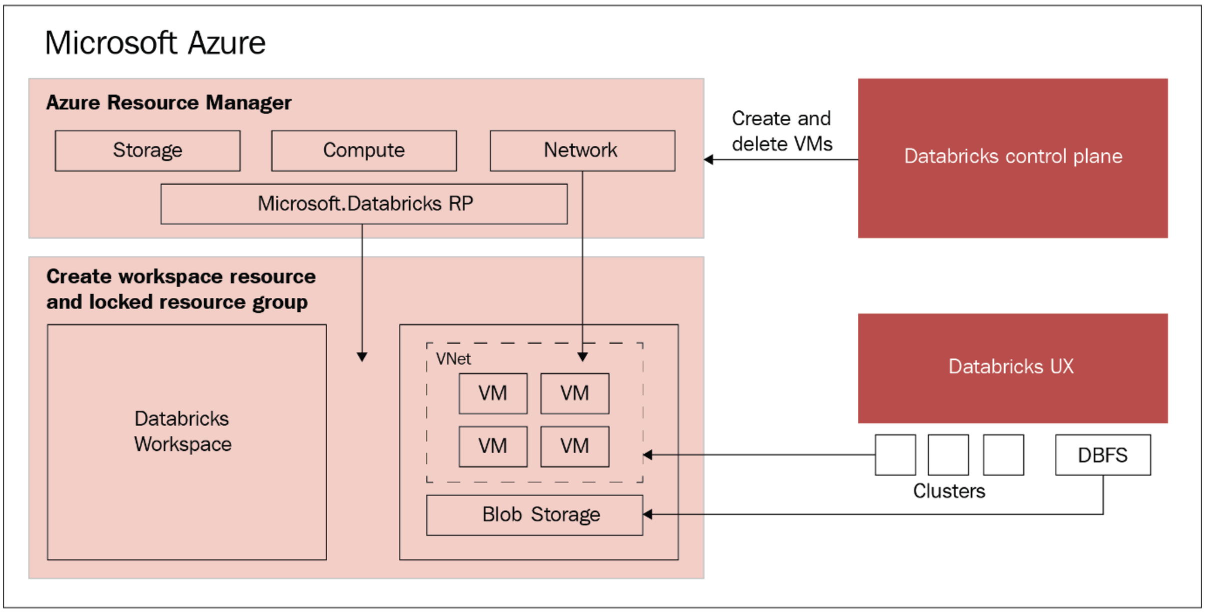
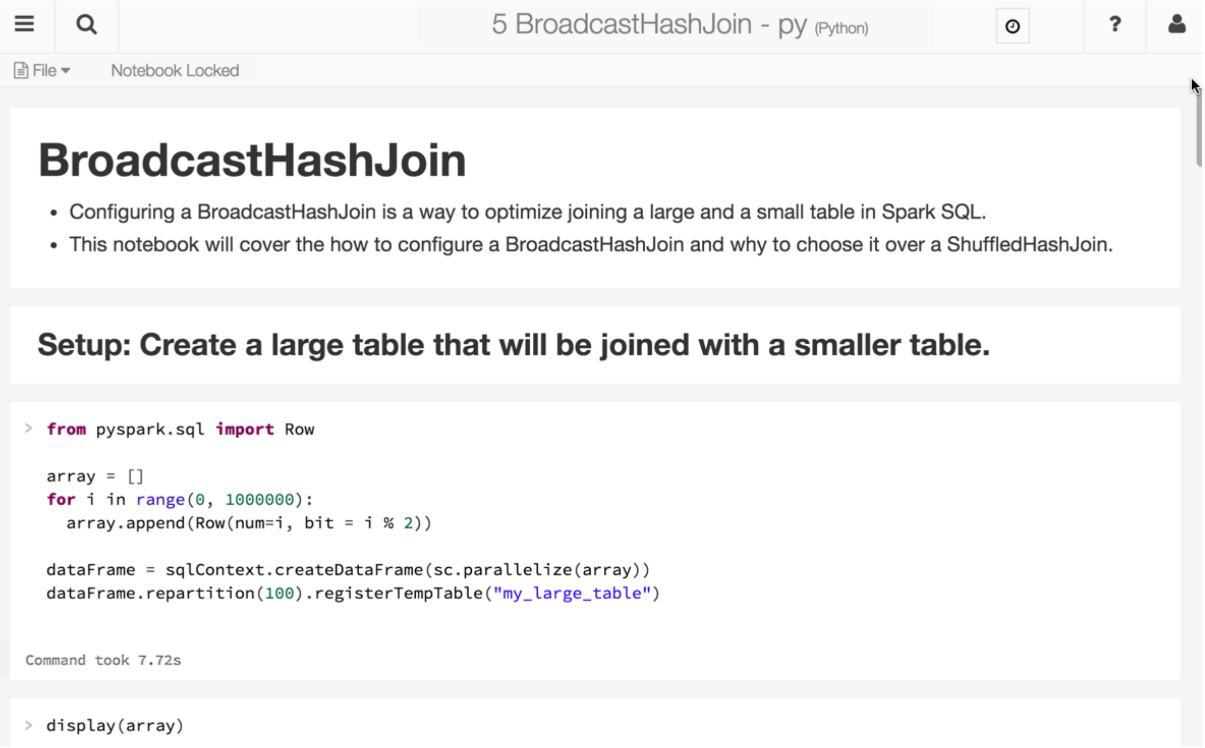
Chapter 1: Introduction to Azure Databricks





BroadcastHashJoin

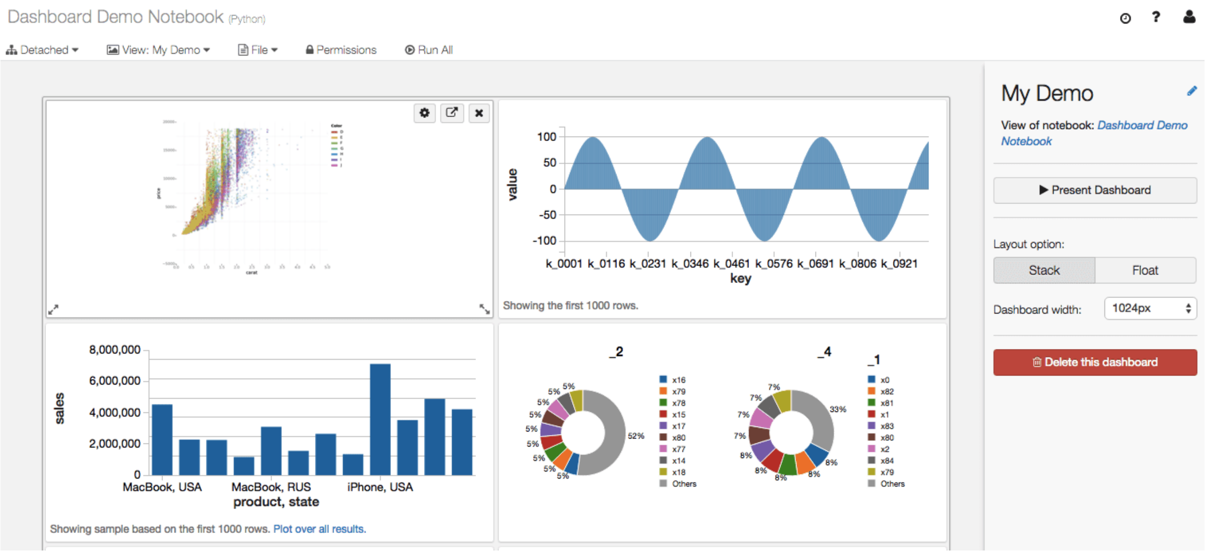
- Configuring a BroadcastHashJoin is a way to optimize joining a large and a small table in Spark SQL.
- This notebook will cover the how to configure a BroadcastHashJoin and why to choose it over a ShuffledHashJoin.

Setup: Create a large table that will be joined with a smaller table.

```
from pyspark.sql import Row

array = []
for i in range(0, 1000000):
  array.append(Row(num=i, bit = i % 2))

dataFrame = sqlContext.createDataFrame(sc.parallelize(array))
dataFrame.repartition(100).registerTempTable("my_large_table")
```

```
Command took 7.72s
```

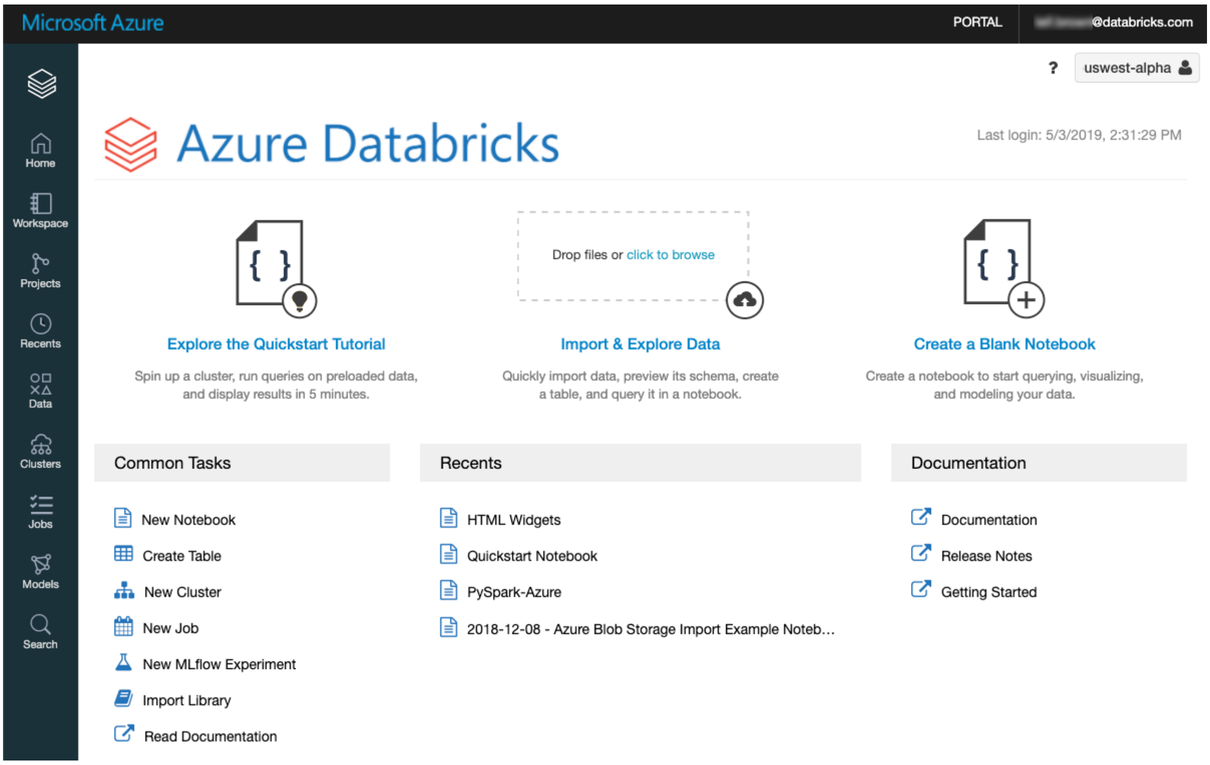
```
display(array)
```



Showing the first 1000 rows.

Showing sample based on the first 1000 rows. Plot over all results.

View of notebook: *Dashboard Demo Notebook*

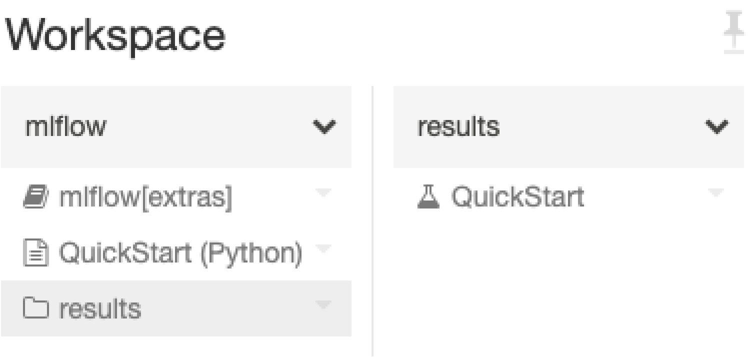


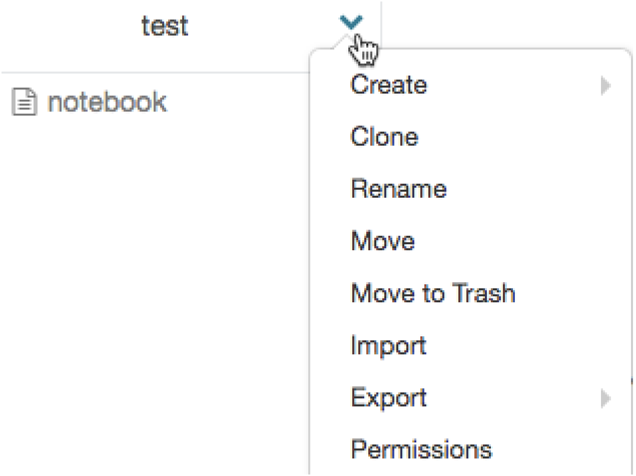
Last login: 5/3/2019, 2:31:29 PM

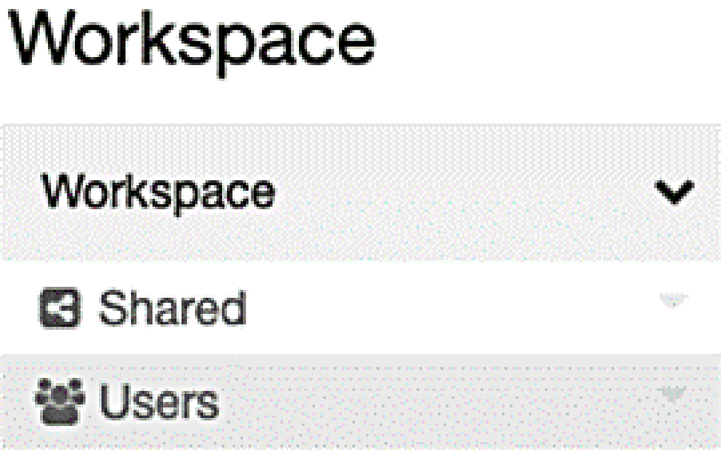
Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

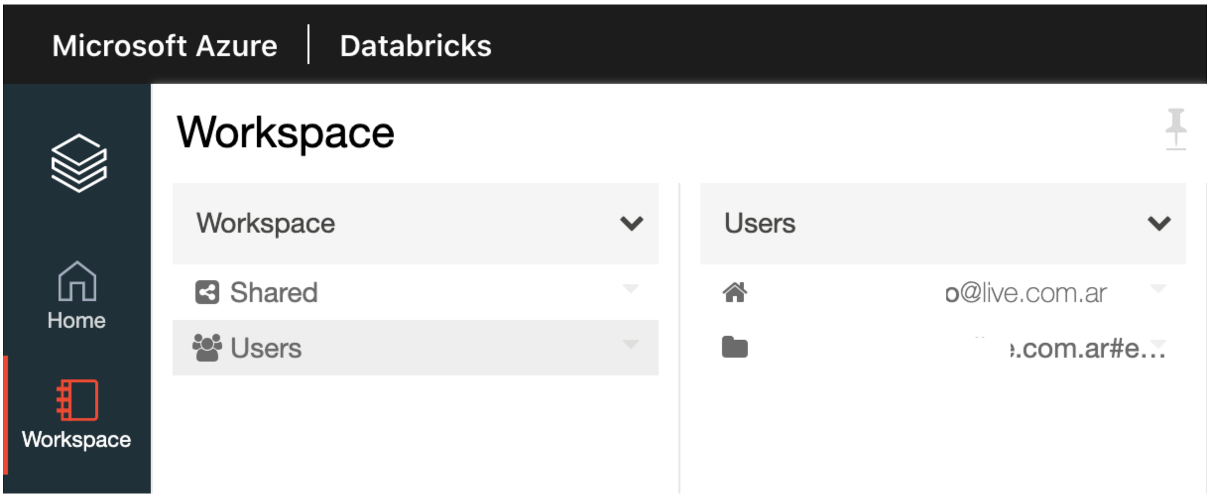
Quickly import data, preview its schema, create a table, and query it in a notebook.

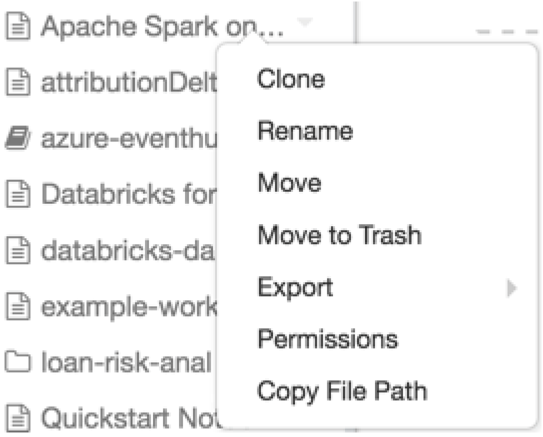
Create a notebook to start querying, visualizing, and modeling your data.

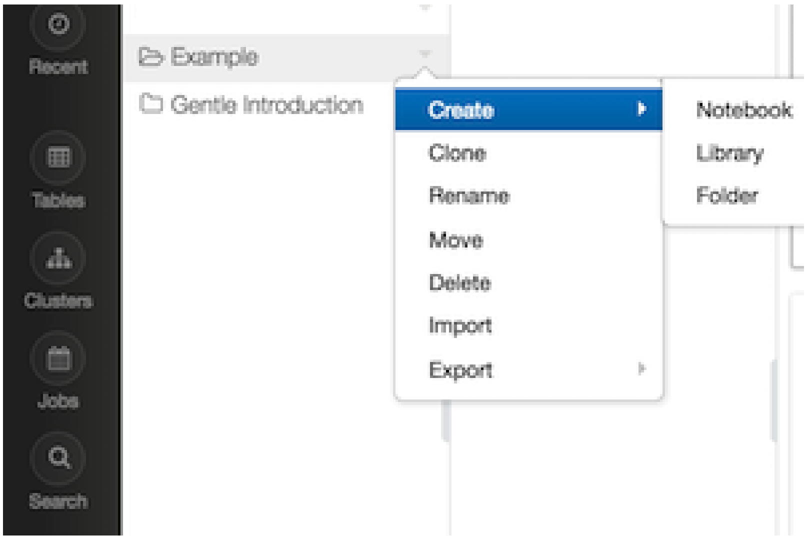


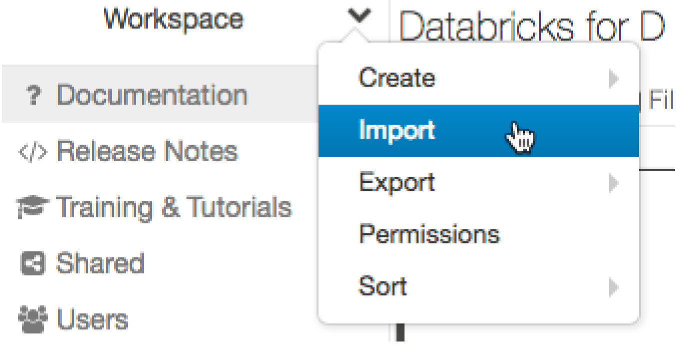

test
notebook
Create
Clone
Rename
Move
Move to Trash
Import
Export
Permissions

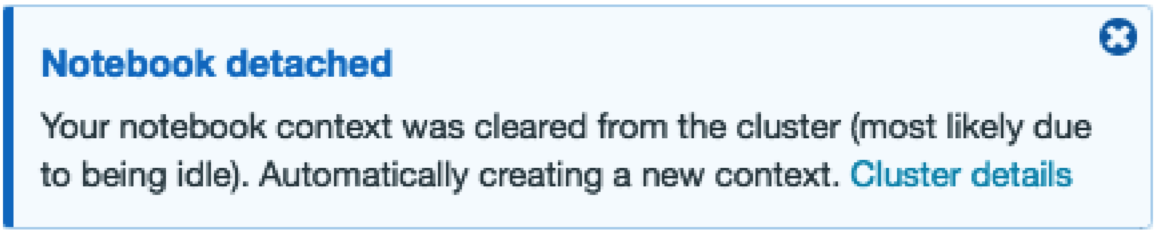

Workspace
Workspace
Shared
Users


Microsoft Azure
Databricks
Home
Workspace
Workspace
Workspace
Shared
Users
Users
o@live.com.ar
.com.ar#e…



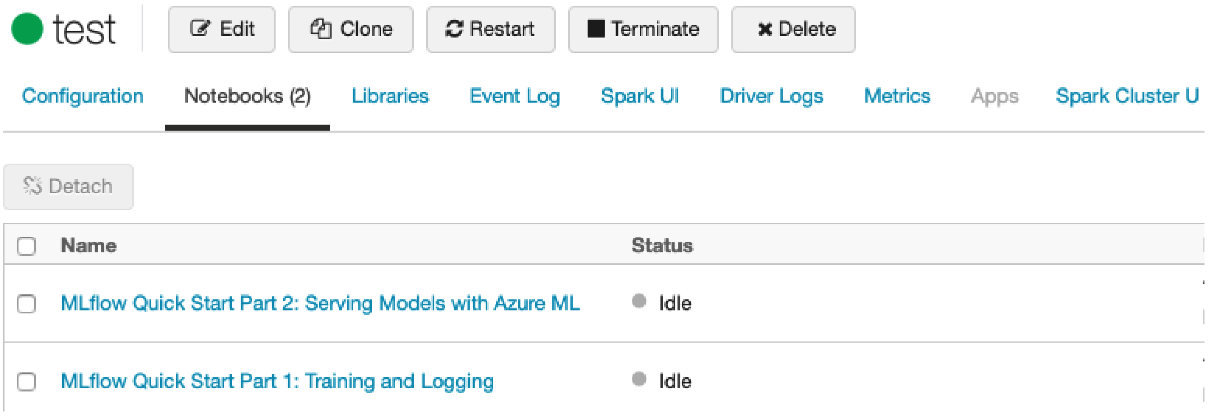




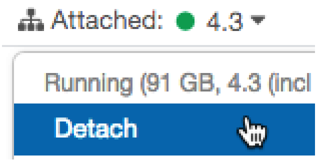

Notebook detached

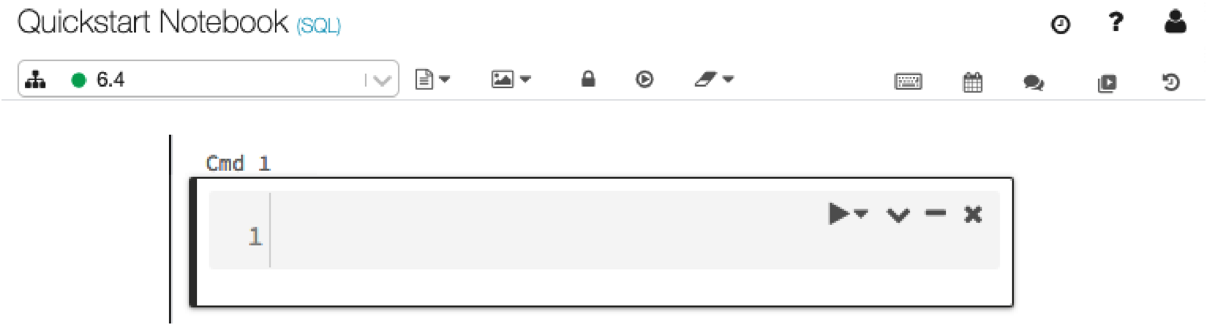
Your notebook context was cleared from the cluster (most likely due to being idle). Automatically creating a new context. Cluster details

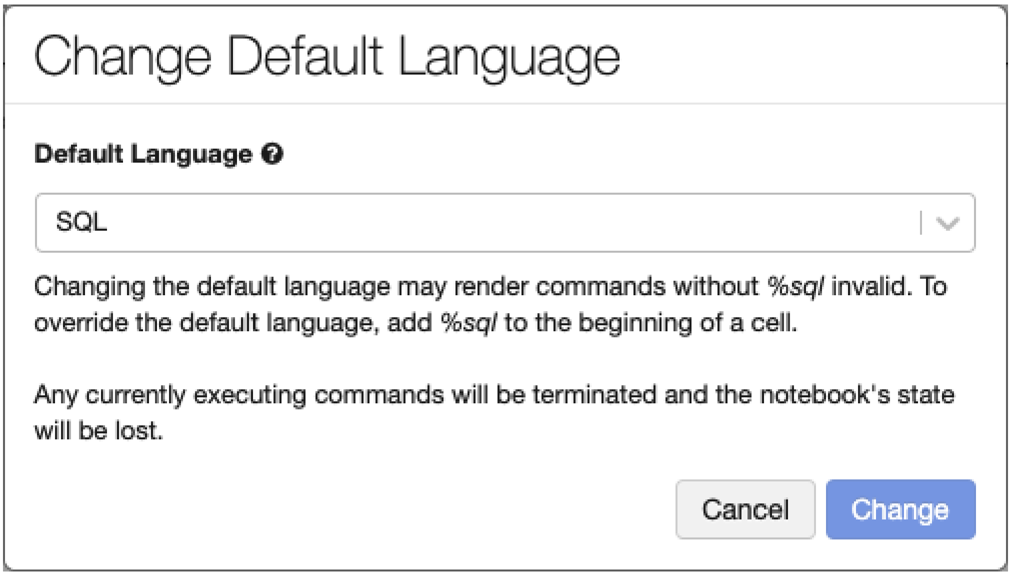
Class	Variable Name
SparkContext	sc
SQLContext/HiveContext	sqlContext
SparkSession (Spark 2.x)	spark



□ Name	Status
□ MLflow Quick Start Part 2: Serving Models with Azure ML	● Idle
□ MLflow Quick Start Part 1: Training and Logging	● Idle







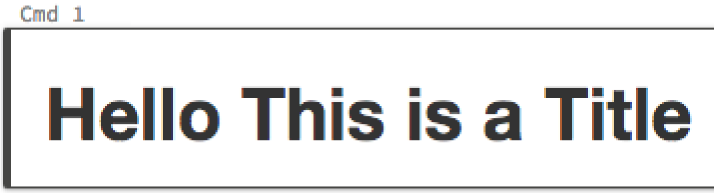
Change Default Language

Default Language

SQL

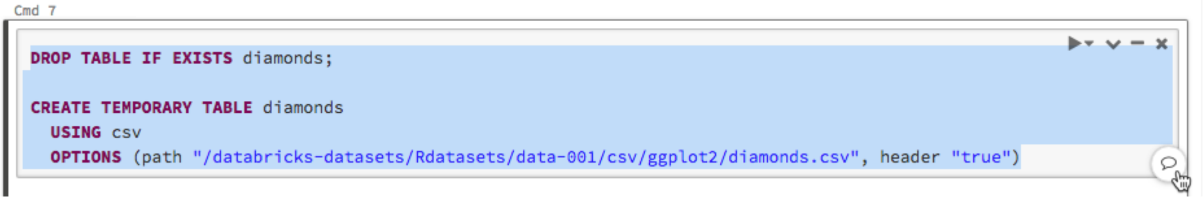
Changing the default language may render commands without *%sql* invalid. To override the default language, add *%sql* to the beginning of a cell.

Any currently executing commands will be terminated and the notebook's state will be lost.



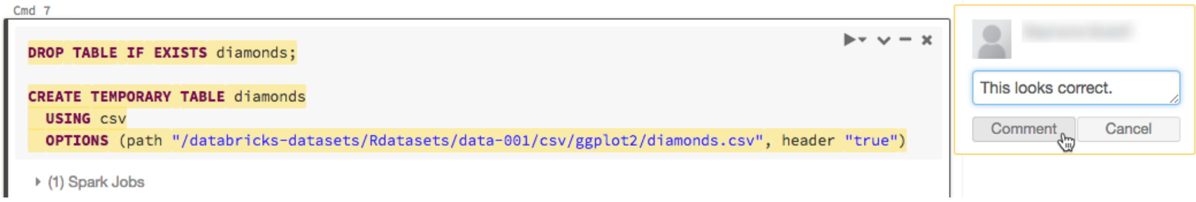
Cmd 1

Hello This is a Title



Cmd 7

```
DROP TABLE IF EXISTS diamonds;

CREATE TEMPORARY TABLE diamonds
  USING csv
  OPTIONS (path "/databricks-datasets/Rdatasets/data-001/csv/ggplot2/diamonds.csv", header "true")
```

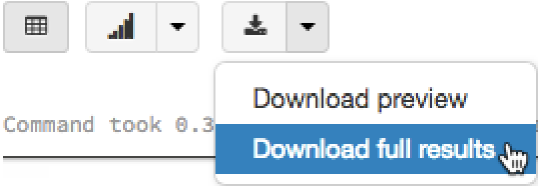


Cmd 7

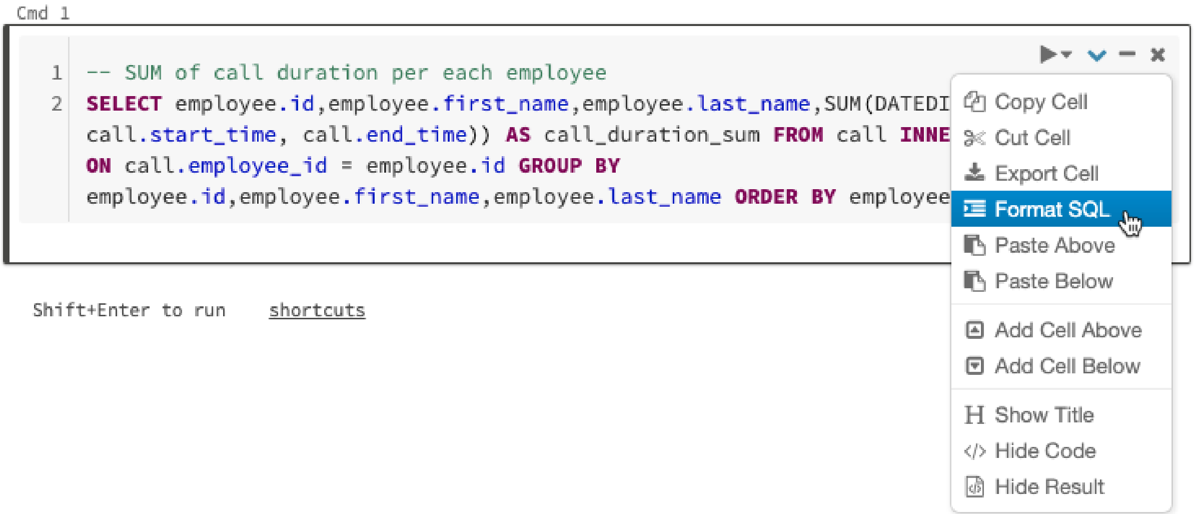
```
DROP TABLE IF EXISTS diamonds;

CREATE TEMPORARY TABLE diamonds
  USING csv
  OPTIONS (path "/databricks-datasets/Rdatasets/data-001/csv/ggplot2/diamonds.csv", header "true")
```

▸ (1) Spark Jobs

This looks correct.

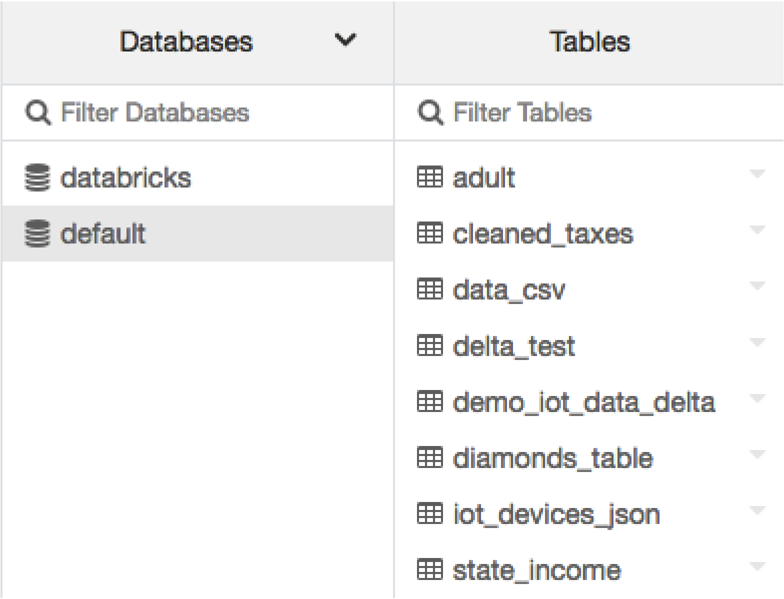


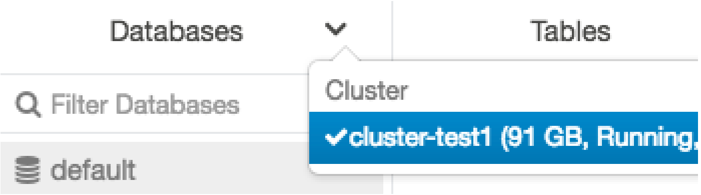
```
Command took 0.3
```

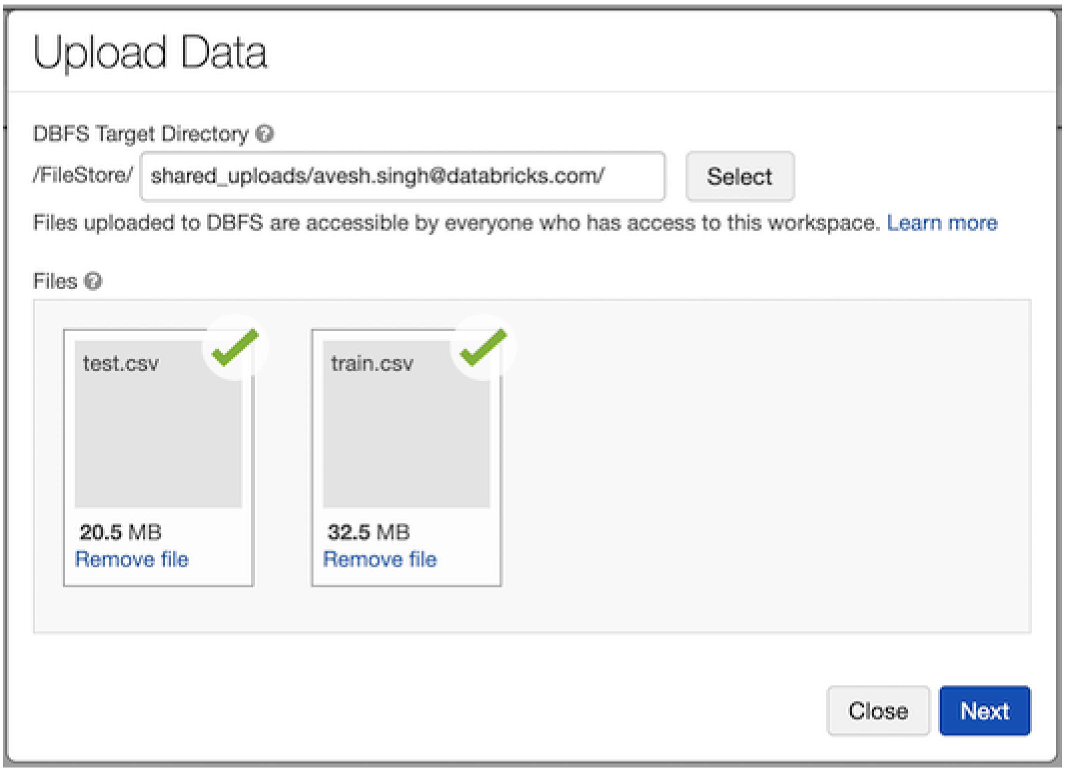


Cmd 1

```
-- SUM of call duration per each employee
SELECT employee.id,employee.first_name,employee.last_name,SUM(DATEDI
call.start_time, call.end_time)) AS call_duration_sum FROM call INNE
ON call.employee_id = employee.id GROUP BY
employee.id,employee.first_name,employee.last_name ORDER BY employee
```







Upload Data

DBFS Target Directory

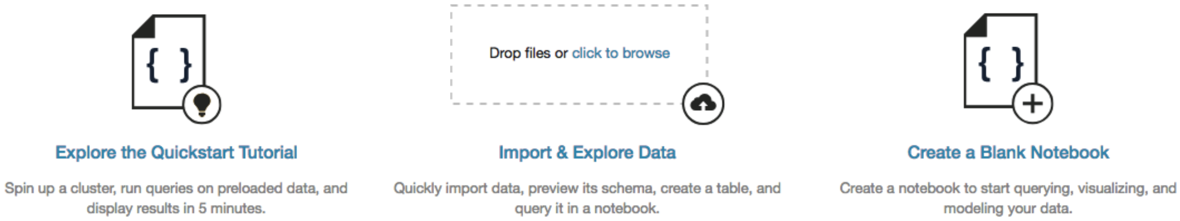
/FileStore/ shared_uploads/avesh.singh@databricks.com/

Files uploaded to DBFS are accessible by everyone who has access to this workspace. Learn more

Files

test.csv 20.5 MB

train.csv 32.5 MB



Drop files or click to browse

Explore the Quickstart Tutorial

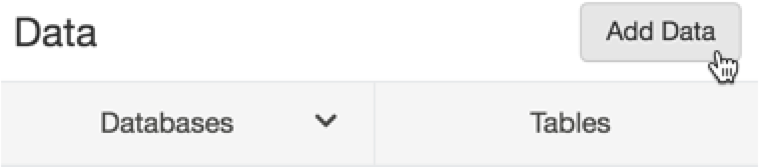
Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

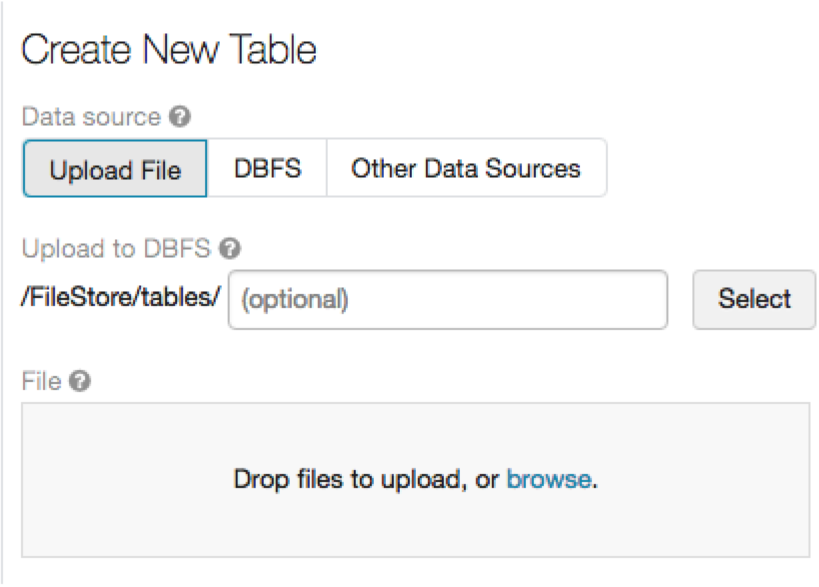
Import & Explore Data

Quickly import data, preview its schema, create a table, and query it in a notebook.

Create a Blank Notebook

Create a notebook to start querying, visualizing, and modeling your data.





Create New Table

Data source

Upload to DBFS

/FileStore/tables/ (optional)

File

Drop files to upload, or browse.

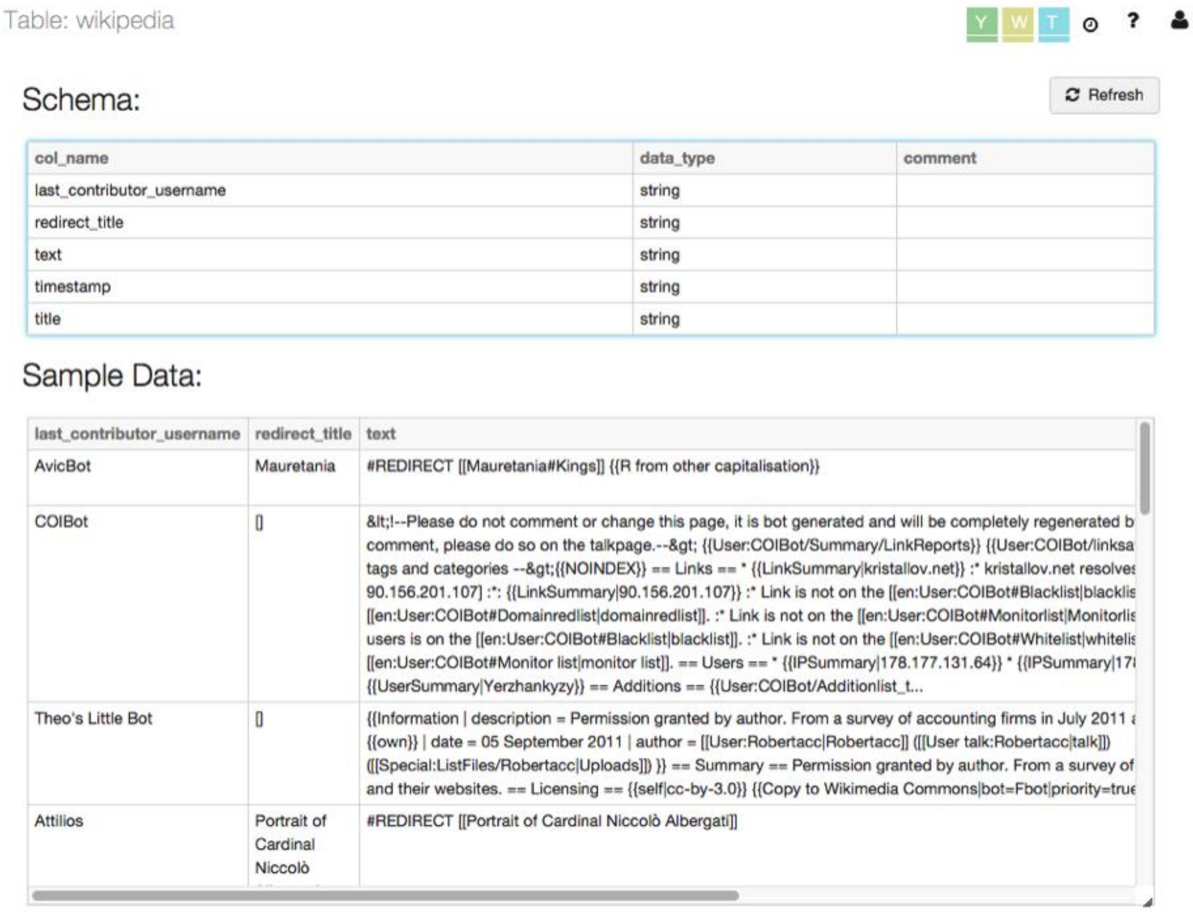


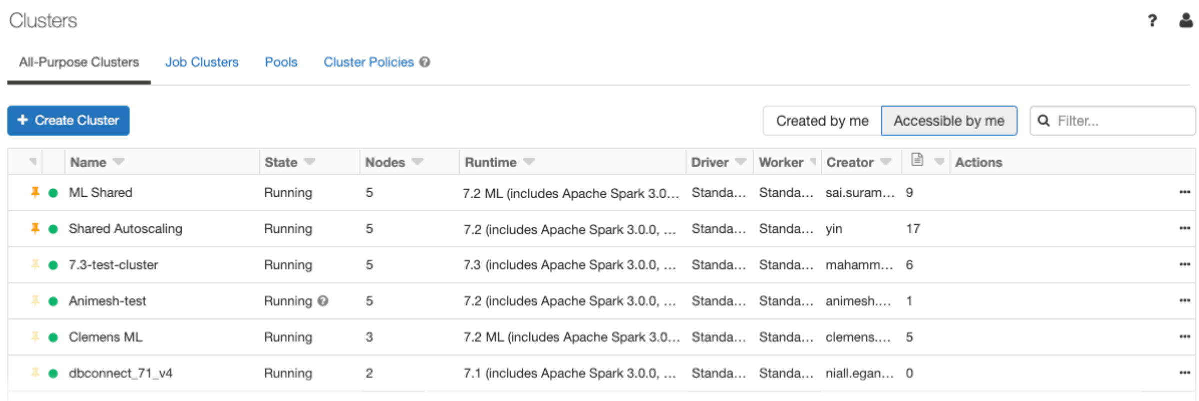
Table: wikipedia

Schema:

col_name	data_type	comment
last_contributor_username	string	
redirect_title	string	
text	string	
timestamp	string	
title	string	

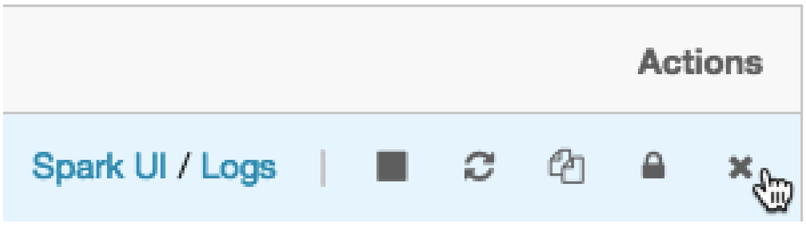
Sample Data:

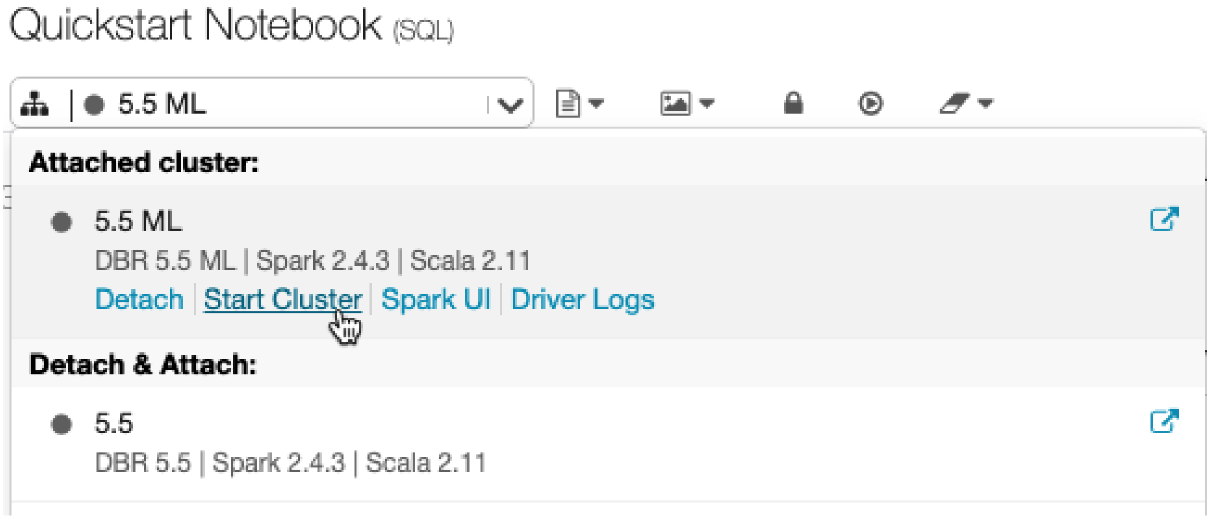
<table>
<tr><th>last_contributor_username</th><th>redirect_title</th><th>text</th></tr>
<tr><td>AvicBot</td><td>Mauretania</td><td>#REDIRECT [[Mauretania#Kings]] {{R from other capitalisation}}</td></tr>
<tr><td>COIBot</td><td>[]</td><td>&lt;!--Please do not comment or change this page, it is bot generated and will be completely regenerated b comment, please do so on the talkpage.--&gt; {{User:COIBot/Summary/LinkReports}} {{User:COIBot/linksa tags and categories --&gt;{{NOINDEX}} == Links == * {{LinkSummary|kristallov.net}} :* kristallov.net resolves 90.156.201.107] :*: {{LinkSummary|90.156.201.107}} :* Link is not on the [[en:User:COIBot#Blacklist|blacklis [[en:User:COIBot#Domainredlist|domainredlist]]. :* Link is not on the [[en:User:COIBot#Monitorlist|Monitorlis users is on the [[en:User:COIBot#Blacklist|blacklist]]. :* Link is not on the [[en:User:COIBot#Whitelist|whitelis [[en:User:COIBot#Monitor list|monitor list]]. == Users == * {{IPSummary|178.177.131.64}} * {{IPSummary|17 {{UserSummary|Yerzhankyzy}} == Additions == {{User:COIBot/Additionlist_t...</td></tr>
<tr><td>Theo's Little Bot</td><td>[]</td><td>{{Information | description = Permission granted by author. From a survey of accounting firms in July 2011 {{own}} | date = 05 September 2011 | author = [[User:Robertacc|Robertacc]] ([[User talk:Robertacc|talk]]) ([[Special:ListFiles/Robertacc|Uploads]]) }} == Summary == Permission granted by author. From a survey of and their websites. == Licensing == {{self|cc-by-3.0}} {{Copy to Wikimedia Commons|bot=Fbot|priority=true</td></tr>
<tr><td>Attilios</td><td>Portrait of Cardinal Niccolò</td><td>#REDIRECT [[Portrait of Cardinal Niccolò Albergati]]</td></tr>
</table>



Clusters

Name	State	Nodes	Runtime	Driver	Worker	Creator	🗎	Actions
● ML Shared	Running	5	7.2 ML (includes Apache Spark 3.0...	Standa...	Standa...	sai.suram...	9	•••
● Shared Autoscaling	Running	5	7.2 (includes Apache Spark 3.0.0, ...	Standa...	Standa...	yin	17	•••
● 7.3-test-cluster	Running	5	7.3 (includes Apache Spark 3.0.0, ...	Standa...	Standa...	mahamm...	6	•••
● Animesh-test	Running	5	7.2 (includes Apache Spark 3.0.0, ...	Standa...	Standa...	animesh....	1	•••
● Clemens ML	Running	3	7.2 ML (includes Apache Spark 3.0...	Standa...	Standa...	clemens....	5	•••
● dbconnect_71_v4	Running	2	7.1 (includes Apache Spark 3.0.0, ...	Standa...	Standa...	niall.egan...	0	•••





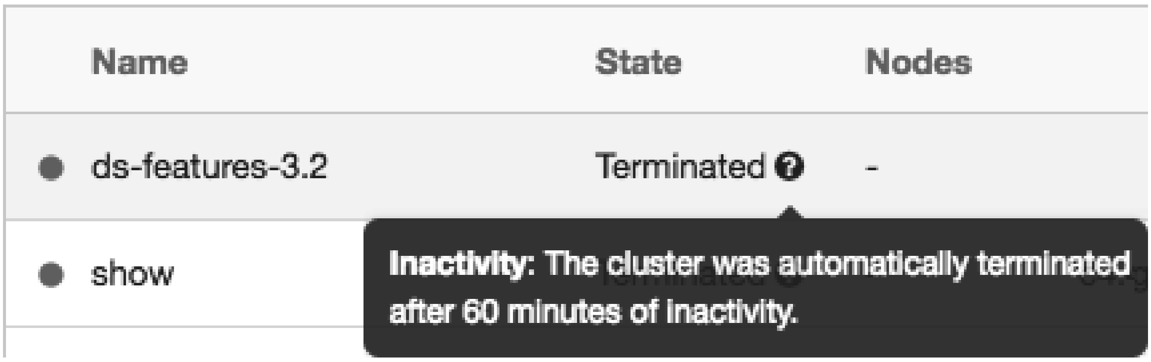
Quickstart Notebook (SQL)

Attached cluster:

● 5.5 ML
DBR 5.5 ML | Spark 2.4.3 | Scala 2.11
Detach | Start Cluster | Spark UI | Driver Logs

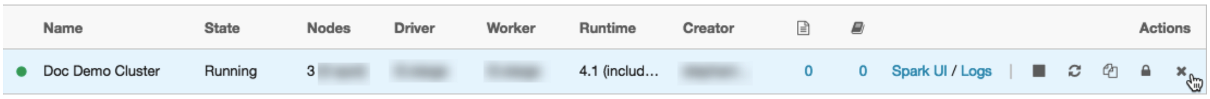
Detach & Attach:

● 5.5
DBR 5.5 | Spark 2.4.3 | Scala 2.11

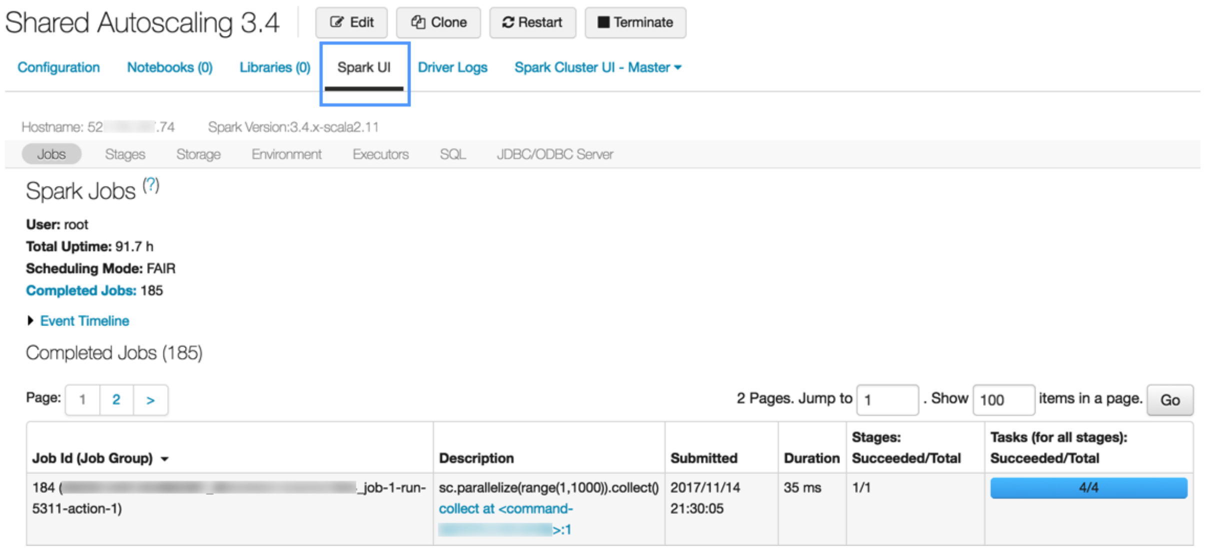


Name	State	Nodes
● ds-features-3.2	Terminated	-
● show		

Inactivity: The cluster was automatically terminated after 60 minutes of inactivity.



