Lab Manual- Azure Data Bricks Provisioning and Data Ingestion Part1

Prepared for:

Date: 18th March 2022

Prepared by:

Document Name: Lab Manual

Document Number AZLabn990

Contributor:

Contents

1. Introduction

Systems are working with massive amounts of data in petabytes or even more and it is still growing at an exponential rate. Big data is present everywhere around us and comes in from different sources like social media sites, sales, customer data, transactional data, etc

Apache Spark is an open-source, fast cluster computing system and a highly popular framework for big data analysis. This framework processes the data in parallel that helps to boost the performance. It is written in Scala, a high-level language, and also supports APIs for Python, SQL, Java and R.

Azure Databricks is the implementation of Apache Spark on Azure. With fully managed Spark clusters, it is used to process large workloads of data and also helps in data engineering, data exploring and also visualizing data using Machine learning.

In this Lab, you do the following:

- Provision Azure Data Bricks
- Configure Azure Data bricks Cluster
- Ingest Data to Azure Databricks with Delta Lake

2. Lab 1: Provision Azure Data Bricks

1. Sign in to Azure Portal
2. In the Search Bar type Databricks and select Azure Databricks
3. Click Create to Create Azure Databricks workspace
4. Use below parameter in the wizard
    - **Resource Group : Your Resource Group**
    - **Virtual Machine Name : Anyname**
    - **Pricing Tier : Standard**
    - **Size : Default**
    - **Username : VMAdmin**
    - **Password : Password@123**
    - **Ports : 80,3389**

Home > Azure Databricks >

Create an Azure Databricks workspace ...

Basics | Networking | Advanced | Tags | Review + create

Project Details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ: Visual Studio Enterprise Subscription

Resource group * ⓘ: bipeen7676

Create new

Instance Details

Workspace name *: bipeendb

Region *: East US

Pricing Tier * ⓘ: Standard (Apache Spark, Secure with Azure AD)

Review + create | < Previous | Next : Networking >

5. Leave all option default and click Next

Home > Azure Databricks >

Create an Azure Databricks workspace ...

Basics | Networking | Advanced | Tags | Review + create

Deploy Azure Databricks workspace with Secure Cluster Connectivity (No Public IP) ⓘ: ○ Yes ◉ No

Deploy Azure Databricks workspace in your own Virtual Network (VNet): ○ Yes ◉ No

6. Leave all option default and click Next

Home > Azure Databricks >

Create an Azure Databricks workspace ...

Basics | Networking | **Advanced** | Tags | Review + create

Enable Infrastructure Encryption ☐

⚠The current pricing tier does not support infrastructure encryption.

7. Click **Review and create**

Home > Azure Databricks >

Create an Azure Databricks workspace ...

✔ Validation Succeeded

Basics | Networking | Advanced | Tags | **Review + create**

Summary

Basics

Workspace name	bipeendb
Subscription	Visual Studio Enterprise Subscription
Resource group	bipeen7676
Region	East US
Pricing Tier	standard

