenshot showing that the download completed.

Task 8 — Reflection

Explore at least **five** items from the left navigation menu in Databricks.
For **each** of the five items:

- Take **one screenshot**, and
- Write **2–5 sentences** explaining what you observed.

Optional Task (For Students With Cloud Access)

This task is optional and only for students who have Azure, AWS, or GCP credits.

Students who have access to a cloud account (Azure for Students credit, AWS free trial, or GCP trial) may complete the following optional task to run **real Spark and machine learning using MLlib**.

Optional: Run Machine Learning on Spark (Cloud Databricks)

If you choose to complete this task:

1. Create a Databricks workspace on one of the cloud providers:
    - Azure Databricks
    - AWS Databricks
    - GCP Databricks
2. Create a **Spark cluster** inside your cloud workspace
3. Create a new **Python notebook**
4. Upload or load the **tips.csv** dataset
5. Using PySpark, complete the following steps:
    - Load the CSV into a Spark DataFrame
    - Select appropriate features and label
    - Assemble features using **VectorAssembler**
    - Train a simple model using Spark MLlib, such as:
        - Logistic Regression
        - Decision Tree
        - Linear Regression
    - Show predictions

    - Evaluate the model using accuracy or RMSE
6. Take screenshots showing:
    - The ML model training
    - The prediction output

Optional Deliverable

A short summary including:

- Which cloud you used
- Your Spark cluster settings
- The ML algorithm you ran
- One screenshot of your model output