Name	State	Nodes	Driver	Worker	Runtime	Creator	🗎			Actions
● Doc Demo Cluster	Running	3 [redacted]	[redacted]	[redacted]	4.1 (includ...	[redacted]	0	0	Spark UI / Logs	



Hostname: 52[redacted].74 Spark Version:3.4.x-scala2.11

Spark Jobs (?)

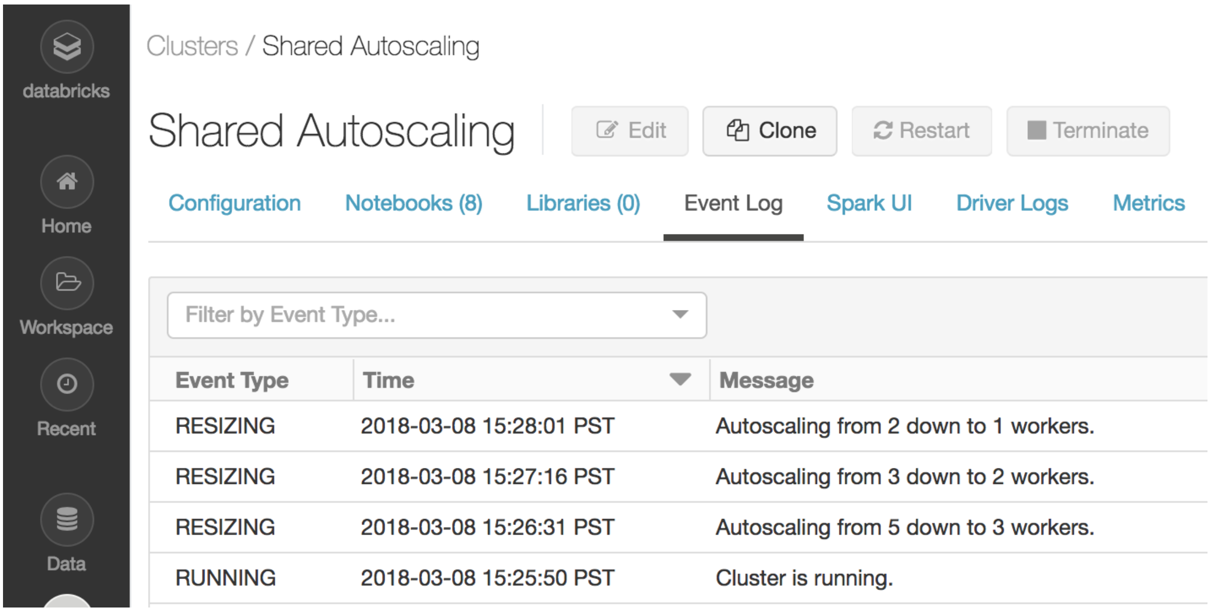
User: root
Total Uptime: 91.7 h
Scheduling Mode: FAIR
Completed Jobs: 185

Event Timeline

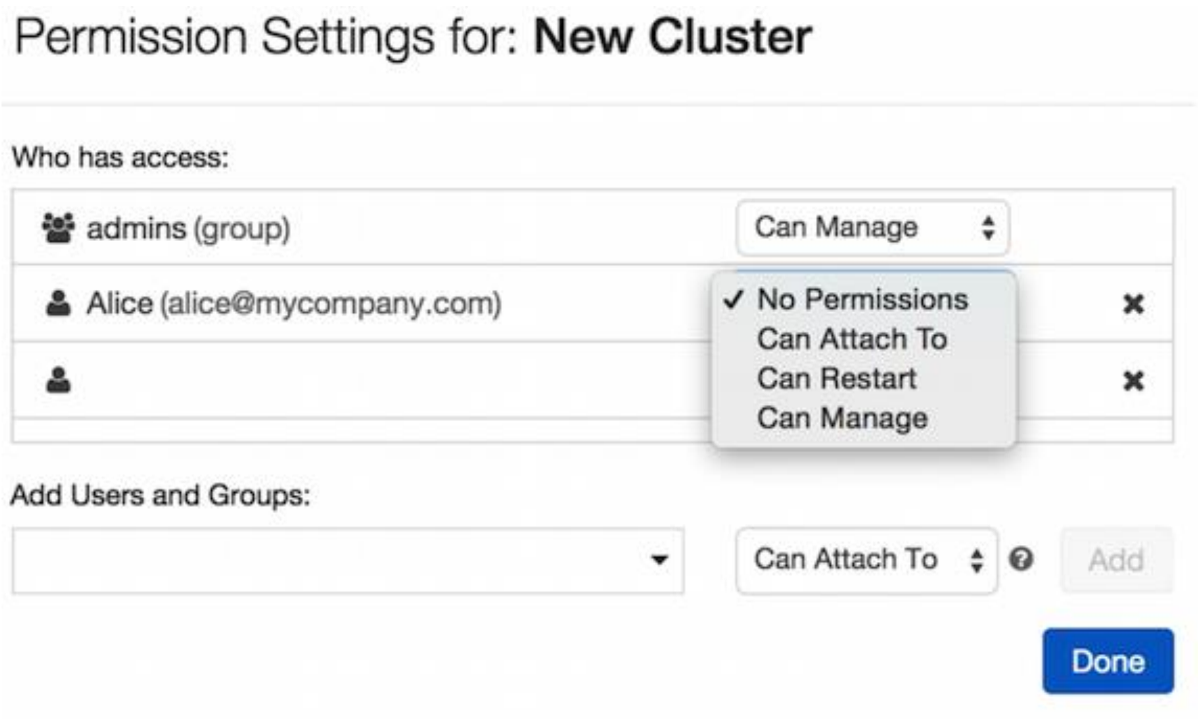
Completed Jobs (185)

Page: 1 2 > 2 Pages. Jump to 1 . Show 100 items in a page. Go

Job Id (Job Group)	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
184 ([redacted]_job-1-run-5311-action-1)	sc.parallelize(range(1,1000)).collect() collect at <command-[redacted]>:1	2017/11/14 21:30:05	35 ms	1/1	4/4



Event Type	Time	Message
RESIZING	2018-03-08 15:28:01 PST	Autoscaling from 2 down to 1 workers.
RESIZING	2018-03-08 15:27:16 PST	Autoscaling from 3 down to 2 workers.
RESIZING	2018-03-08 15:26:31 PST	Autoscaling from 5 down to 3 workers.
RUNNING	2018-03-08 15:25:50 PST	Cluster is running.



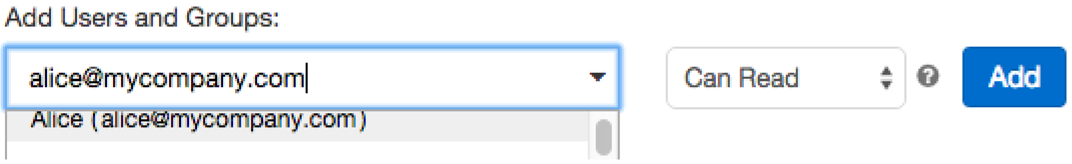
Permission Settings for: **New Cluster**

Who has access:

- admins (group): Can Manage
- Alice (alice@mycompany.com)

Add Users and Groups:

Can Attach To

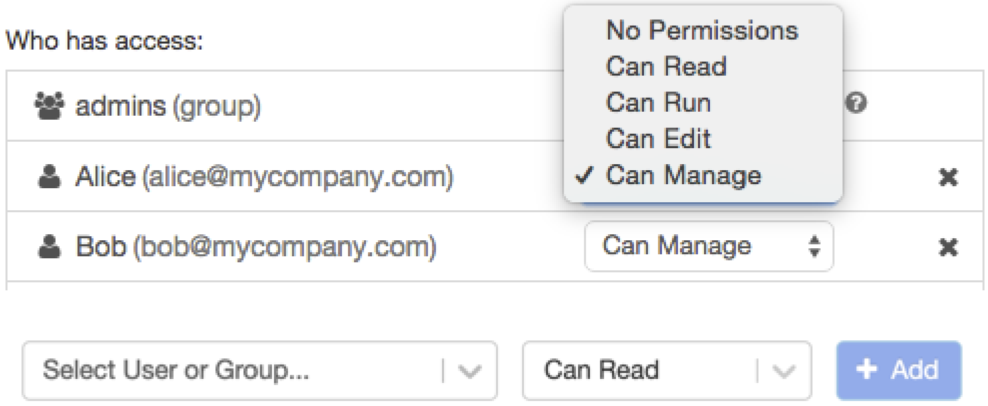


Add Users and Groups:

alice@mycompany.com

Alice (alice@mycompany.com)

Can Read

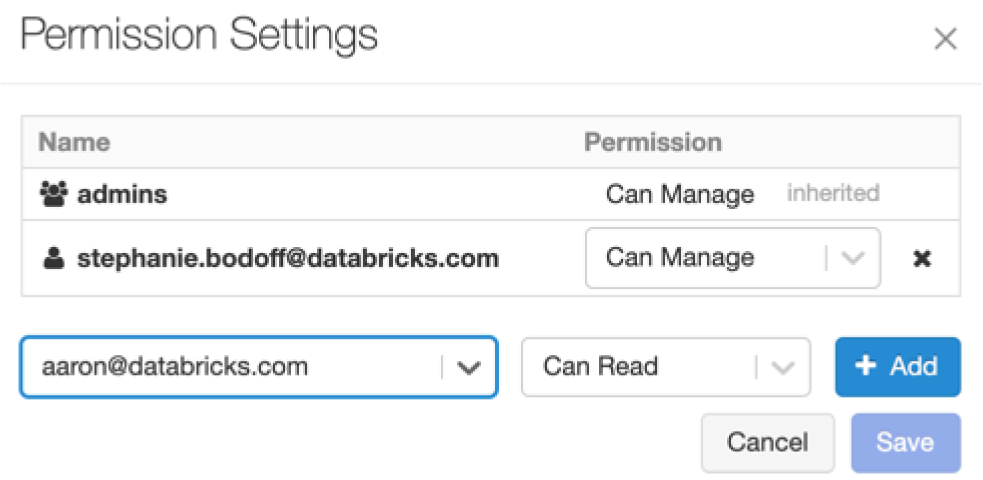


Who has access:

- admins (group)
- Alice (alice@mycompany.com)
- Bob (bob@mycompany.com): Can Manage

Select User or Group...

Can Read



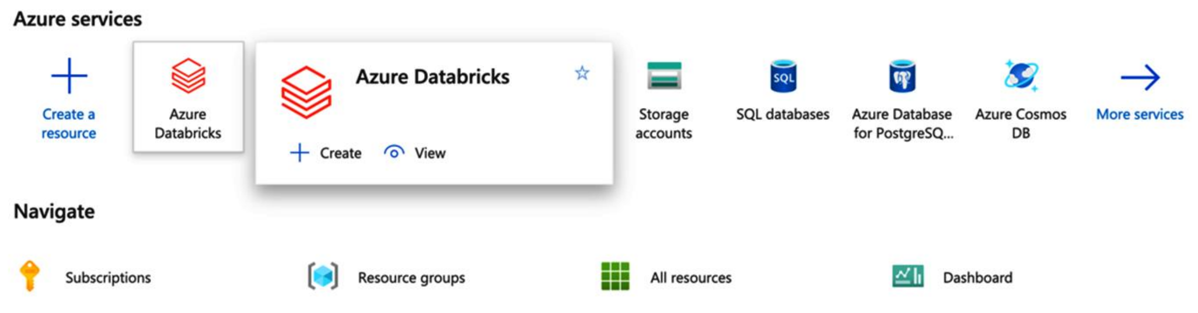
Permission Settings

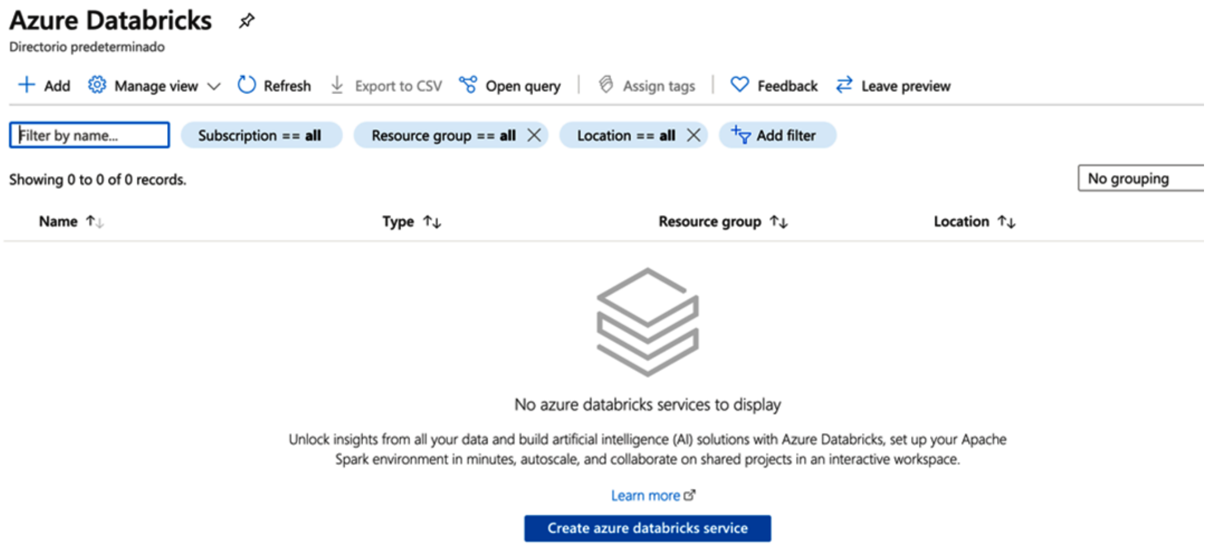
Name	Permission	
admins	Can Manage	inherited
stephanie.bodoff@databricks.com	Can Manage	✖

aaron@databricks.com

Can Read

Chapter 2: Creating an Azure Databricks Workspace



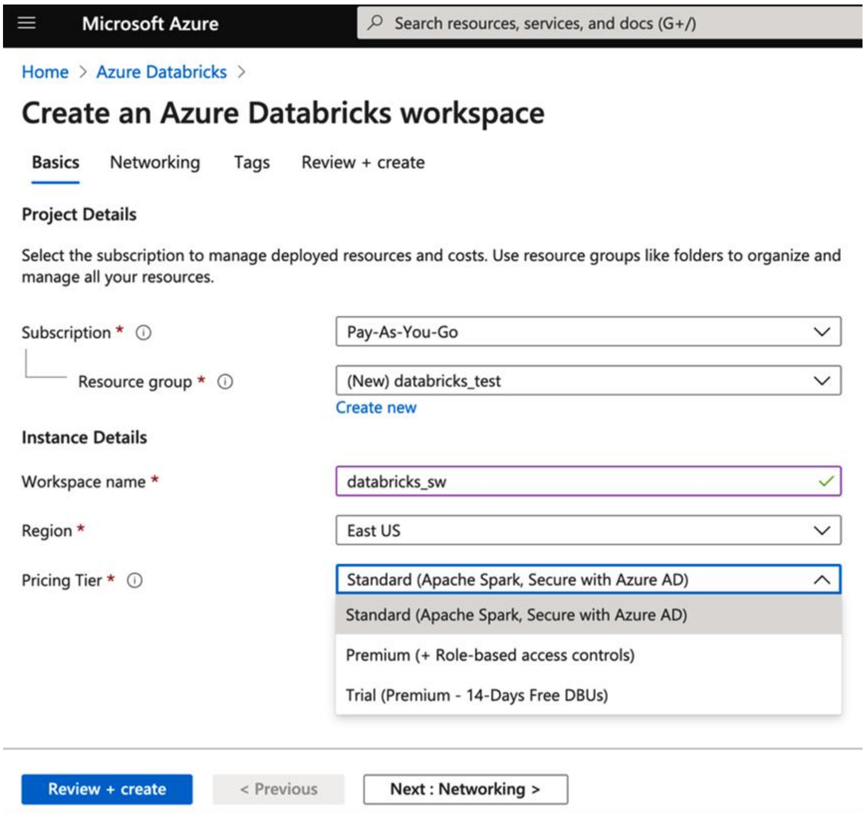


Showing 0 to 0 of 0 records.

Name	Type	Resource group	Location

No azure databricks services to display

Unlock insights from all your data and build artificial intelligence (AI) solutions with Azure Databricks, set up your Apache Spark environment in minutes, autoscale, and collaborate on shared projects in an interactive workspace.



Create an Azure Databricks workspace

Project Details

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

Subscription * : Pay-As-You-Go

Resource group * : (New) databricks_test

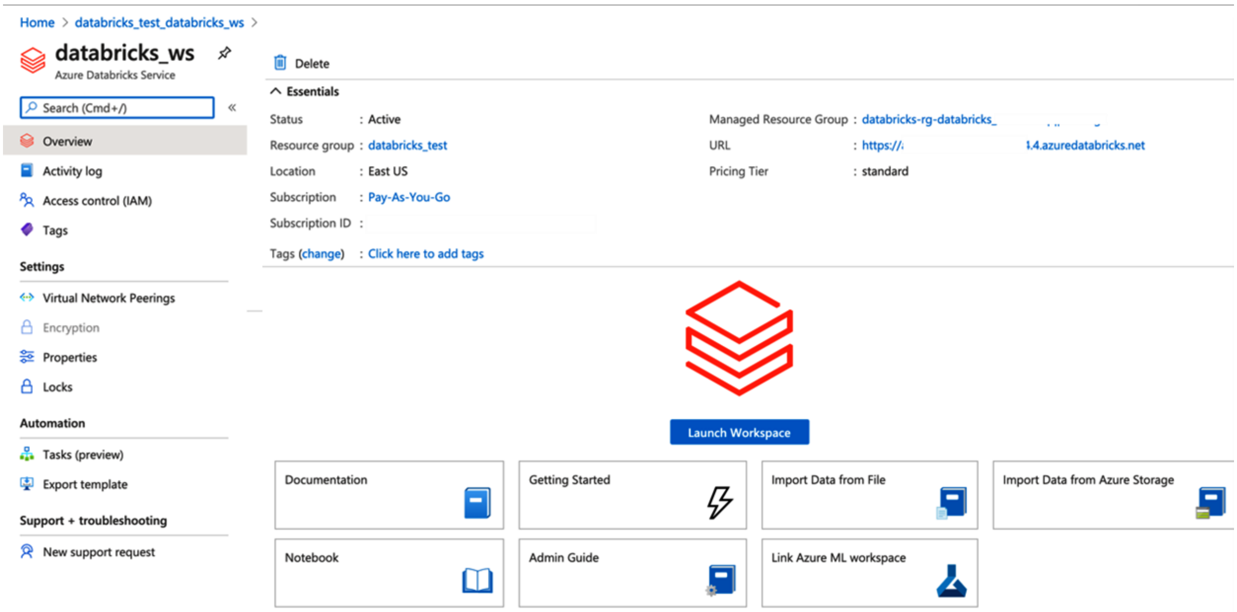
Instance Details

Workspace name * : databricks_sw

Region * : East US

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

- Standard (Apache Spark, Secure with Azure AD)
- Premium (+ Role-based access controls)
- Trial (Premium - 14-Days Free DBUs)



Home > databricks_test_databricks_ws >

databricks_ws

Azure Databricks Service

Search (Cmd+/)

- Overview
- Activity log
- Access control (IAM)
- Tags

Settings

- Virtual Network Peerings
- Encryption
- Properties
- Locks

Automation

- Tasks (preview)
- Export template

Support + troubleshooting

- New support request

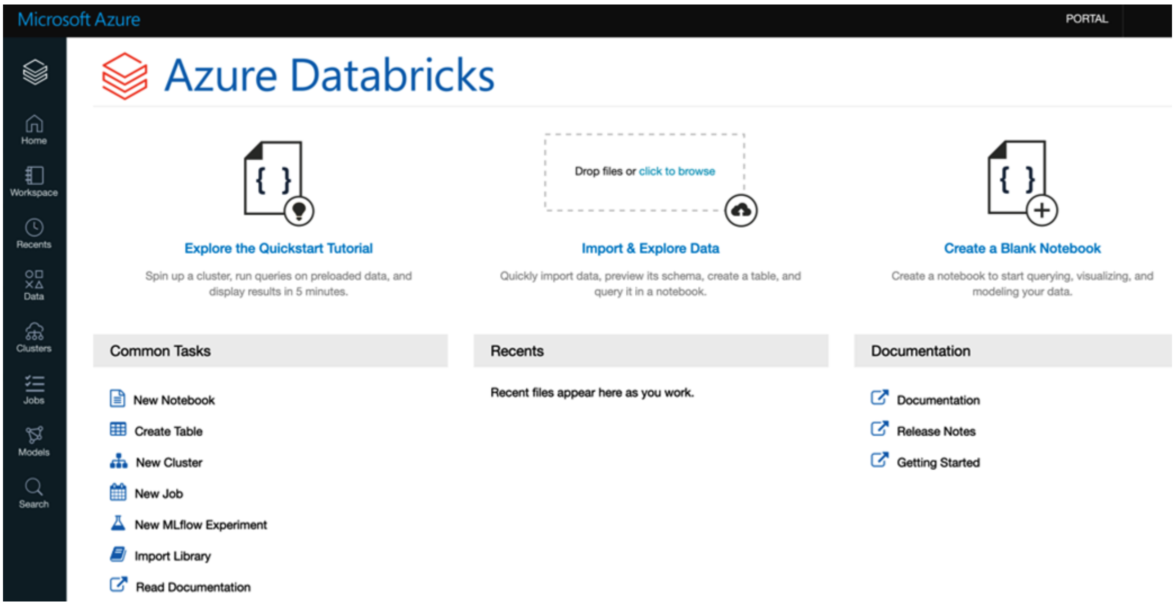
Delete

Essentials

Status	: Active	Managed Resource Group	: databricks-rg-databricks_
Resource group	: databricks_test	URL	: https:// .4.azuredatabricks.net
Location	: East US	Pricing Tier	: standard
Subscription	: Pay-As-You-Go		
Subscription ID	:		
Tags (change)	: Click here to add tags		

Launch Workspace

Documentation	Getting Started	Import Data from File	Import Data from Azure Storage
Notebook	Admin Guide	Link Azure ML workspace	



Microsoft Azure PORTAL

Azure Databricks

Drop files or click to browse

Explore the Quickstart Tutorial

Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

Import & Explore Data

Quickly import data, preview its schema, create a table, and query it in a notebook.

Create a Blank Notebook

Create a notebook to start querying, visualizing, and modeling your data.

Common Tasks

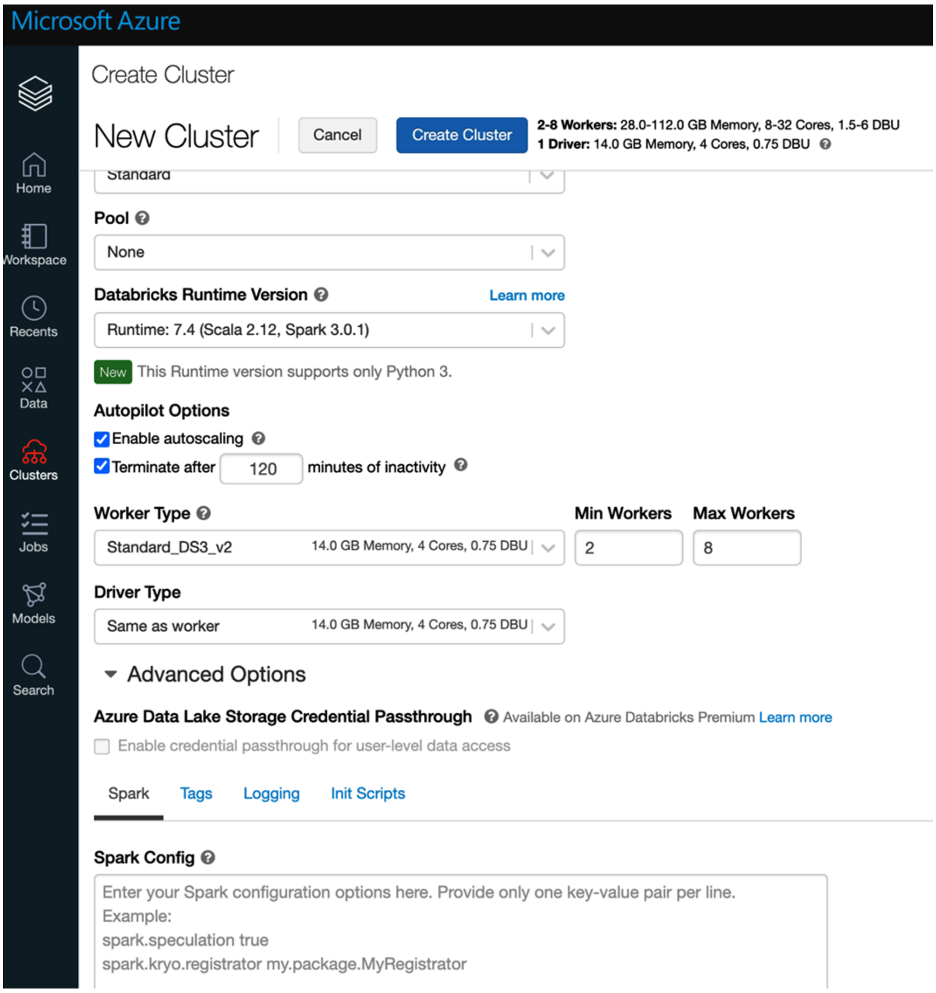
- New Notebook
- Create Table
- New Cluster
- New Job
- New MLflow Experiment
- Import Library
- Read Documentation

Recents

Recent files appear here as you work.

Documentation

- Documentation
- Release Notes
- Getting Started



Create Cluster

New Cluster   Cancel   Create Cluster

2-8 Workers: 28.0-112.0 GB Memory, 8-32 Cores, 1.5-6 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU

Standard

Pool
None

Databricks Runtime Version   Learn more
Runtime: 7.4 (Scala 2.12, Spark 3.0.1)

New This Runtime version supports only Python 3.

Autopilot Options
☑ Enable autoscaling
☑ Terminate after 120 minutes of inactivity

Worker Type
Standard_DS3_v2   14.0 GB Memory, 4 Cores, 0.75 DBU

Min Workers 2
Max Workers 8

Driver Type
Same as worker   14.0 GB Memory, 4 Cores, 0.75 DBU

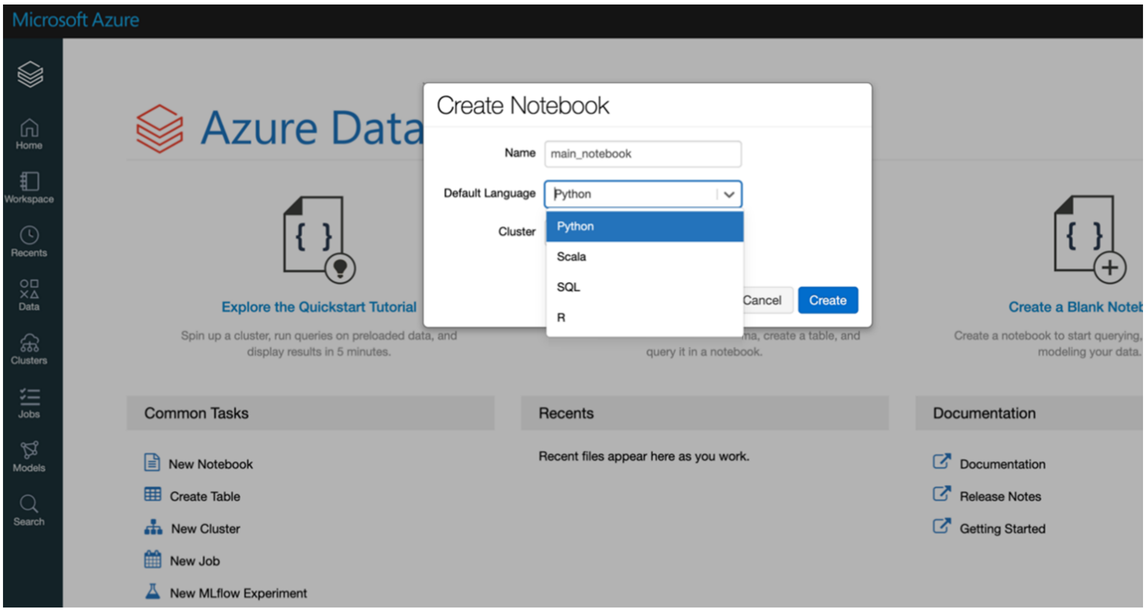
Advanced Options

Azure Data Lake Storage Credential Passthrough Available on Azure Databricks Premium Learn more
□ Enable credential passthrough for user-level data access

Spark   Tags   Logging   Init Scripts

Spark Config

Enter your Spark configuration options here. Provide only one key-value pair per line.
Example:
spark.speculation true
spark.kryo.registrator my.package.MyRegistrator



Create Notebook

Name: main_notebook
Default Language: Python
Cluster

Python
Scala
SQL
R

Cancel   Create

Explore the Quickstart Tutorial
Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

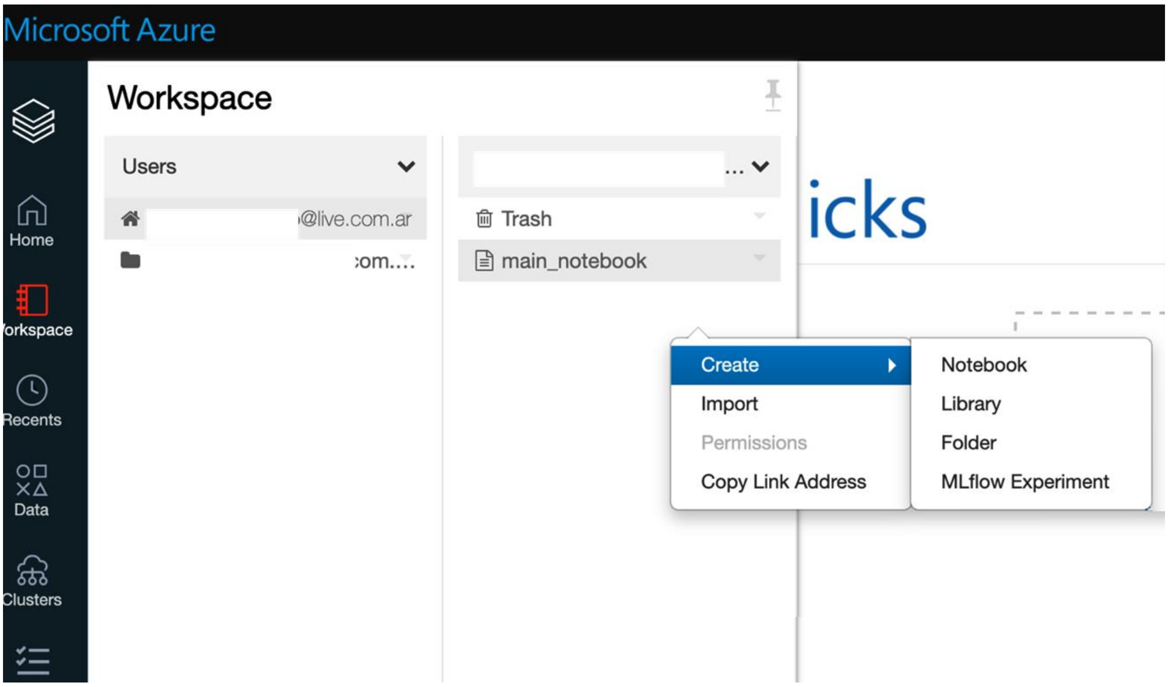
ma, create a table, and query it in a notebook.

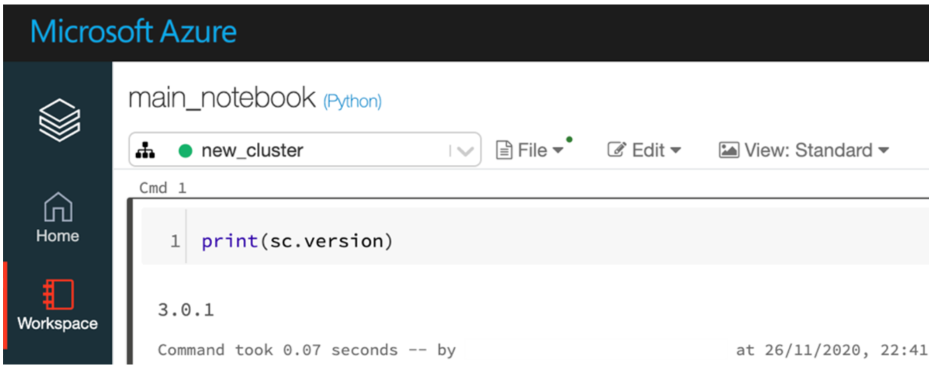
Create a Blank Noteb
Create a notebook to start querying, modeling your data.

Common Tasks
- New Notebook
- Create Table
- New Cluster
- New Job
- New MLflow Experiment

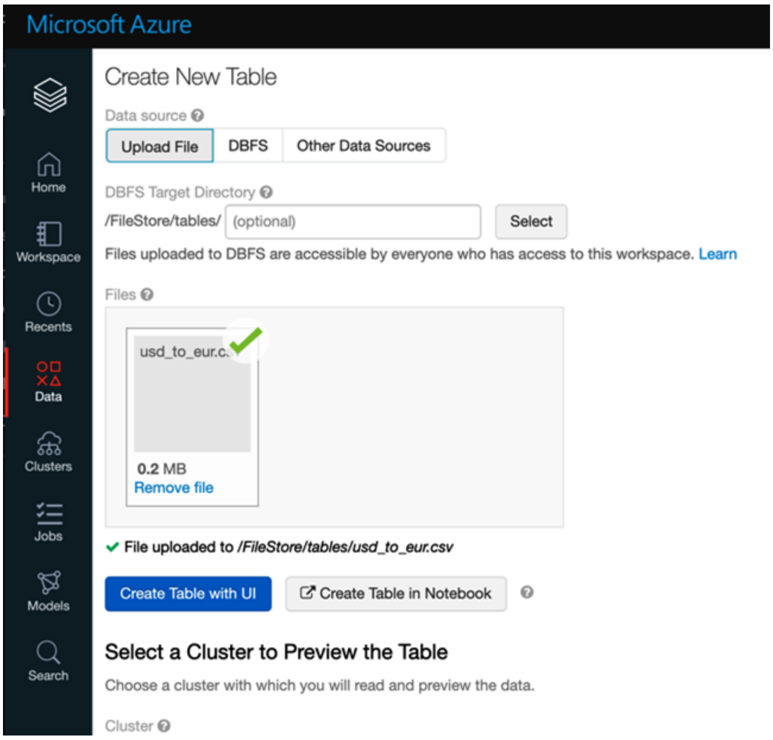
Recents
Recent files appear here as you work.

Documentation
- Documentation
- Release Notes
- Getting Started





```
Cmd 1

print(sc.version)

3.0.1

Command took 0.07 seconds -- by                                   at 26/11/2020, 22:41
```



Create New Table

Data source

DBFS Target Directory

/FileStore/tables/ (optional)

Files uploaded to DBFS are accessible by everyone who has access to this workspace. Learn

Files

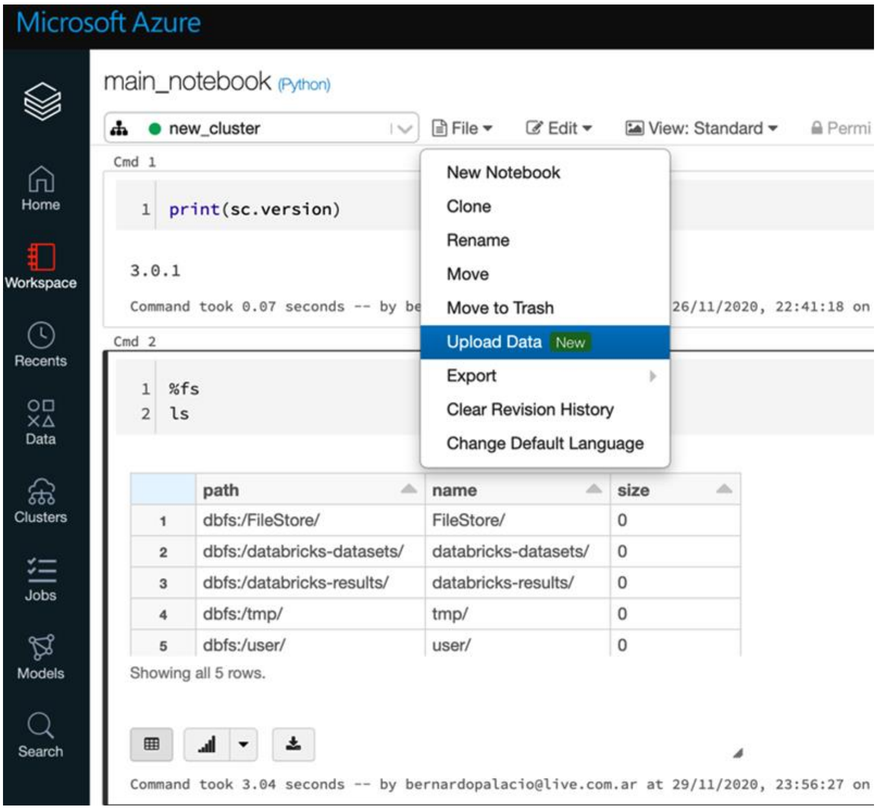
0.2 MB

✓ File uploaded to */FileStore/tables/usd_to_eur.csv*

Select a Cluster to Preview the Table

Choose a cluster with which you will read and preview the data.

Cluster



Cmd 1

```
print(sc.version)
```

```
3.0.1

Command took 0.07 seconds -- by be
26/11/2020, 22:41:18 on
```

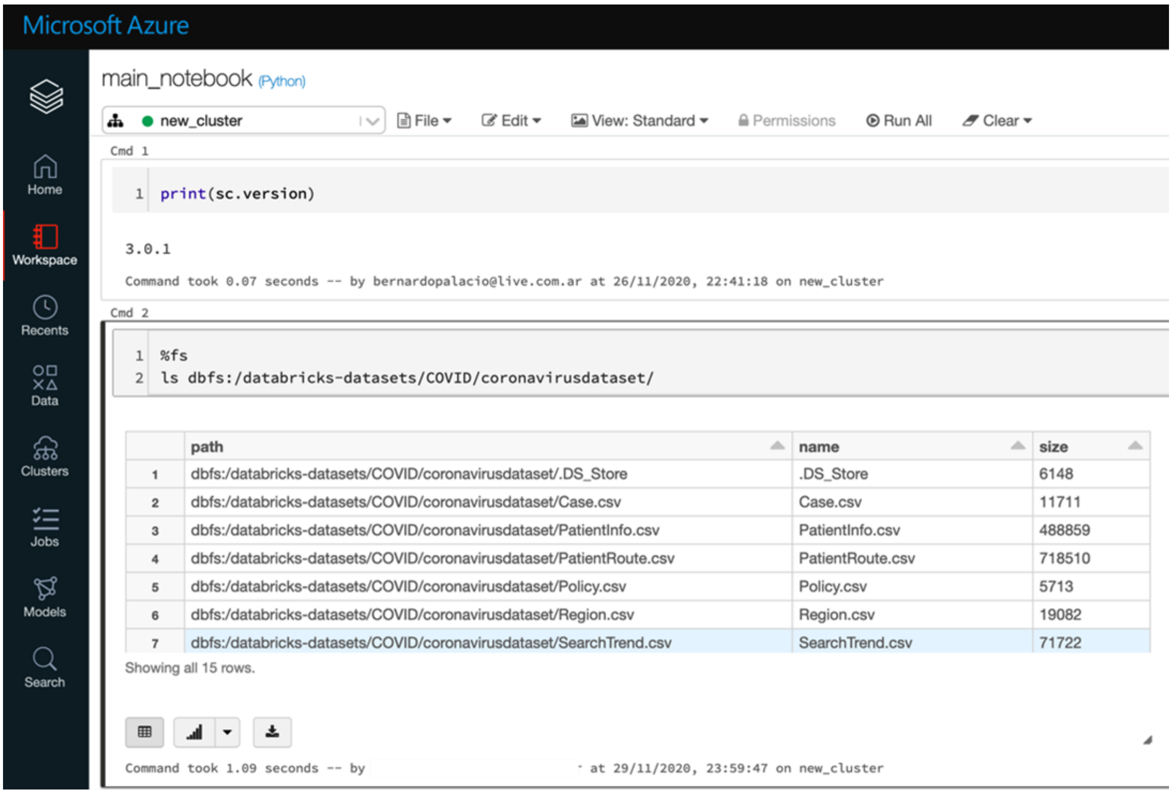
Cmd 2

```
%fs
ls
```

	path	name	size
1	dbfs:/FileStore/	FileStore/	0
2	dbfs:/databricks-datasets/	databricks-datasets/	0
3	dbfs:/databricks-results/	databricks-results/	0
4	dbfs:/tmp/	tmp/	0
5	dbfs:/user/	user/	0

Showing all 5 rows.

```
Command took 3.04 seconds -- by bernardopalacio@live.com.ar at 29/11/2020, 23:56:27 on
```



Cmd 1

```
print(sc.version)
```

```
3.0.1

Command took 0.07 seconds -- by bernardopalacio@live.com.ar at 26/11/2020, 22:41:18 on new_cluster
```

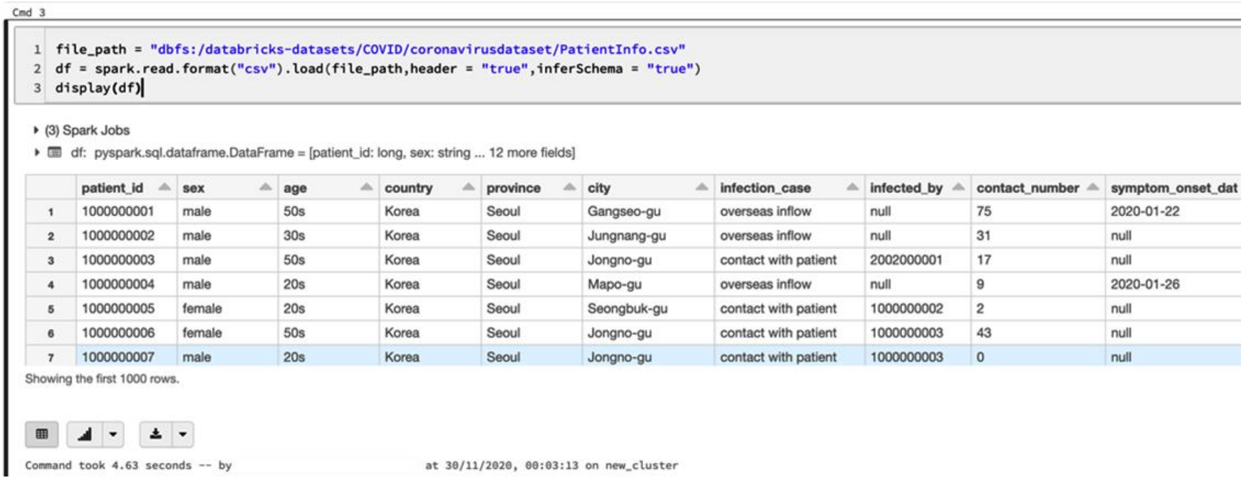
Cmd 2

```
%fs
ls dbfs:/databricks-datasets/COVID/coronavirusdataset/
```

	path	name	size
1	dbfs:/databricks-datasets/COVID/coronavirusdataset/.DS_Store	.DS_Store	6148
2	dbfs:/databricks-datasets/COVID/coronavirusdataset/Case.csv	Case.csv	11711
3	dbfs:/databricks-datasets/COVID/coronavirusdataset/PatientInfo.csv	PatientInfo.csv	488859
4	dbfs:/databricks-datasets/COVID/coronavirusdataset/PatientRoute.csv	PatientRoute.csv	718510
5	dbfs:/databricks-datasets/COVID/coronavirusdataset/Policy.csv	Policy.csv	5713
6	dbfs:/databricks-datasets/COVID/coronavirusdataset/Region.csv	Region.csv	19082
7	dbfs:/databricks-datasets/COVID/coronavirusdataset/SearchTrend.csv	SearchTrend.csv	71722

Showing all 15 rows.

```
Command took 1.09 seconds -- by                     · at 29/11/2020, 23:59:47 on new_cluster
```



Cmd 3

```
file_path = "dbfs:/databricks-datasets/COVID/coronavirusdataset/PatientInfo.csv"
df = spark.read.format("csv").load(file_path,header = "true",inferSchema = "true")
display(df)
```

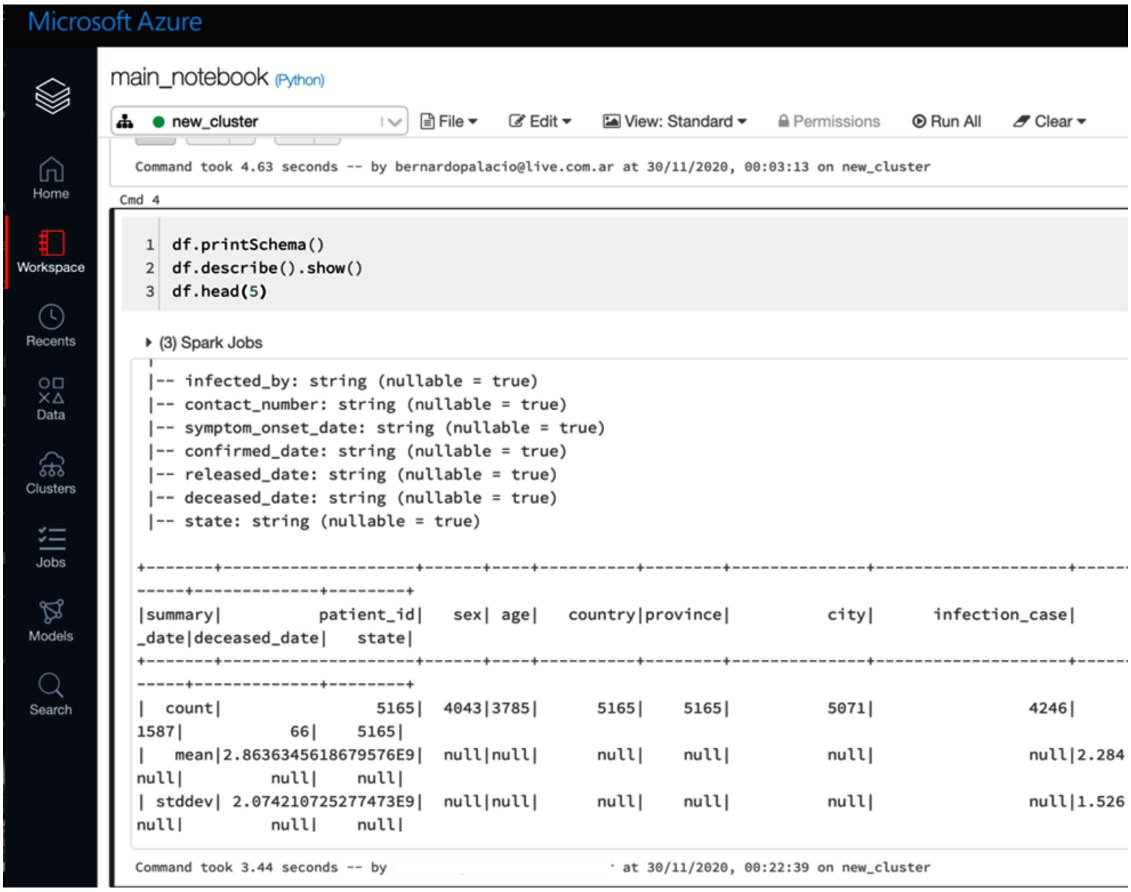
▸ (3) Spark Jobs

▸ df: pyspark.sql.dataframe.DataFrame = [patient_id: long, sex: string ... 12 more fields]

	patient_id	sex	age	country	province	city	infection_case	infected_by	contact_number	symptom_onset_dat
1	1000000001	male	50s	Korea	Seoul	Gangseo-gu	overseas inflow	null	75	2020-01-22
2	1000000002	male	30s	Korea	Seoul	Jungnang-gu	overseas inflow	null	31	null
3	1000000003	male	50s	Korea	Seoul	Jongno-gu	contact with patient	2002000001	17	null
4	1000000004	male	20s	Korea	Seoul	Mapo-gu	overseas inflow	null	9	2020-01-26
5	1000000005	female	20s	Korea	Seoul	Seongbuk-gu	contact with patient	1000000002	2	null
6	1000000006	female	50s	Korea	Seoul	Jongno-gu	contact with patient	1000000003	43	null
7	1000000007	male	20s	Korea	Seoul	Jongno-gu	contact with patient	1000000003	0	null

Showing the first 1000 rows.