Networking

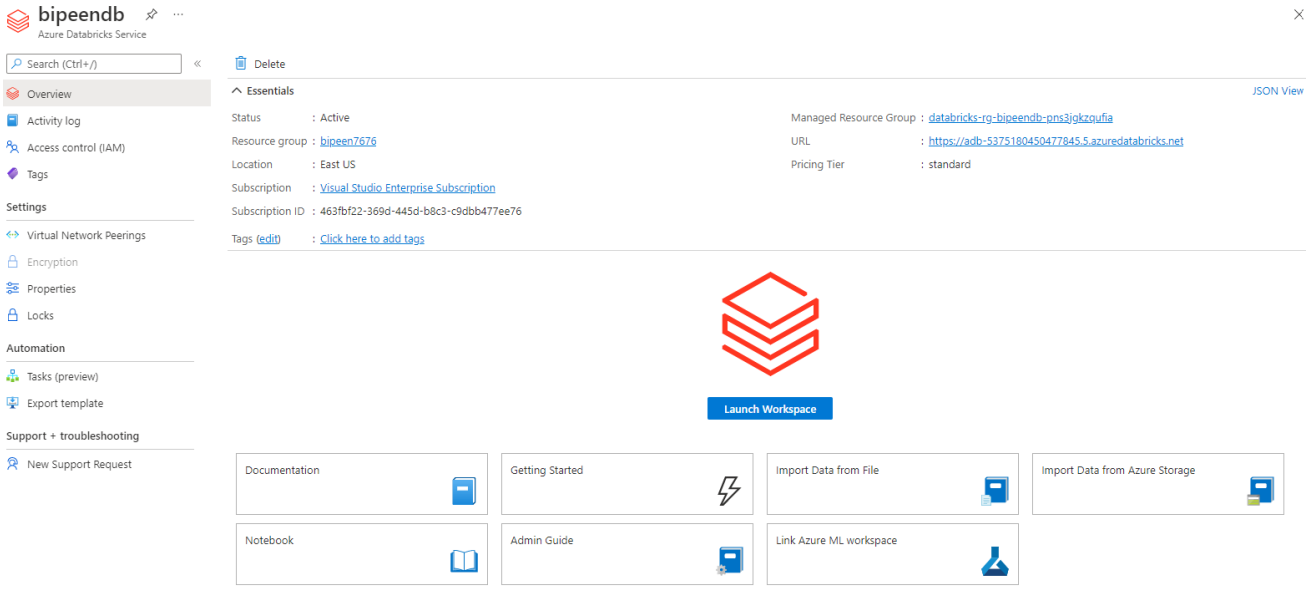
Deploy Azure Databricks workspace with Secure Cluster Connectivity (No Public IP)	No
Deploy Azure Databricks workspace in your own Virtual Network (VNet)	No