Command took 4.63 seconds -- by at 30/11/2020, 00:03:13 on new_cluster



Command took 4.63 seconds -- by bernardopalacio@live.com.ar at 30/11/2020, 00:03:13 on new_cluster

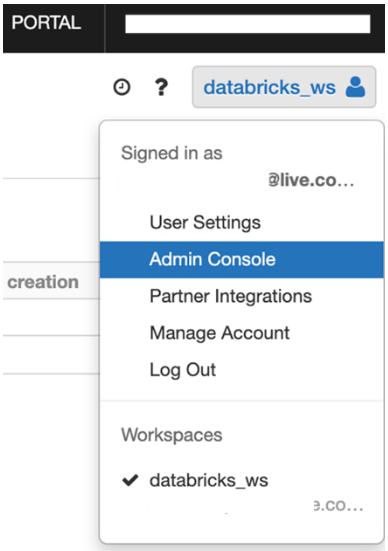
Cmd 4

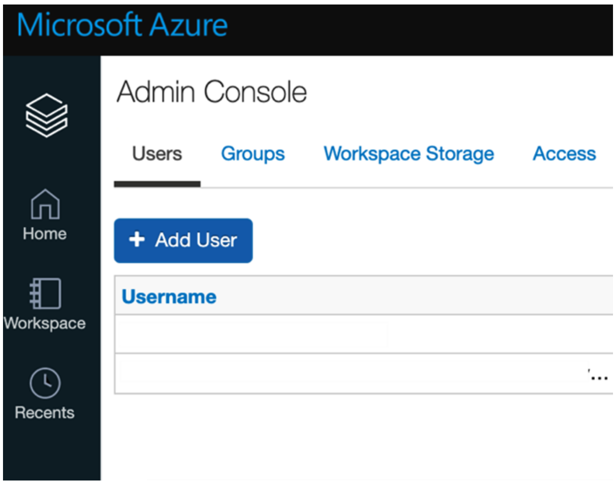
```
df.printSchema()
df.describe().show()
df.head(5)
```

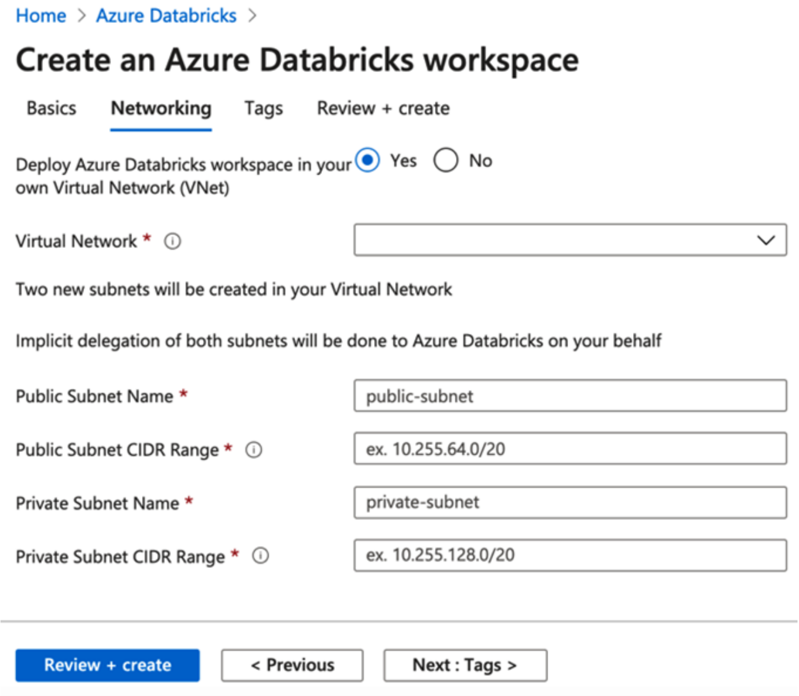
▸ (3) Spark Jobs

```
 |-- infected_by: string (nullable = true)
 |-- contact_number: string (nullable = true)
 |-- symptom_onset_date: string (nullable = true)
 |-- confirmed_date: string (nullable = true)
 |-- released_date: string (nullable = true)
 |-- deceased_date: string (nullable = true)
 |-- state: string (nullable = true)

+-------+--------------------+------+----+----------+--------+--------------+--------------------+-----
-----+-------------+--------+
|summary|          patient_id|   sex| age|   country|province|          city|      infection_case|
_date|deceased_date|   state|
+-------+--------------------+------+----+----------+--------+--------------+--------------------+-----
-----+-------------+--------+
|  count|                5165|  4043|3785|      5165|    5165|          5071|                4246|
1587|           66|    5165|
|   mean|2.8636345618679576E9|  null|null|      null|    null|          null|                null|2.284
null|         null|    null|
| stddev| 2.074210725277473E9|  null|null|      null|    null|          null|                null|1.526
null|         null|    null|
```

Command took 3.44 seconds -- by at 30/11/2020, 00:22:39 on new_cluster






Home > Azure Databricks >

Create an Azure Databricks workspace

Basics | Networking | Tags | Review + create

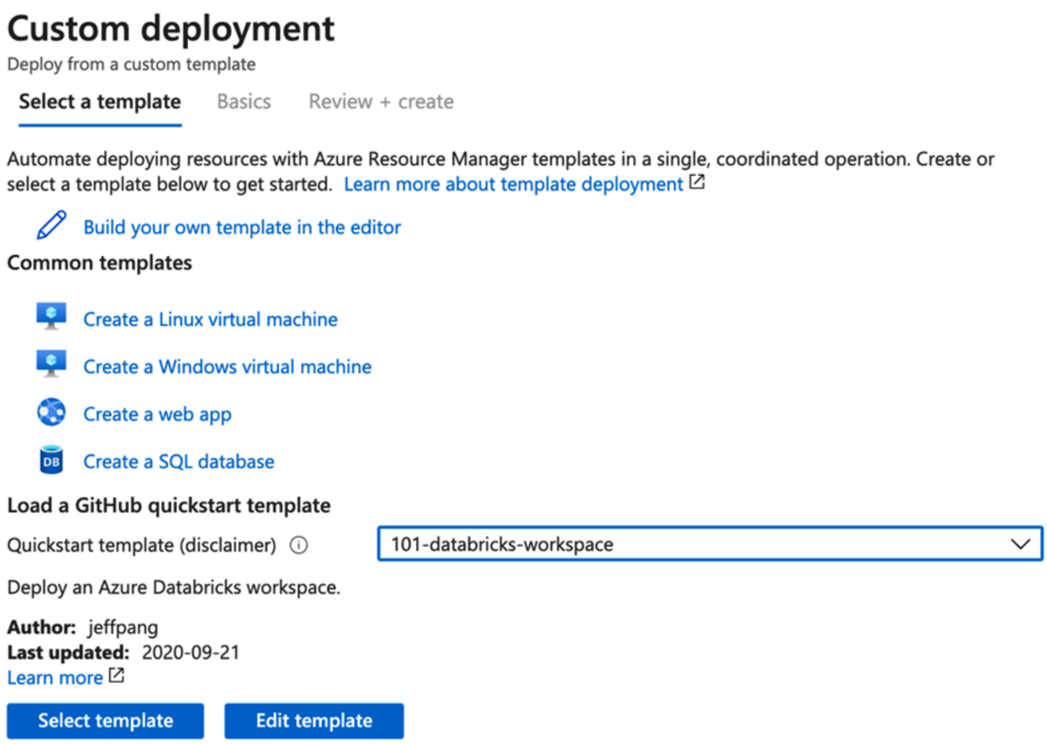
Deploy Azure Databricks workspace in your own Virtual Network (VNet) ● Yes ○ No

Virtual Network * ⓘ	

Two new subnets will be created in your Virtual Network

Implicit delegation of both subnets will be done to Azure Databricks on your behalf

Public Subnet Name *	public-subnet
Public Subnet CIDR Range * ⓘ	ex. 10.255.64.0/20
Private Subnet Name *	private-subnet
Private Subnet CIDR Range * ⓘ	ex. 10.255.128.0/20



Custom deployment

Deploy from a custom template

Select a template
Basics
Review + create

Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Create or select a template below to get started. Learn more about template deployment

Build your own template in the editor

Common templates

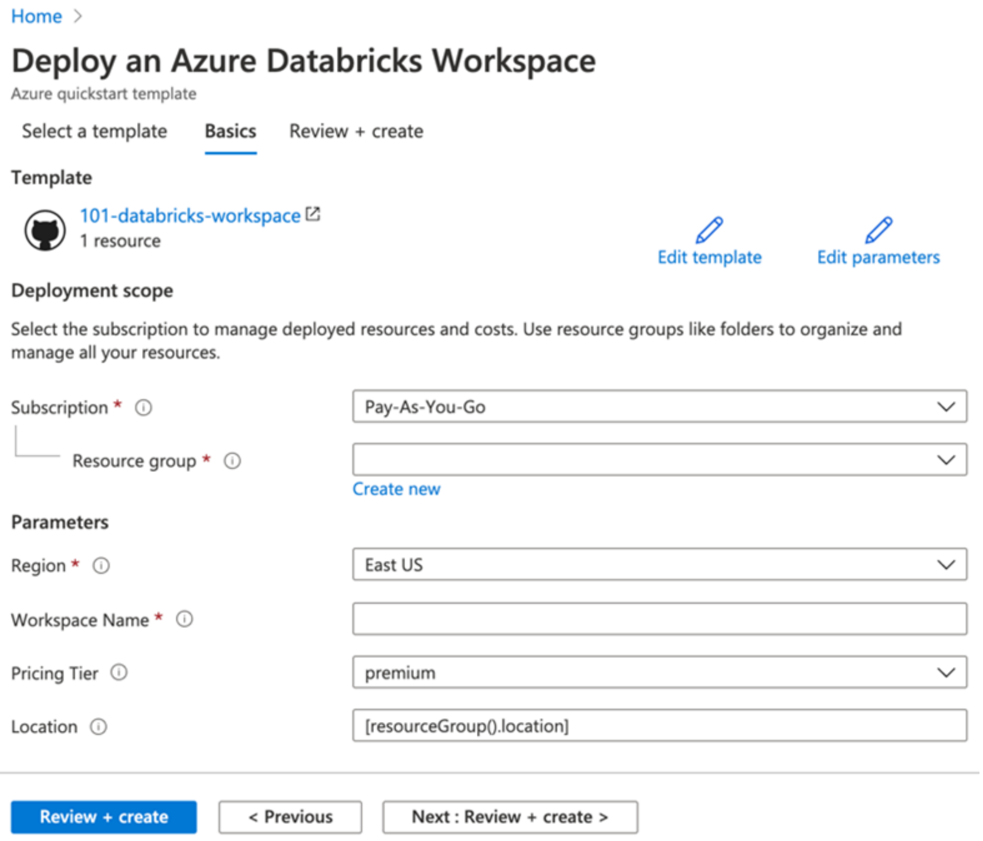
Create a Linux virtual machine
Create a Windows virtual machine
Create a web app
Create a SQL database

Load a GitHub quickstart template

Quickstart template (disclaimer): 101-databricks-workspace

Deploy an Azure Databricks workspace.

Author: jeffpang
Last updated: 2020-09-21
Learn more



Deploy an Azure Databricks Workspace

Azure quickstart template

Template

101-databricks-workspace
1 resource

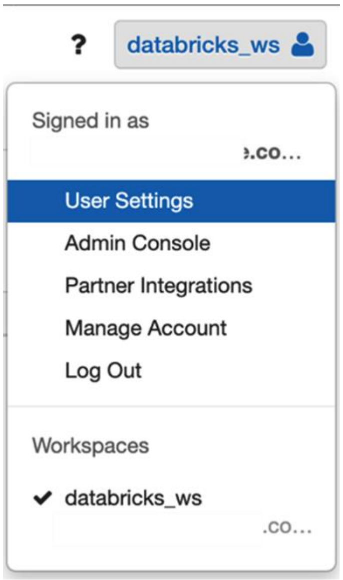
Deployment scope

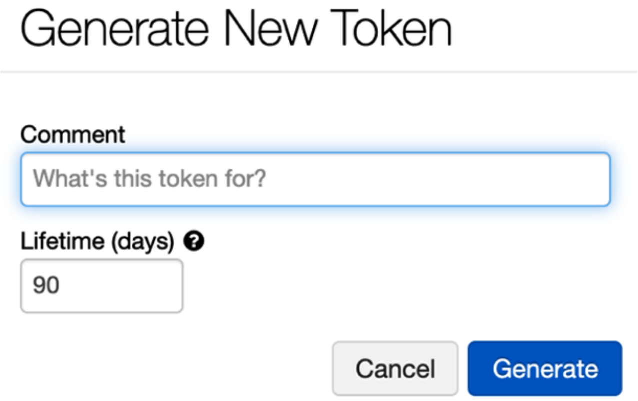
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *: Pay-As-You-Go
Resource group *:
Create new

Parameters

Region *: East US
Workspace Name *:
Pricing Tier: premium
Location: [resourceGroup().location]





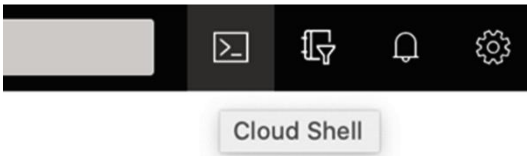
Generate New Token

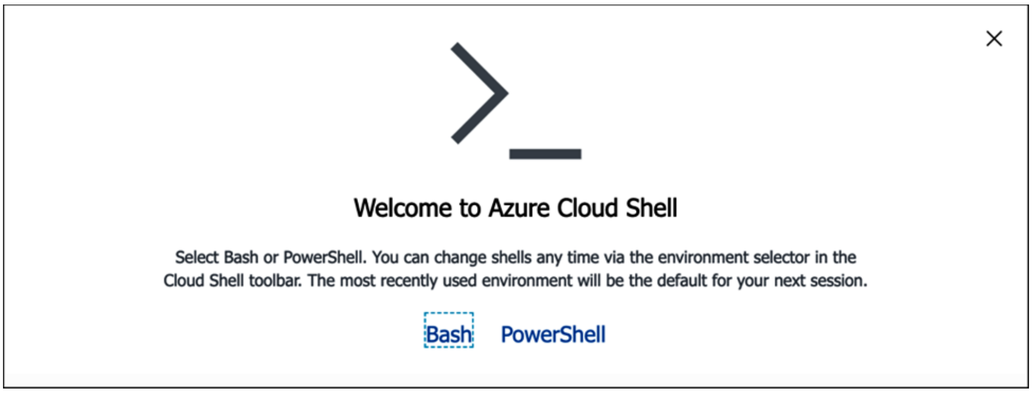
Comment

What's this token for?

Lifetime (days)

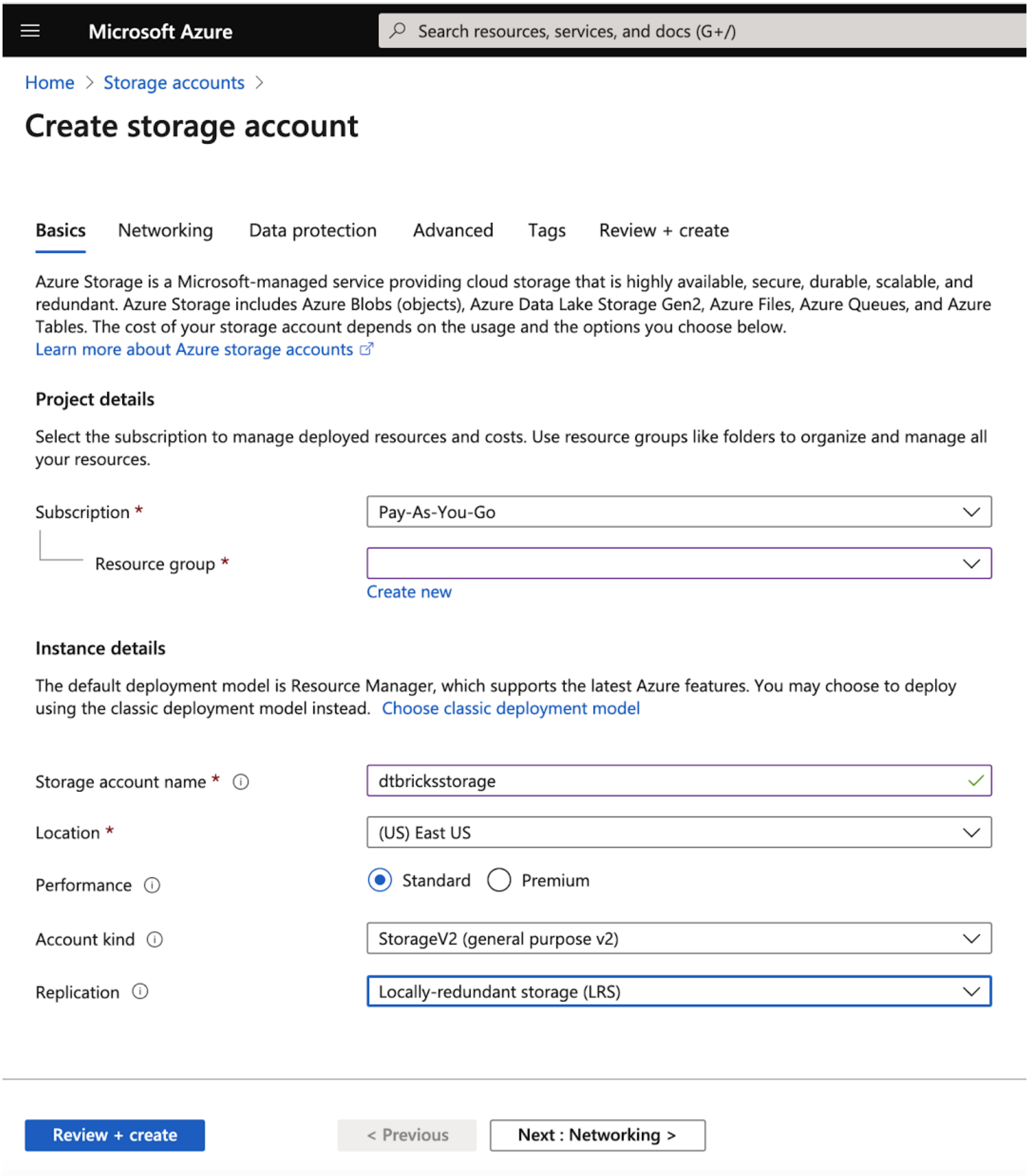
90





Welcome to Azure Cloud Shell

Select Bash or PowerShell. You can change shells any time via the environment selector in the Cloud Shell toolbar. The most recently used environment will be the default for your next session.



Create storage account

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below.
Learn more about Azure storage accounts

Project details

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

Subscription * : Pay-As-You-Go

Resource group * :

Instance details

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead.

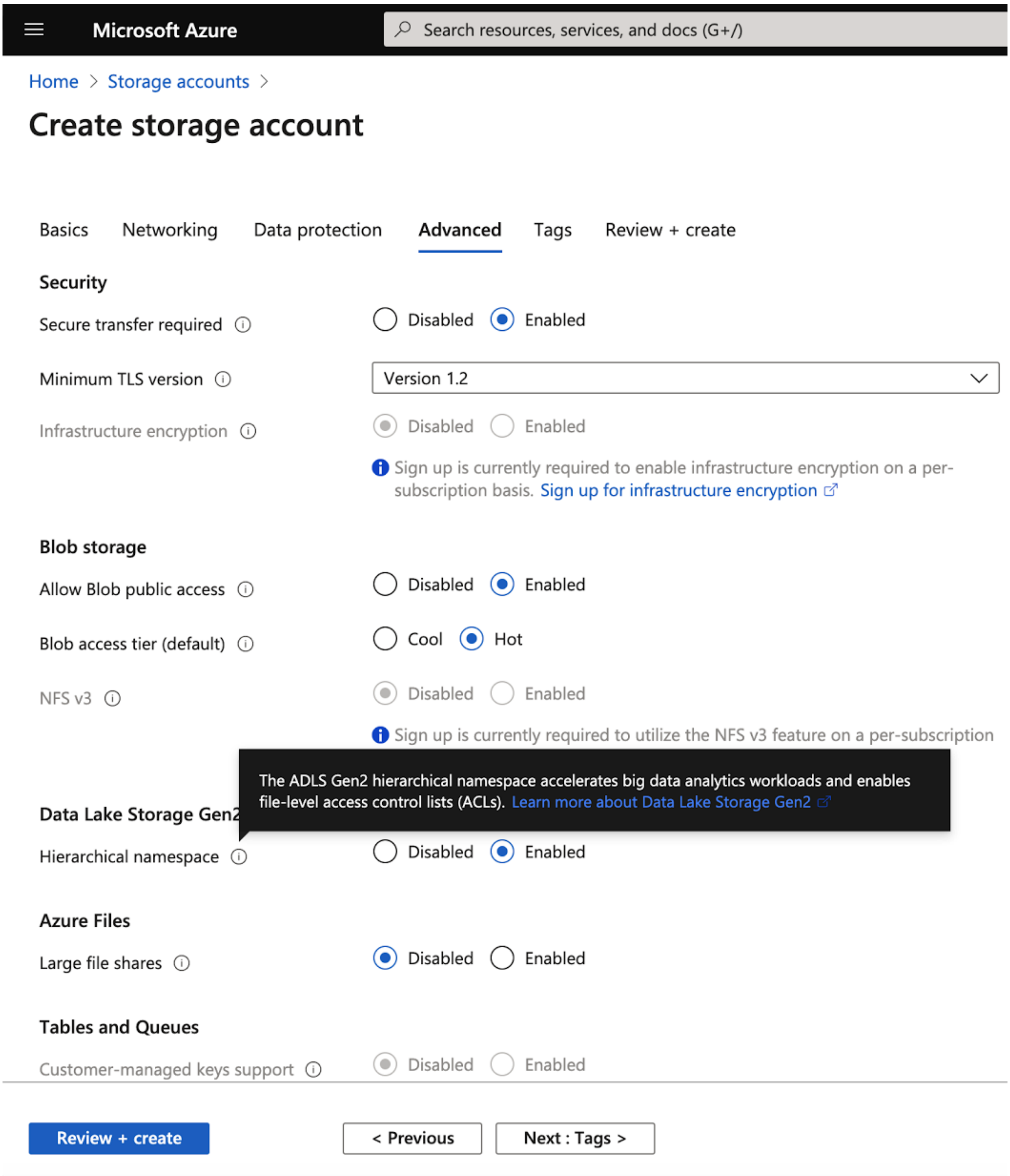
Storage account name * : dtbricksstorage ✓

Location * : (US) East US

Performance: ● Standard ○ Premium

Account kind: StorageV2 (general purpose v2)

Replication: Locally-redundant storage (LRS)



Home > Storage accounts >

Create storage account

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

Security

Secure transfer required ⓘ ○ Disabled ● Enabled

Minimum TLS version ⓘ Version 1.2

Infrastructure encryption ⓘ ● Disabled ○ Enabled

ℹ Sign up is currently required to enable infrastructure encryption on a per-subscription basis. Sign up for infrastructure encryption

Blob storage

Allow Blob public access ⓘ ○ Disabled ● Enabled

Blob access tier (default) ⓘ ○ Cool ● Hot

NFS v3 ⓘ ● Disabled ○ Enabled

ℹ Sign up is currently required to utilize the NFS v3 feature on a per-subscription

The ADLS Gen2 hierarchical namespace accelerates big data analytics workloads and enables file-level access control lists (ACLs). Learn more about Data Lake Storage Gen2

Data Lake Storage Gen2

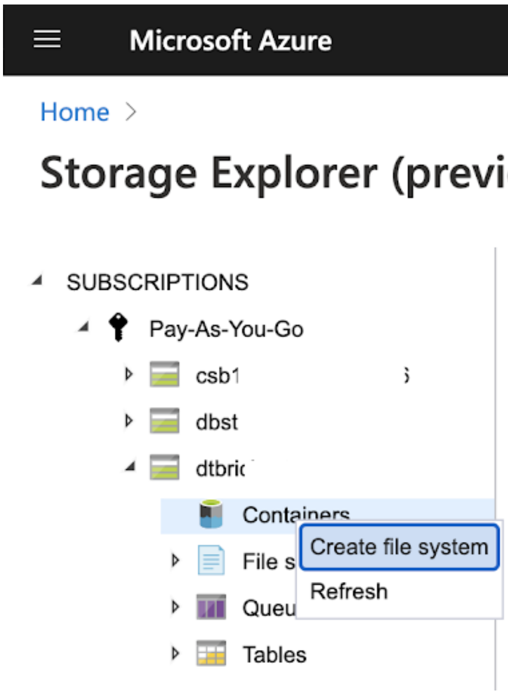
Hierarchical namespace ⓘ ○ Disabled ● Enabled

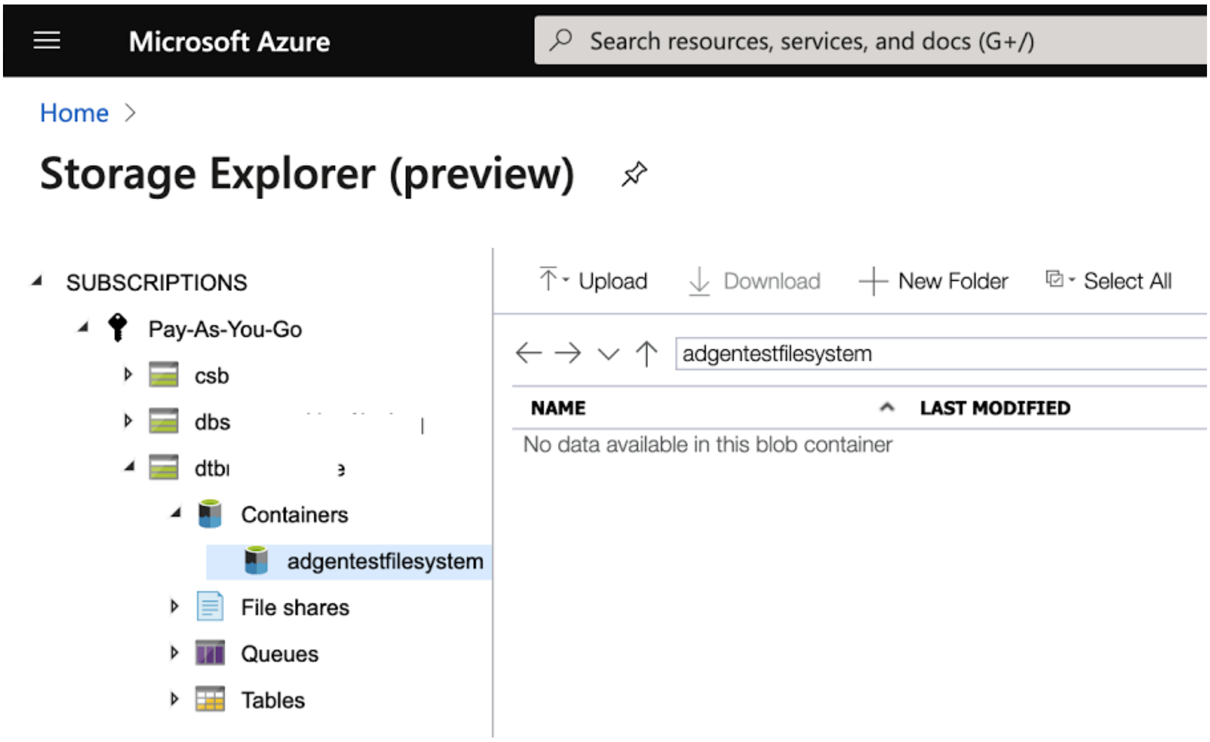
Azure Files

Large file shares ⓘ ● Disabled ○ Enabled

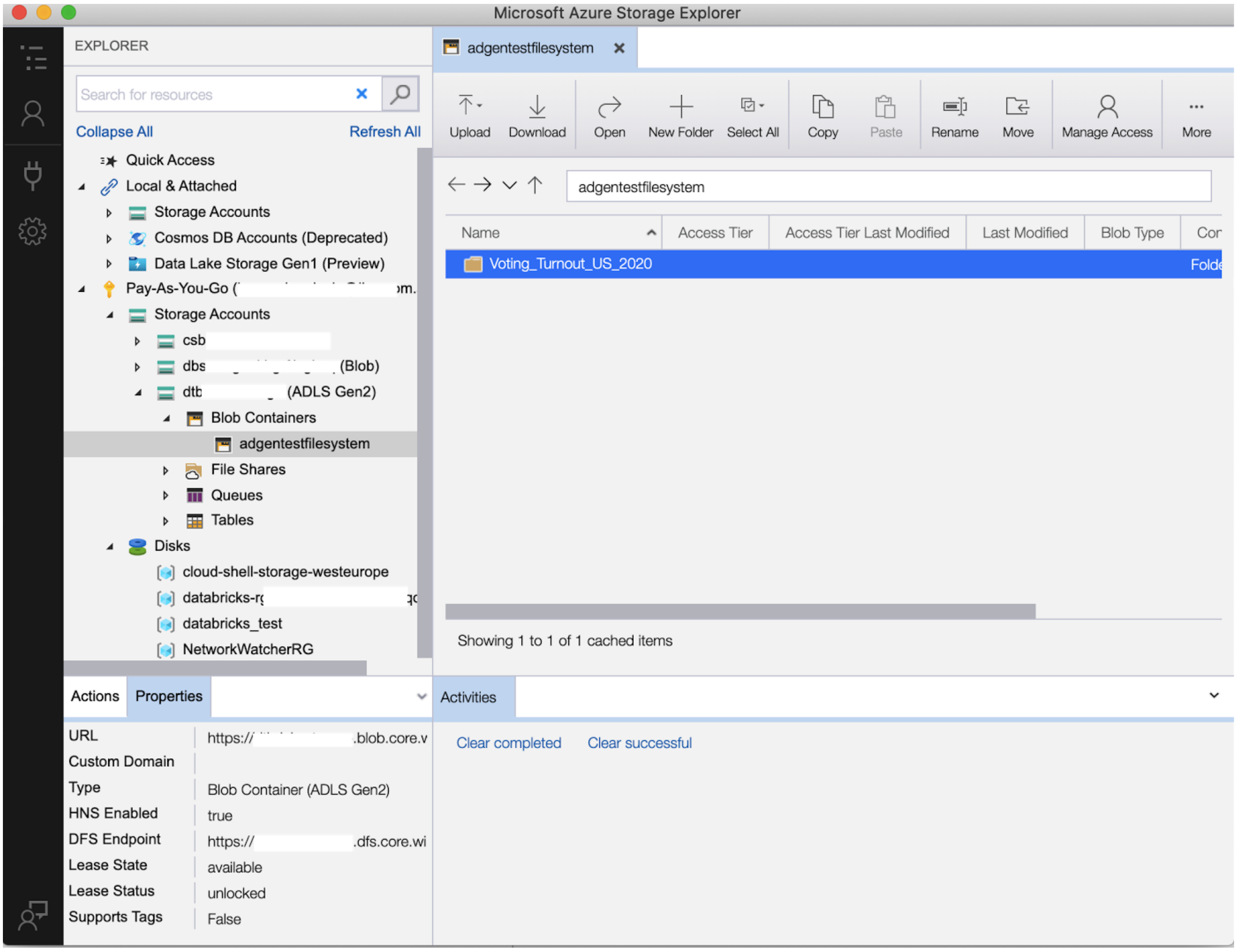
Tables and Queues

Customer-managed keys support ⓘ ● Disabled ○ Enabled





No data available in this blob container



Microsoft Azure Storage Explorer

EXPLORER

Search for resources

Collapse All   Refresh All

- Quick Access
- Local & Attached
  - Storage Accounts
  - Cosmos DB Accounts (Deprecated)
  - Data Lake Storage Gen1 (Preview)
- Pay-As-You-Go ([redacted]m.
  - Storage Accounts
    - csb[redacted]
    - dbs[redacted] (Blob)
    - dtb[redacted] (ADLS Gen2)
      - Blob Containers
        - adgentestfilesystem
      - File Shares
      - Queues
      - Tables
  - Disks
    - cloud-shell-storage-westeurope
    - databricks-r[redacted]
    - databricks_test
    - NetworkWatcherRG

adgentestfilesystem

Upload | Download | Open | New Folder | Select All | Copy | Paste | Rename | Move | Manage Access | More

adgentestfilesystem

Name	Access Tier	Access Tier Last Modified	Last Modified	Blob Type	Cor
Voting_Turnout_US_2020					Folde

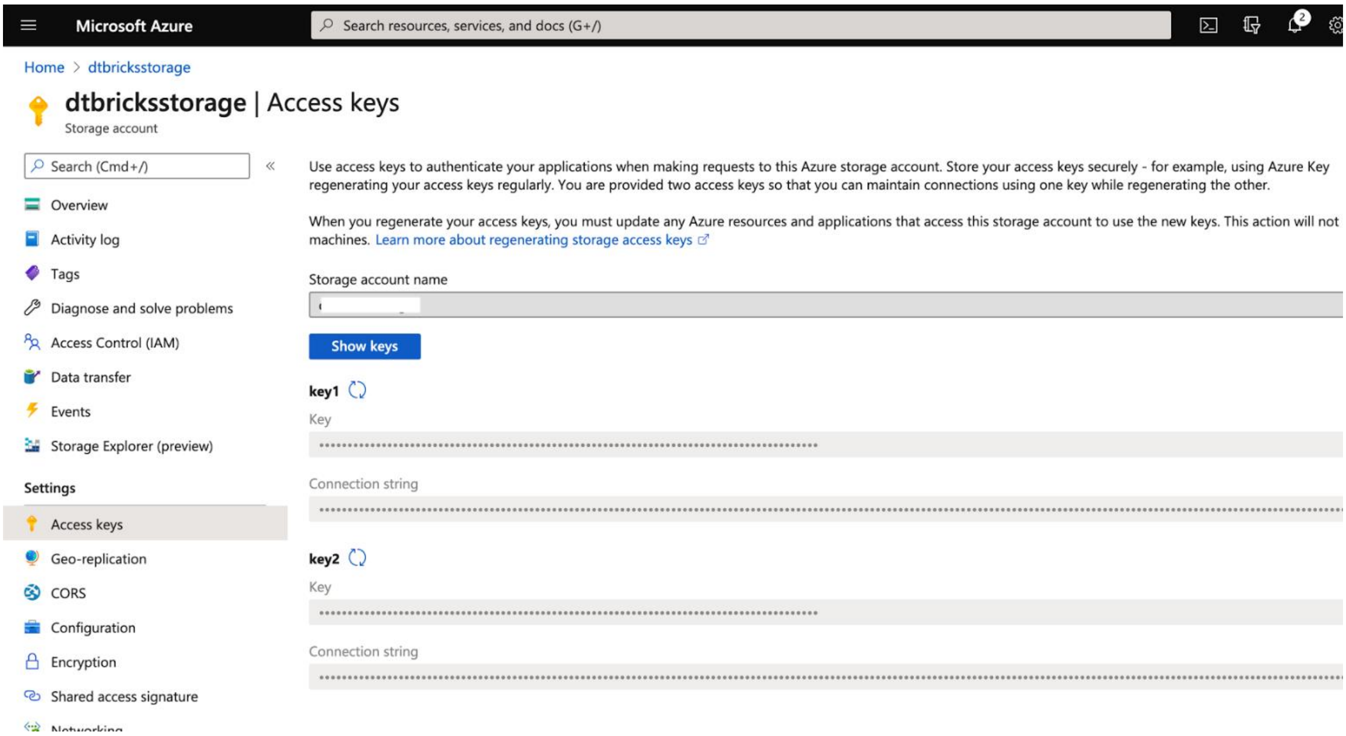
Showing 1 to 1 of 1 cached items

Actions | Properties

URL	https://[redacted].blob.core.v
Custom Domain	
Type	Blob Container (ADLS Gen2)
HNS Enabled	true
DFS Endpoint	https://[redacted].dfs.core.wi
Lease State	available
Lease Status	unlocked
Supports Tags	False

Activities

Clear completed   Clear successful



Microsoft Azure   Search resources, services, and docs (G+/)

Home > dtbricksstorage

dtbricksstorage | Access keys

Storage account

Search (Cmd+/)

- Overview
- Activity log
- Tags
- Diagnose and solve problems
- Access Control (IAM)
- Data transfer
- Events
- Storage Explorer (preview)

Settings

- Access keys
- Geo-replication
- CORS
- Configuration
- Encryption
- Shared access signature
- Networking

Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not machines. Learn more about regenerating storage access keys

Storage account name

[redacted]

Show keys

key1

Key

••

Connection string

••

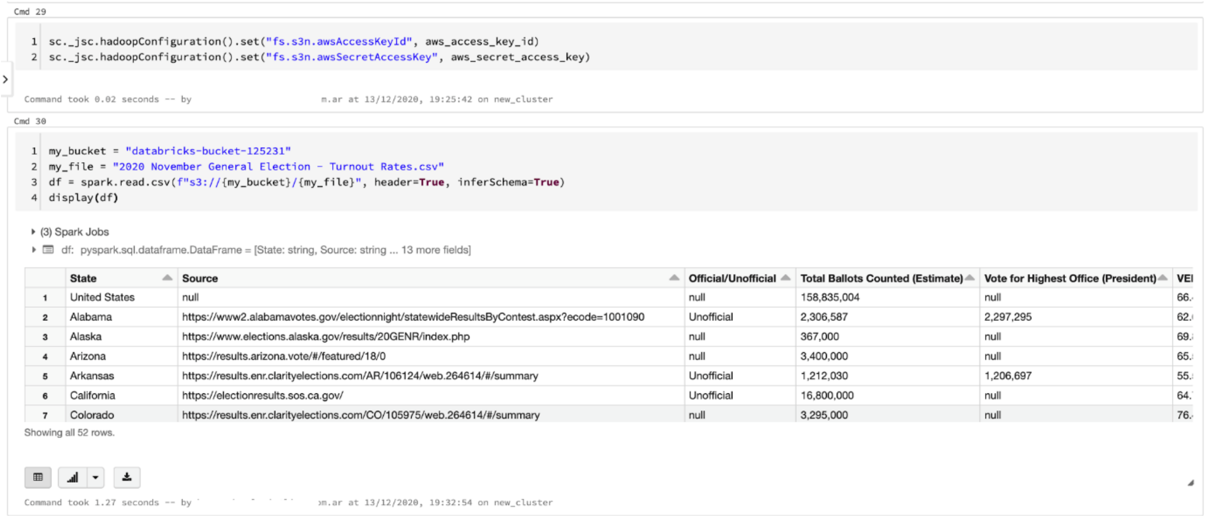
key2

Key

••

Connection string

••



Cmd 29

```
sc._jsc.hadoopConfiguration().set("fs.s3n.awsAccessKeyId", aws_access_key_id)
sc._jsc.hadoopConfiguration().set("fs.s3n.awsSecretAccessKey", aws_secret_access_key)
```

```
Command took 0.02 seconds -- by                    m.ar at 13/12/2020, 19:25:42 on new_cluster
```

Cmd 30

```
my_bucket = "databricks-bucket-125231"
my_file = "2020 November General Election - Turnout Rates.csv"
df = spark.read.csv(f"s3://{my_bucket}/{my_file}", header=True, inferSchema=True)
display(df)
```

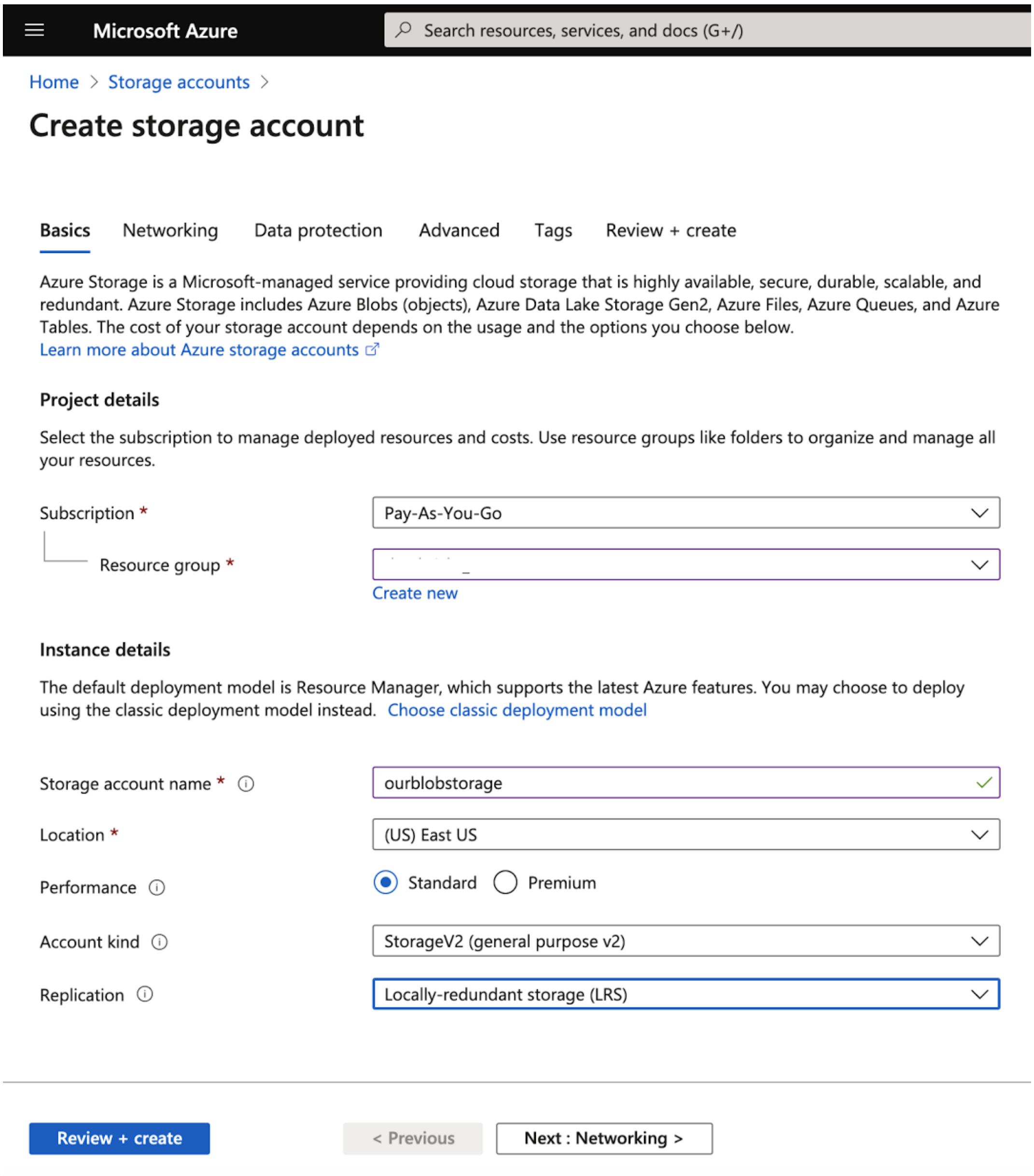
▸ (3) Spark Jobs

▸ df: pyspark.sql.dataframe.DataFrame = [State: string, Source: string ... 13 more fields]

	State	Source	Official/Unofficial	Total Ballots Counted (Estimate)	Vote for Highest Office (President)	VEI
1	United States	null	null	158,835,004	null	66.
2	Alabama	https://www2.alabamavotes.gov/electionnight/statewideResultsByContest.aspx?ecode=1001090	Unofficial	2,306,587	2,297,295	62.
3	Alaska	https://www.elections.alaska.gov/results/20GENR/index.php	null	367,000	null	69.
4	Arizona	https://results.arizona.vote/#/featured/18/0	null	3,400,000	null	65.
5	Arkansas	https://results.enr.clarityelections.com/AR/106124/web.264614/#/summary	Unofficial	1,212,030	1,206,697	55.
6	California	https://electionresults.sos.ca.gov/	Unofficial	16,800,000	null	64.
7	Colorado	https://results.enr.clarityelections.com/CO/105975/web.264614/#/summary	null	3,295,000	null	76.

Showing all 52 rows.

```
Command took 1.27 seconds -- by                    om.ar at 13/12/2020, 19:32:54 on new_cluster
```



Microsoft Azure
Search resources, services, and docs (G+/)
Home > Storage accounts >

Create storage account

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below.

Learn more about Azure storage accounts

Project details

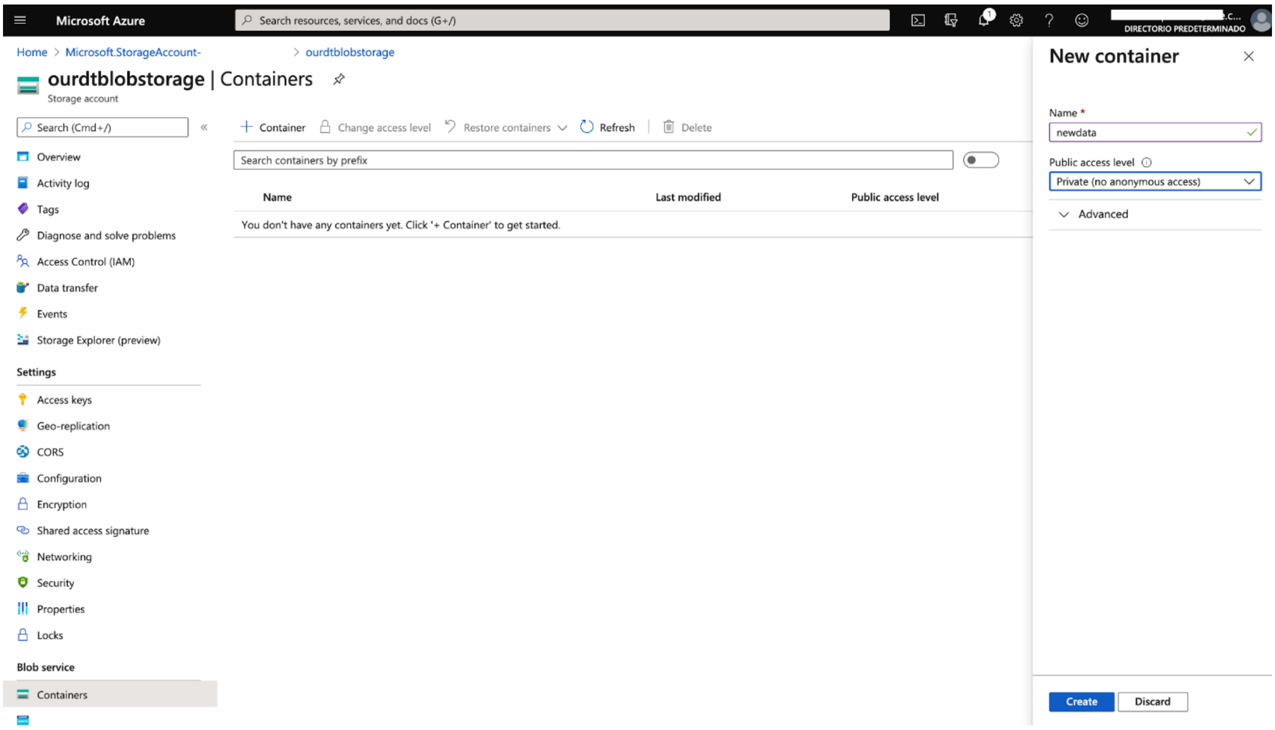
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Pay-As-You-Go
Resource group *	 Create new

Instance details

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. Choose classic deployment model

Storage account name * ⓘ	ourblobstorage ✓
Location *	(US) East US
Performance ⓘ	● Standard ○ Premium
Account kind ⓘ	StorageV2 (general purpose v2)
Replication ⓘ	Locally-redundant storage (LRS)



ourdtblobstorage | Containers

Storage account

Search containers by prefix

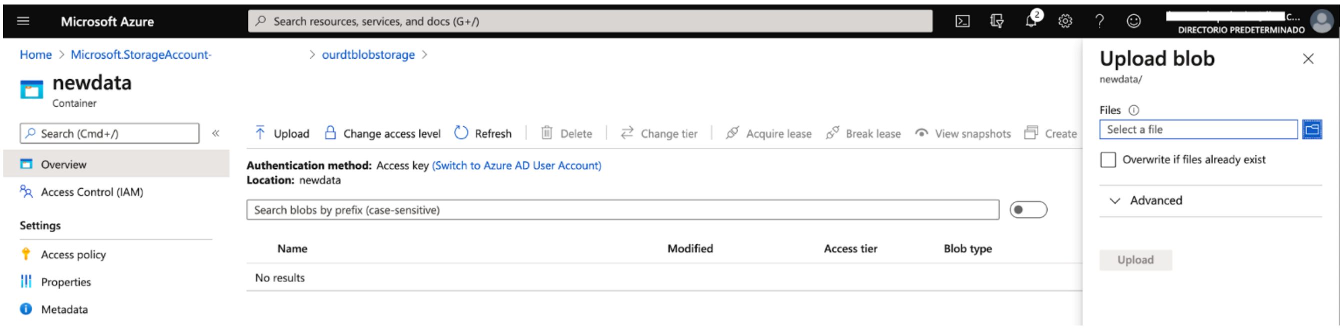
Name	Last modified	Public access level
You don't have any containers yet. Click '+ Container' to get started.		

New container

Name *: newdata

Public access level: Private (no anonymous access)

Advanced



newdata

Container

Authentication method: Access key (Switch to Azure AD User Account)

Location: newdata

Search blobs by prefix (case-sensitive)

Name	Modified	Access tier	Blob type
No results			

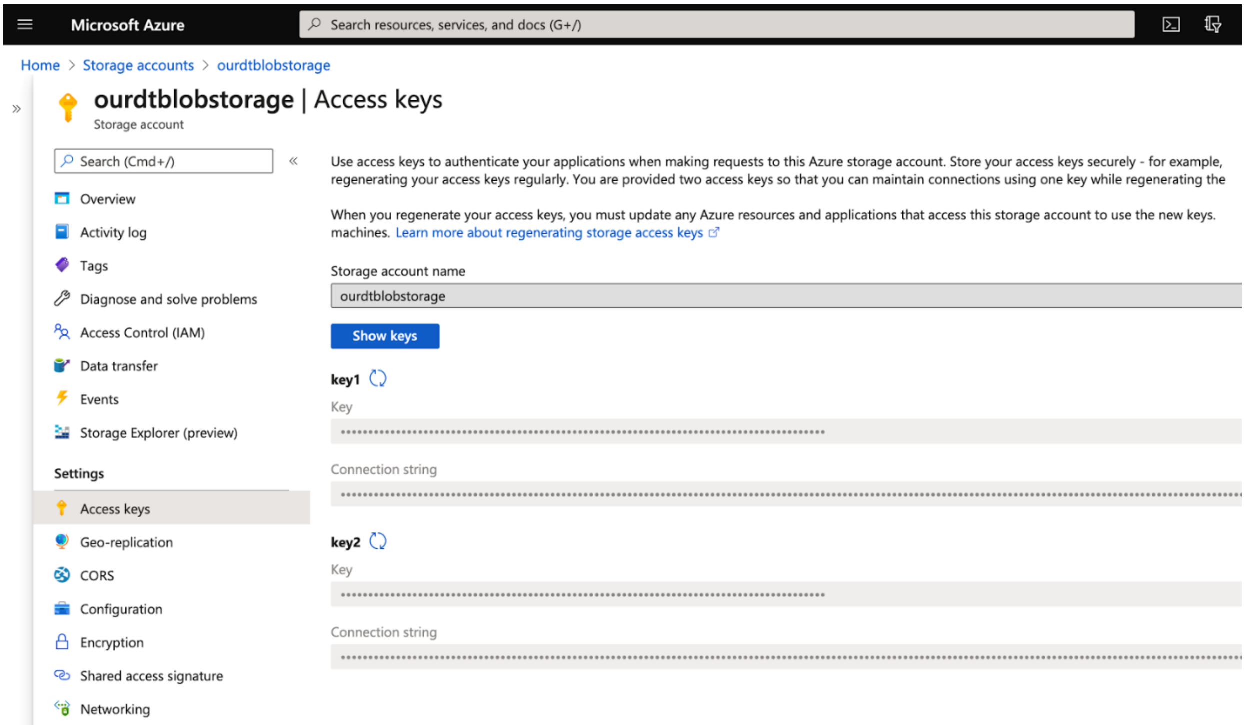
Upload blob

newdata/

Files: Select a file

□ Overwrite if files already exist

Advanced



ourdtblobstorage | Access keys

Storage account

Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. machines. Learn more about regenerating storage access keys

Storage account name: ourdtblobstorage

key1

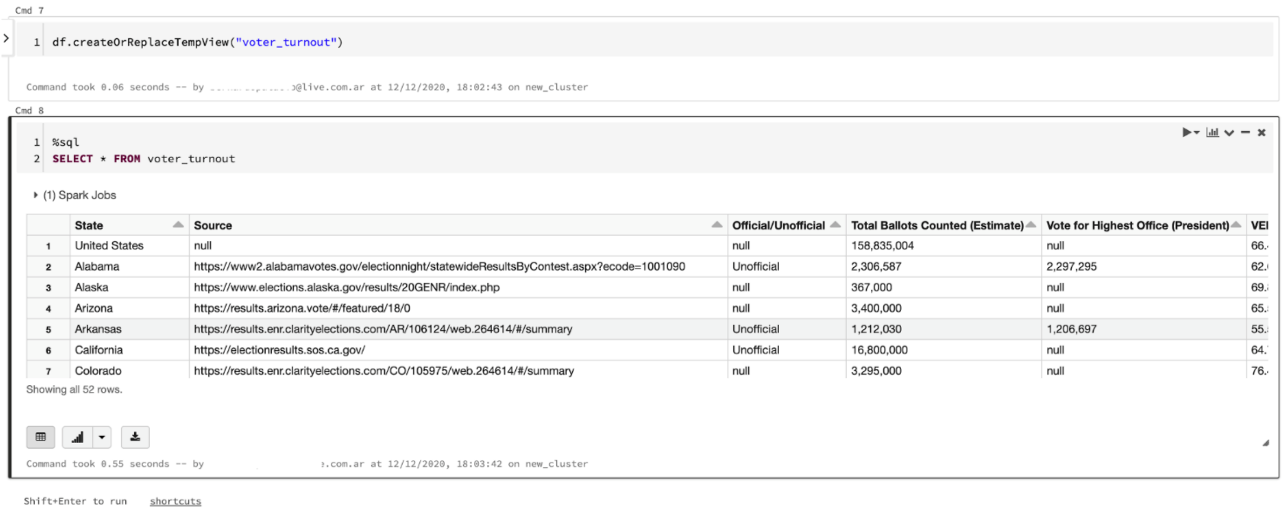
Key: ••

Connection string: •••

key2

Key: ••

Connection string: •••



Cmd 7

```
df.createOrReplaceTempView("voter_turnout")
```

Command took 0.06 seconds -- by [redacted]@live.com.ar at 12/12/2020, 18:02:43 on new_cluster

Cmd 8

```
%sql
SELECT * FROM voter_turnout
```

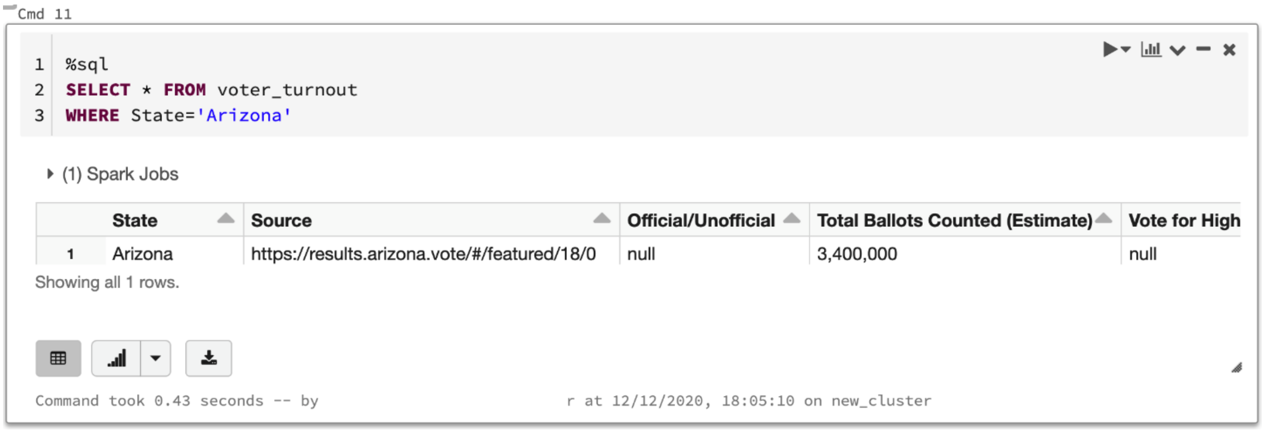
(1) Spark Jobs

	State	Source	Official/Unofficial	Total Ballots Counted (Estimate)	Vote for Highest Office (President)	VEI
1	United States	null	null	158,835,004	null	66.
2	Alabama	https://www2.alabamavotes.gov/electionnight/statewideResultsByContest.aspx?ecode=1001090	Unofficial	2,306,587	2,297,295	62.
3	Alaska	https://www.elections.alaska.gov/results/20GENR/index.php	null	367,000	null	69.
4	Arizona	https://results.arizona.vote/#/featured/18/0	null	3,400,000	null	65.
5	Arkansas	https://results.enr.clarityelections.com/AR/106124/web.264614/#/summary	Unofficial	1,212,030	1,206,697	55.
6	California	https://electionresults.sos.ca.gov/	Unofficial	16,800,000	null	64.
7	Colorado	https://results.enr.clarityelections.com/CO/105975/web.264614/#/summary	null	3,295,000	null	76.

Showing all 52 rows.

Command took 0.55 seconds -- by [redacted].com.ar at 12/12/2020, 18:03:42 on new_cluster

Shift+Enter to run shortcuts



Cmd 11

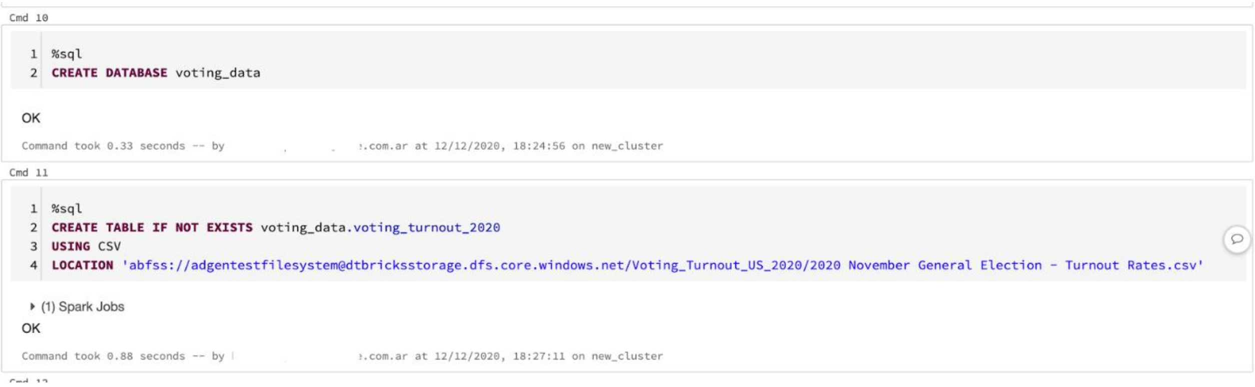
```
%sql
SELECT * FROM voter_turnout
WHERE State='Arizona'
```

(1) Spark Jobs

	State	Source	Official/Unofficial	Total Ballots Counted (Estimate)	Vote for High
1	Arizona	https://results.arizona.vote/#/featured/18/0	null	3,400,000	null

Showing all 1 rows.

Command took 0.43 seconds -- by [redacted]r at 12/12/2020, 18:05:10 on new_cluster



Cmd 10

```
%sql
CREATE DATABASE voting_data
```

OK

Command took 0.33 seconds -- by [redacted].com.ar at 12/12/2020, 18:24:56 on new_cluster

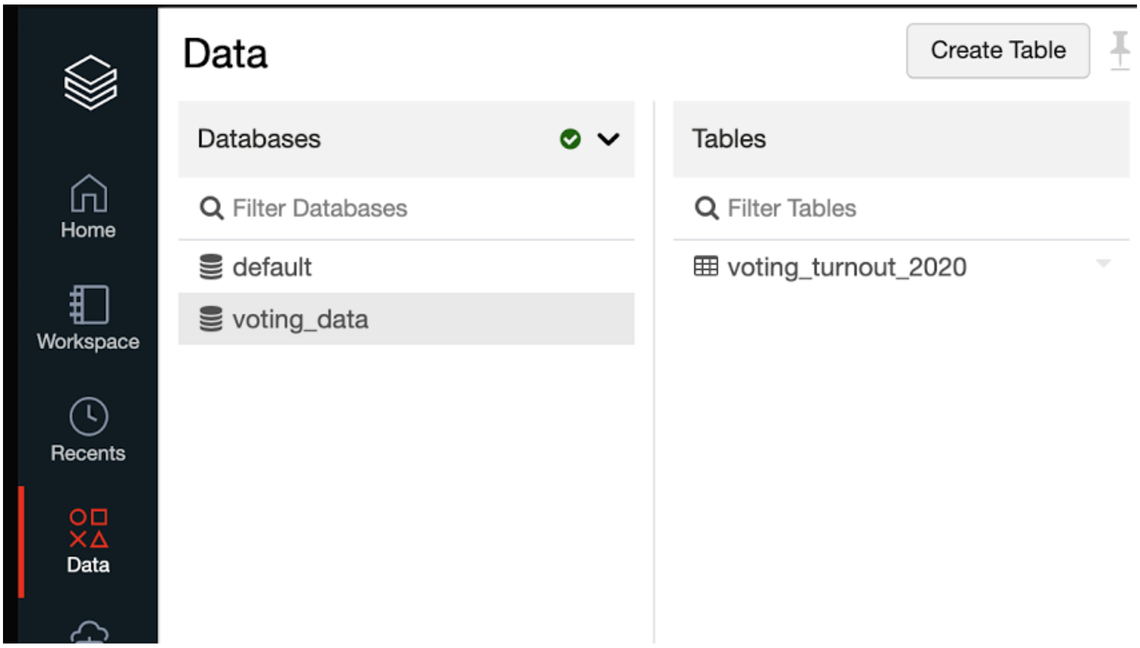
Cmd 11

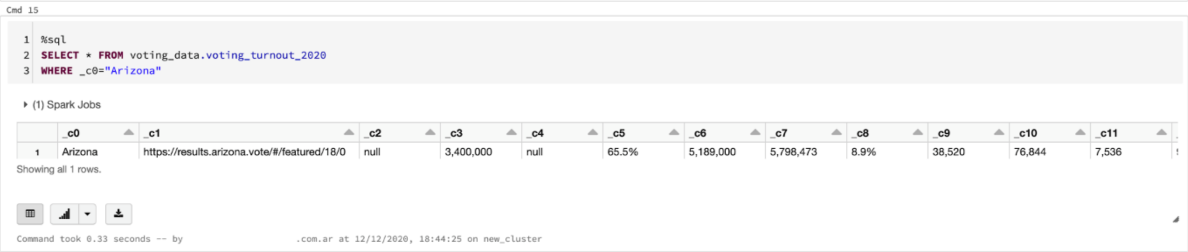
```
%sql
CREATE TABLE IF NOT EXISTS voting_data.voting_turnout_2020
USING CSV
LOCATION 'abfss://adgentestfilesystem@dtbricksstorage.dfs.core.windows.net/Voting_Turnout_US_2020/2020 November General Election - Turnout Rates.csv'
```

(1) Spark Jobs

OK

Command took 0.88 seconds -- by [redacted].com.ar at 12/12/2020, 18:27:11 on new_cluster





Cmd 15

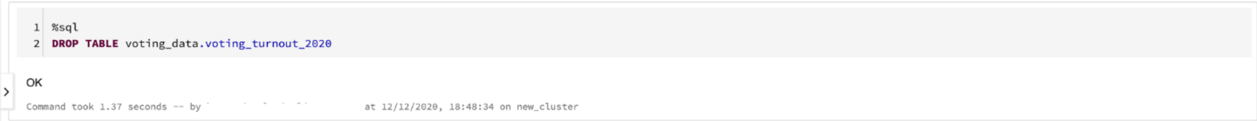
```
%sql
SELECT * FROM voting_data.voting_turnout_2020
WHERE _c0="Arizona"
```

(1) Spark Jobs

	_c0	_c1	_c2	_c3	_c4	_c5	_c6	_c7	_c8	_c9	_c10	_c11
1	Arizona	https://results.arizona.vote/#/featured/18/0	null	3,400,000	null	65.5%	5,189,000	5,798,473	8.9%	38,520	76,844	7,536

Showing all 1 rows.

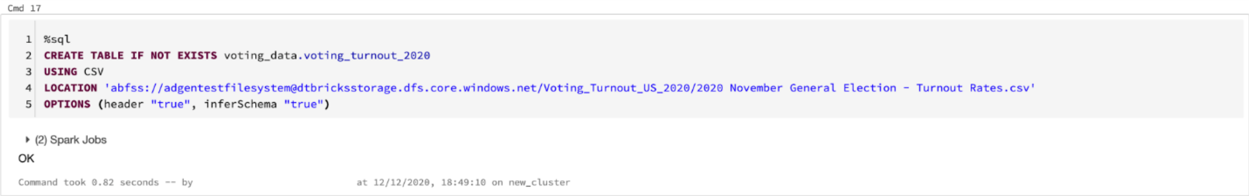
```
Command took 0.33 seconds -- by                    .com.ar at 12/12/2020, 18:44:25 on new_cluster
```



```
%sql
DROP TABLE voting_data.voting_turnout_2020
```

OK

```
Command took 1.37 seconds -- by                    at 12/12/2020, 18:48:34 on new_cluster
```



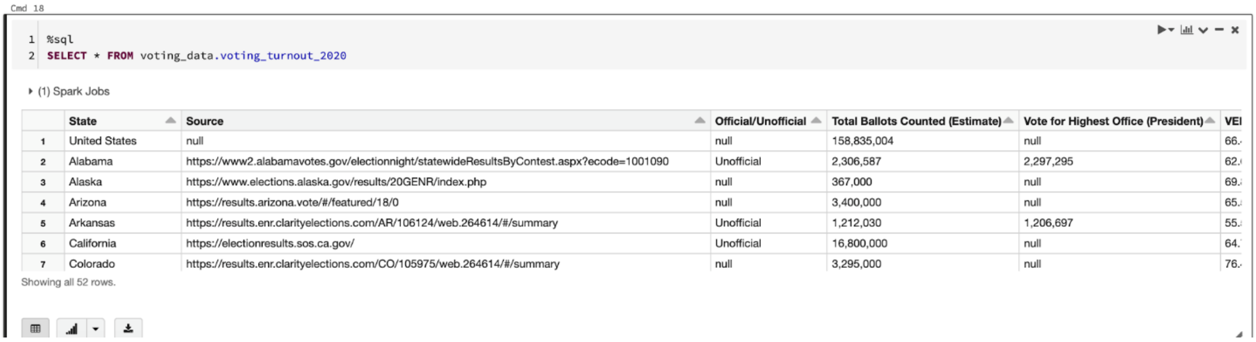
Cmd 17

```
%sql
CREATE TABLE IF NOT EXISTS voting_data.voting_turnout_2020
USING CSV
LOCATION 'abfss://adgentestfilesystem@dtbricksstorage.dfs.core.windows.net/Voting_Turnout_US_2020/2020 November General Election - Turnout Rates.csv'
OPTIONS (header "true", inferSchema "true")
```

(2) Spark Jobs

OK

```
Command took 0.82 seconds -- by                    at 12/12/2020, 18:49:10 on new_cluster
```



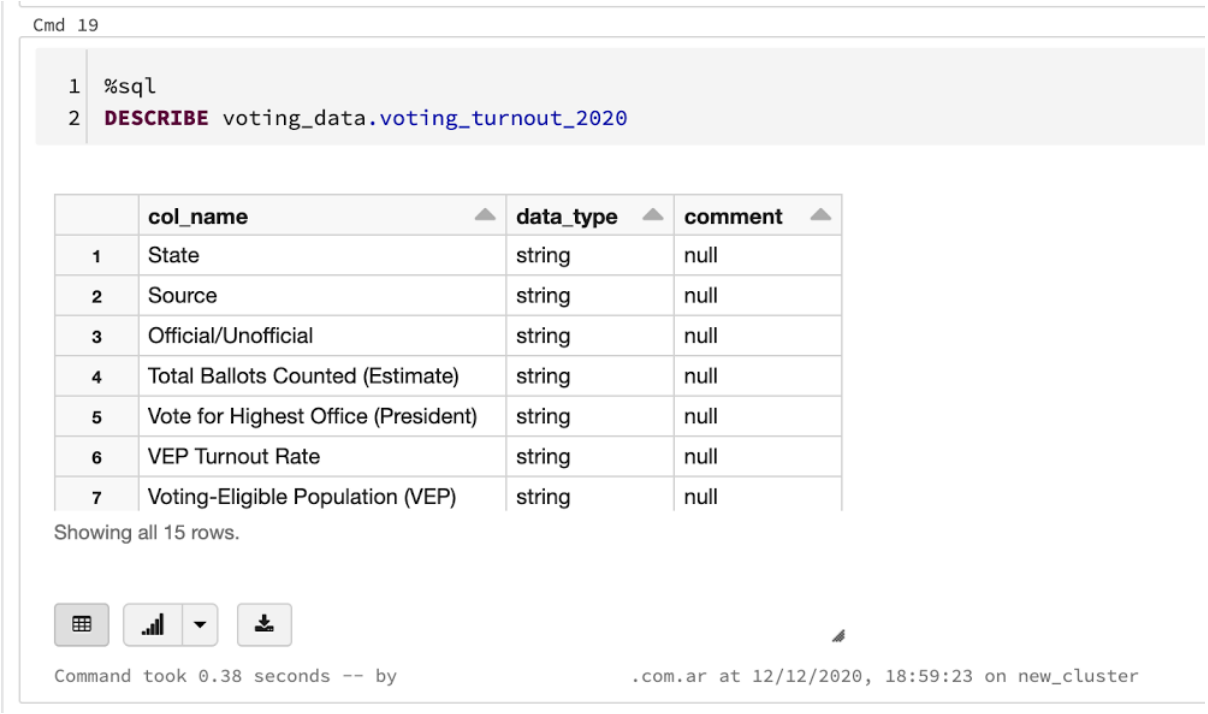
Cmd 18

```
%sql
SELECT * FROM voting_data.voting_turnout_2020
```

(1) Spark Jobs

	State	Source	Official/Unofficial	Total Ballots Counted (Estimate)	Vote for Highest Office (President)	VEI
1	United States	null	null	158,835,004	null	66.
2	Alabama	https://www2.alabamavotes.gov/electionnight/statewideResultsByContest.aspx?ecode=1001090	Unofficial	2,306,587	2,297,295	62.
3	Alaska	https://www.elections.alaska.gov/results/20GENR/index.php	null	367,000	null	69.
4	Arizona	https://results.arizona.vote/#/featured/18/0	null	3,400,000	null	65.
5	Arkansas	https://results.enr.clarityelections.com/AR/106124/web.264614/#/summary	Unofficial	1,212,030	1,206,697	55.
6	California	https://electionresults.sos.ca.gov/	Unofficial	16,800,000	null	64.
7	Colorado	https://results.enr.clarityelections.com/CO/105975/web.264614/#/summary	null	3,295,000	null	76.

Showing all 52 rows.



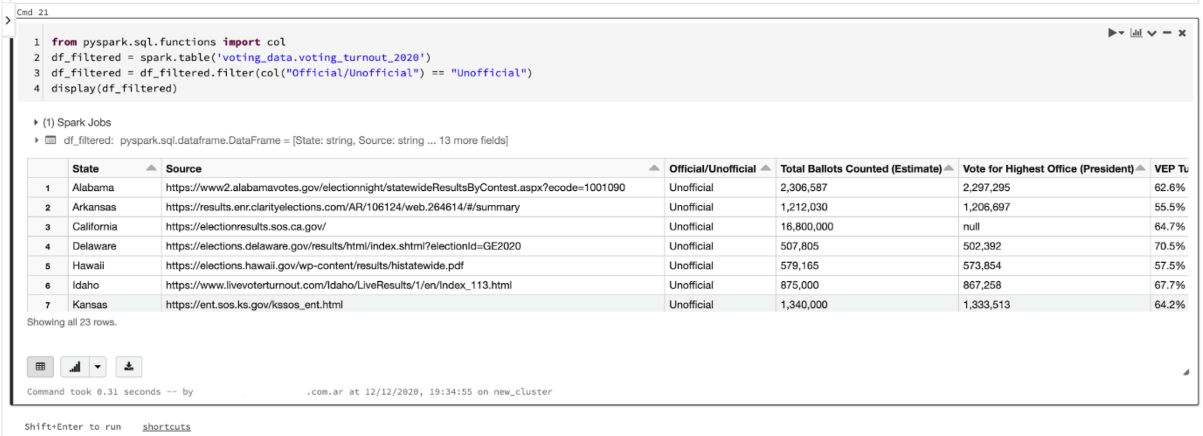
Cmd 19

```
%sql
DESCRIBE voting_data.voting_turnout_2020
```

	col_name	data_type	comment
1	State	string	null
2	Source	string	null
3	Official/Unofficial	string	null
4	Total Ballots Counted (Estimate)	string	null
5	Vote for Highest Office (President)	string	null
6	VEP Turnout Rate	string	null
7	Voting-Eligible Population (VEP)	string	null

Showing all 15 rows.

Command took 0.38 seconds -- by .com.ar at 12/12/2020, 18:59:23 on new_cluster



Cmd 21

```
from pyspark.sql.functions import col
df_filtered = spark.table('voting_data.voting_turnout_2020')
df_filtered = df_filtered.filter(col("Official/Unofficial") == "Unofficial")
display(df_filtered)
```

(1) Spark Jobs

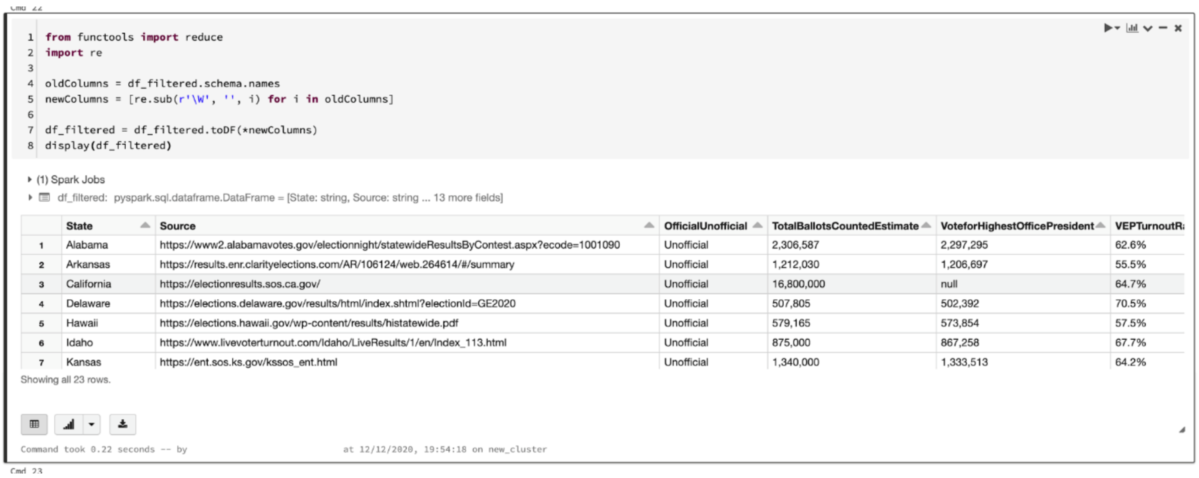
df_filtered: pyspark.sql.dataframe.DataFrame = [State: string, Source: string ... 13 more fields]

	State	Source	Official/Unofficial	Total Ballots Counted (Estimate)	Vote for Highest Office (President)	VEP Tu
1	Alabama	https://www2.alabamavotes.gov/electionnight/statewideResultsByContest.aspx?ecode=1001090	Unofficial	2,306,587	2,297,295	62.6%
2	Arkansas	https://results.enr.clarityelections.com/AR/106124/web.264614/#/summary	Unofficial	1,212,030	1,206,697	55.5%
3	California	https://electionresults.sos.ca.gov/	Unofficial	16,800,000	null	64.7%
4	Delaware	https://elections.delaware.gov/results/html/index.shtml?electionId=GE2020	Unofficial	507,805	502,392	70.5%
5	Hawaii	https://elections.hawaii.gov/wp-content/results/histatewide.pdf	Unofficial	579,165	573,854	57.5%
6	Idaho	https://www.livevoterturnout.com/Idaho/LiveResults/1/en/Index_113.html	Unofficial	875,000	867,258	67.7%
7	Kansas	https://ent.sos.ks.gov/kssos_ent.html	Unofficial	1,340,000	1,333,513	64.2%

Showing all 23 rows.

Command took 0.31 seconds -- by .com.ar at 12/12/2020, 19:34:55 on new_cluster

Shift+Enter to run shortcuts



Cmd 22

```
from functools import reduce
import re

oldColumns = df_filtered.schema.names
newColumns = [re.sub(r'\W', '', i) for i in oldColumns]

df_filtered = df_filtered.toDF(*newColumns)
display(df_filtered)
```

(1) Spark Jobs

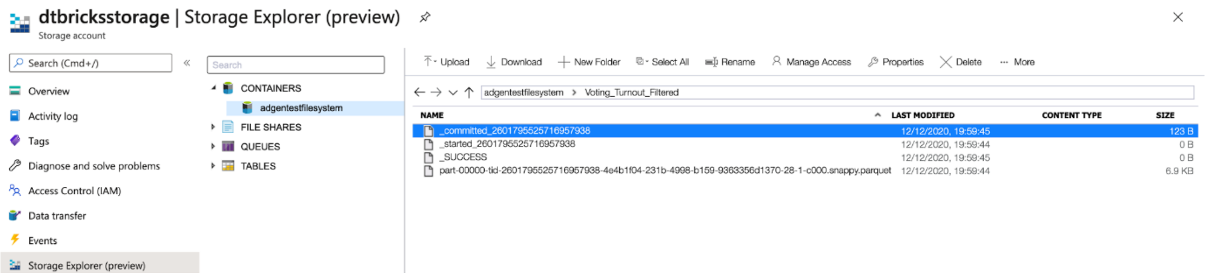
df_filtered: pyspark.sql.dataframe.DataFrame = [State: string, Source: string ... 13 more fields]

	State	Source	OfficialUnofficial	TotalBallotsCountedEstimate	VoteforHighestOfficePresident	VEPTurnoutRa
1	Alabama	https://www2.alabamavotes.gov/electionnight/statewideResultsByContest.aspx?ecode=1001090	Unofficial	2,306,587	2,297,295	62.6%
2	Arkansas	https://results.enr.clarityelections.com/AR/106124/web.264614/#/summary	Unofficial	1,212,030	1,206,697	55.5%
3	California	https://electionresults.sos.ca.gov/	Unofficial	16,800,000	null	64.7%
4	Delaware	https://elections.delaware.gov/results/html/index.shtml?electionId=GE2020	Unofficial	507,805	502,392	70.5%
5	Hawaii	https://elections.hawaii.gov/wp-content/results/histatewide.pdf	Unofficial	579,165	573,854	57.5%
6	Idaho	https://www.livevoterturnout.com/Idaho/LiveResults/1/en/Index_113.html	Unofficial	875,000	867,258	67.7%
7	Kansas	https://ent.sos.ks.gov/kssos_ent.html	Unofficial	1,340,000	1,333,513	64.2%

Showing all 23 rows.

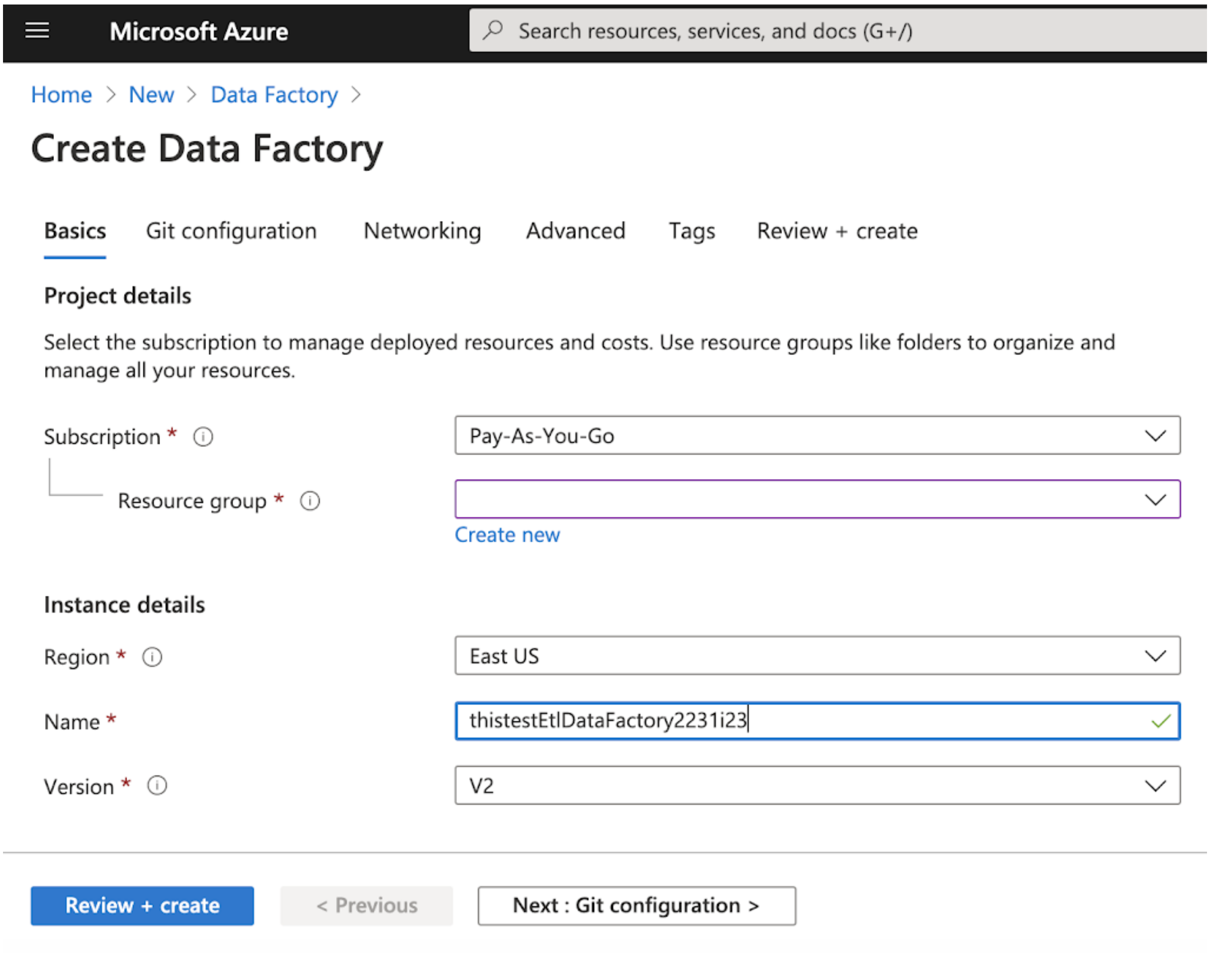
Command took 0.22 seconds -- by at 12/12/2020, 19:54:18 on new_cluster

Cmd 23



adgentestfilesystem > Voting_Turnout_Filtered

NAME	LAST MODIFIED	CONTENT TYPE	SIZE
_committed_2601795525716957938	12/12/2020, 19:59:45		123 B
_started_2601795525716957938	12/12/2020, 19:59:44		0 B
_SUCCESS	12/12/2020, 19:59:45		0 B
part-00000-tid-2601795525716957938-4e4b1f04-231b-4998-b159-9363356d1370-28-1-c000.snappy.parquet	12/12/2020, 19:59:44		6.9 KB


Microsoft Azure
Search resources, services, and docs (G+/)
Home > New > Data Factory >

Create Data Factory

Project details

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

Subscription * : Pay-As-You-Go

Resource group * :

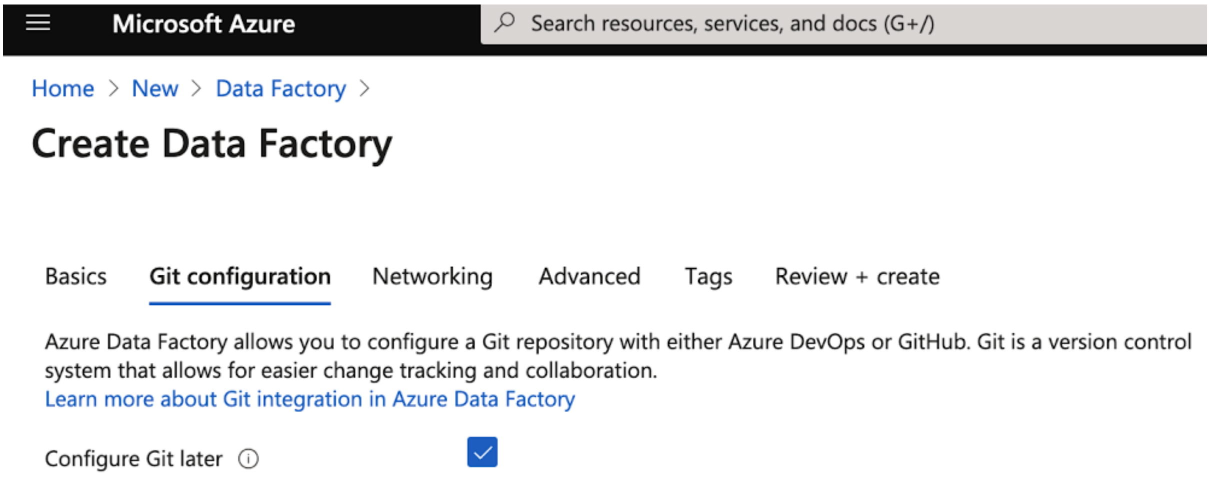
Create new

Instance details

Region * : East US

Name * : thistestEtlDataFactory2231i23

Version * : V2

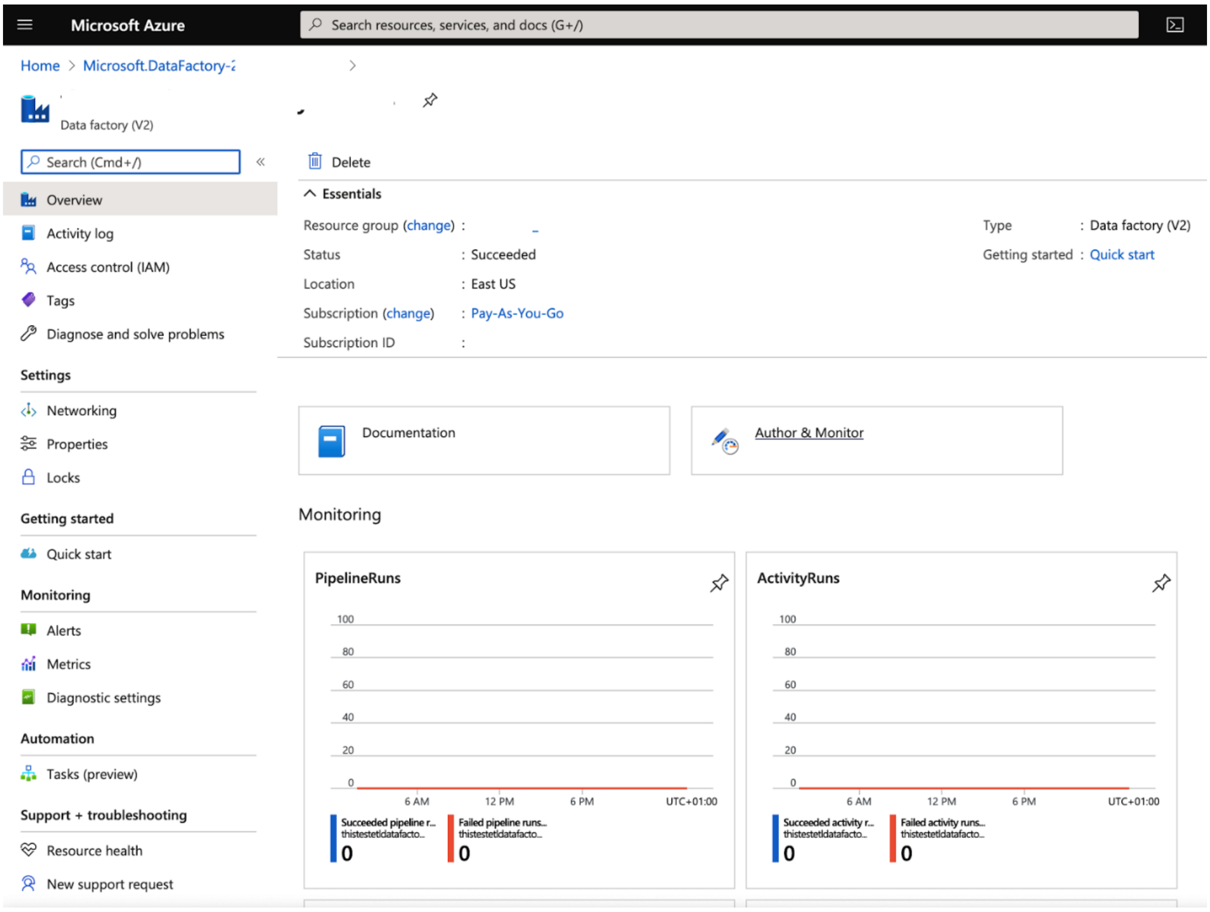

Microsoft Azure
Search resources, services, and docs (G+/)
Home > New > Data Factory >

Create Data Factory

Azure Data Factory allows you to configure a Git repository with either Azure DevOps or GitHub. Git is a version control system that allows for easier change tracking and collaboration.

Learn more about Git integration in Azure Data Factory

Configure Git later ☑



Resource group (change) :

Status : Succeeded

Location : East US

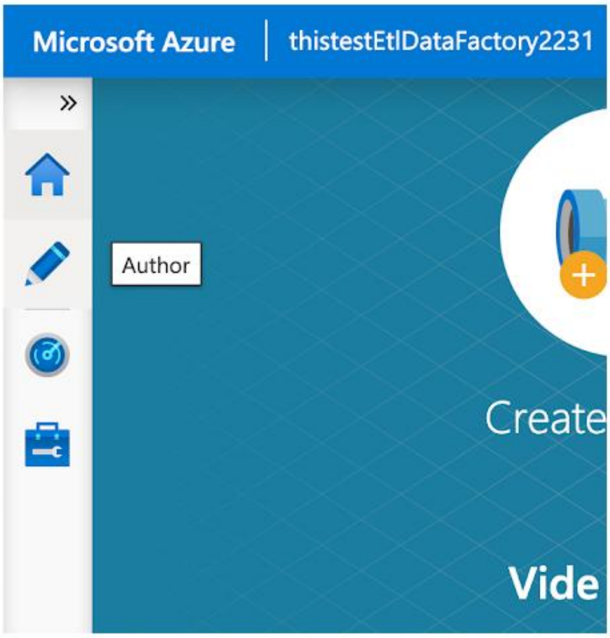
Subscription (change) : Pay-As-You-Go

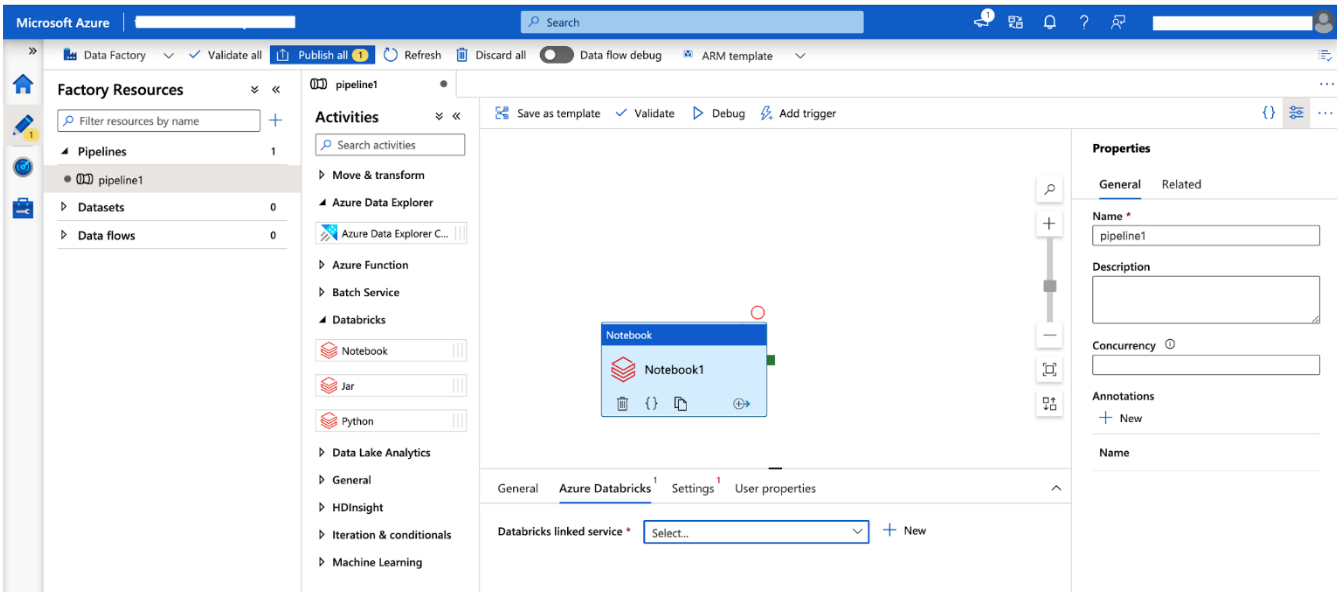
Subscription ID :

Type : Data factory (V2)

Getting started : Quick start

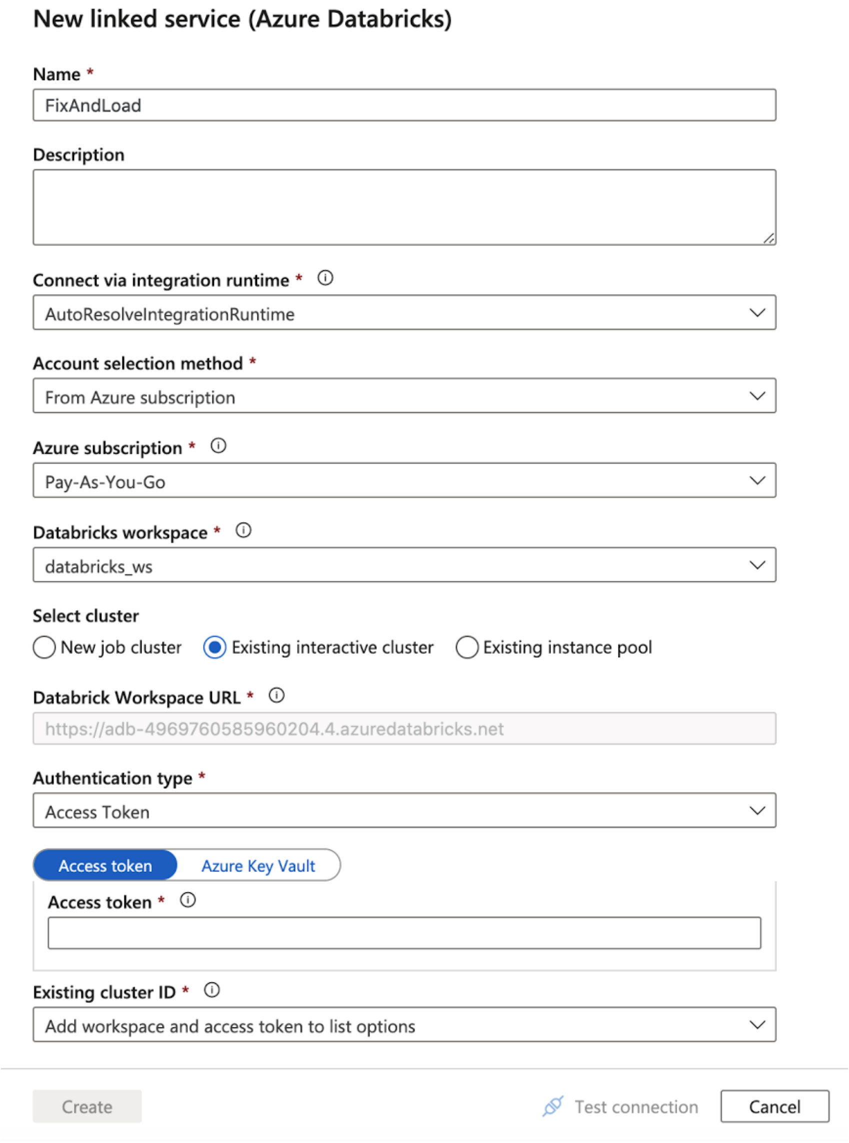
Documentation
Author & Monitor
Monitoring
PipelineRuns
100
80
60
40
20
0
6 AM
12 PM
6 PM
UTC+01:00
Succeeded pipeline r... thistestetldatafacto...
0
Failed pipeline runs... thistestetldatafacto...
0
ActivityRuns
100
80
60
40
20
0
6 AM
12 PM
6 PM
UTC+01:00
Succeeded activity r... thistestetldatafacto...
0
Failed activity runs... thistestetldatafacto...
0





Name * pipeline1

Databricks linked service * Select...



New linked service (Azure Databricks)

Name * FixAndLoad

Description

Connect via integration runtime * AutoResolveIntegrationRuntime

Account selection method * From Azure subscription

Azure subscription * Pay-As-You-Go

Databricks workspace * databricks_ws

Select cluster

○ New job cluster ● Existing interactive cluster ○ Existing instance pool

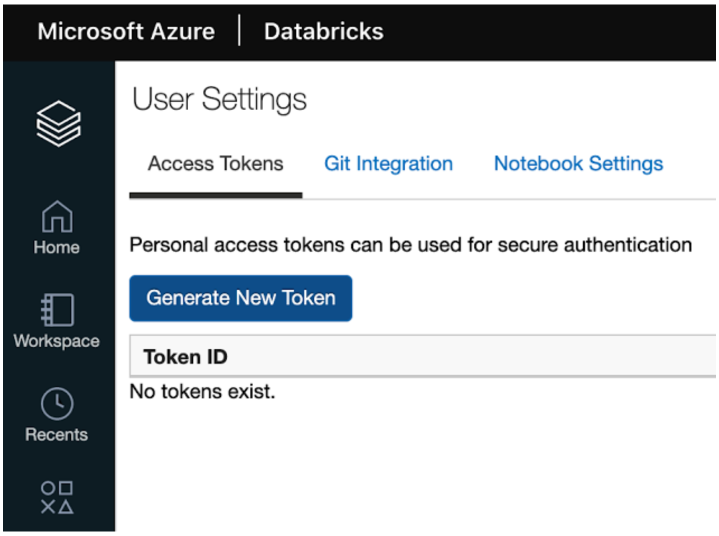
Databrick Workspace URL * https://adb-4969760585960204.4.azuredatabricks.net

Authentication type * Access Token

Access token | Azure Key Vault

Access token *

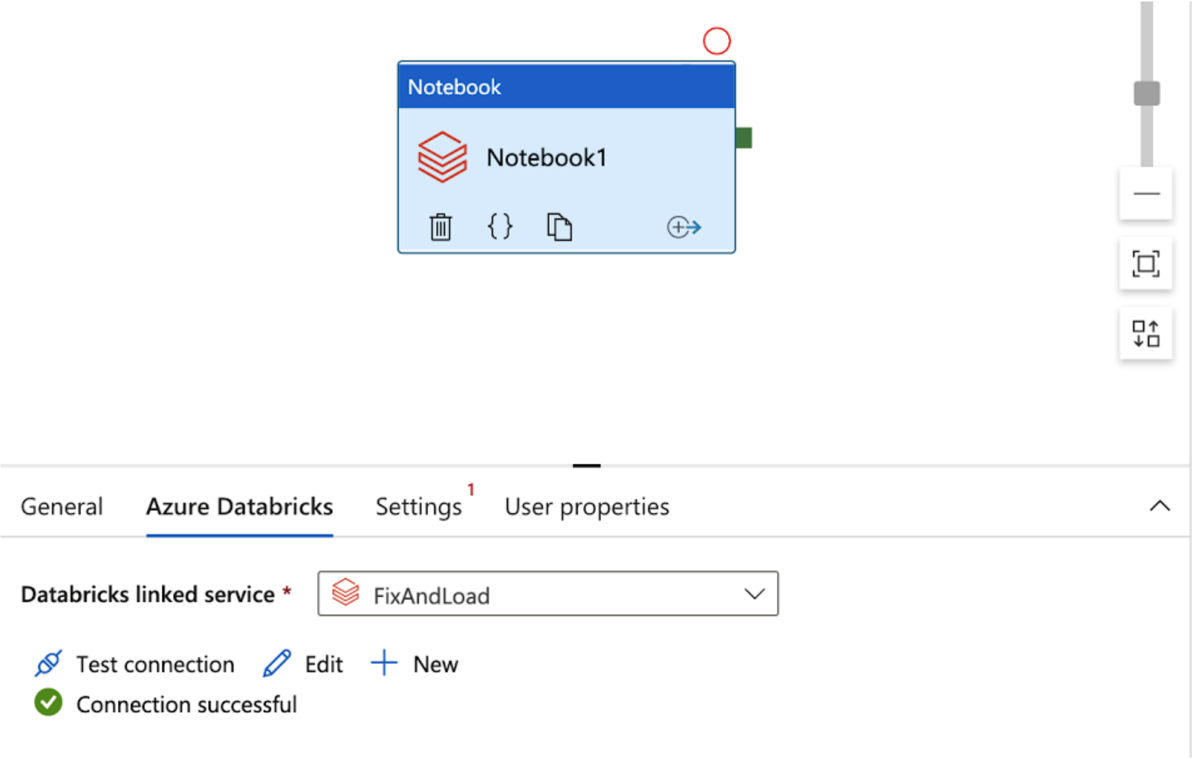
Existing cluster ID * Add workspace and access token to list options



User Settings

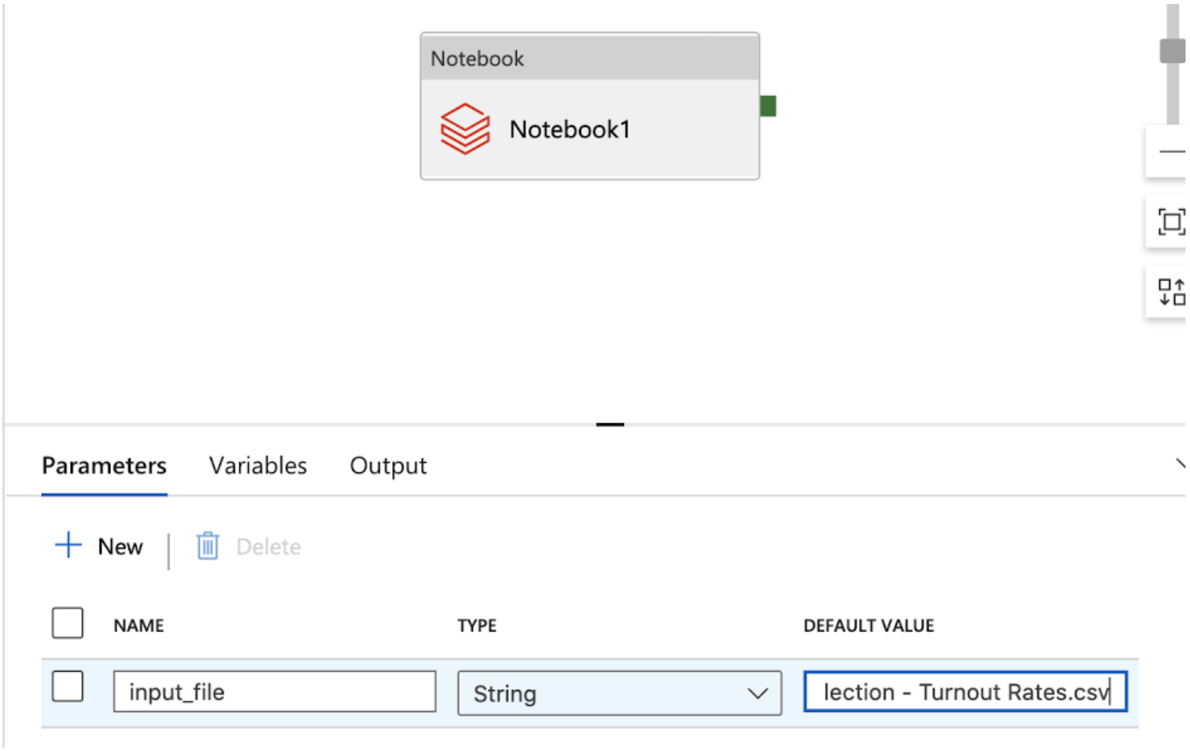
Personal access tokens can be used for secure authentication

Token ID
No tokens exist.