Advanced

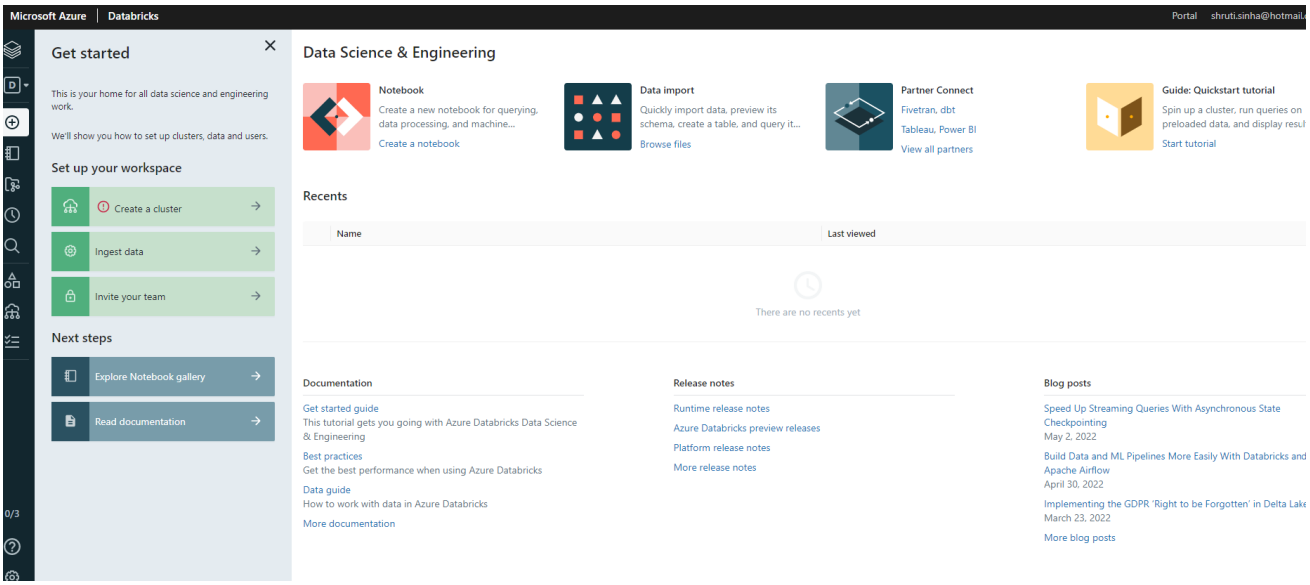
Enable Infrastructure Encryption	No

3. Lab 2: Configure Databricks Cluster with Apace Spark

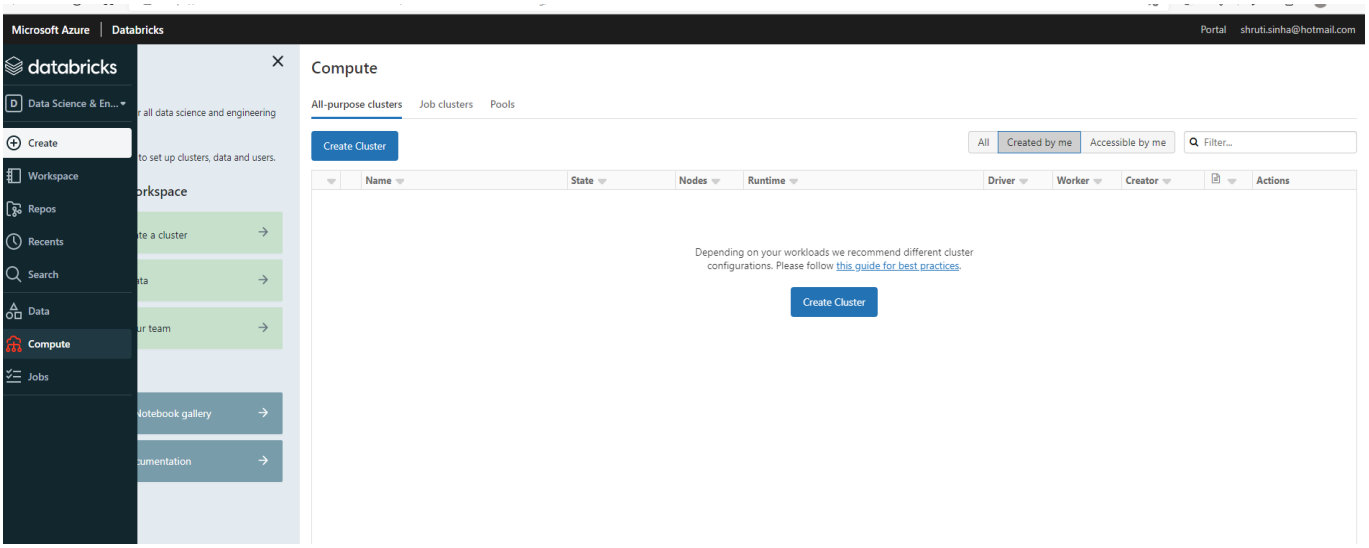
- Click your workspace to open it.



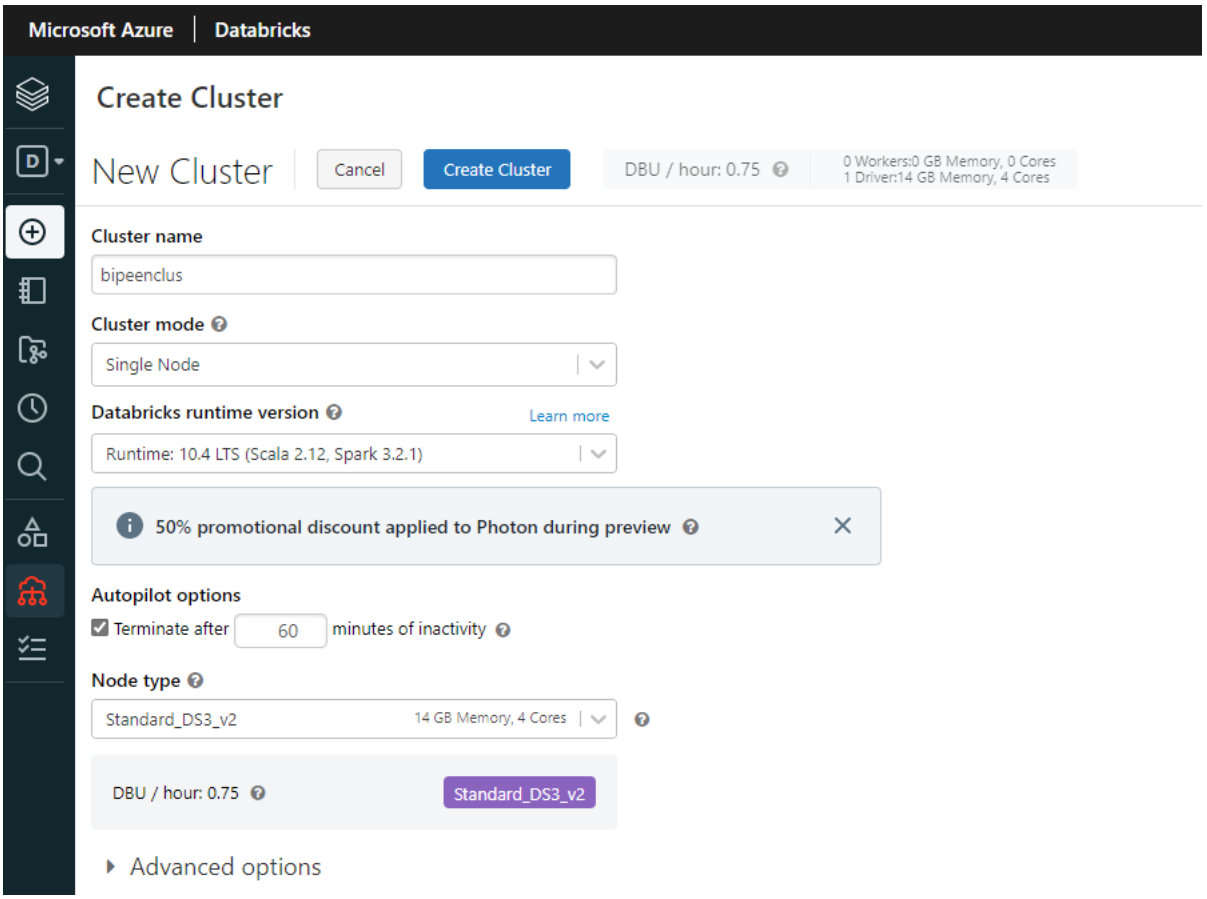
- Click Launch workspace to open **Databricks studio**