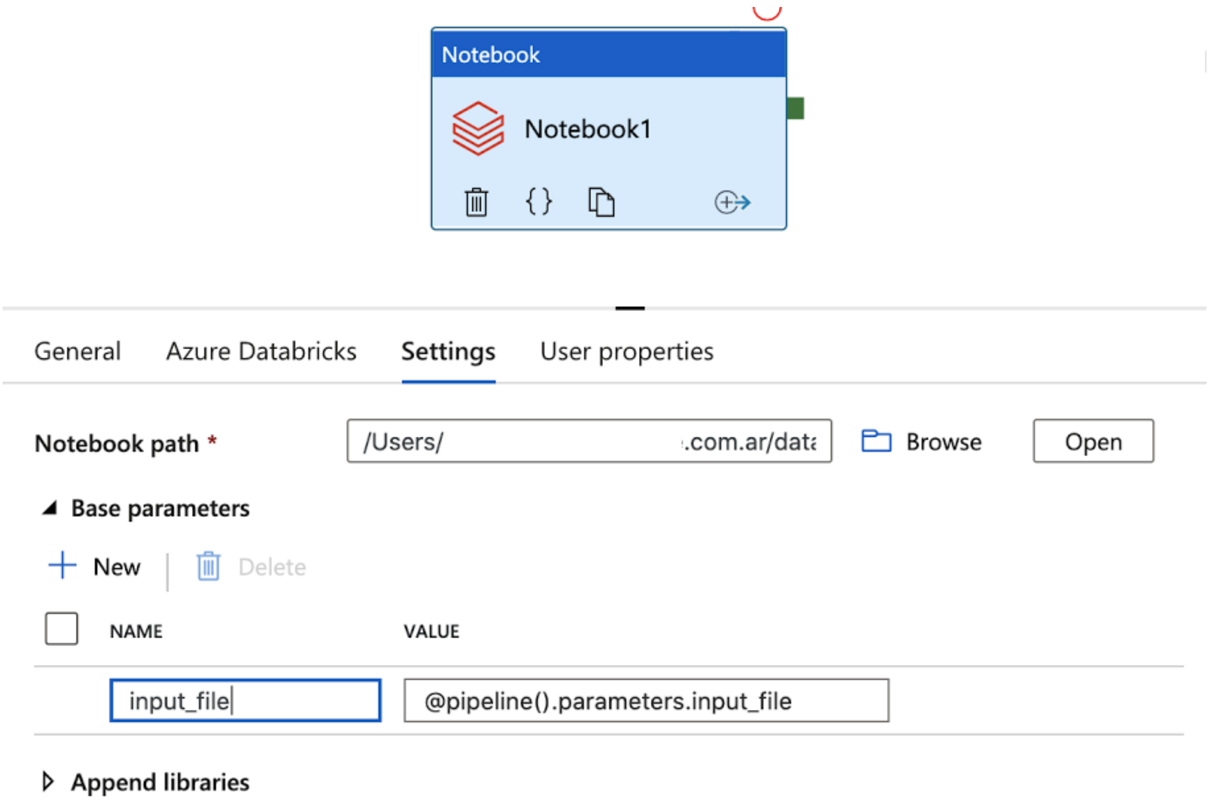


Databricks linked service * FixAndLoad

Connection successful



NAME	TYPE	DEFAULT VALUE
input_file	String	lection - Turnout Rates.csv

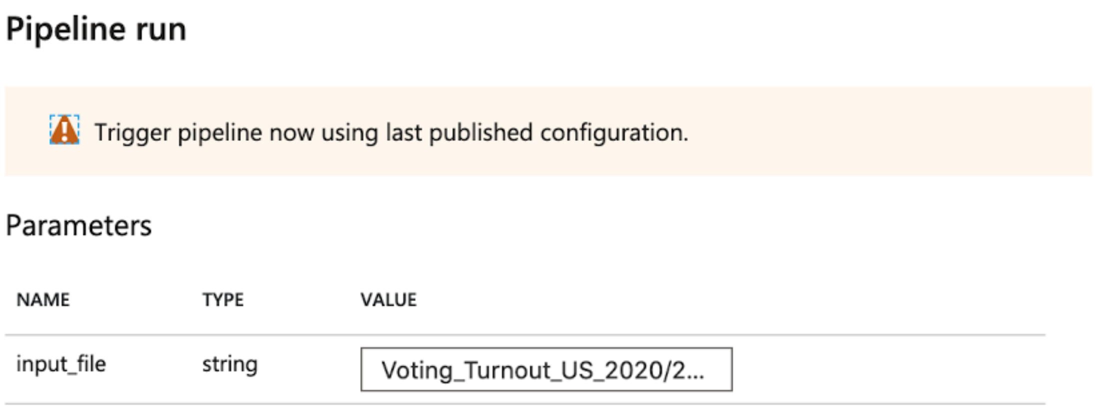


Notebook path * /Users/ .com.ar/data

Base parameters

NAME	VALUE
input_file	@pipeline().parameters.input_file

Append libraries

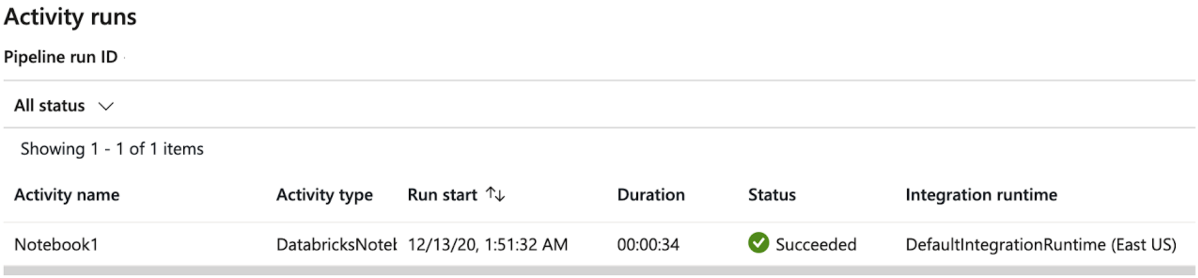


Pipeline run

⚠ Trigger pipeline now using last published configuration.

Parameters

NAME	TYPE	VALUE
input_file	string	Voting_Turnout_US_2020/2...



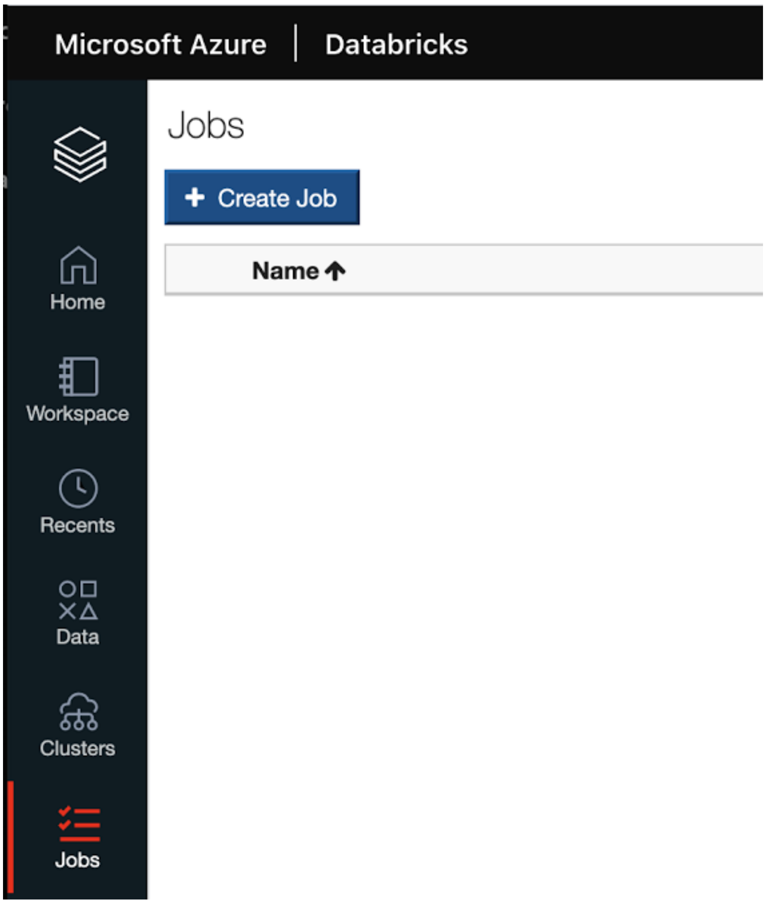
Activity runs

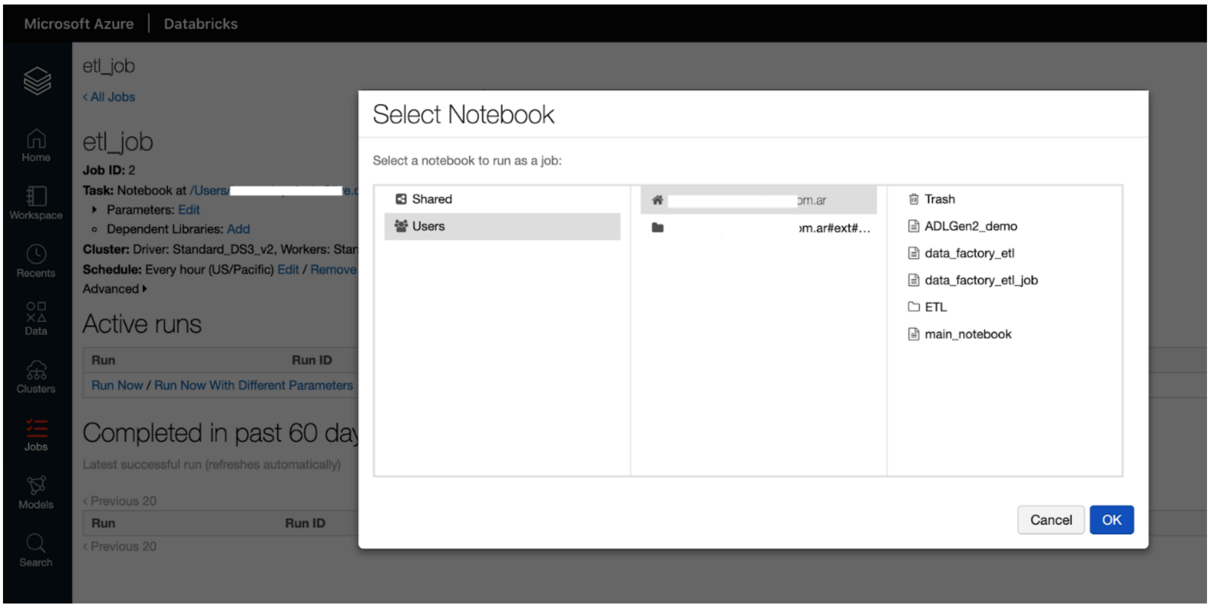
Pipeline run ID

All status

Showing 1 - 1 of 1 items

Activity name	Activity type	Run start ↑↓	Duration	Status	Integration runtime
Notebook1	DatabricksNotel	12/13/20, 1:51:32 AM	00:00:34	✓ Succeeded	DefaultIntegrationRuntime (East US)





etl_job

‹ All Jobs

etl_job

Job ID: 2

Task: Notebook at /Users/[redacted]e.c

- Parameters: Edit
- Dependent Libraries: Add

Cluster: Driver: Standard_DS3_v2, Workers: Star

Schedule: Every hour (US/Pacific) Edit / Remove

Advanced ▸

Active runs

Run	Run ID
Run Now / Run Now With Different Parameters	

Completed in past 60 day

Latest successful run (refreshes automatically)

‹ Previous 20

Run	Run ID

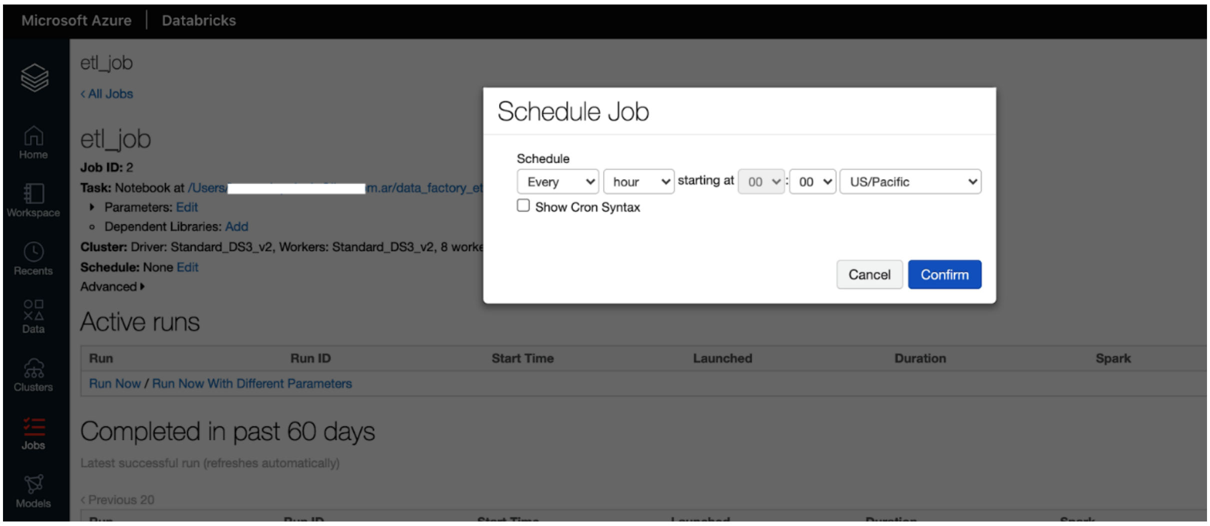
‹ Previous 20

Select Notebook

Select a notebook to run as a job:

Shared	[redacted]om.ar	Trash
Users	[redacted]om.ar#ext#…	ADLGen2_demo
		data_factory_etl
		data_factory_etl_job
		ETL
		main_notebook

Cancel OK



etl_job

‹ All Jobs

etl_job

Job ID: 2

Task: Notebook at /Users/[redacted]m.ar/data_factory_et

- Parameters: Edit
- Dependent Libraries: Add

Cluster: Driver: Standard_DS3_v2, Workers: Standard_DS3_v2, 8 worke

Schedule: None Edit

Advanced ▸

Active runs

Run	Run ID	Start Time	Launched	Duration	Spark
Run Now / Run Now With Different Parameters					

Completed in past 60 days

Latest successful run (refreshes automatically)

‹ Previous 20

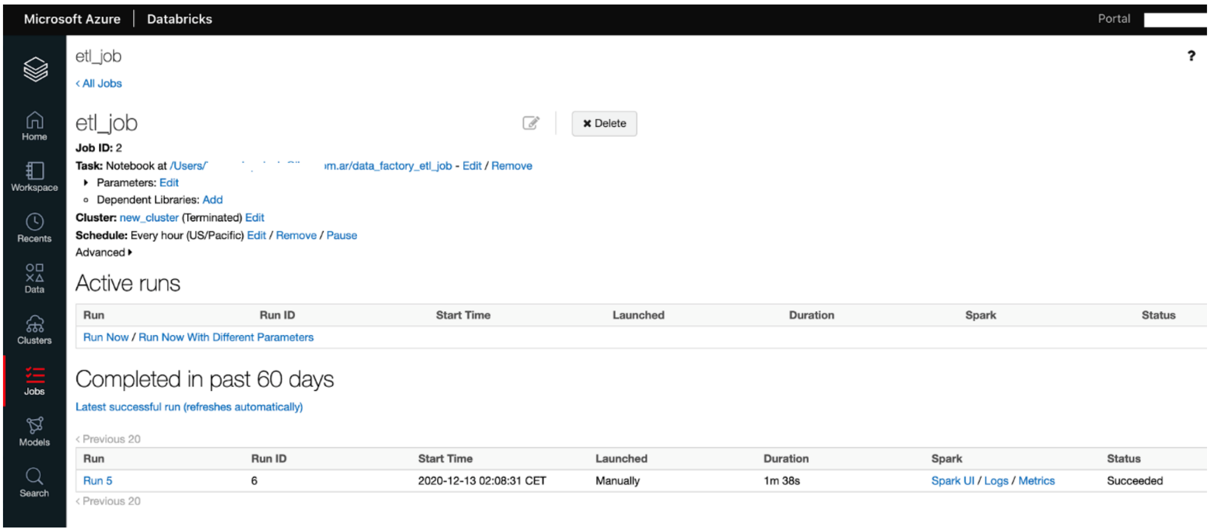
Schedule Job

Schedule

Every hour starting at 00 : 00 US/Pacific

□ Show Cron Syntax

Cancel Confirm



etl_job

‹ All Jobs

etl_job ✖ Delete

Job ID: 2

Task: Notebook at /Users/[redacted]om.ar/data_factory_etl_job - Edit / Remove

- Parameters: Edit
- Dependent Libraries: Add

Cluster: new_cluster (Terminated) Edit

Schedule: Every hour (US/Pacific) Edit / Remove / Pause

Advanced ▸

Active runs

Run	Run ID	Start Time	Launched	Duration	Spark	Status
Run Now / Run Now With Different Parameters						

Completed in past 60 days

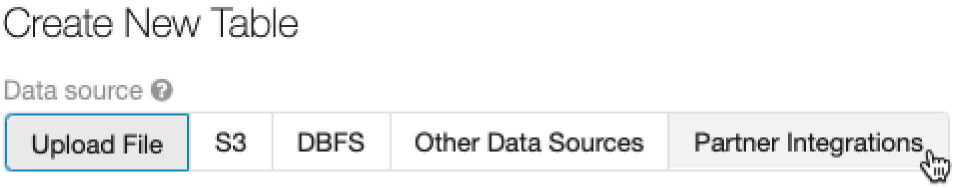
Latest successful run (refreshes automatically)

‹ Previous 20

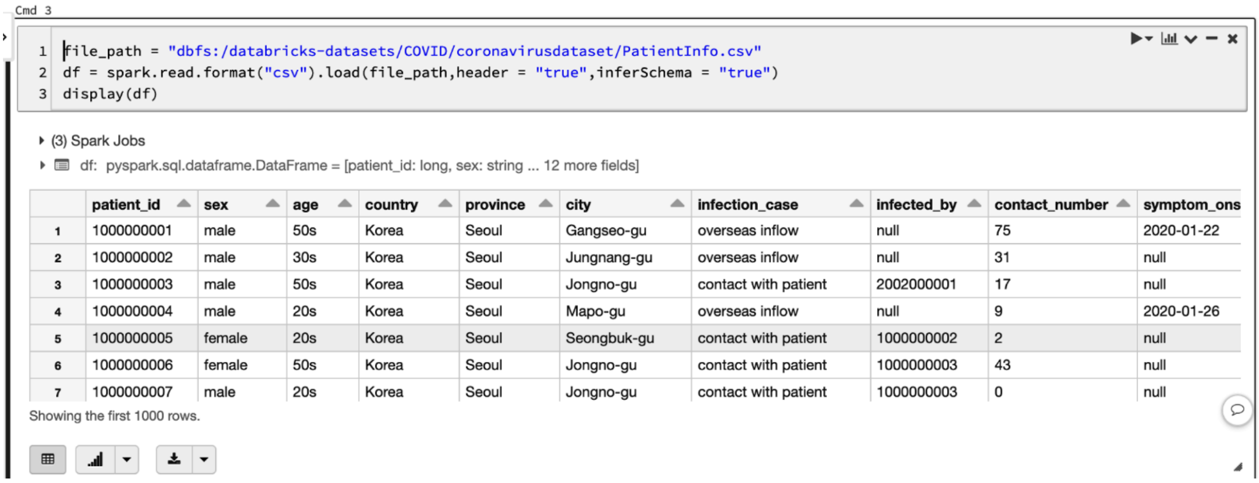
Run	Run ID	Start Time	Launched	Duration	Spark	Status
Run 5	6	2020-12-13 02:08:31 CET	Manually	1m 38s	Spark UI / Logs / Metrics	Succeeded

‹ Previous 20

Chapter 4: Delta Lake with Azure Databricks


Create New Table

Data source


Cmd 3

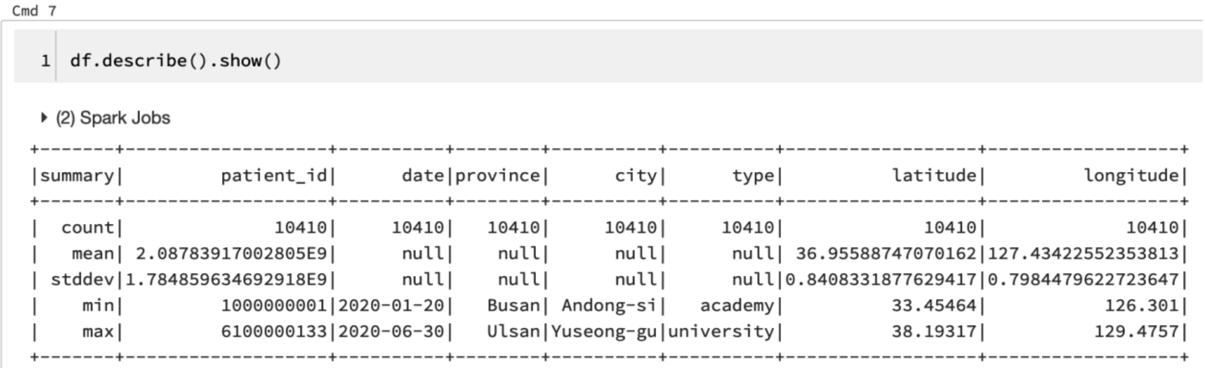
```
file_path = "dbfs:/databricks-datasets/COVID/coronavirusdataset/PatientInfo.csv"
df = spark.read.format("csv").load(file_path,header = "true",inferSchema = "true")
display(df)
```

(3) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [patient_id: long, sex: string ... 12 more fields]

	patient_id	sex	age	country	province	city	infection_case	infected_by	contact_number	symptom_ons
1	1000000001	male	50s	Korea	Seoul	Gangseo-gu	overseas inflow	null	75	2020-01-22
2	1000000002	male	30s	Korea	Seoul	Jungnang-gu	overseas inflow	null	31	null
3	1000000003	male	50s	Korea	Seoul	Jongno-gu	contact with patient	2002000001	17	null
4	1000000004	male	20s	Korea	Seoul	Mapo-gu	overseas inflow	null	9	2020-01-26
5	1000000005	female	20s	Korea	Seoul	Seongbuk-gu	contact with patient	1000000002	2	null
6	1000000006	female	50s	Korea	Seoul	Jongno-gu	contact with patient	1000000003	43	null
7	1000000007	male	20s	Korea	Seoul	Jongno-gu	contact with patient	1000000003	0	null

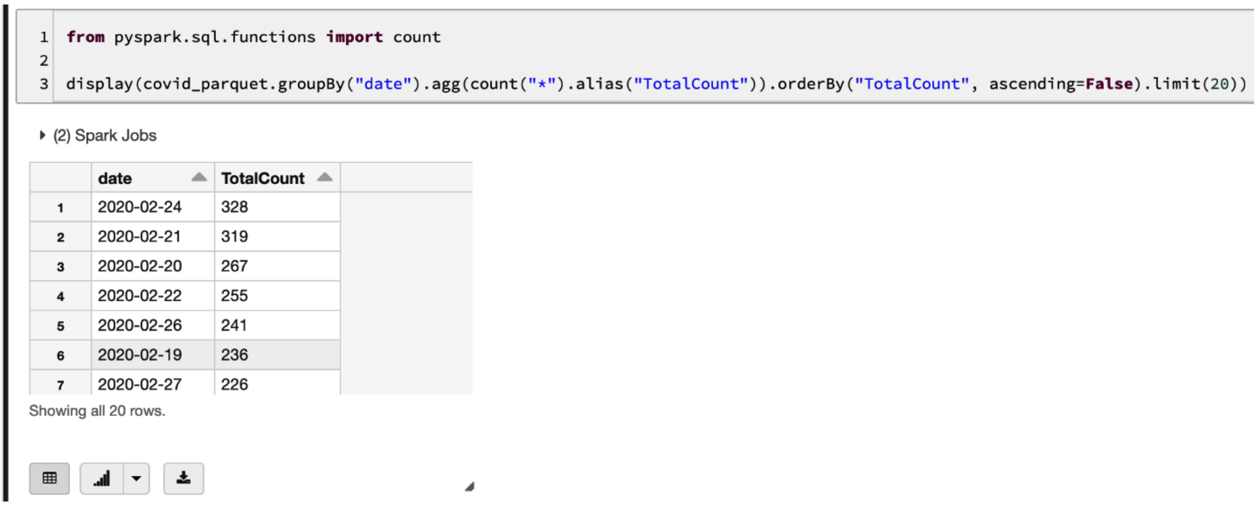
Showing the first 1000 rows.


Cmd 7

```
df.describe().show()
```

(2) Spark Jobs

```
+-------+-------------------+----------+--------+----------+----------+------------------+------------------+
|summary|         patient_id|      date|province|      city|      type|          latitude|         longitude|
+-------+-------------------+----------+--------+----------+----------+------------------+------------------+
|  count|              10410|     10410|   10410|     10410|     10410|             10410|             10410|
|   mean| 2.08783917002805E9|      null|    null|      null|      null| 36.95588747070162|127.43422552353813|
| stddev|1.784859634692918E9|      null|    null|      null|      null|0.8408331877629417|0.7984479622723647|
|    min|         1000000001|2020-01-20|   Busan| Andong-si|   academy|          33.45464|           126.301|
|    max|         6100000133|2020-06-30|   Ulsan|Yuseong-gu|university|          38.19317|          129.4757|
+-------+-------------------+----------+--------+----------+----------+------------------+------------------+
```

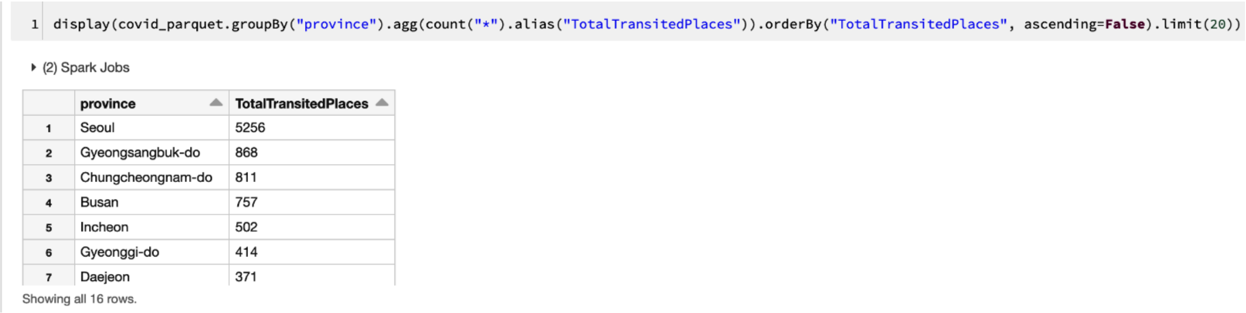


```
from pyspark.sql.functions import count

display(covid_parquet.groupBy("date").agg(count("*").alias("TotalCount")).orderBy("TotalCount", ascending=False).limit(20))
```

(2) Spark Jobs

	date	TotalCount
1	2020-02-24	328
2	2020-02-21	319
3	2020-02-20	267
4	2020-02-22	255
5	2020-02-26	241
6	2020-02-19	236
7	2020-02-27	226

Showing all 20 rows.

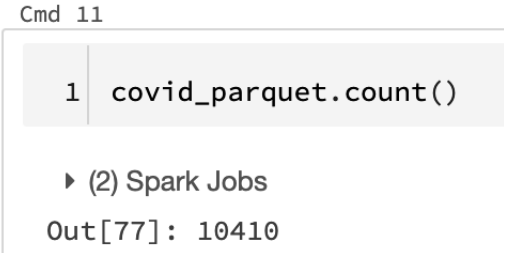


```
display(covid_parquet.groupBy("province").agg(count("*").alias("TotalTransitedPlaces")).orderBy("TotalTransitedPlaces", ascending=False).limit(20))
```

▸ (2) Spark Jobs

	province	TotalTransitedPlaces
1	Seoul	5256
2	Gyeongsangbuk-do	868
3	Chungcheongnam-do	811
4	Busan	757
5	Incheon	502
6	Gyeonggi-do	414
7	Daejeon	371

Showing all 16 rows.

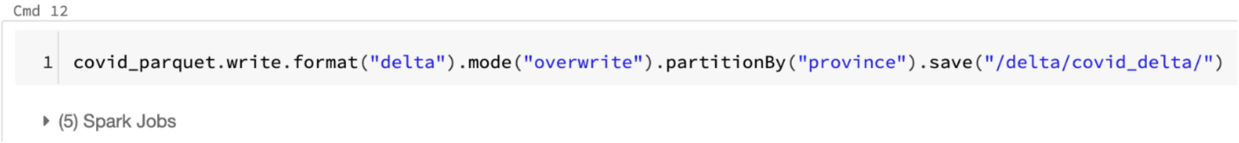


Cmd 11

```
covid_parquet.count()
```

▸ (2) Spark Jobs

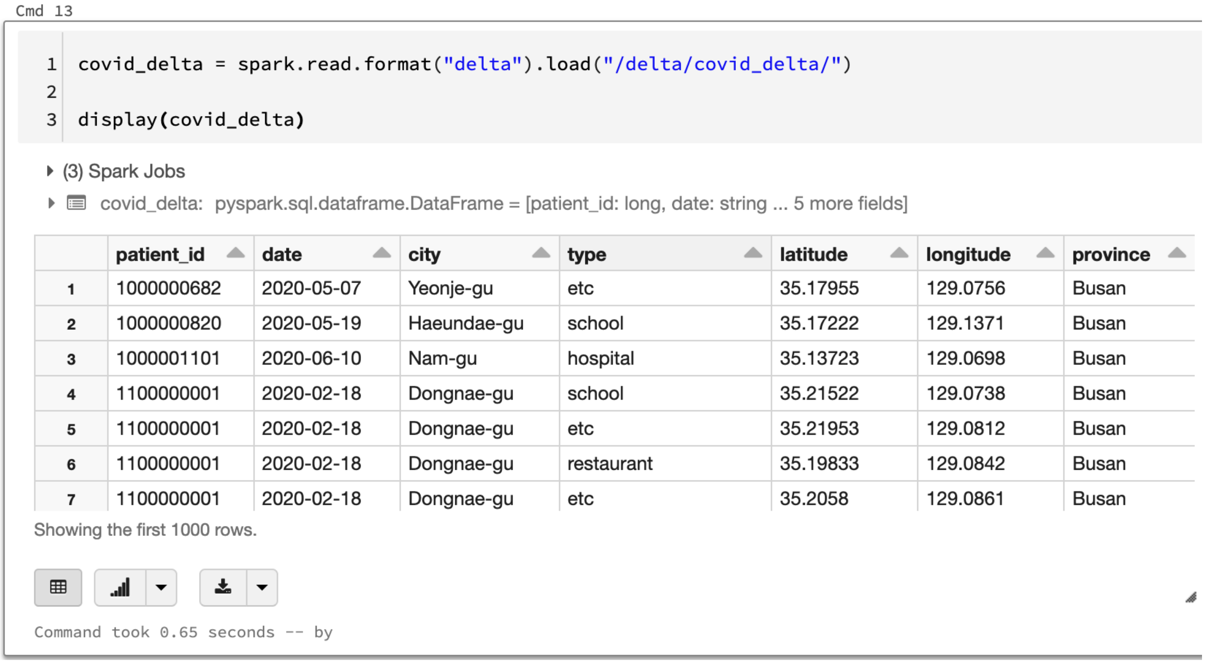
```
Out[77]: 10410
```



Cmd 12

```
covid_parquet.write.format("delta").mode("overwrite").partitionBy("province").save("/delta/covid_delta/")
```

▸ (5) Spark Jobs



Cmd 13

```
covid_delta = spark.read.format("delta").load("/delta/covid_delta/")

display(covid_delta)
```

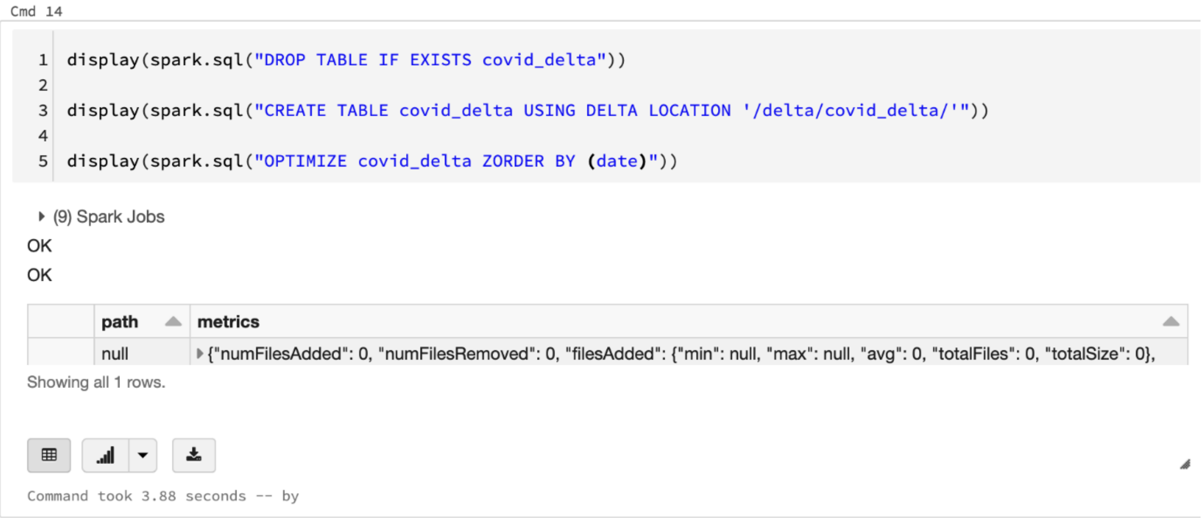
▸ (3) Spark Jobs

▸ covid_delta: pyspark.sql.dataframe.DataFrame = [patient_id: long, date: string ... 5 more fields]

	patient_id	date	city	type	latitude	longitude	province
1	1000000682	2020-05-07	Yeonje-gu	etc	35.17955	129.0756	Busan
2	1000000820	2020-05-19	Haeundae-gu	school	35.17222	129.1371	Busan
3	1000001101	2020-06-10	Nam-gu	hospital	35.13723	129.0698	Busan
4	1100000001	2020-02-18	Dongnae-gu	school	35.21522	129.0738	Busan
5	1100000001	2020-02-18	Dongnae-gu	etc	35.21953	129.0812	Busan
6	1100000001	2020-02-18	Dongnae-gu	restaurant	35.19833	129.0842	Busan
7	1100000001	2020-02-18	Dongnae-gu	etc	35.2058	129.0861	Busan

Showing the first 1000 rows.

```
Command took 0.65 seconds -- by
```



Cmd 14

```
display(spark.sql("DROP TABLE IF EXISTS covid_delta"))

display(spark.sql("CREATE TABLE covid_delta USING DELTA LOCATION '/delta/covid_delta/'"))

display(spark.sql("OPTIMIZE covid_delta ZORDER BY (date)"))
```

▸ (9) Spark Jobs

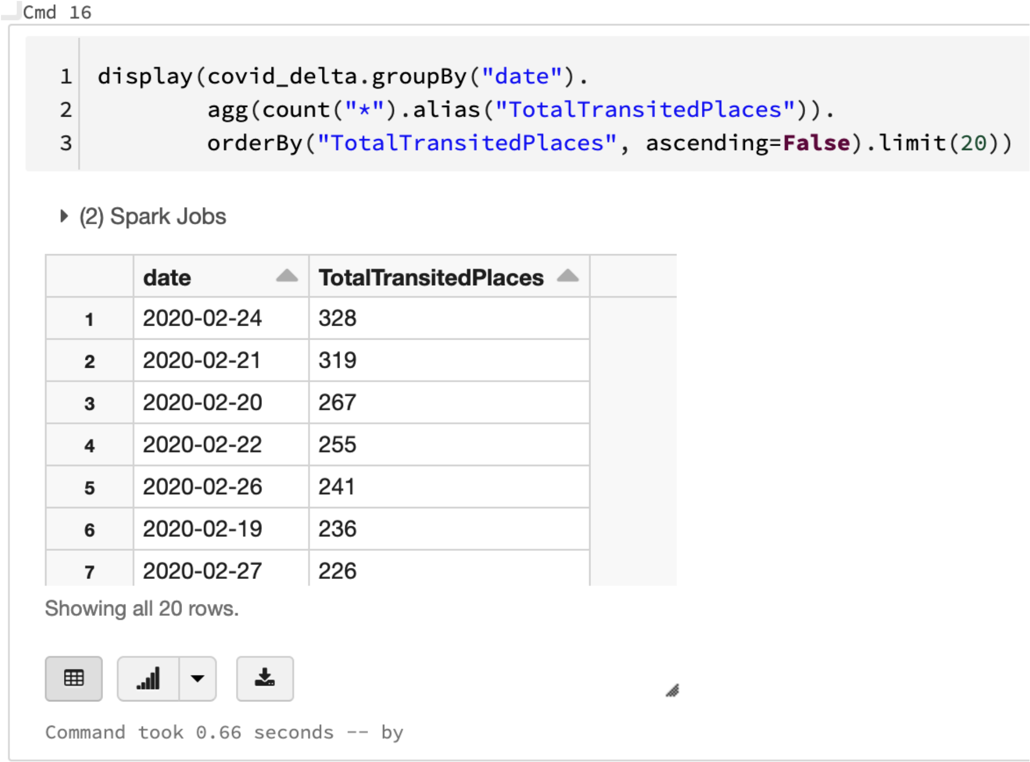
OK

OK

	path	metrics
	null	▸ {"numFilesAdded": 0, "numFilesRemoved": 0, "filesAdded": {"min": null, "max": null, "avg": 0, "totalFiles": 0, "totalSize": 0},

Showing all 1 rows.

Command took 3.88 seconds -- by



Cmd 16

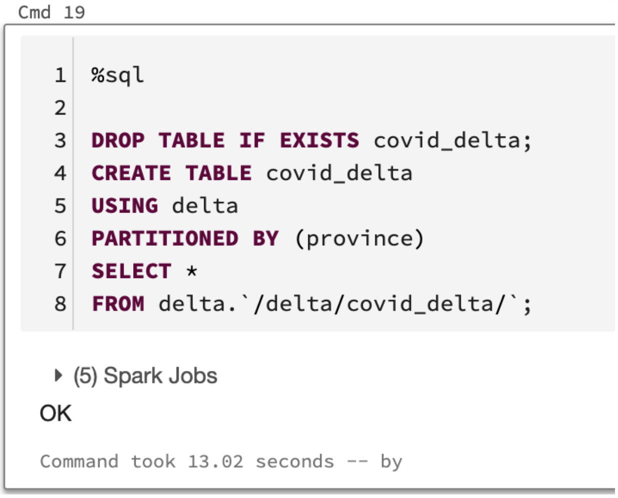
```
display(covid_delta.groupBy("date").
        agg(count("*").alias("TotalTransitedPlaces")).
        orderBy("TotalTransitedPlaces", ascending=False).limit(20))
```

▸ (2) Spark Jobs

	date	TotalTransitedPlaces
1	2020-02-24	328
2	2020-02-21	319
3	2020-02-20	267
4	2020-02-22	255
5	2020-02-26	241
6	2020-02-19	236
7	2020-02-27	226

Showing all 20 rows.

Command took 0.66 seconds -- by



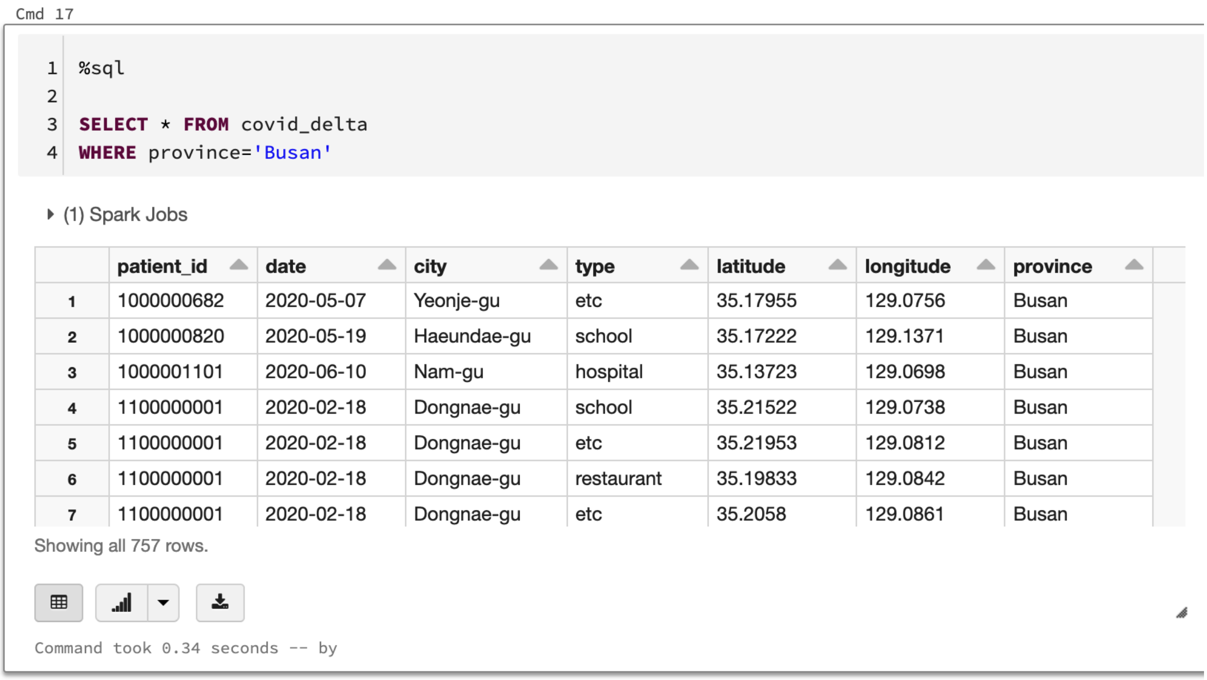
Cmd 19

```
%sql

DROP TABLE IF EXISTS covid_delta;
CREATE TABLE covid_delta
USING delta
PARTITIONED BY (province)
SELECT *
FROM delta.`/delta/covid_delta/`;
```

▸ (5) Spark Jobs

OK

Command took 13.02 seconds -- by



Cmd 17

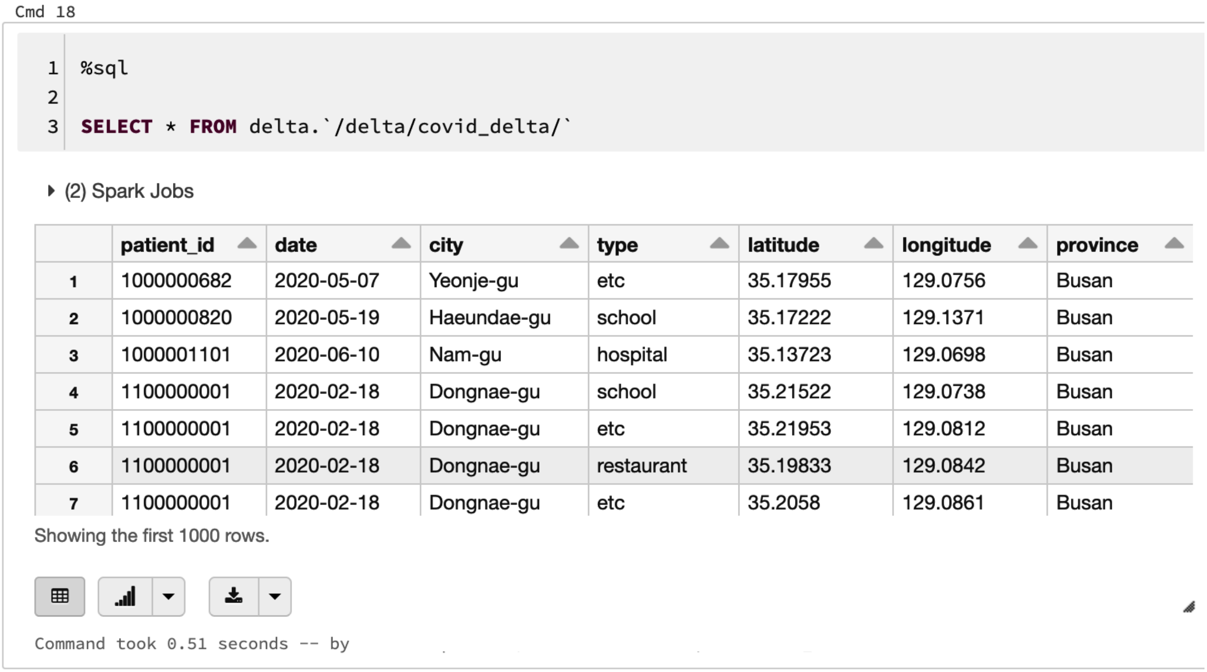
```
%sql

SELECT * FROM covid_delta
WHERE province='Busan'
```

(1) Spark Jobs

	patient_id	date	city	type	latitude	longitude	province
1	1000000682	2020-05-07	Yeonje-gu	etc	35.17955	129.0756	Busan
2	1000000820	2020-05-19	Haeundae-gu	school	35.17222	129.1371	Busan
3	1000001101	2020-06-10	Nam-gu	hospital	35.13723	129.0698	Busan
4	1100000001	2020-02-18	Dongnae-gu	school	35.21522	129.0738	Busan
5	1100000001	2020-02-18	Dongnae-gu	etc	35.21953	129.0812	Busan
6	1100000001	2020-02-18	Dongnae-gu	restaurant	35.19833	129.0842	Busan
7	1100000001	2020-02-18	Dongnae-gu	etc	35.2058	129.0861	Busan

Showing all 757 rows.

```
Command took 0.34 seconds -- by
```



Cmd 18

```
%sql

SELECT * FROM delta.`/delta/covid_delta/`
```

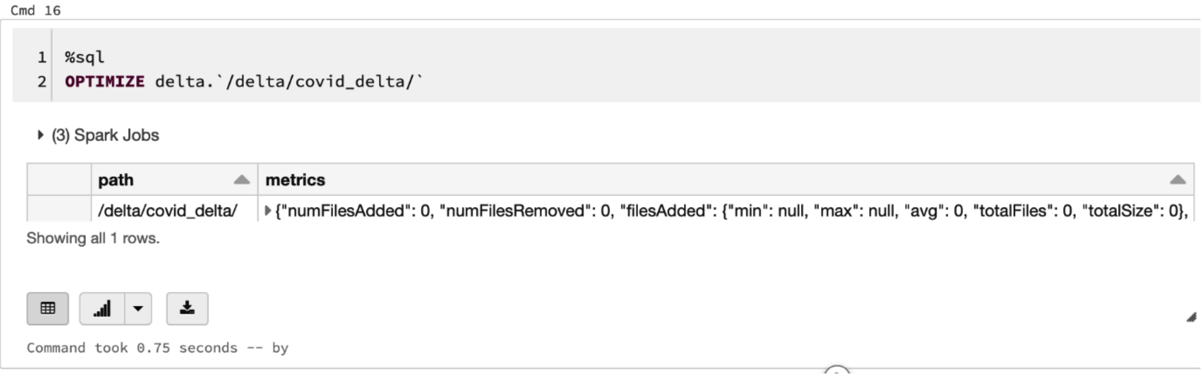
(2) Spark Jobs

	patient_id	date	city	type	latitude	longitude	province
1	1000000682	2020-05-07	Yeonje-gu	etc	35.17955	129.0756	Busan
2	1000000820	2020-05-19	Haeundae-gu	school	35.17222	129.1371	Busan
3	1000001101	2020-06-10	Nam-gu	hospital	35.13723	129.0698	Busan
4	1100000001	2020-02-18	Dongnae-gu	school	35.21522	129.0738	Busan
5	1100000001	2020-02-18	Dongnae-gu	etc	35.21953	129.0812	Busan
6	1100000001	2020-02-18	Dongnae-gu	restaurant	35.19833	129.0842	Busan
7	1100000001	2020-02-18	Dongnae-gu	etc	35.2058	129.0861	Busan

Showing the first 1000 rows.

```
Command took 0.51 seconds -- by
```

Chapter 5: Introducing Delta Engine



Cmd 16

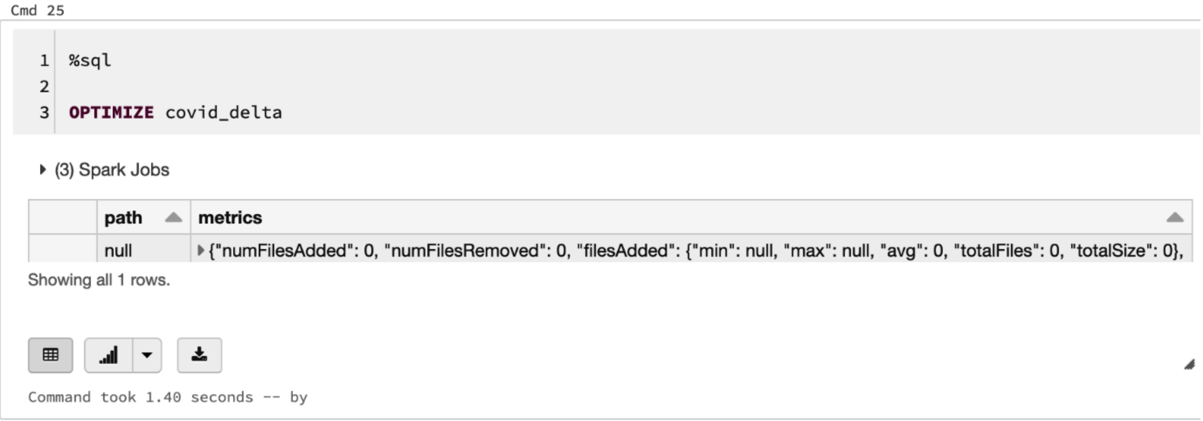
```
%sql
OPTIMIZE delta.`/delta/covid_delta/`
```

▸ (3) Spark Jobs

	path	metrics
	/delta/covid_delta/	▸ {"numFilesAdded": 0, "numFilesRemoved": 0, "filesAdded": {"min": null, "max": null, "avg": 0, "totalFiles": 0, "totalSize": 0},

Showing all 1 rows.