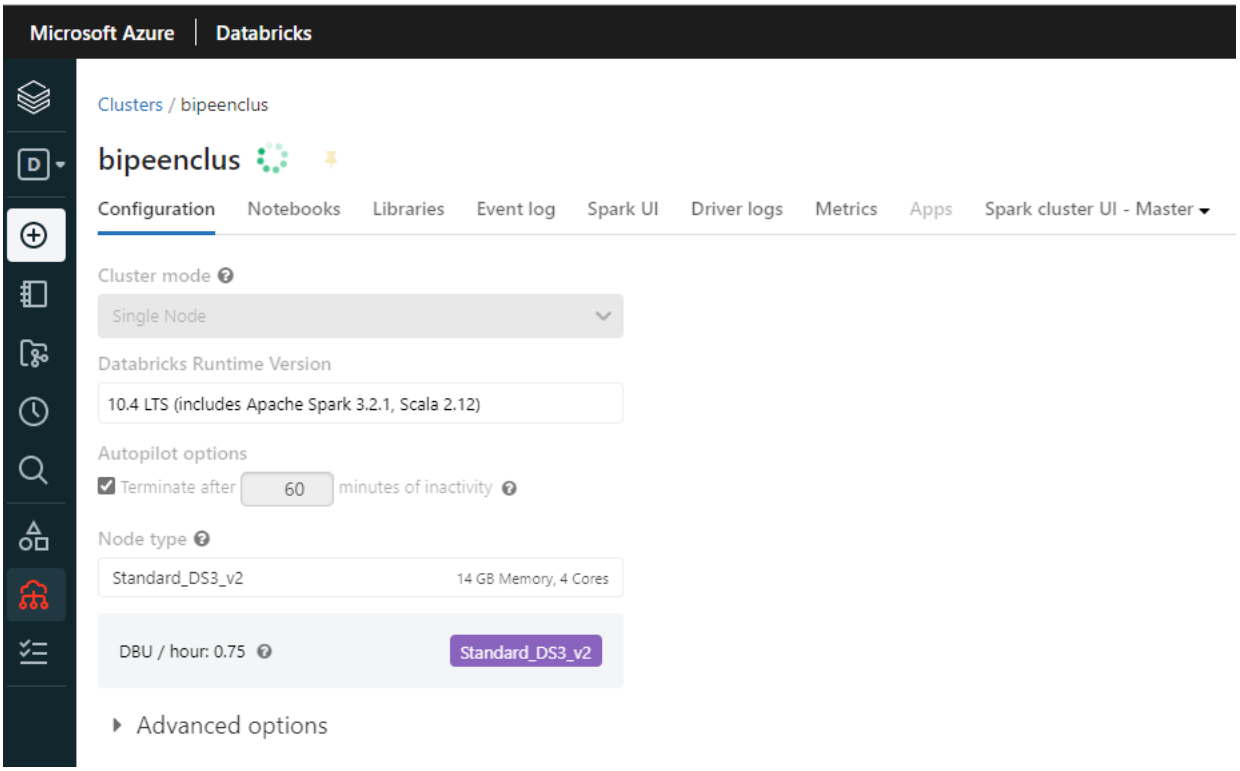
- In the Databricks Studio, Click **Compute** from left hand side menu



- Click Create Cluster and type your **cluster name**
- In the **cluster Node** select **Single Node**
- In the **terminat**e after type **60**

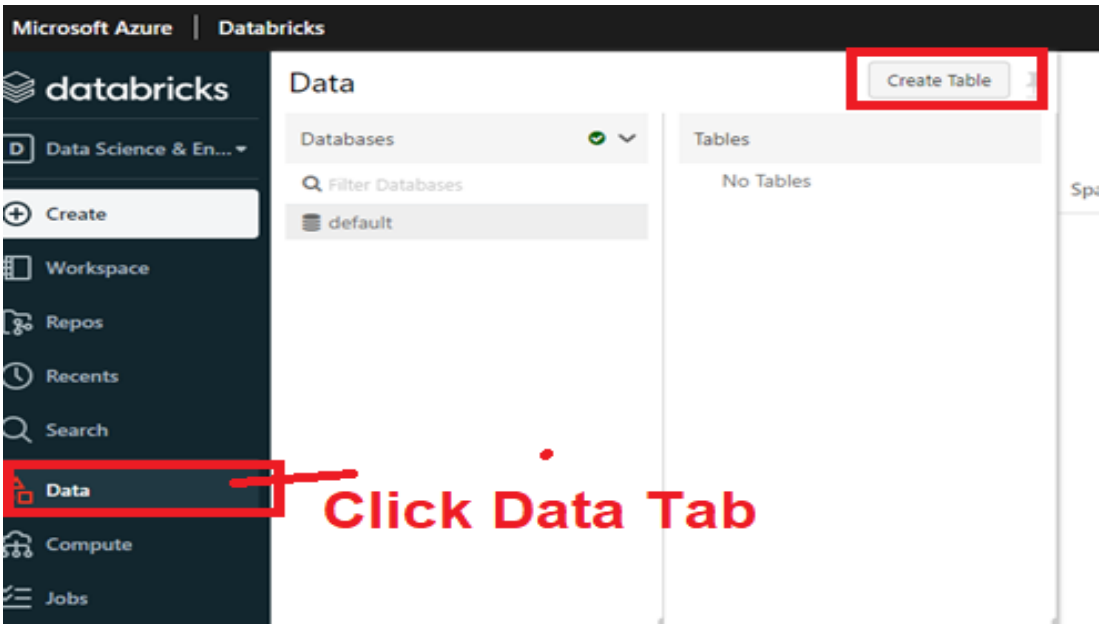


- Click **Create Cluster**. It will take 5-10 Minute

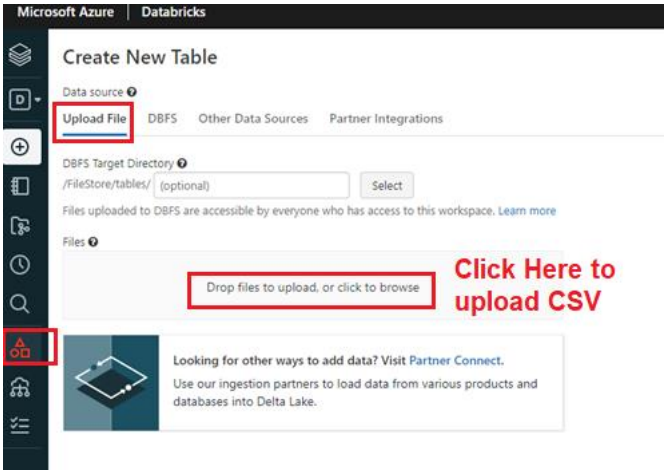


4. Lab 3: Data Ingestion using Delta lake Table

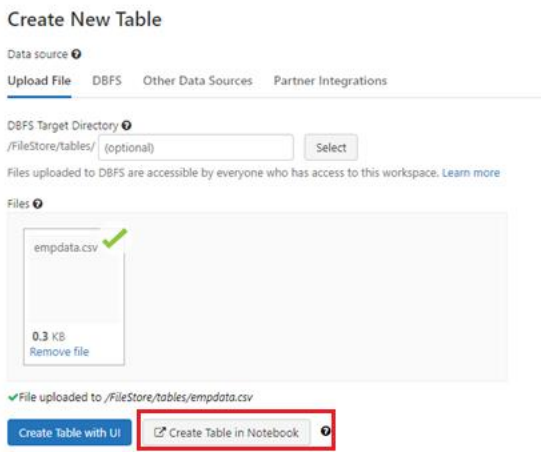
- Click **Data** menu from Left side Pane and Click **Create Table**