Command took 0.75 seconds -- by



Cmd 25

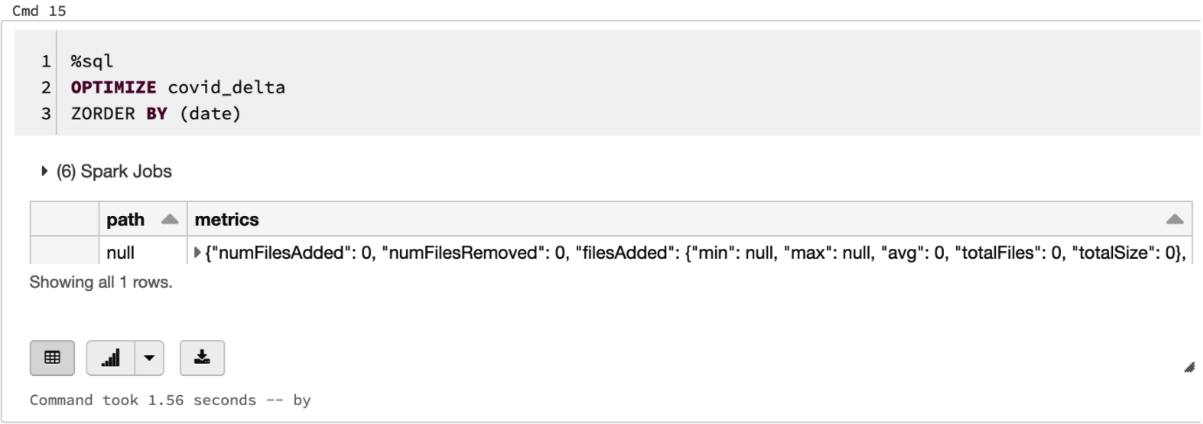
```
%sql

OPTIMIZE covid_delta
```

▸ (3) Spark Jobs

	path	metrics
	null	▸ {"numFilesAdded": 0, "numFilesRemoved": 0, "filesAdded": {"min": null, "max": null, "avg": 0, "totalFiles": 0, "totalSize": 0},

Showing all 1 rows.

Command took 1.40 seconds -- by



Cmd 15

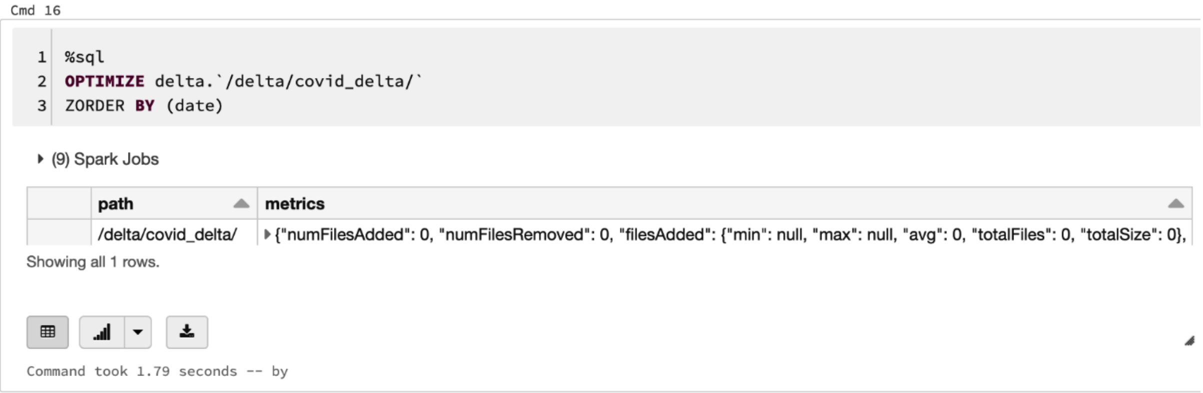
```
%sql
OPTIMIZE covid_delta
ZORDER BY (date)
```

▸ (6) Spark Jobs

	path	metrics
	null	▸ {"numFilesAdded": 0, "numFilesRemoved": 0, "filesAdded": {"min": null, "max": null, "avg": 0, "totalFiles": 0, "totalSize": 0},

Showing all 1 rows.

Command took 1.56 seconds -- by



Cmd 16

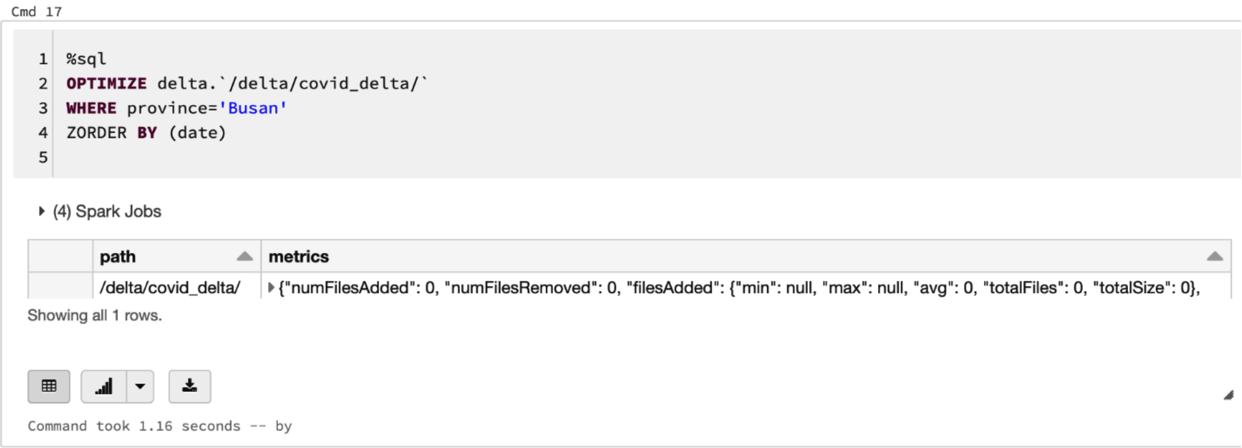
```
%sql
OPTIMIZE delta.`/delta/covid_delta/`
ZORDER BY (date)
```

▸ (9) Spark Jobs

	path	metrics
	/delta/covid_delta/	▸ {"numFilesAdded": 0, "numFilesRemoved": 0, "filesAdded": {"min": null, "max": null, "avg": 0, "totalFiles": 0, "totalSize": 0},

Showing all 1 rows.

Command took 1.79 seconds -- by



Cmd 17

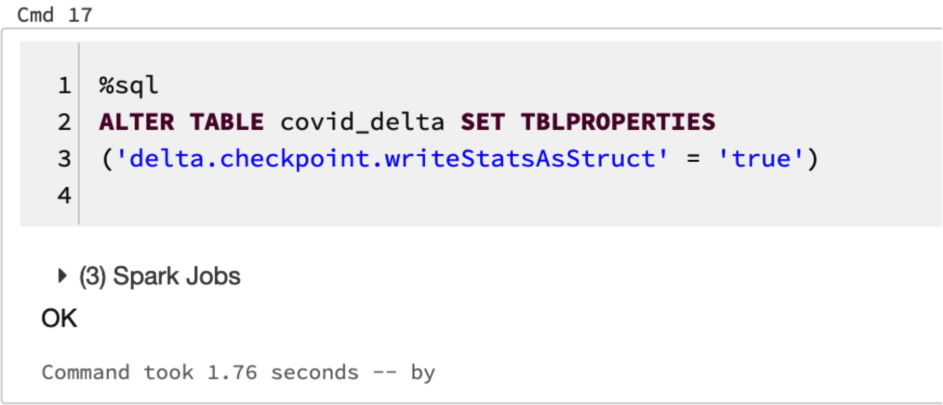
```
%sql
OPTIMIZE delta.`/delta/covid_delta/`
WHERE province='Busan'
ZORDER BY (date)

```

(4) Spark Jobs

	path	metrics
	/delta/covid_delta/	{"numFilesAdded": 0, "numFilesRemoved": 0, "filesAdded": {"min": null, "max": null, "avg": 0, "totalFiles": 0, "totalSize": 0},

Showing all 1 rows.

```
Command took 1.16 seconds -- by
```



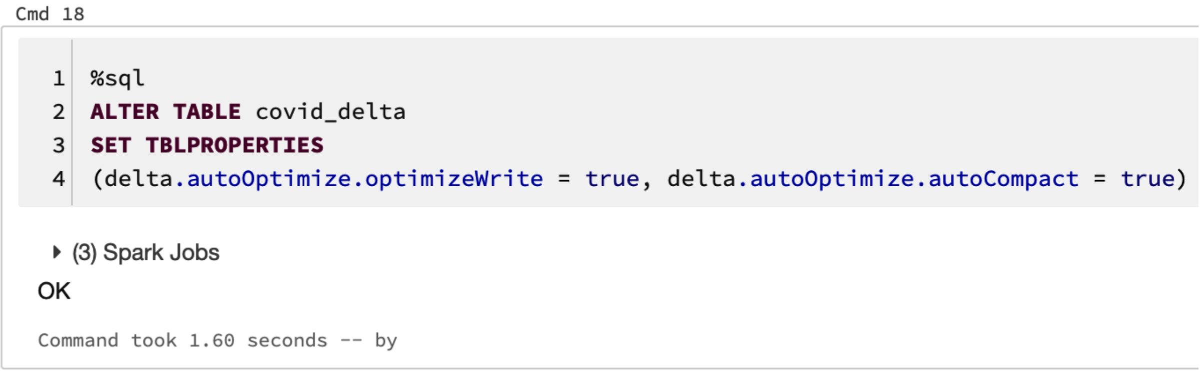
Cmd 17

```
%sql
ALTER TABLE covid_delta SET TBLPROPERTIES
('delta.checkpoint.writeStatsAsStruct' = 'true')

```

(3) Spark Jobs

OK

```
Command took 1.76 seconds -- by
```



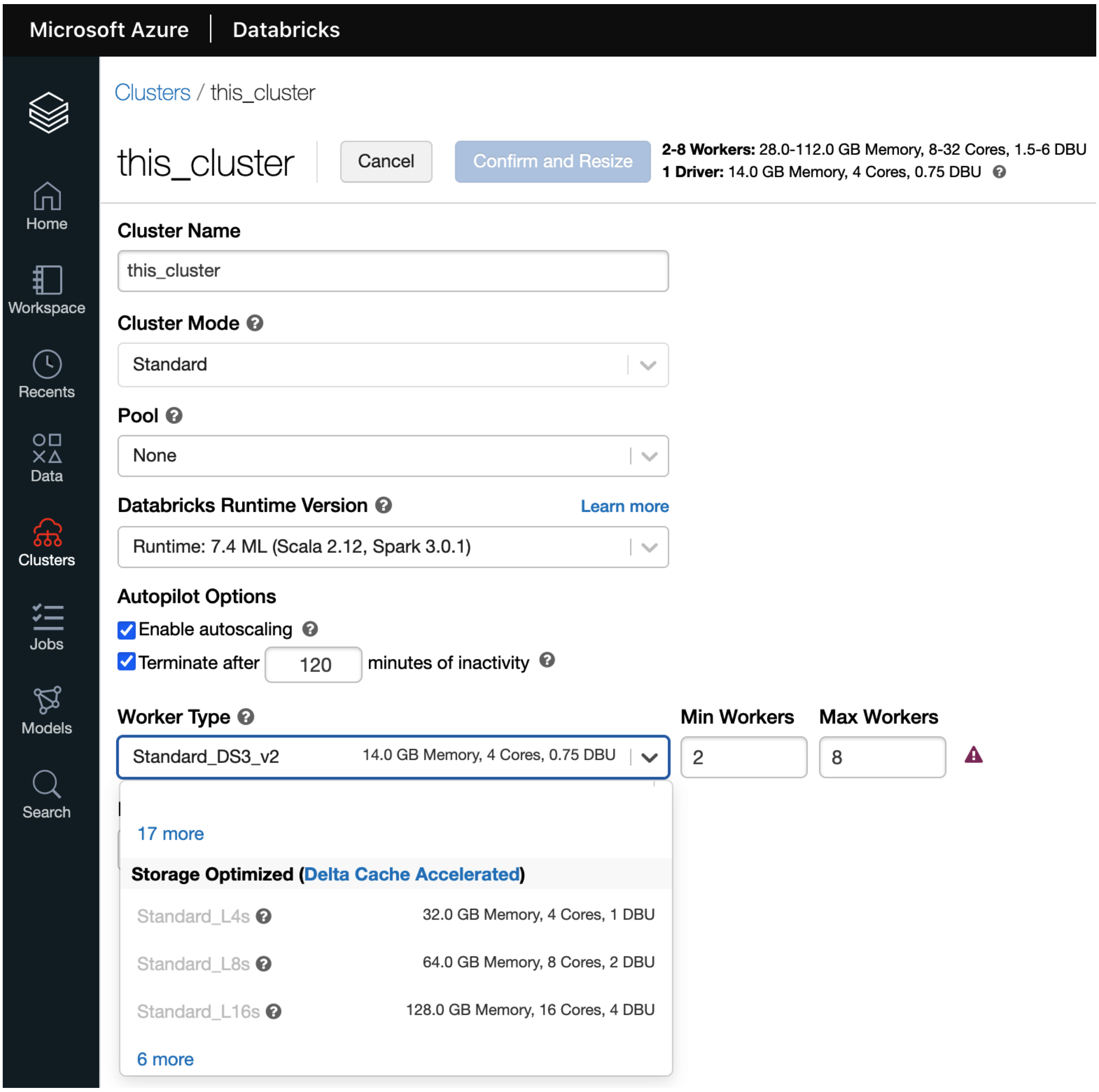
Cmd 18

```
%sql
ALTER TABLE covid_delta
SET TBLPROPERTIES
(delta.autoOptimize.optimizeWrite = true, delta.autoOptimize.autoCompact = true)
```

(3) Spark Jobs

OK

```
Command took 1.60 seconds -- by
```



this_cluster

2-8 Workers: 28.0-112.0 GB Memory, 8-32 Cores, 1.5-6 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU

Cluster Name: this_cluster

Cluster Mode: Standard

Pool: None

Databricks Runtime Version: Runtime: 7.4 ML (Scala 2.12, Spark 3.0.1)

Autopilot Options

☑ Enable autoscaling

☑ Terminate after 120 minutes of inactivity

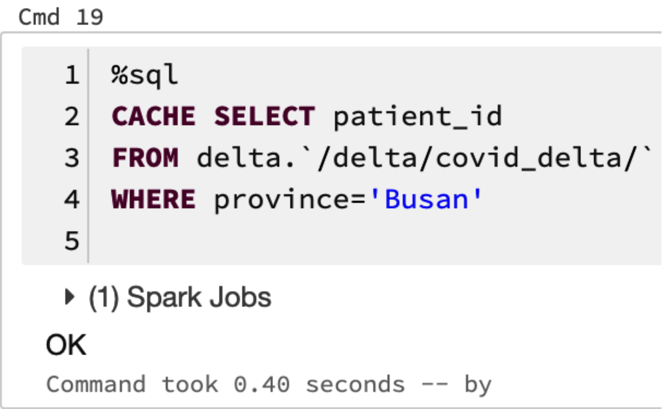
Worker Type: Standard_DS3_v2 14.0 GB Memory, 4 Cores, 0.75 DBU

Min Workers: 2

Max Workers: 8

Storage Optimized (Delta Cache Accelerated)

Standard_L4s	32.0 GB Memory, 4 Cores, 1 DBU
Standard_L8s	64.0 GB Memory, 8 Cores, 2 DBU
Standard_L16s	128.0 GB Memory, 16 Cores, 4 DBU



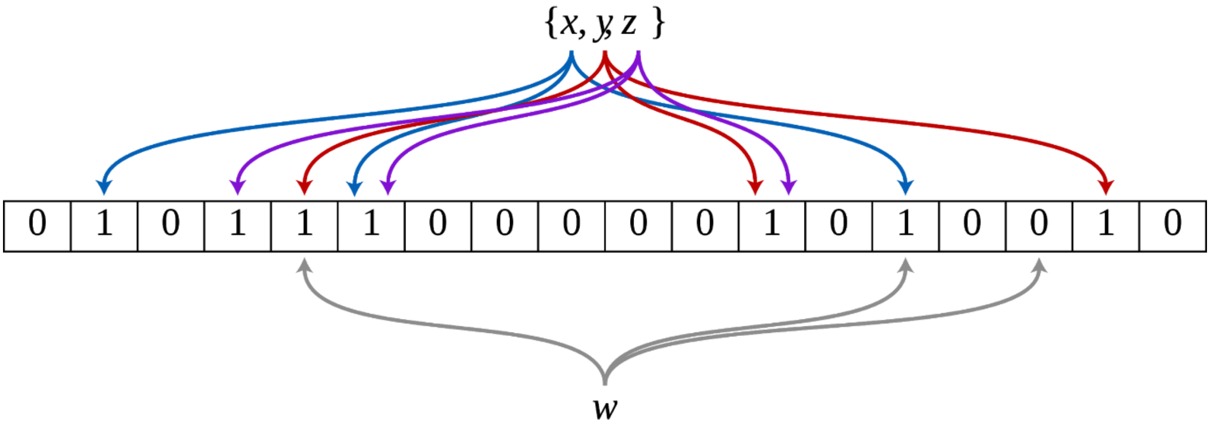
Cmd 19

```
%sql
CACHE SELECT patient_id
FROM delta.`/delta/covid_delta/`
WHERE province='Busan'
```

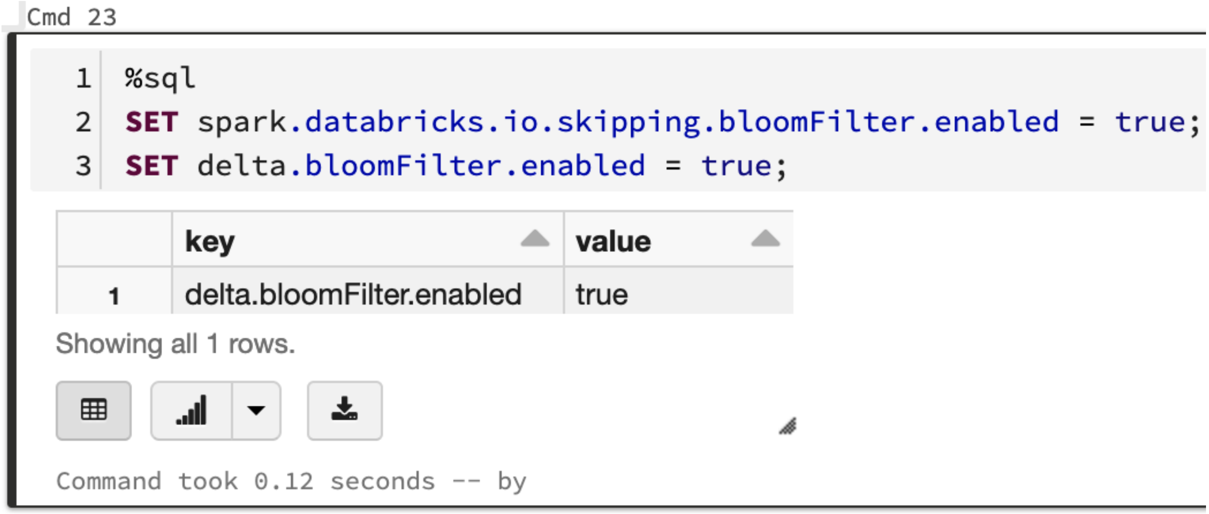
▸ (1) Spark Jobs

OK

Command took 0.40 seconds -- by



0	1	0	1	1	1	0	0	0	0	0	1	0	1	0	0	1	0

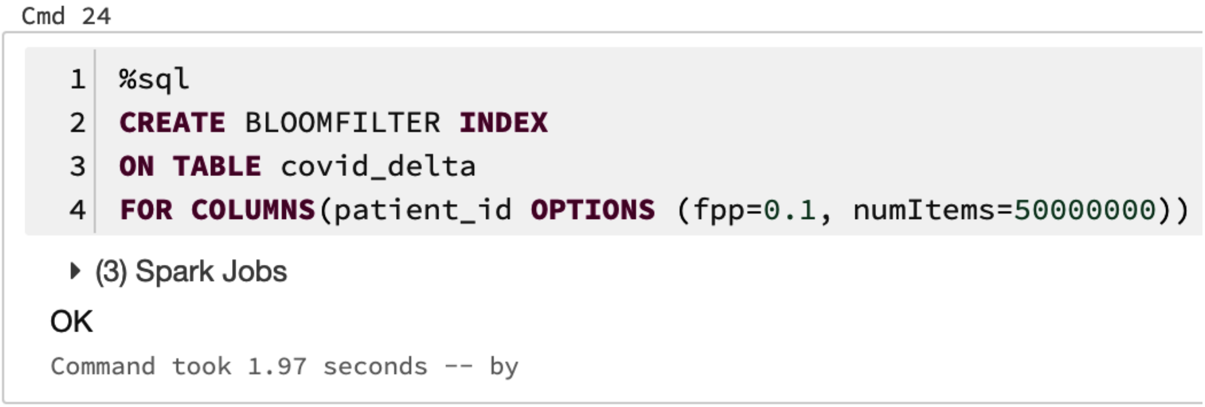

Cmd 23

```
%sql
SET spark.databricks.io.skipping.bloomFilter.enabled = true;
SET delta.bloomFilter.enabled = true;
```

	key	value
1	delta.bloomFilter.enabled	true

Showing all 1 rows.

Command took 0.12 seconds -- by


Cmd 24

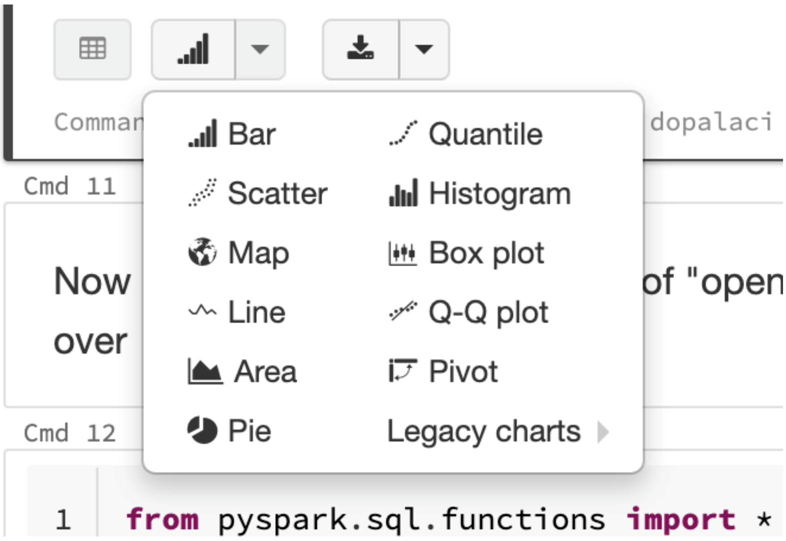
```
%sql
CREATE BLOOMFILTER INDEX
ON TABLE covid_delta
FOR COLUMNS(patient_id OPTIONS (fpp=0.1, numItems=50000000))
```

(3) Spark Jobs

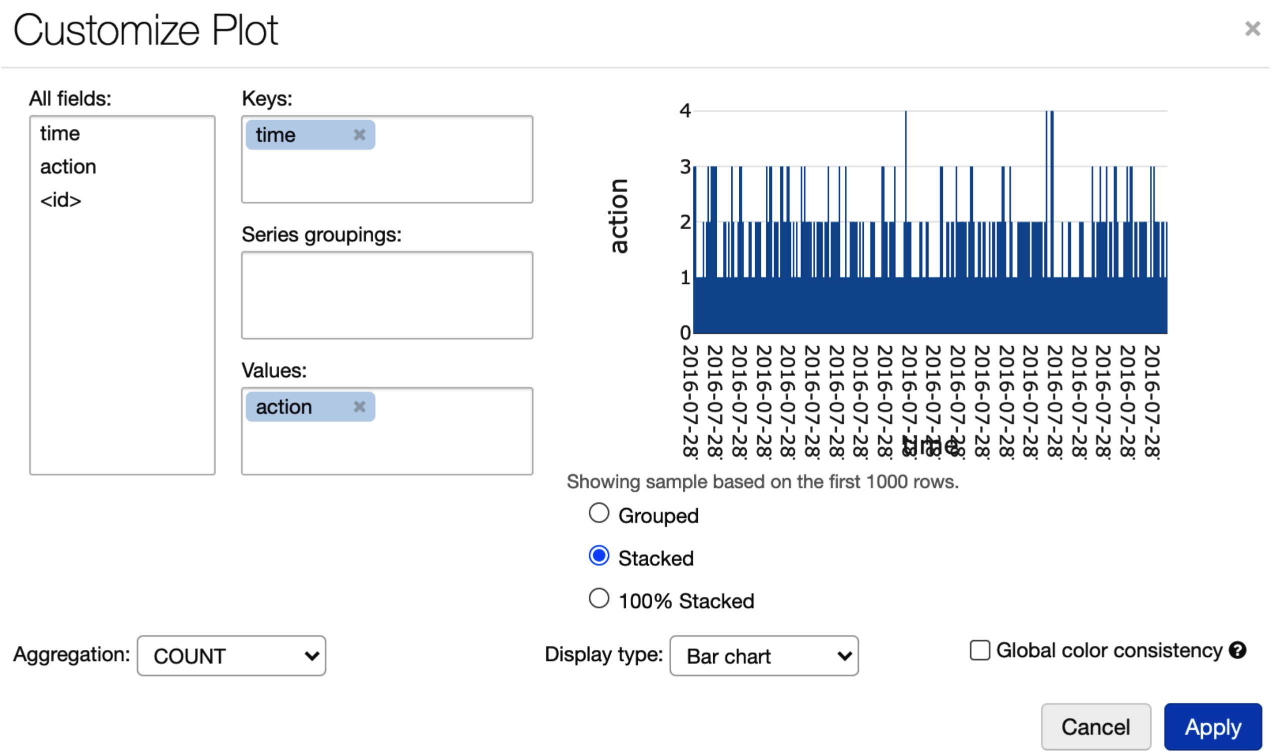
OK

Command took 1.97 seconds -- by

Chapter 6: Introducing Structured Streaming



```
from pyspark.sql.functions import *
```



Customize Plot

All fields:

- time
- action
- <id>

Keys: time

Series groupings:

Values: action

Showing sample based on the first 1000 rows.

○ Grouped

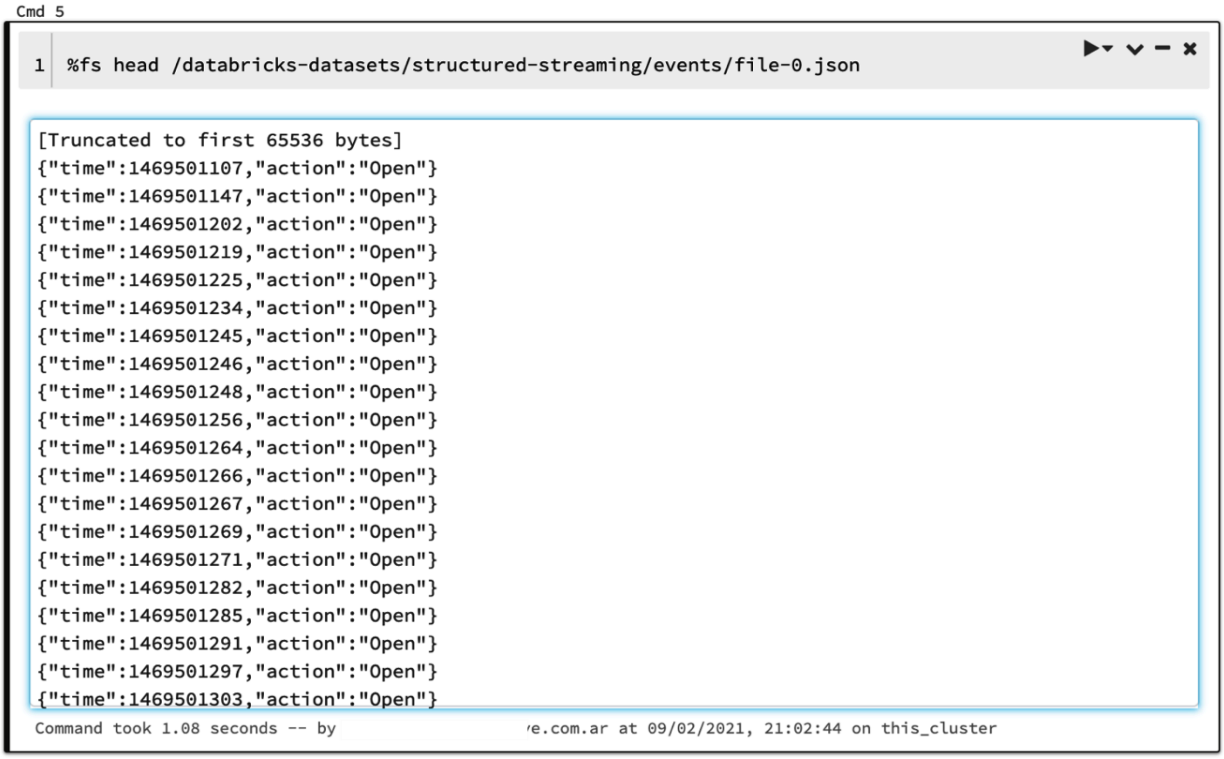
● Stacked

○ 100% Stacked

Aggregation: COUNT

Display type: Bar chart

□ Global color consistency

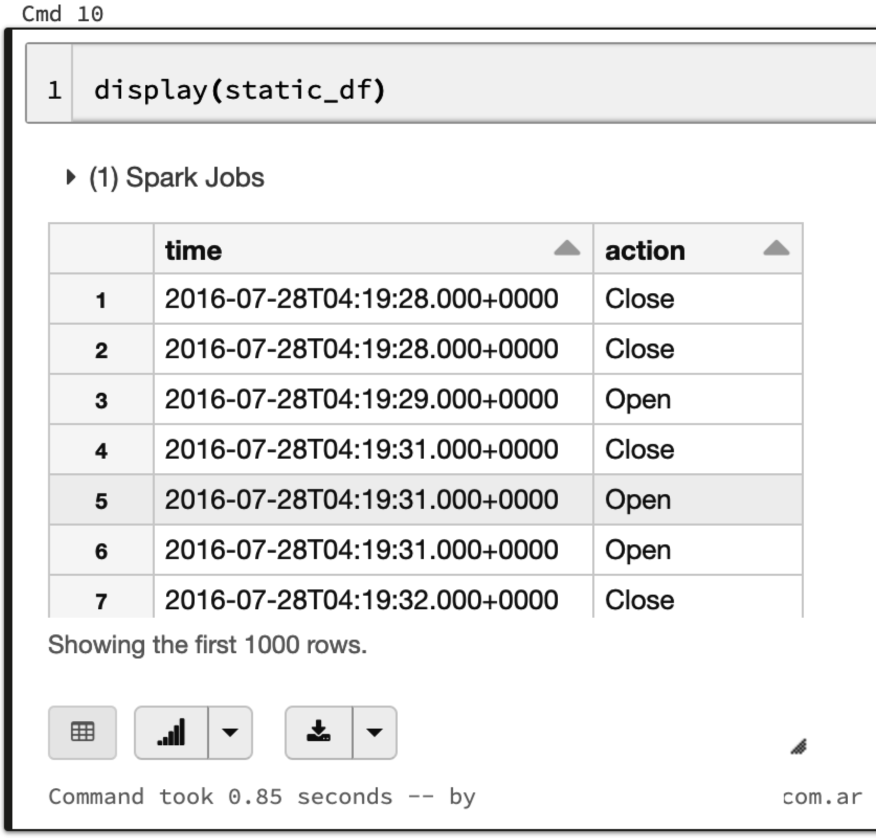


Cmd 5

```
%fs head /databricks-datasets/structured-streaming/events/file-0.json
```

```
[Truncated to first 65536 bytes]
{"time":1469501107,"action":"Open"}
{"time":1469501147,"action":"Open"}
{"time":1469501202,"action":"Open"}
{"time":1469501219,"action":"Open"}
{"time":1469501225,"action":"Open"}
{"time":1469501234,"action":"Open"}
{"time":1469501245,"action":"Open"}
{"time":1469501246,"action":"Open"}
{"time":1469501248,"action":"Open"}
{"time":1469501256,"action":"Open"}
{"time":1469501264,"action":"Open"}
{"time":1469501266,"action":"Open"}
{"time":1469501267,"action":"Open"}
{"time":1469501269,"action":"Open"}
{"time":1469501271,"action":"Open"}
{"time":1469501282,"action":"Open"}
{"time":1469501285,"action":"Open"}
{"time":1469501291,"action":"Open"}
{"time":1469501297,"action":"Open"}
{"time":1469501303,"action":"Open"}
```

```
Command took 1.08 seconds -- by                    'e.com.ar at 09/02/2021, 21:02:44 on this_cluster
```



Cmd 10

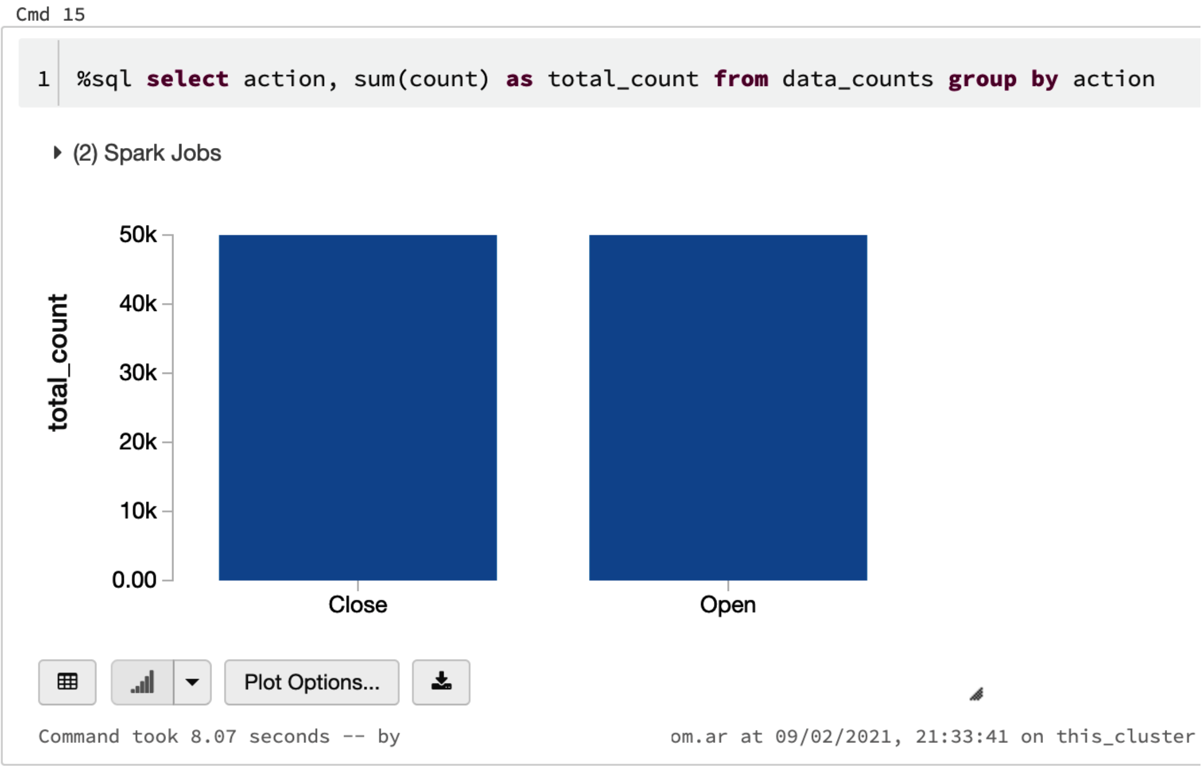
```
display(static_df)
```

▸ (1) Spark Jobs

	time	action
1	2016-07-28T04:19:28.000+0000	Close
2	2016-07-28T04:19:28.000+0000	Close
3	2016-07-28T04:19:29.000+0000	Open
4	2016-07-28T04:19:31.000+0000	Close
5	2016-07-28T04:19:31.000+0000	Open
6	2016-07-28T04:19:31.000+0000	Open
7	2016-07-28T04:19:32.000+0000	Close

Showing the first 1000 rows.

```
Command took 0.85 seconds -- by                    com.ar
```

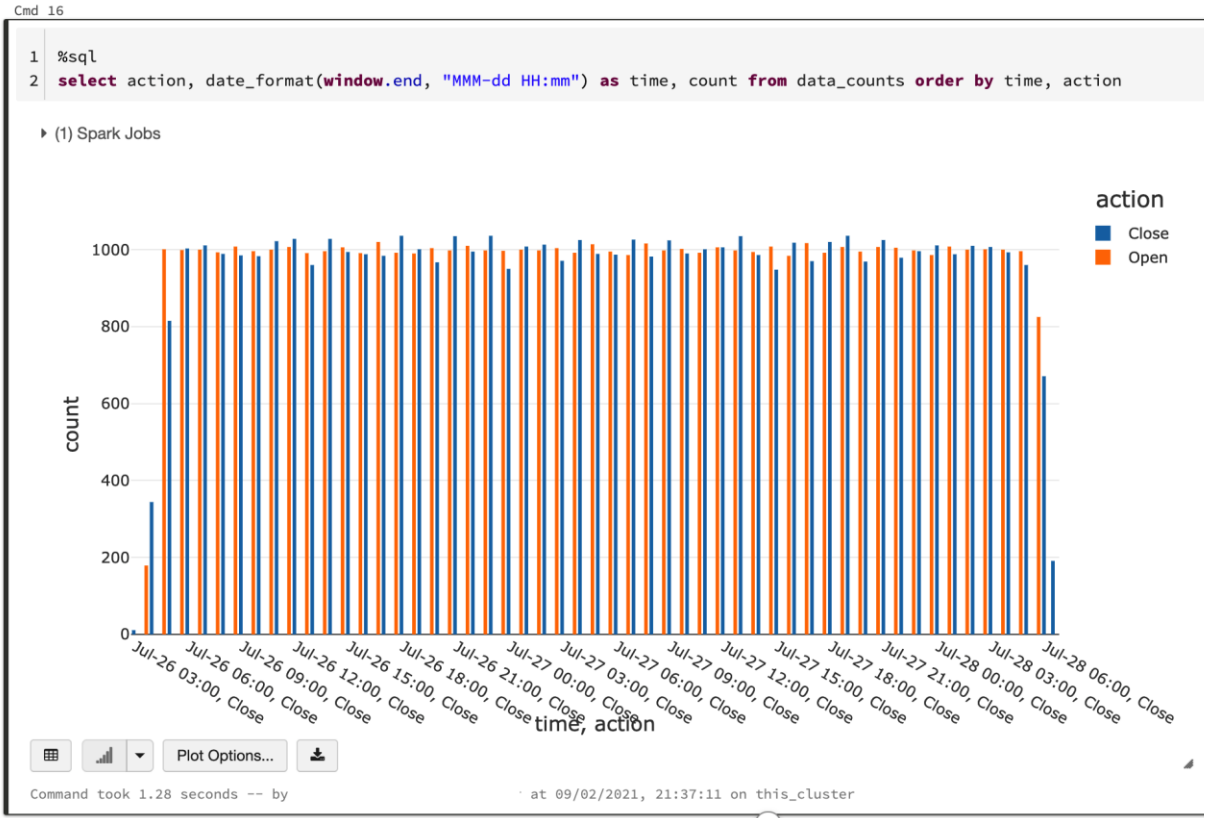


Cmd 15

```
%sql select action, sum(count) as total_count from data_counts group by action
```

▸ (2) Spark Jobs

Command took 8.07 seconds -- by om.ar at 09/02/2021, 21:33:41 on this_cluster

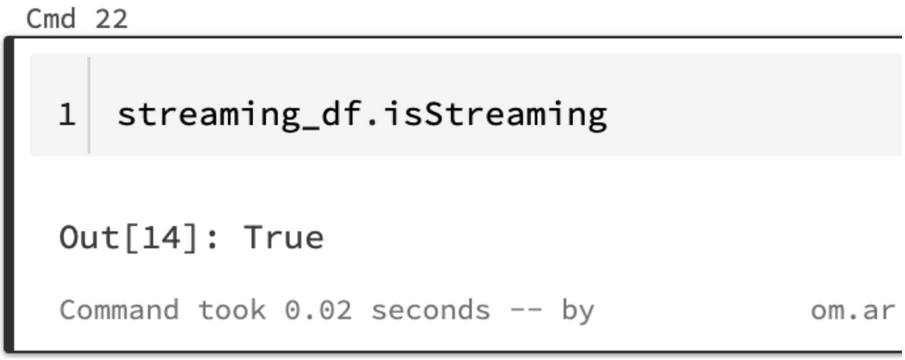


Cmd 16

```
%sql
select action, date_format(window.end, "MMM-dd HH:mm") as time, count from data_counts order by time, action
```

▸ (1) Spark Jobs

Command took 1.28 seconds -- by · at 09/02/2021, 21:37:11 on this_cluster

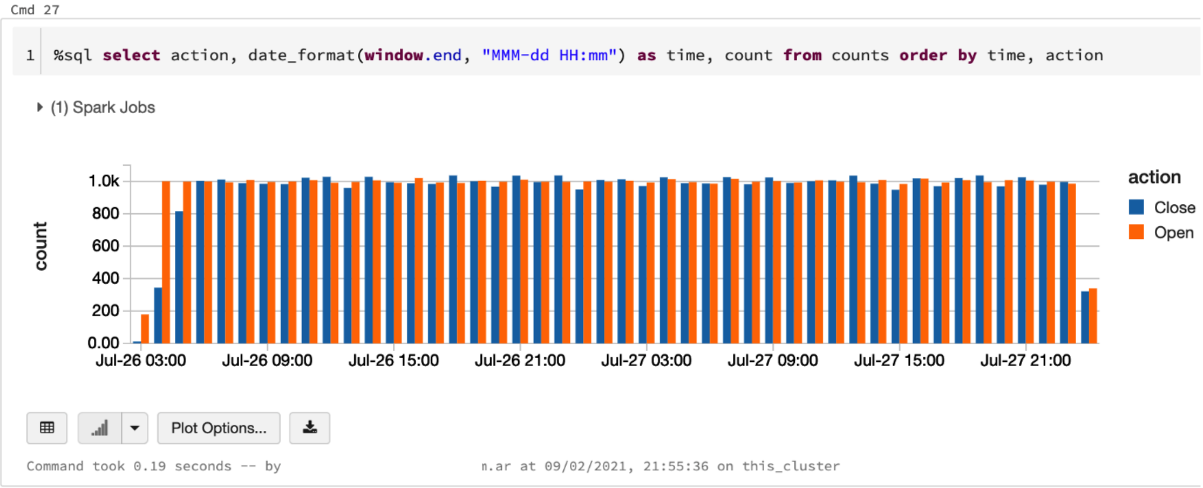


Cmd 22

```
streaming_df.isStreaming
```

```
Out[14]: True
```

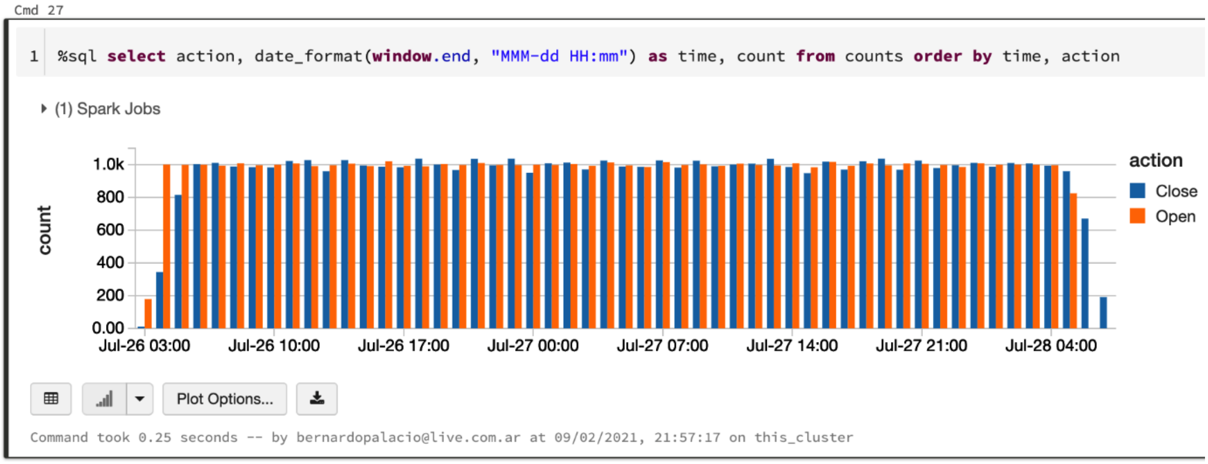
```
Command took 0.02 seconds -- by          om.ar
```



Cmd 27

```
%sql select action, date_format(window.end, "MMM-dd HH:mm") as time, count from counts order by time, action
```

```
Command took 0.19 seconds -- by                    n.ar at 09/02/2021, 21:55:36 on this_cluster
```

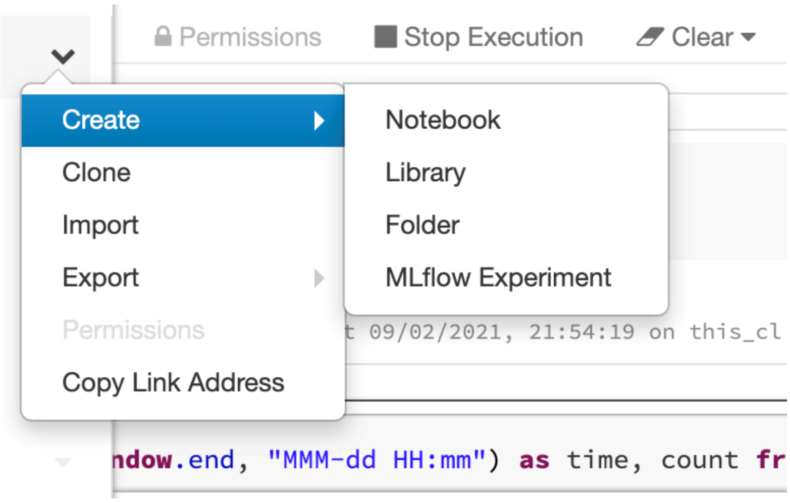


Cmd 27

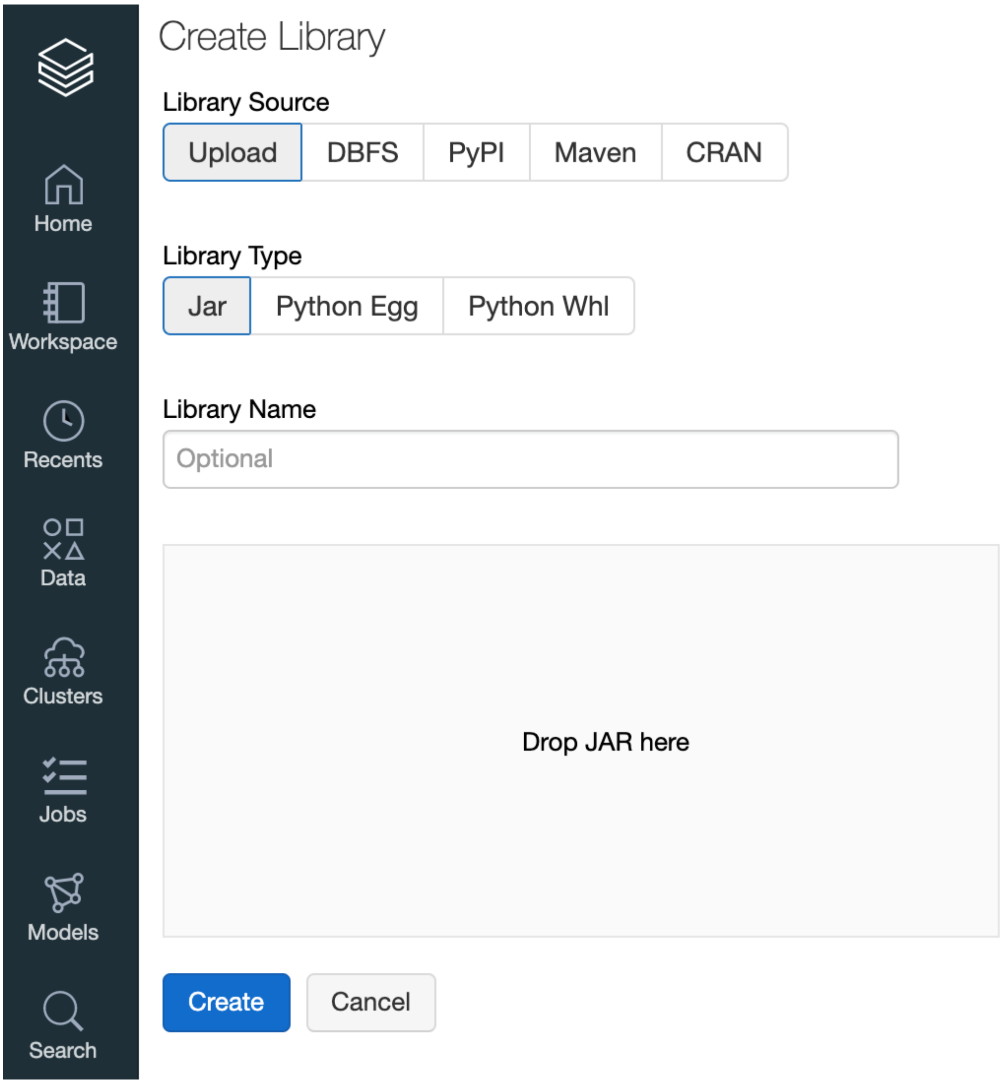
```
%sql select action, date_format(window.end, "MMM-dd HH:mm") as time, count from counts order by time, action
```