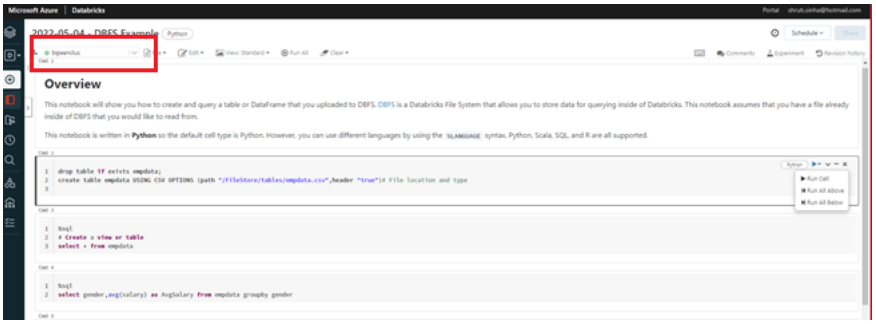
- **Upload** your CSV File



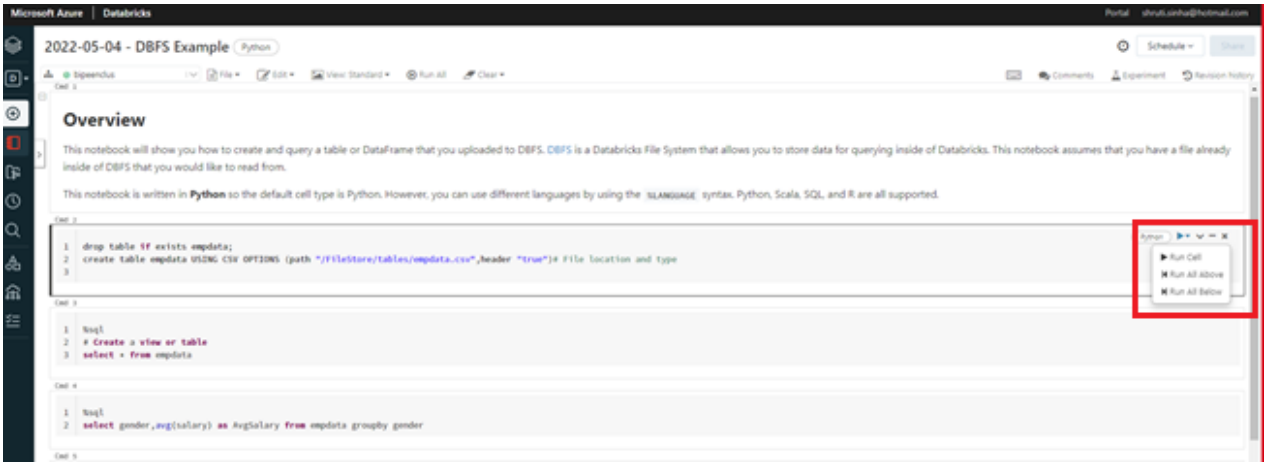
- Once your CSV Uploaded click "**Create Table in Notebook**"



- In The notebook first **attach your cluster**



- In The notebook in each cell, change the type from **Python** to **SQL**



- In The notebook in **First Cell** Remove existing code and type below **SQL Script** to create Table and Click **Run Cell**

drop table if exists empdata;
create table empdata USING CSV OPTIONS (path "/FileStore/tables/empdata.csv",header "true")

- In The notebook in **2nd Cell** Remove existing code and type below SQL Script to show the data from just created table and click run cell. It will show the content of your CSV Data as Table

select * from empdata

```
%sql
drop table if exists empdata;
create table empdata USING CSV OPTIONS (path "/FileStore/tables/empdata.csv",header "true")
```

(1) Spark Jobs

OK

Command took 4.78 seconds -- by shruti.sinha@hotmail.com at 5/4/2022, 11:11:25 AM on bipeenclus

Cmd 3

```
%sql
select * from empdata
```

(1) Spark Jobs

Table | Data Profile

	empid	empname	gender	salary	dsepartment
1	1	Mohit	Male	5000	IT
2	2	Rohit	Male	7000	Sales
3	3	John	Male	3000	IT
4	4	Peter	Male	5000	MKT
5	5	Kevin	Male	6500	IT
6	6	Riya	Female	9000	Sales
7	7	Swetha	Female	6700	MKT

Showing all 9 rows.

Command took 0.61 seconds -- by shruti.sinha@hotmail.com at 5/4/2022, 11:15:12 AM on bipeenclus

- In The notebook in **3rd Cell** Remove existing code and type below SQL Script to show the data from just created table and click run cell. It will show the content of your CSV Data as Table with condition

select * from empdata where dsepartment='IT'

Cmd 4

```
%sql
select * from empdata where dsepartment='IT'
```

(1) Spark Jobs

Table Data Profile

	empid	empname	gender	salary	dsepartment
1	1	Mohit	Male	5000	IT
2	3	John	Male	3000	IT
3	5	Kevin	Male	6500	IT
4	9	Jonson	Male	3000	IT

Showing all 4 rows.

Command took 0.45 seconds -- by shruti.sinha@hotmail.com at 5/4/2022, 11:27:23 AM on bipeenclus