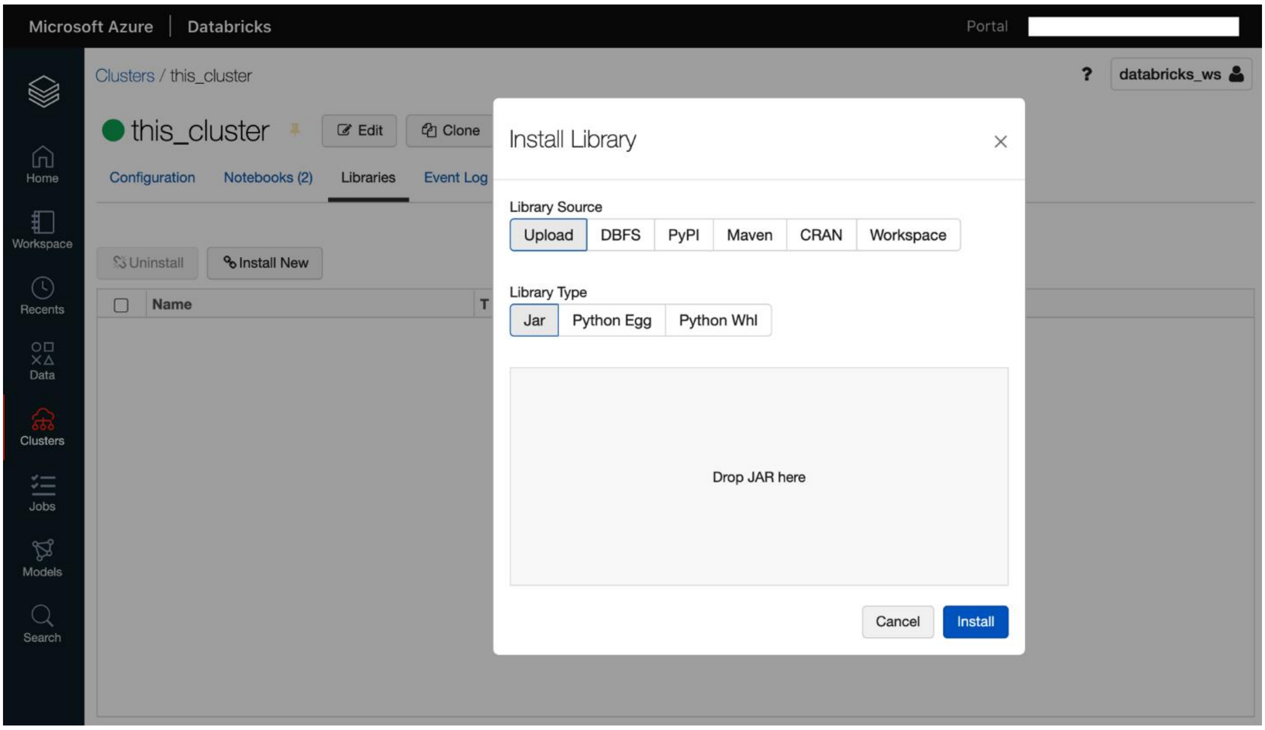
```
Command took 0.25 seconds -- by bernardopalacio@live.com.ar at 09/02/2021, 21:57:17 on this_cluster
```

Chapter 7: Using Python Libraries in Azure Databricks



```
t 09/02/2021, 21:54:19 on this_cl
ndow.end, "MMM-dd HH:mm") as time, count fr
```





Install Library

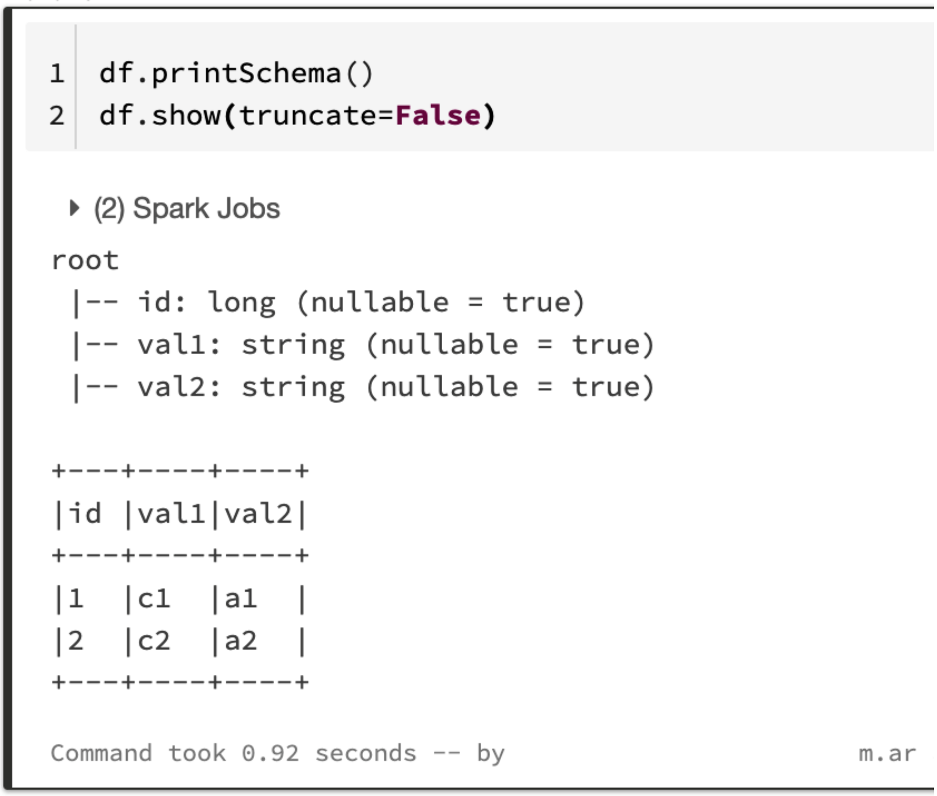
Library Source

Upload | DBFS | PyPI | Maven | CRAN | Workspace

Library Type

Jar | Python Egg | Python Whl

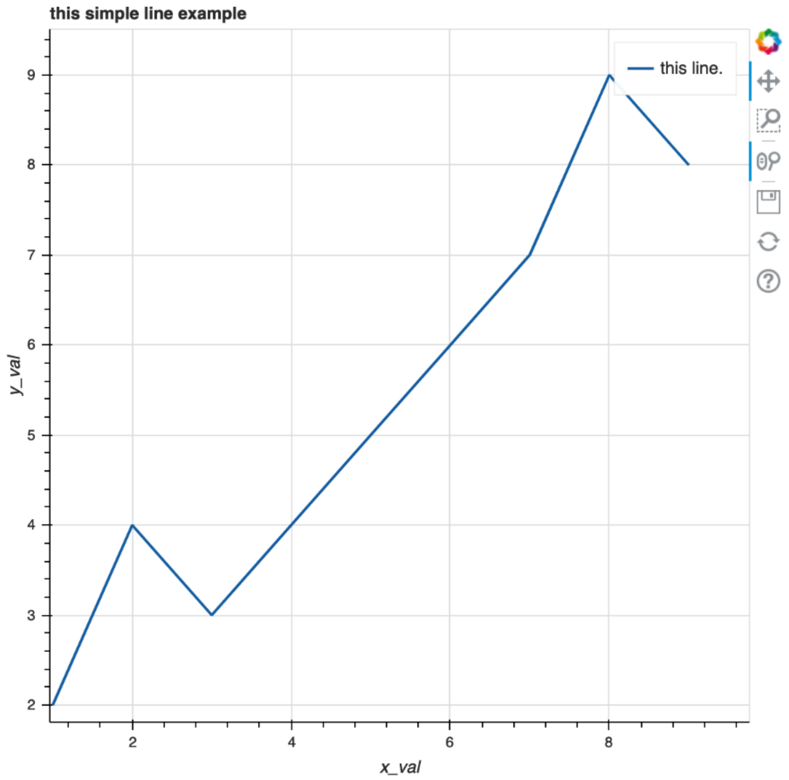
Drop JAR here

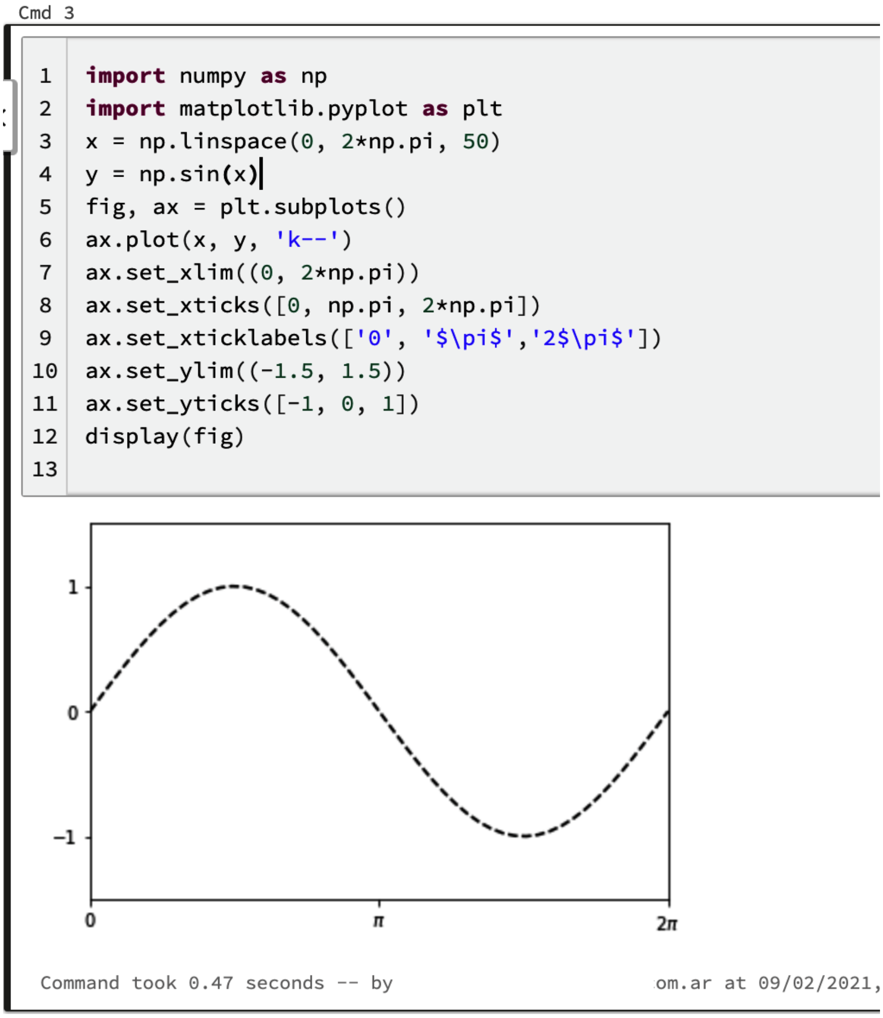


```
df.printSchema()
df.show(truncate=False)
```

(2) Spark Jobs

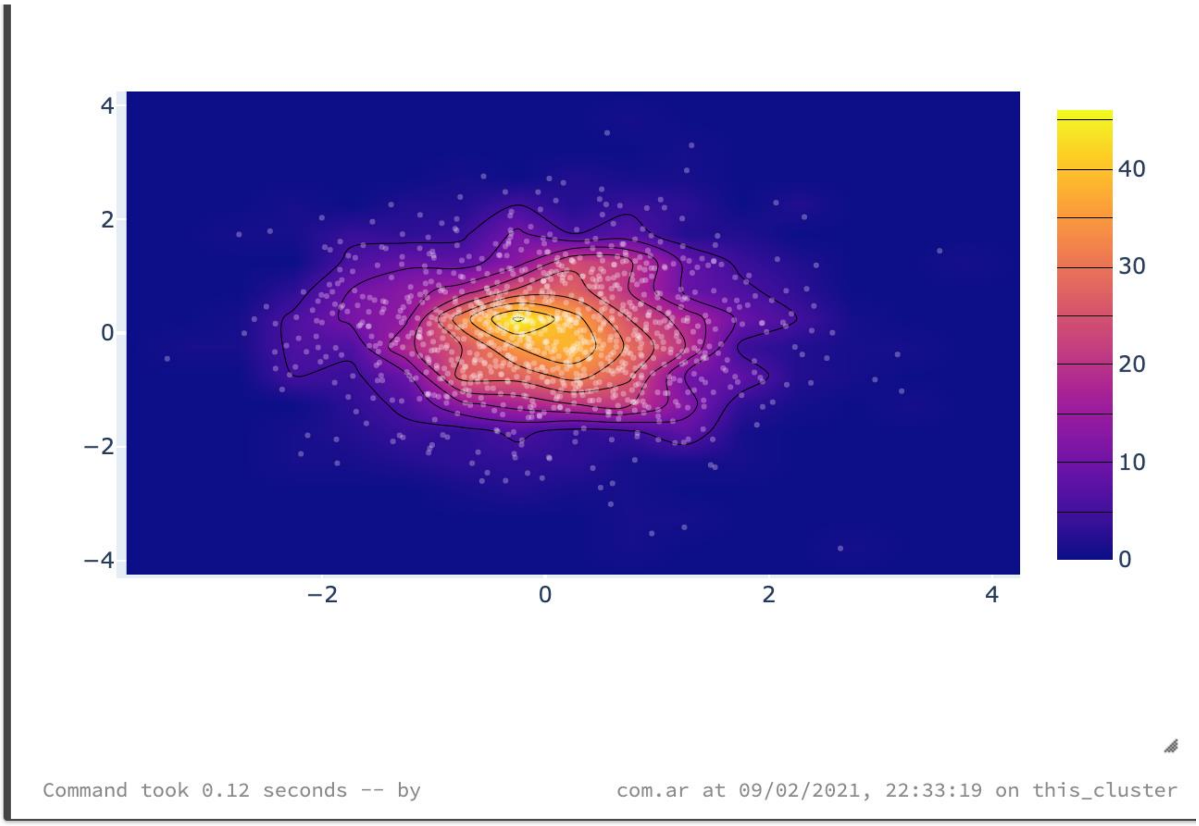
```
root
 |-- id: long (nullable = true)
 |-- val1: string (nullable = true)
 |-- val2: string (nullable = true)

+---+----+----+
|id |val1|val2|
+---+----+----+
|1  |c1  |a1  |
|2  |c2  |a2  |
+---+----+----+

Command took 0.92 seconds -- by               m.ar
```





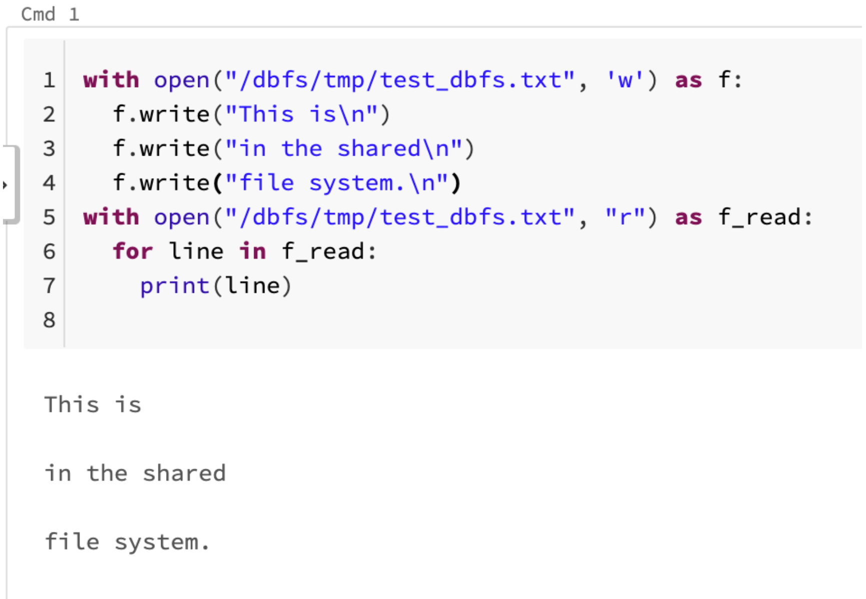
```
import numpy as np
import matplotlib.pyplot as plt
x = np.linspace(0, 2*np.pi, 50)
y = np.sin(x)
fig, ax = plt.subplots()
ax.plot(x, y, 'k--')
ax.set_xlim((0, 2*np.pi))
ax.set_xticks([0, np.pi, 2*np.pi])
ax.set_xticklabels(['0', '$\pi$','2$\pi$'])
ax.set_ylim((-1.5, 1.5))
ax.set_yticks([-1, 0, 1])
display(fig)
```

```
Command took 0.47 seconds -- by                              om.ar at 09/02/2021,
```



```
Command took 0.12 seconds -- by             com.ar at 09/02/2021, 22:33:19 on this_cluster
```

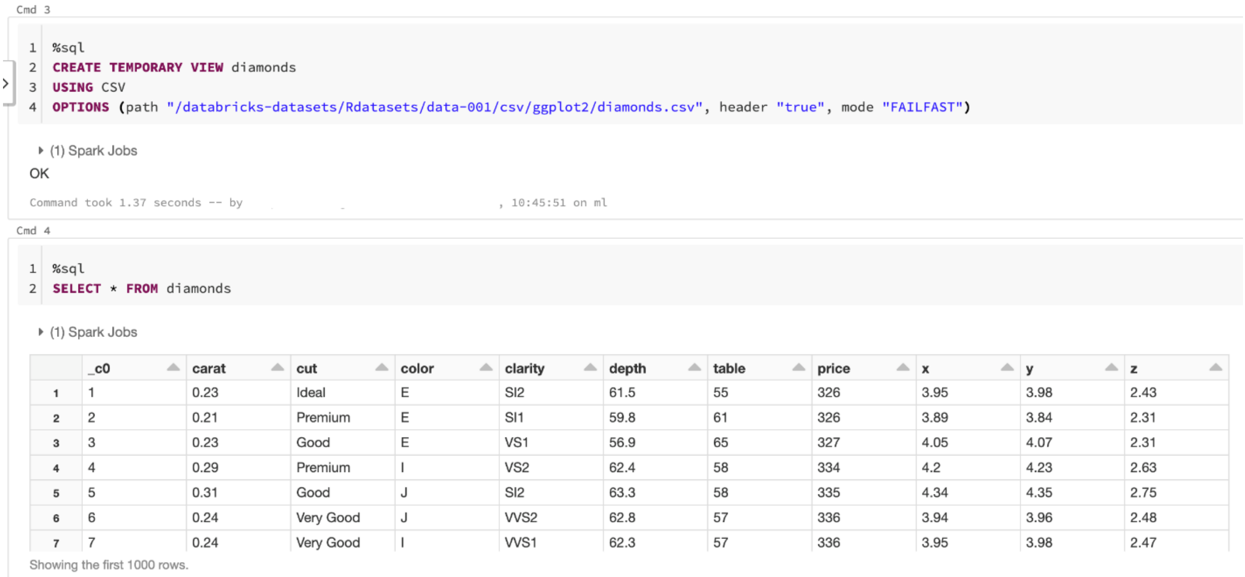
Chapter 8: Databricks Runtime for Machine Learning



Cmd 1

```
with open("/dbfs/tmp/test_dbfs.txt", 'w') as f:
  f.write("This is\n")
  f.write("in the shared\n")
  f.write("file system.\n")
with open("/dbfs/tmp/test_dbfs.txt", "r") as f_read:
  for line in f_read:
    print(line)

```

```
This is

in the shared

file system.
```



Cmd 3

```
%sql
CREATE TEMPORARY VIEW diamonds
USING CSV
OPTIONS (path "/databricks-datasets/Rdatasets/data-001/csv/ggplot2/diamonds.csv", header "true", mode "FAILFAST")
```

```
OK
```

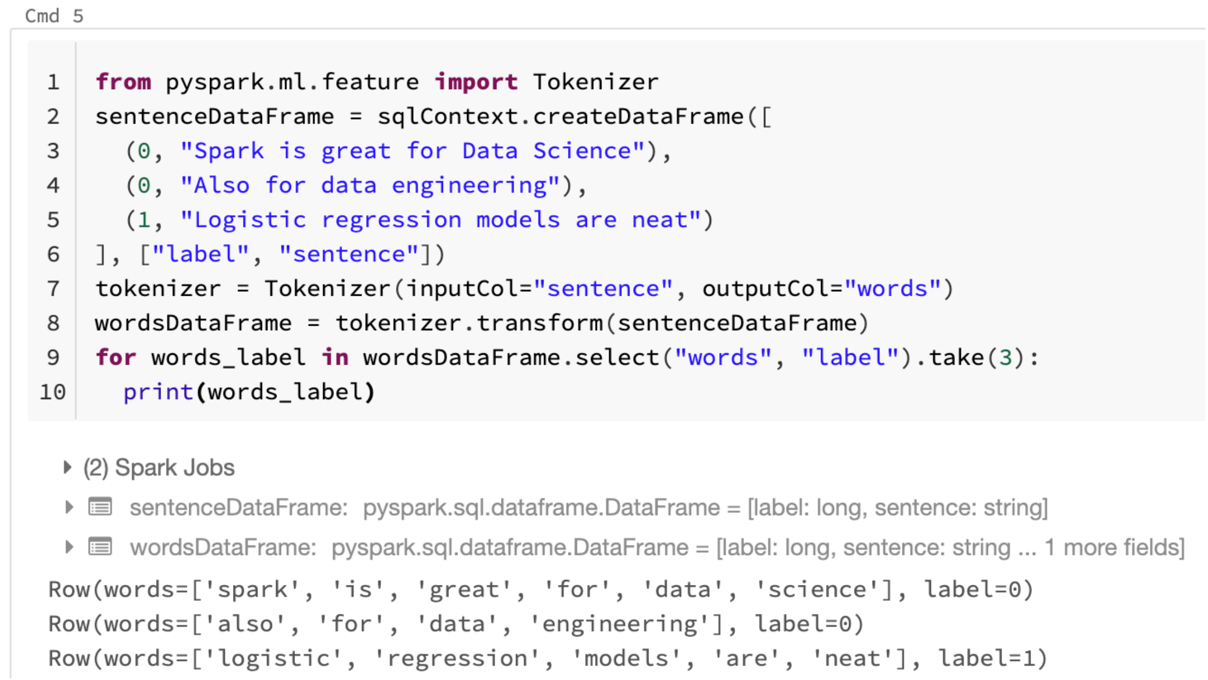
Command took 1.37 seconds -- by , 10:45:51 on ml

Cmd 4

```
%sql
SELECT * FROM diamonds
```

	_c0	carat	cut	color	clarity	depth	table	price	x	y	z
1	1	0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
2	2	0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
3	3	0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
4	4	0.29	Premium	I	VS2	62.4	58	334	4.2	4.23	2.63
5	5	0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75
6	6	0.24	Very Good	J	VVS2	62.8	57	336	3.94	3.96	2.48
7	7	0.24	Very Good	I	VVS1	62.3	57	336	3.95	3.98	2.47

Showing the first 1000 rows.

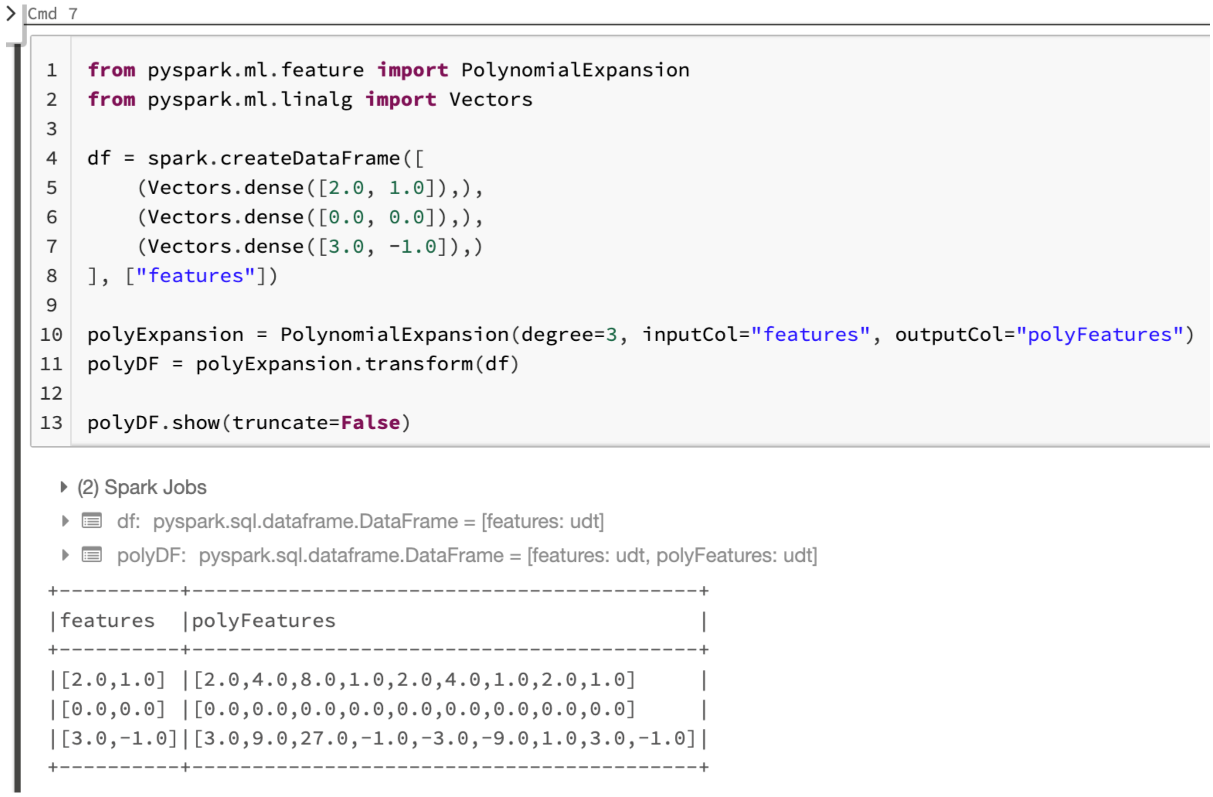


Cmd 5

```python
from pyspark.ml.feature import Tokenizer
sentenceDataFrame = sqlContext.createDataFrame([
  (0, "Spark is great for Data Science"),
  (0, "Also for data engineering"),
  (1, "Logistic regression models are neat")
], ["label", "sentence"])
tokenizer = Tokenizer(inputCol="sentence", outputCol="words")
wordsDataFrame = tokenizer.transform(sentenceDataFrame)
for words_label in wordsDataFrame.select("words", "label").take(3):
  print(words_label)
```

(2) Spark Jobs
sentenceDataFrame: pyspark.sql.dataframe.DataFrame = [label: long, sentence: string]
wordsDataFrame: pyspark.sql.dataframe.DataFrame = [label: long, sentence: string ... 1 more fields]

```
Row(words=['spark', 'is', 'great', 'for', 'data', 'science'], label=0)
Row(words=['also', 'for', 'data', 'engineering'], label=0)
Row(words=['logistic', 'regression', 'models', 'are', 'neat'], label=1)
```

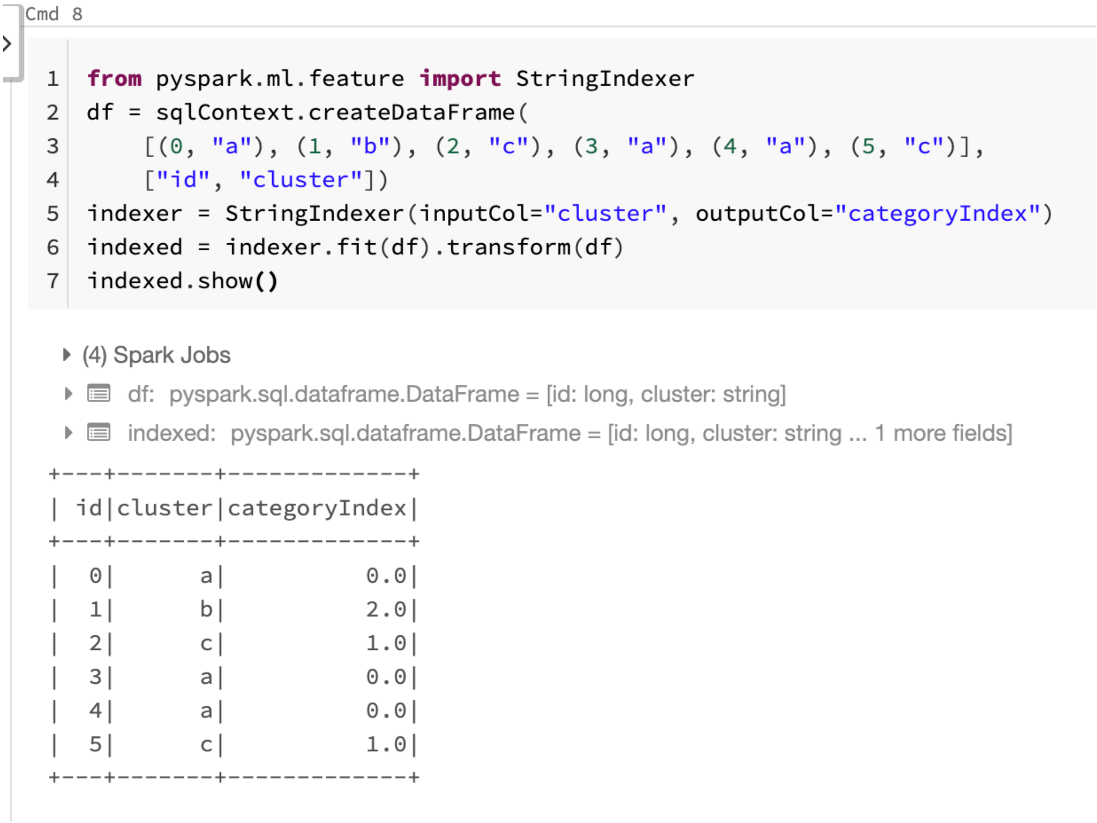


Cmd 7

```python
from pyspark.ml.feature import PolynomialExpansion
from pyspark.ml.linalg import Vectors

df = spark.createDataFrame([
    (Vectors.dense([2.0, 1.0]),),
    (Vectors.dense([0.0, 0.0]),),
    (Vectors.dense([3.0, -1.0]),)
], ["features"])

polyExpansion = PolynomialExpansion(degree=3, inputCol="features", outputCol="polyFeatures")
polyDF = polyExpansion.transform(df)

polyDF.show(truncate=False)
```

(2) Spark Jobs
df: pyspark.sql.dataframe.DataFrame = [features: udt]
polyDF: pyspark.sql.dataframe.DataFrame = [features: udt, polyFeatures: udt]

```
+----------+------------------------------------------+
|features  |polyFeatures                              |
+----------+------------------------------------------+
|[2.0,1.0] |[2.0,4.0,8.0,1.0,2.0,4.0,1.0,2.0,1.0]     |
|[0.0,0.0] |[0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0]     |
|[3.0,-1.0]|[3.0,9.0,27.0,-1.0,-3.0,-9.0,1.0,3.0,-1.0]|
+----------+------------------------------------------+
```



Cmd 8

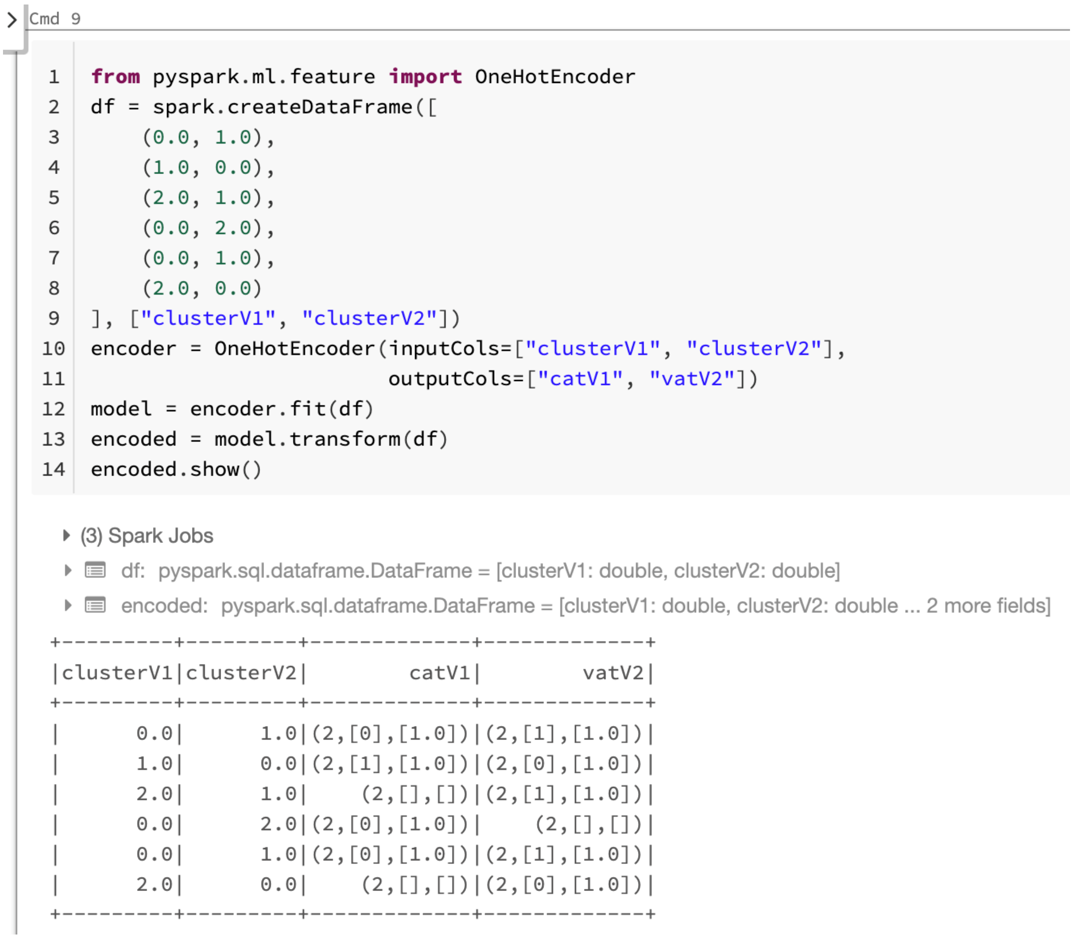
```python
from pyspark.ml.feature import StringIndexer
df = sqlContext.createDataFrame(
    [(0, "a"), (1, "b"), (2, "c"), (3, "a"), (4, "a"), (5, "c")],
    ["id", "cluster"])
indexer = StringIndexer(inputCol="cluster", outputCol="categoryIndex")
indexed = indexer.fit(df).transform(df)
indexed.show()
```

(4) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [id: long, cluster: string]

indexed: pyspark.sql.dataframe.DataFrame = [id: long, cluster: string ... 1 more fields]

```
+---+-------+-------------+
| id|cluster|categoryIndex|
+---+-------+-------------+
|  0|      a|          0.0|
|  1|      b|          2.0|
|  2|      c|          1.0|
|  3|      a|          0.0|
|  4|      a|          0.0|
|  5|      c|          1.0|
+---+-------+-------------+
```



Cmd 9

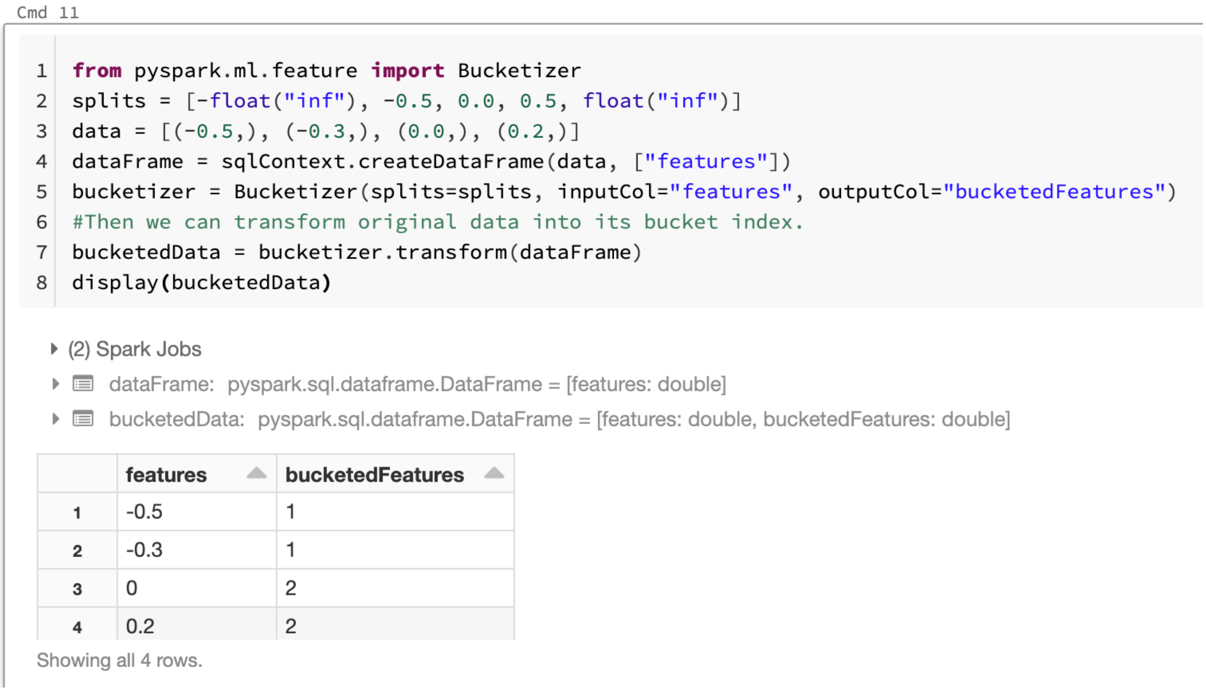
```python
from pyspark.ml.feature import OneHotEncoder
df = spark.createDataFrame([
    (0.0, 1.0),
    (1.0, 0.0),
    (2.0, 1.0),
    (0.0, 2.0),
    (0.0, 1.0),
    (2.0, 0.0)
], ["clusterV1", "clusterV2"])
encoder = OneHotEncoder(inputCols=["clusterV1", "clusterV2"],
                        outputCols=["catV1", "vatV2"])
model = encoder.fit(df)
encoded = model.transform(df)
encoded.show()
```

(3) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [clusterV1: double, clusterV2: double]

encoded: pyspark.sql.dataframe.DataFrame = [clusterV1: double, clusterV2: double ... 2 more fields]

```
+---------+---------+-------------+-------------+
|clusterV1|clusterV2|        catV1|        vatV2|
+---------+---------+-------------+-------------+
|      0.0|      1.0|(2,[0],[1.0])|(2,[1],[1.0])|
|      1.0|      0.0|(2,[1],[1.0])|(2,[0],[1.0])|
|      2.0|      1.0|    (2,[],[])|(2,[1],[1.0])|
|      0.0|      2.0|(2,[0],[1.0])|    (2,[],[])|
|      0.0|      1.0|(2,[0],[1.0])|(2,[1],[1.0])|
|      2.0|      0.0|    (2,[],[])|(2,[0],[1.0])|
+---------+---------+-------------+-------------+
```



Cmd 11

```
from pyspark.ml.feature import Bucketizer
splits = [-float("inf"), -0.5, 0.0, 0.5, float("inf")]
data = [(-0.5,), (-0.3,), (0.0,), (0.2,)]
dataFrame = sqlContext.createDataFrame(data, ["features"])
bucketizer = Bucketizer(splits=splits, inputCol="features", outputCol="bucketedFeatures")
#Then we can transform original data into its bucket index.
bucketedData = bucketizer.transform(dataFrame)
display(bucketedData)
```

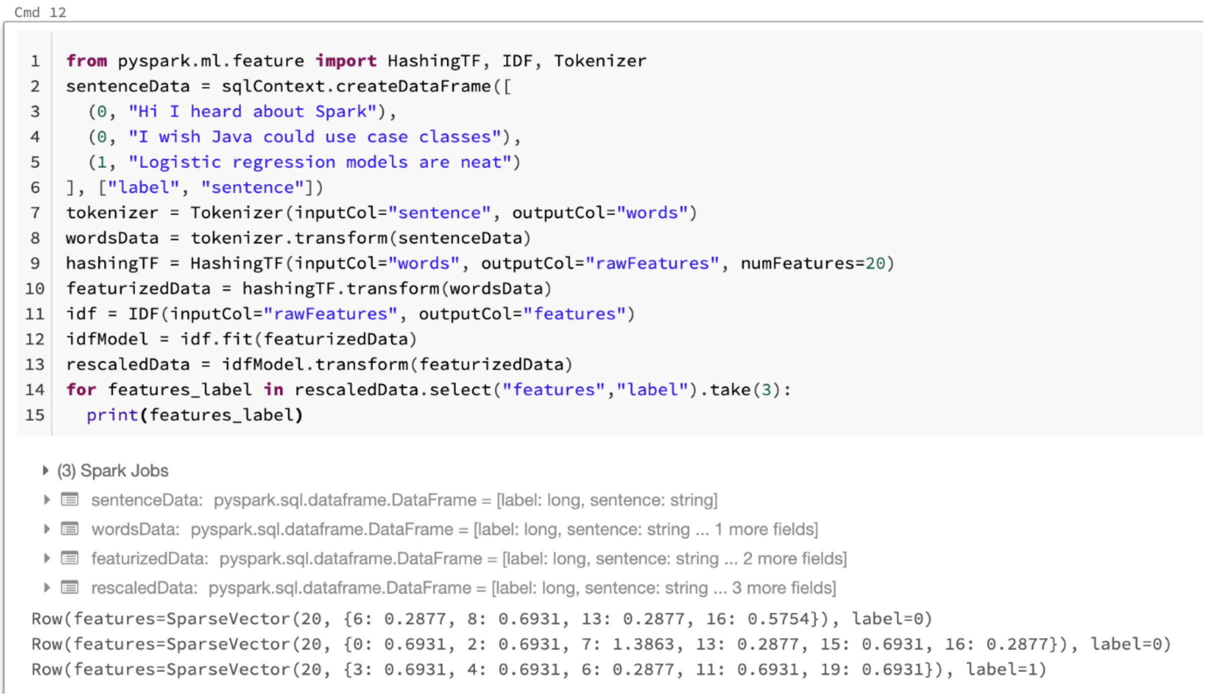
▸ (2) Spark Jobs

▸ dataFrame: pyspark.sql.dataframe.DataFrame = [features: double]

▸ bucketedData: pyspark.sql.dataframe.DataFrame = [features: double, bucketedFeatures: double]

	features	bucketedFeatures
1	-0.5	1
2	-0.3	1
3	0	2
4	0.2	2

Showing all 4 rows.



Cmd 12

```
from pyspark.ml.feature import HashingTF, IDF, Tokenizer
sentenceData = sqlContext.createDataFrame([
  (0, "Hi I heard about Spark"),
  (0, "I wish Java could use case classes"),
  (1, "Logistic regression models are neat")
], ["label", "sentence"])
tokenizer = Tokenizer(inputCol="sentence", outputCol="words")
wordsData = tokenizer.transform(sentenceData)
hashingTF = HashingTF(inputCol="words", outputCol="rawFeatures", numFeatures=20)
featurizedData = hashingTF.transform(wordsData)
idf = IDF(inputCol="rawFeatures", outputCol="features")
idfModel = idf.fit(featurizedData)
rescaledData = idfModel.transform(featurizedData)
for features_label in rescaledData.select("features","label").take(3):
  print(features_label)
```

▸ (3) Spark Jobs

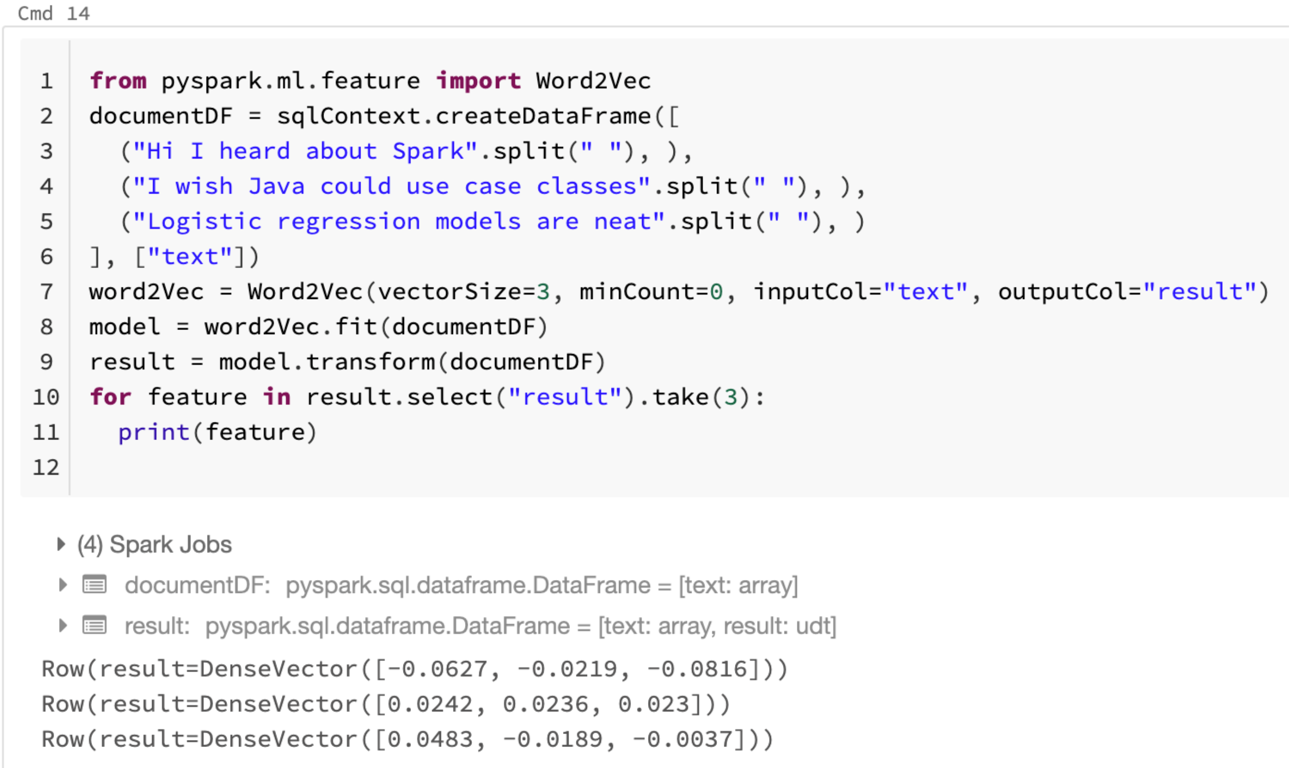
▸ sentenceData: pyspark.sql.dataframe.DataFrame = [label: long, sentence: string]

▸ wordsData: pyspark.sql.dataframe.DataFrame = [label: long, sentence: string ... 1 more fields]

▸ featurizedData: pyspark.sql.dataframe.DataFrame = [label: long, sentence: string ... 2 more fields]

▸ rescaledData: pyspark.sql.dataframe.DataFrame = [label: long, sentence: string ... 3 more fields]

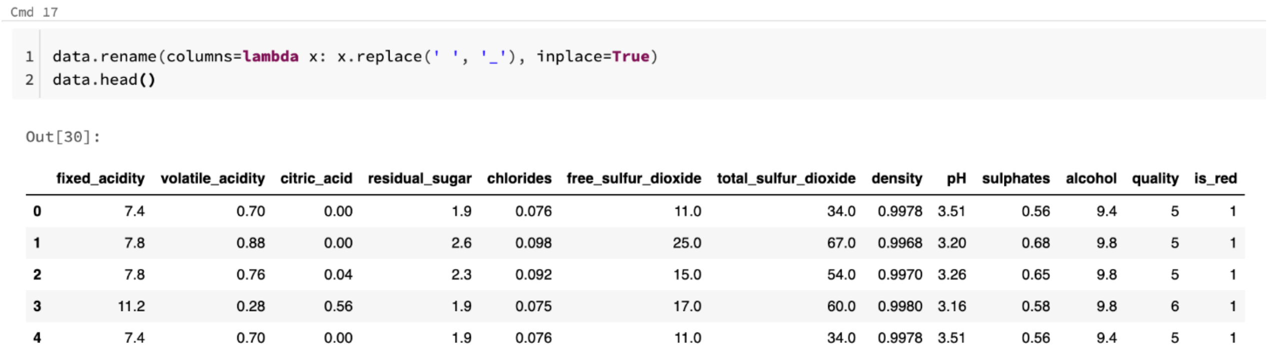
```
Row(features=SparseVector(20, {6: 0.2877, 8: 0.6931, 13: 0.2877, 16: 0.5754}), label=0)
Row(features=SparseVector(20, {0: 0.6931, 2: 0.6931, 7: 1.3863, 13: 0.2877, 15: 0.6931, 16: 0.2877}), label=0)
Row(features=SparseVector(20, {3: 0.6931, 4: 0.6931, 6: 0.2877, 11: 0.6931, 19: 0.6931}), label=1)
```



Cmd 14

```python
from pyspark.ml.feature import Word2Vec
documentDF = sqlContext.createDataFrame([
  ("Hi I heard about Spark".split(" "), ),
  ("I wish Java could use case classes".split(" "), ),
  ("Logistic regression models are neat".split(" "), )
], ["text"])
word2Vec = Word2Vec(vectorSize=3, minCount=0, inputCol="text", outputCol="result")
model = word2Vec.fit(documentDF)
result = model.transform(documentDF)
for feature in result.select("result").take(3):
  print(feature)

```

```
▸ (4) Spark Jobs
▸ documentDF:  pyspark.sql.dataframe.DataFrame = [text: array]
▸ result:  pyspark.sql.dataframe.DataFrame = [text: array, result: udt]
Row(result=DenseVector([-0.0627, -0.0219, -0.0816]))
Row(result=DenseVector([0.0242, 0.0236, 0.023]))
Row(result=DenseVector([0.0483, -0.0189, -0.0037]))
```

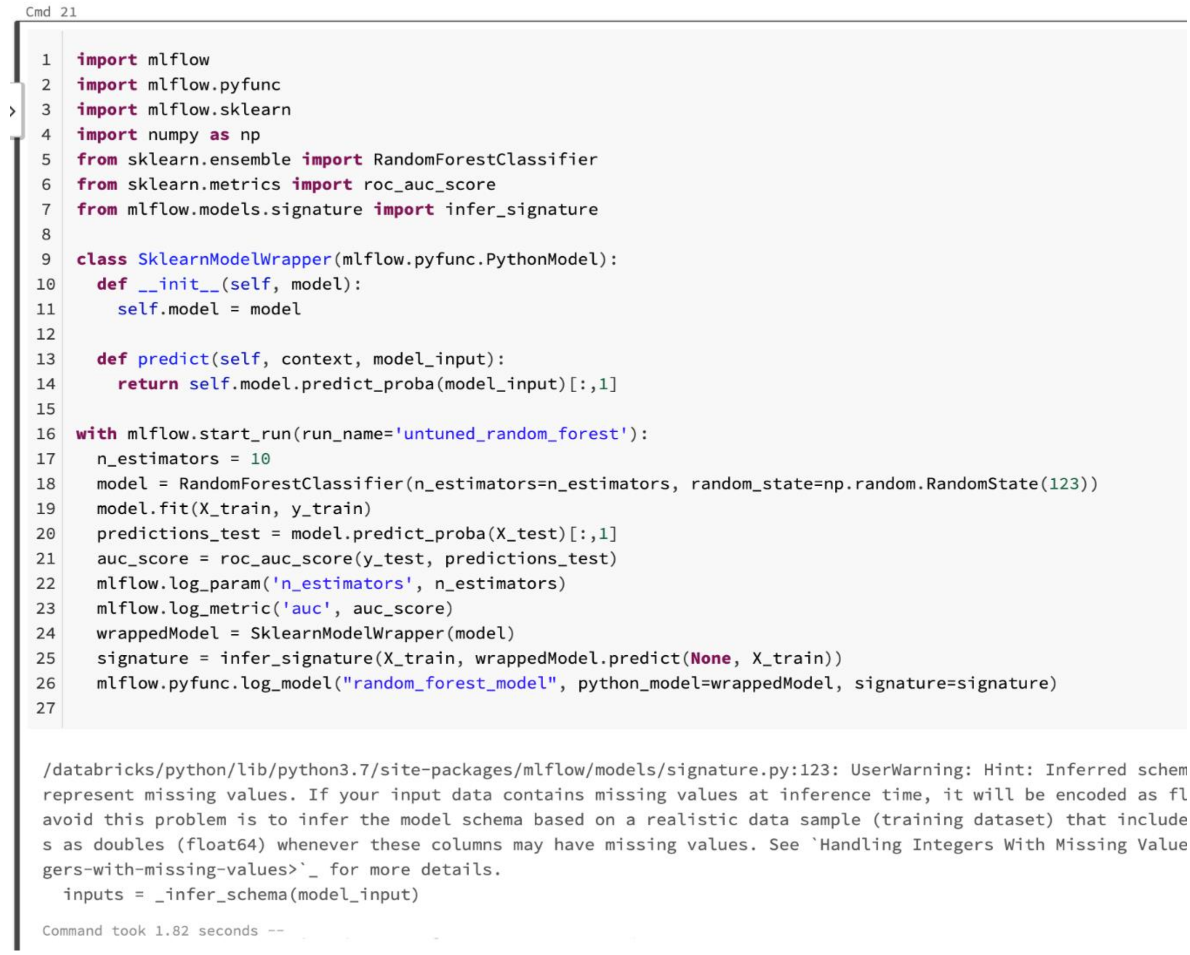


Cmd 17

```python
data.rename(columns=lambda x: x.replace(' ', '_'), inplace=True)
data.head()
```

Out[30]:

	fixed_acidity	volatile_acidity	citric_acid	residual_sugar	chlorides	free_sulfur_dioxide	total_sulfur_dioxide	density	pH	sulphates	alcohol	quality	is_red
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5	1
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5	1
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5	1
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6	1
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5	1

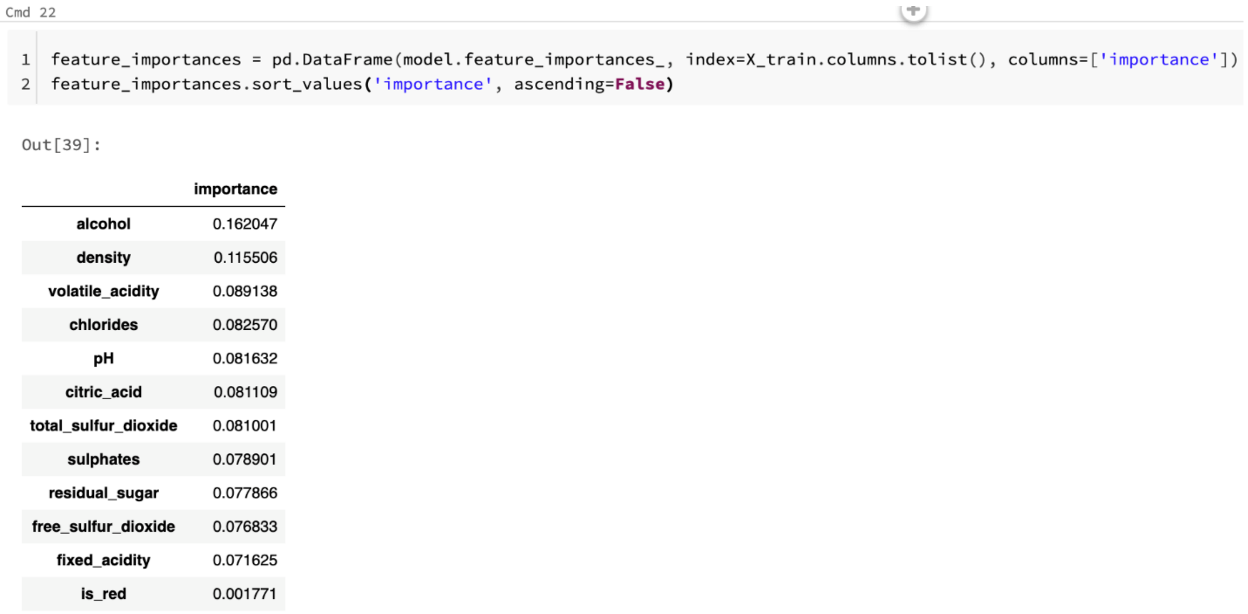


Cmd 21

```
import mlflow
import mlflow.pyfunc
import mlflow.sklearn
import numpy as np
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import roc_auc_score
from mlflow.models.signature import infer_signature

class SklearnModelWrapper(mlflow.pyfunc.PythonModel):
  def __init__(self, model):
    self.model = model

  def predict(self, context, model_input):
    return self.model.predict_proba(model_input)[:,1]

with mlflow.start_run(run_name='untuned_random_forest'):
  n_estimators = 10
  model = RandomForestClassifier(n_estimators=n_estimators, random_state=np.random.RandomState(123))
  model.fit(X_train, y_train)
  predictions_test = model.predict_proba(X_test)[:,1]
  auc_score = roc_auc_score(y_test, predictions_test)
  mlflow.log_param('n_estimators', n_estimators)
  mlflow.log_metric('auc', auc_score)
  wrappedModel = SklearnModelWrapper(model)
  signature = infer_signature(X_train, wrappedModel.predict(None, X_train))
  mlflow.pyfunc.log_model("random_forest_model", python_model=wrappedModel, signature=signature)
```

```
/databricks/python/lib/python3.7/site-packages/mlflow/models/signature.py:123: UserWarning: Hint: Inferred schem
represent missing values. If your input data contains missing values at inference time, it will be encoded as fl
avoid this problem is to infer the model schema based on a realistic data sample (training dataset) that include
s as doubles (float64) whenever these columns may have missing values. See `Handling Integers With Missing Value
gers-with-missing-values>`_ for more details.
  inputs = _infer_schema(model_input)
```

Command took 1.82 seconds --

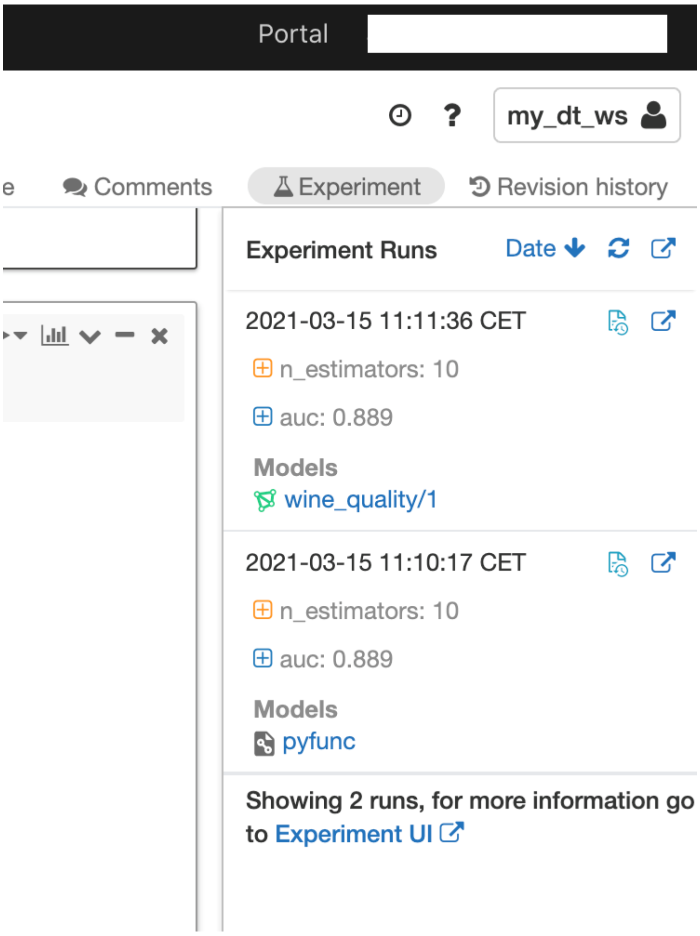


Cmd 22

```
feature_importances = pd.DataFrame(model.feature_importances_, index=X_train.columns.tolist(), columns=['importance'])
feature_importances.sort_values('importance', ascending=False)
```

Out[39]:

	importance
alcohol	0.162047
density	0.115506
volatile_acidity	0.089138
chlorides	0.082570
pH	0.081632
citric_acid	0.081109
total_sulfur_dioxide	0.081001
sulphates	0.078901
residual_sugar	0.077866
free_sulfur_dioxide	0.076833
fixed_acidity	0.071625
is_red	0.001771



Experiment Runs

2021-03-15 11:11:36 CET

n_estimators: 10

auc: 0.889

Models

wine_quality/1

2021-03-15 11:10:17 CET

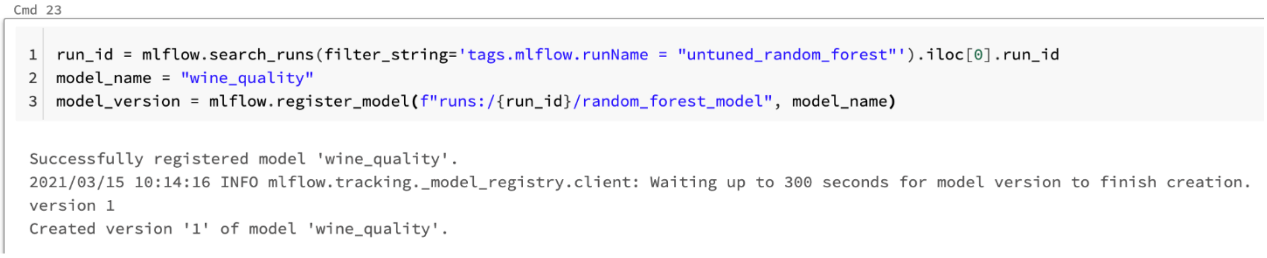
n_estimators: 10

auc: 0.889

Models

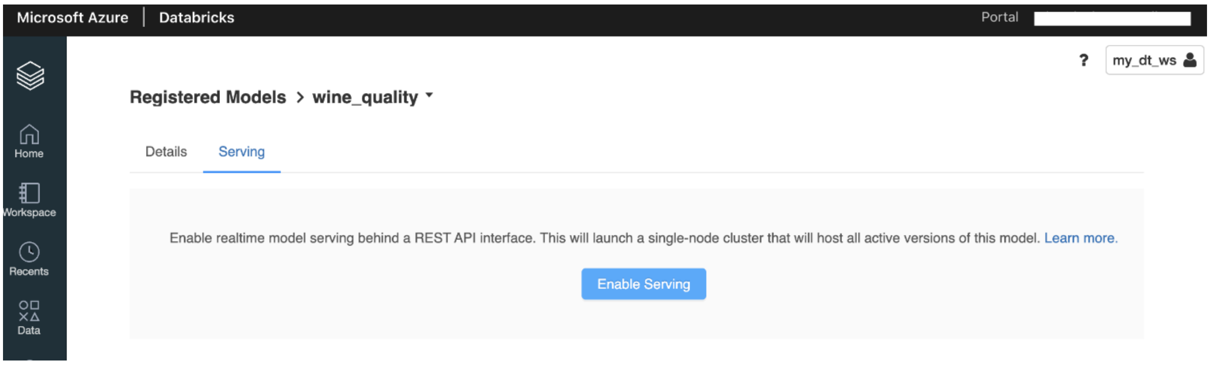
pyfunc

Showing 2 runs, for more information go to Experiment UI


Cmd 23

```
run_id = mlflow.search_runs(filter_string='tags.mlflow.runName = "untuned_random_forest"').iloc[0].run_id
model_name = "wine_quality"
model_version = mlflow.register_model(f"runs:/{run_id}/random_forest_model", model_name)
```

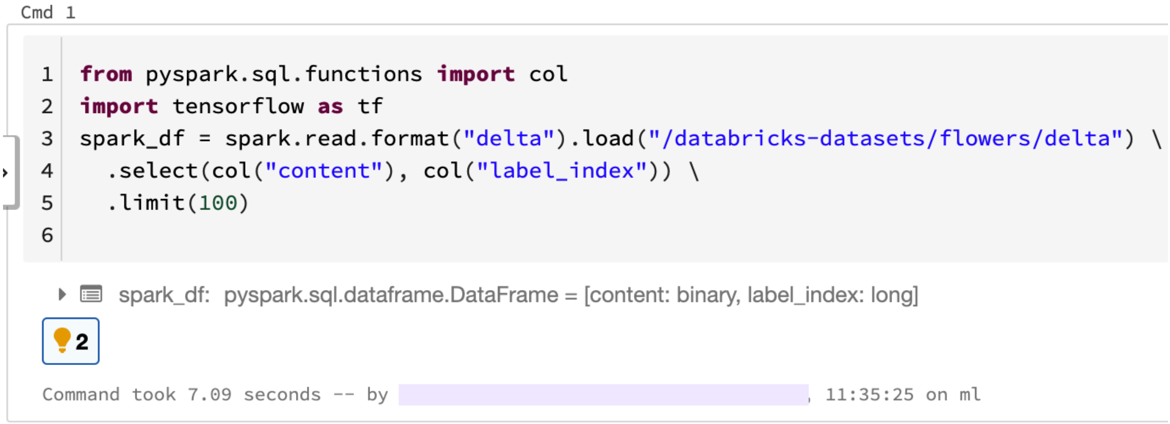
```
Successfully registered model 'wine_quality'.
2021/03/15 10:14:16 INFO mlflow.tracking._model_registry.client: Waiting up to 300 seconds for model version to finish creation.
version 1
Created version '1' of model 'wine_quality'.
```



Registered Models > wine_quality

Enable realtime model serving behind a REST API interface. This will launch a single-node cluster that will host all active versions of this model. Learn more.

Chapter 9: Databricks Runtime for Deep Learning



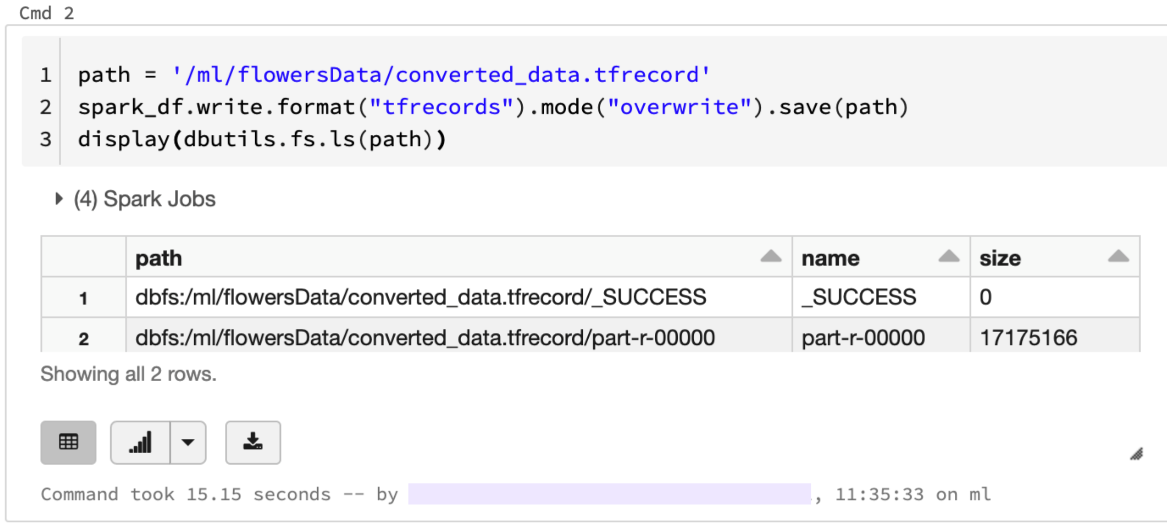
Cmd 1

```
from pyspark.sql.functions import col
import tensorflow as tf
spark_df = spark.read.format("delta").load("/databricks-datasets/flowers/delta") \
  .select(col("content"), col("label_index")) \
  .limit(100)

```

spark_df: pyspark.sql.dataframe.DataFrame = [content: binary, label_index: long]

2

Command took 7.09 seconds -- by [redacted], 11:35:25 on ml



Cmd 2

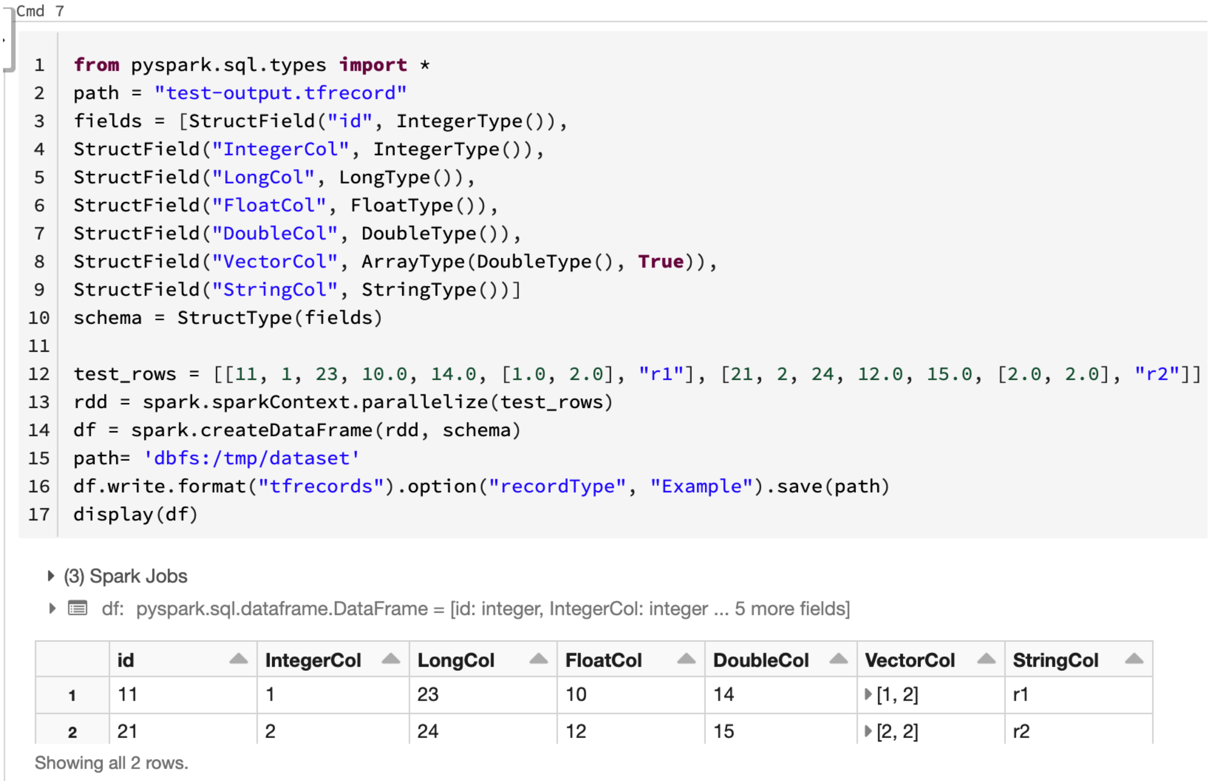
```
path = '/ml/flowersData/converted_data.tfrecord'
spark_df.write.format("tfrecords").mode("overwrite").save(path)
display(dbutils.fs.ls(path))
```

(4) Spark Jobs

	path	name	size
1	dbfs:/ml/flowersData/converted_data.tfrecord/_SUCCESS	_SUCCESS	0
2	dbfs:/ml/flowersData/converted_data.tfrecord/part-r-00000	part-r-00000	17175166

Showing all 2 rows.

Command took 15.15 seconds -- by [redacted], 11:35:33 on ml



Cmd 7

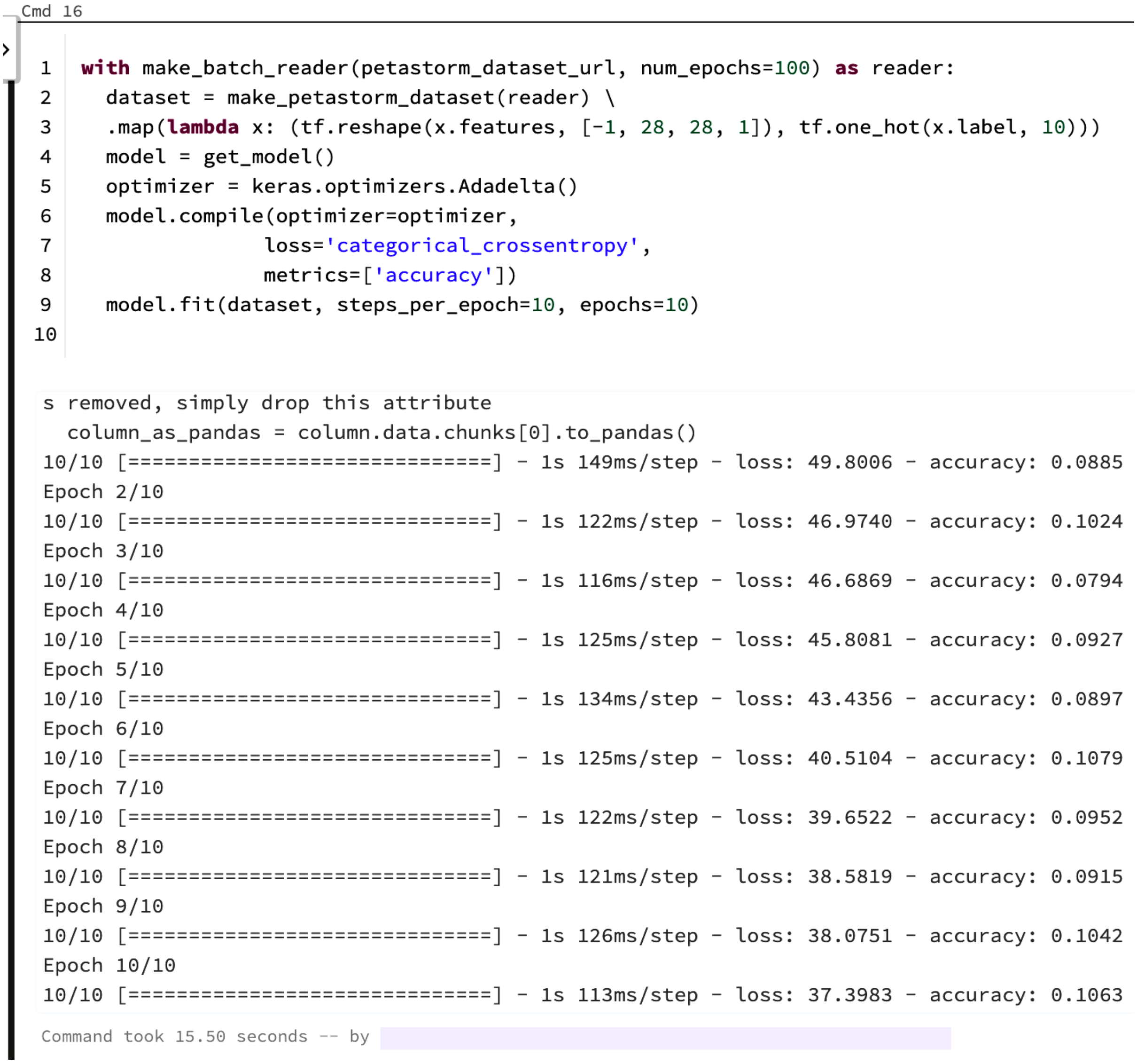
```
from pyspark.sql.types import *
path = "test-output.tfrecord"
fields = [StructField("id", IntegerType()),
StructField("IntegerCol", IntegerType()),
StructField("LongCol", LongType()),
StructField("FloatCol", FloatType()),
StructField("DoubleCol", DoubleType()),
StructField("VectorCol", ArrayType(DoubleType(), True)),
StructField("StringCol", StringType())]
schema = StructType(fields)

test_rows = [[11, 1, 23, 10.0, 14.0, [1.0, 2.0], "r1"], [21, 2, 24, 12.0, 15.0, [2.0, 2.0], "r2"]]
rdd = spark.sparkContext.parallelize(test_rows)
df = spark.createDataFrame(rdd, schema)
path= 'dbfs:/tmp/dataset'
df.write.format("tfrecords").option("recordType", "Example").save(path)
display(df)
```

(3) Spark Jobs

df: pyspark.sql.dataframe.DataFrame = [id: integer, IntegerCol: integer ... 5 more fields]

	id	IntegerCol	LongCol	FloatCol	DoubleCol	VectorCol	StringCol
1	11	1	23	10	14	[1, 2]	r1
2	21	2	24	12	15	[2, 2]	r2

Showing all 2 rows.

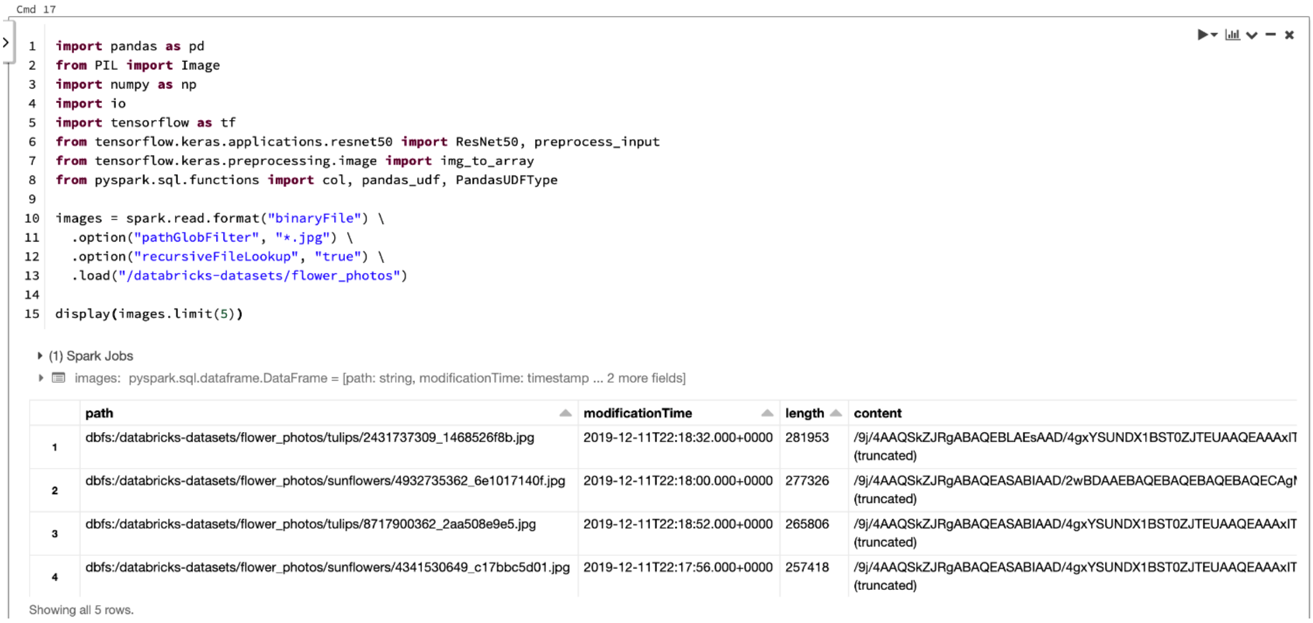


Cmd 16

```
with make_batch_reader(petastorm_dataset_url, num_epochs=100) as reader:
  dataset = make_petastorm_dataset(reader) \
  .map(lambda x: (tf.reshape(x.features, [-1, 28, 28, 1]), tf.one_hot(x.label, 10)))
  model = get_model()
  optimizer = keras.optimizers.Adadelta()
  model.compile(optimizer=optimizer,
                loss='categorical_crossentropy',
                metrics=['accuracy'])
  model.fit(dataset, steps_per_epoch=10, epochs=10)

```

```
s removed, simply drop this attribute
  column_as_pandas = column.data.chunks[0].to_pandas()
10/10 [==============================] - 1s 149ms/step - loss: 49.8006 - accuracy: 0.0885
Epoch 2/10
10/10 [==============================] - 1s 122ms/step - loss: 46.9740 - accuracy: 0.1024
Epoch 3/10
10/10 [==============================] - 1s 116ms/step - loss: 46.6869 - accuracy: 0.0794
Epoch 4/10
10/10 [==============================] - 1s 125ms/step - loss: 45.8081 - accuracy: 0.0927
Epoch 5/10
10/10 [==============================] - 1s 134ms/step - loss: 43.4356 - accuracy: 0.0897
Epoch 6/10
10/10 [==============================] - 1s 125ms/step - loss: 40.5104 - accuracy: 0.1079
Epoch 7/10
10/10 [==============================] - 1s 122ms/step - loss: 39.6522 - accuracy: 0.0952
Epoch 8/10
10/10 [==============================] - 1s 121ms/step - loss: 38.5819 - accuracy: 0.0915
Epoch 9/10
10/10 [==============================] - 1s 126ms/step - loss: 38.0751 - accuracy: 0.1042
Epoch 10/10
10/10 [==============================] - 1s 113ms/step - loss: 37.3983 - accuracy: 0.1063
```

Command took 15.50 seconds -- by



Cmd 17

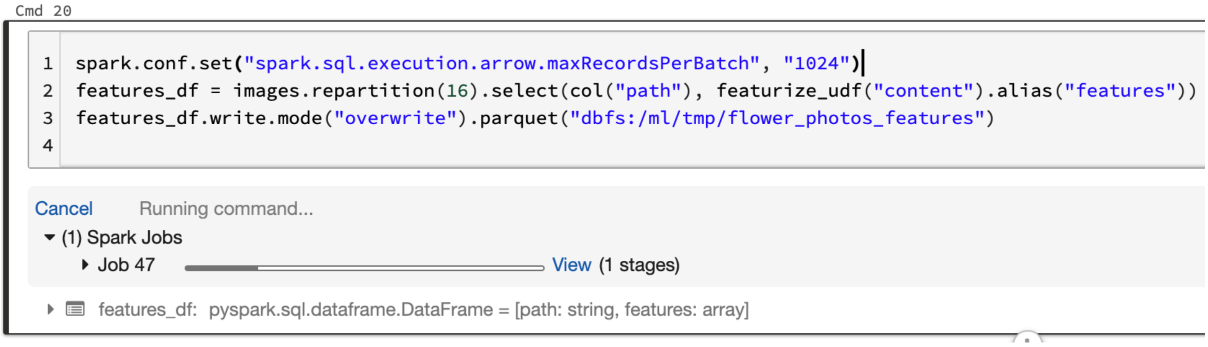
```
import pandas as pd
from PIL import Image
import numpy as np
import io
import tensorflow as tf
from tensorflow.keras.applications.resnet50 import ResNet50, preprocess_input
from tensorflow.keras.preprocessing.image import img_to_array
from pyspark.sql.functions import col, pandas_udf, PandasUDFType

images = spark.read.format("binaryFile") \
  .option("pathGlobFilter", "*.jpg") \
  .option("recursiveFileLookup", "true") \
  .load("/databricks-datasets/flower_photos")

display(images.limit(5))
```

(1) Spark Jobs

images: pyspark.sql.dataframe.DataFrame = [path: string, modificationTime: timestamp ... 2 more fields]

	path	modificationTime	length	content
1	dbfs:/databricks-datasets/flower_photos/tulips/2431737309_1468526f8b.jpg	2019-12-11T22:18:32.000+0000	281953	/9j/4AAQSkZJRgABAQEBLAEsAAD/4gxYSUNDX1BST0ZJTEUAAQEAAAxIT (truncated)
2	dbfs:/databricks-datasets/flower_photos/sunflowers/4932735362_6e1017140f.jpg	2019-12-11T22:18:00.000+0000	277326	/9j/4AAQSkZJRgABAQEASABIAAD/2wBDAAEBAQEBAQEBAQEBAQECAgI (truncated)
3	dbfs:/databricks-datasets/flower_photos/tulips/8717900362_2aa508e9e5.jpg	2019-12-11T22:18:52.000+0000	265806	/9j/4AAQSkZJRgABAQEASABIAAD/4gxYSUNDX1BST0ZJTEUAAQEAAAxIT (truncated)
4	dbfs:/databricks-datasets/flower_photos/sunflowers/4341530649_c17bbc5d01.jpg	2019-12-11T22:17:56.000+0000	257418	/9j/4AAQSkZJRgABAQEASABIAAD/4gxYSUNDX1BST0ZJTEUAAQEAAAxIT (truncated)

Showing all 5 rows.

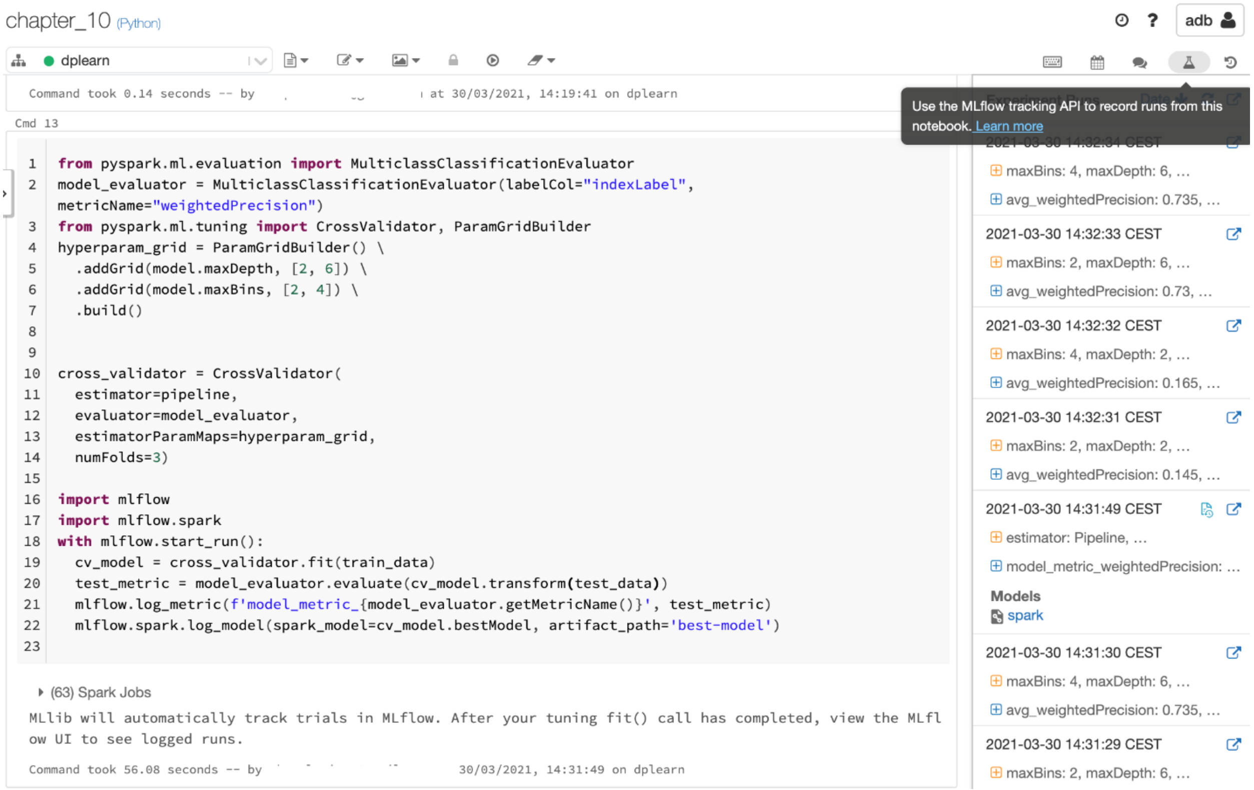


Cmd 20

```
spark.conf.set("spark.sql.execution.arrow.maxRecordsPerBatch", "1024")
features_df = images.repartition(16).select(col("path"), featurize_udf("content").alias("features"))
features_df.write.mode("overwrite").parquet("dbfs:/ml/tmp/flower_photos_features")

```

```
Cancel      Running command...
▾ (1) Spark Jobs
    ▸ Job 47                                   View (1 stages)
▸ features_df:  pyspark.sql.dataframe.DataFrame = [path: string, features: array]
```

Chapter 10: Model Tracking and Tuning in Azure Databricks



```
Command took 0.14 seconds -- by [redacted] at 30/03/2021, 14:19:41 on dplearn
Cmd 13
```

```python
from pyspark.ml.evaluation import MulticlassClassificationEvaluator
model_evaluator = MulticlassClassificationEvaluator(labelCol="indexLabel",
metricName="weightedPrecision")
from pyspark.ml.tuning import CrossValidator, ParamGridBuilder
hyperparam_grid = ParamGridBuilder() \
  .addGrid(model.maxDepth, [2, 6]) \
  .addGrid(model.maxBins, [2, 4]) \
  .build()


cross_validator = CrossValidator(
  estimator=pipeline,
  evaluator=model_evaluator,
  estimatorParamMaps=hyperparam_grid,
  numFolds=3)

import mlflow
import mlflow.spark
with mlflow.start_run():
  cv_model = cross_validator.fit(train_data)
  test_metric = model_evaluator.evaluate(cv_model.transform(test_data))
  mlflow.log_metric(f'model_metric_{model_evaluator.getMetricName()}', test_metric)
  mlflow.spark.log_model(spark_model=cv_model.bestModel, artifact_path='best-model')
```

```
▸ (63) Spark Jobs
MLlib will automatically track trials in MLflow. After your tuning fit() call has completed, view the MLfl
ow UI to see logged runs.
Command took 56.08 seconds -- by [redacted] 30/03/2021, 14:31:49 on dplearn
```

Use the MLflow tracking API to record runs from this notebook. Learn more

2021-03-30 14:32:34 CEST
⊞ maxBins: 4, maxDepth: 6, ...
⊞ avg_weightedPrecision: 0.735, ...

2021-03-30 14:32:33 CEST
⊞ maxBins: 2, maxDepth: 6, ...
⊞ avg_weightedPrecision: 0.73, ...

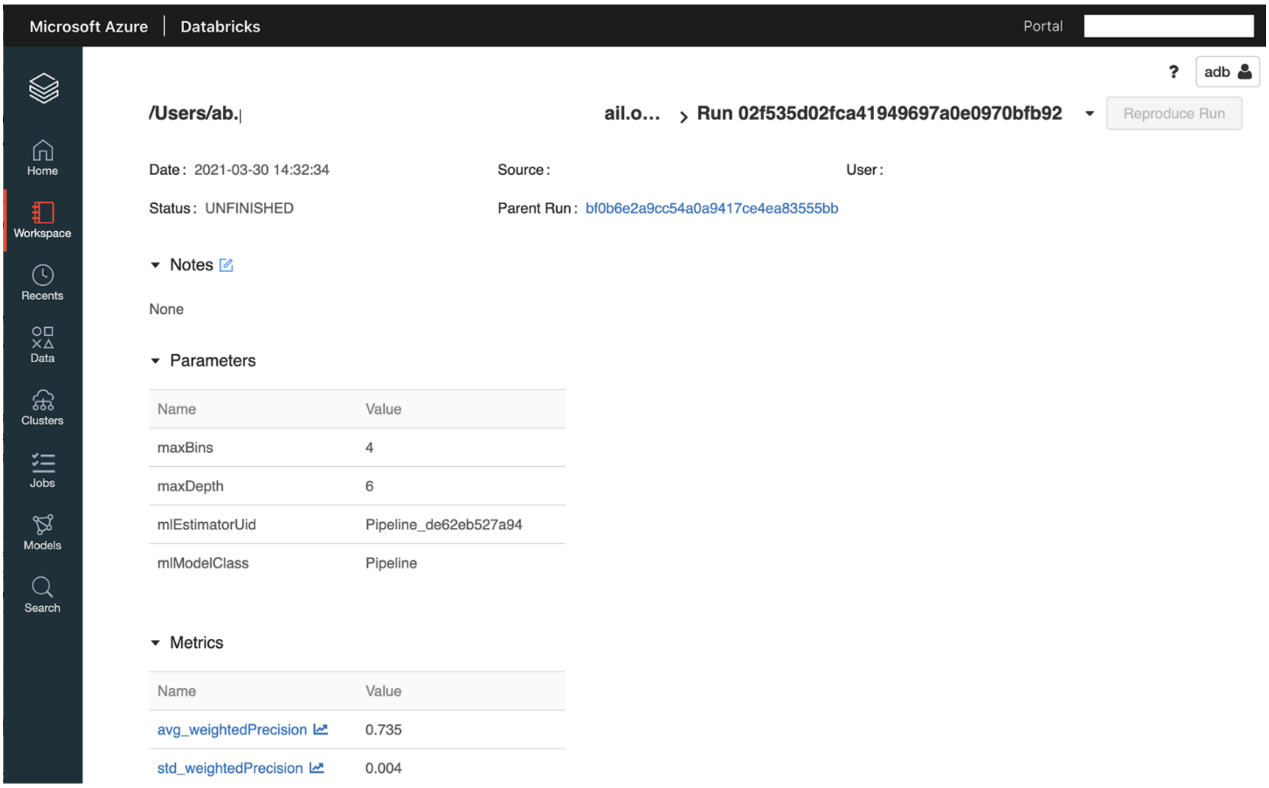
2021-03-30 14:32:32 CEST
⊞ maxBins: 4, maxDepth: 2, ...
⊞ avg_weightedPrecision: 0.165, ...

2021-03-30 14:32:31 CEST
⊞ maxBins: 2, maxDepth: 2, ...
⊞ avg_weightedPrecision: 0.145, ...

2021-03-30 14:31:49 CEST
⊞ estimator: Pipeline, ...
⊞ model_metric_weightedPrecision: ...
Models
spark

2021-03-30 14:31:30 CEST
⊞ maxBins: 4, maxDepth: 6, ...
⊞ avg_weightedPrecision: 0.735, ...

2021-03-30 14:31:29 CEST
⊞ maxBins: 2, maxDepth: 6, ...



/Users/ab.[redacted]ail.o... › Run 02f535d02fca41949697a0e0970bfb92

Date : 2021-03-30 14:32:34

Source :

User :

Status : UNFINISHED

Parent Run : bf0b6e2a9cc54a0a9417ce4ea83555bb

▾ Notes

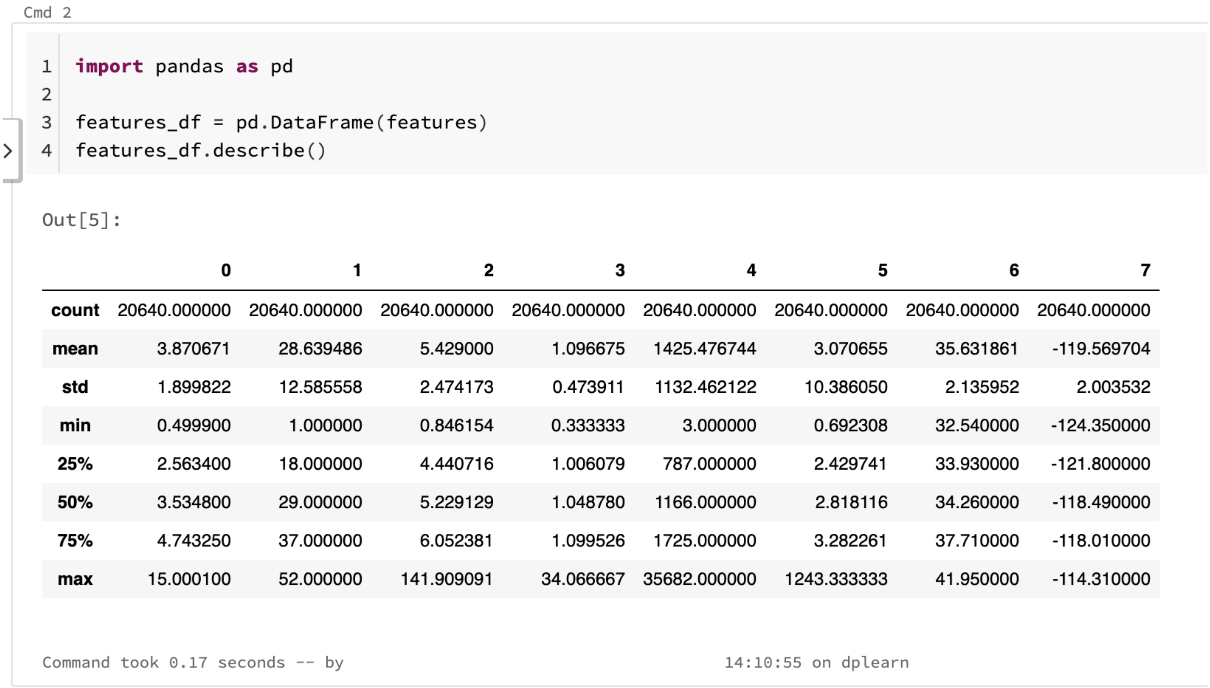
None

▾ Parameters

Name	Value
maxBins	4
maxDepth	6
mlEstimatorUid	Pipeline_de62eb527a94
mlModelClass	Pipeline

▾ Metrics

Name	Value
avg_weightedPrecision	0.735
std_weightedPrecision	0.004



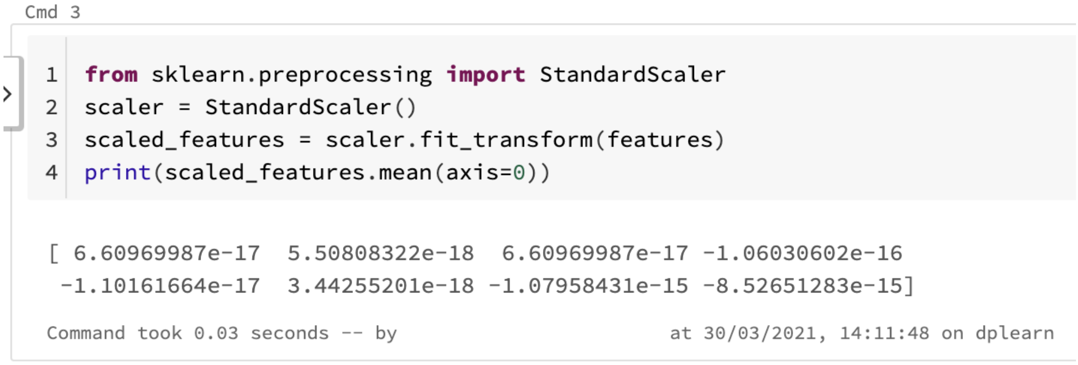
Cmd 2

```python
import pandas as pd

features_df = pd.DataFrame(features)
features_df.describe()
```

Out[5]:

	0	1	2	3	4	5	6	7
count	20640.000000	20640.000000	20640.000000	20640.000000	20640.000000	20640.000000	20640.000000	20640.000000
mean	3.870671	28.639486	5.429000	1.096675	1425.476744	3.070655	35.631861	-119.569704
std	1.899822	12.585558	2.474173	0.473911	1132.462122	10.386050	2.135952	2.003532
min	0.499900	1.000000	0.846154	0.333333	3.000000	0.692308	32.540000	-124.350000
25%	2.563400	18.000000	4.440716	1.006079	787.000000	2.429741	33.930000	-121.800000
50%	3.534800	29.000000	5.229129	1.048780	1166.000000	2.818116	34.260000	-118.490000
75%	4.743250	37.000000	6.052381	1.099526	1725.000000	3.282261	37.710000	-118.010000
max	15.000100	52.000000	141.909091	34.066667	35682.000000	1243.333333	41.950000	-114.310000

```
Command took 0.17 seconds -- by                                    14:10:55 on dplearn
```

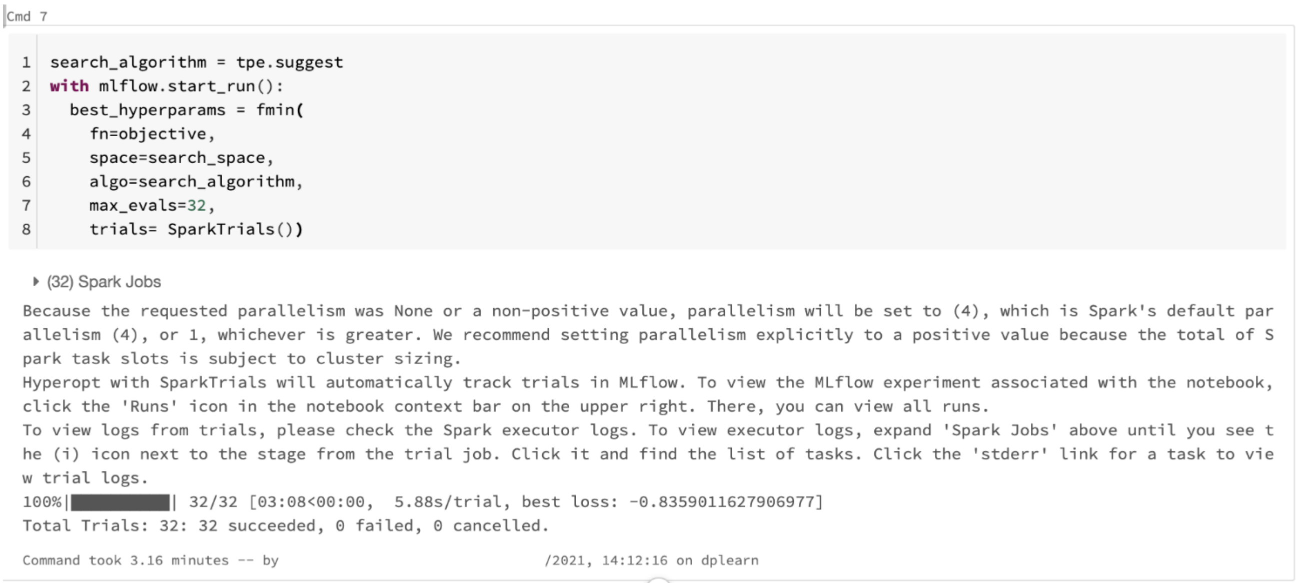


Cmd 3

```python
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
scaled_features = scaler.fit_transform(features)
print(scaled_features.mean(axis=0))
```

```
[ 6.60969987e-17  5.50808322e-18  6.60969987e-17 -1.06030602e-16
 -1.10161664e-17  3.44255201e-18 -1.07958431e-15 -8.52651283e-15]
```

```
Command took 0.03 seconds -- by                       at 30/03/2021, 14:11:48 on dplearn
```



Cmd 7

```python
search_algorithm = tpe.suggest
with mlflow.start_run():
  best_hyperparams = fmin(
    fn=objective,
    space=search_space,
    algo=search_algorithm,
    max_evals=32,
    trials= SparkTrials())
```

▸ (32) Spark Jobs

```
Because the requested parallelism was None or a non-positive value, parallelism will be set to (4), which is Spark's default par
allelism (4), or 1, whichever is greater. We recommend setting parallelism explicitly to a positive value because the total of S
park task slots is subject to cluster sizing.
Hyperopt with SparkTrials will automatically track trials in MLflow. To view the MLflow experiment associated with the notebook,
click the 'Runs' icon in the notebook context bar on the upper right. There, you can view all runs.
To view logs from trials, please check the Spark executor logs. To view executor logs, expand 'Spark Jobs' above until you see t
he (i) icon next to the stage from the trial job. Click it and find the list of tasks. Click the 'stderr' link for a task to vie
w trial logs.
100%|██████████| 32/32 [03:08<00:00,  5.88s/trial, best loss: -0.8359011627906977]
Total Trials: 32: 32 succeeded, 0 failed, 0 cancelled.
```

```
Command took 3.16 minutes -- by                            /2021, 14:12:16 on dplearn
```

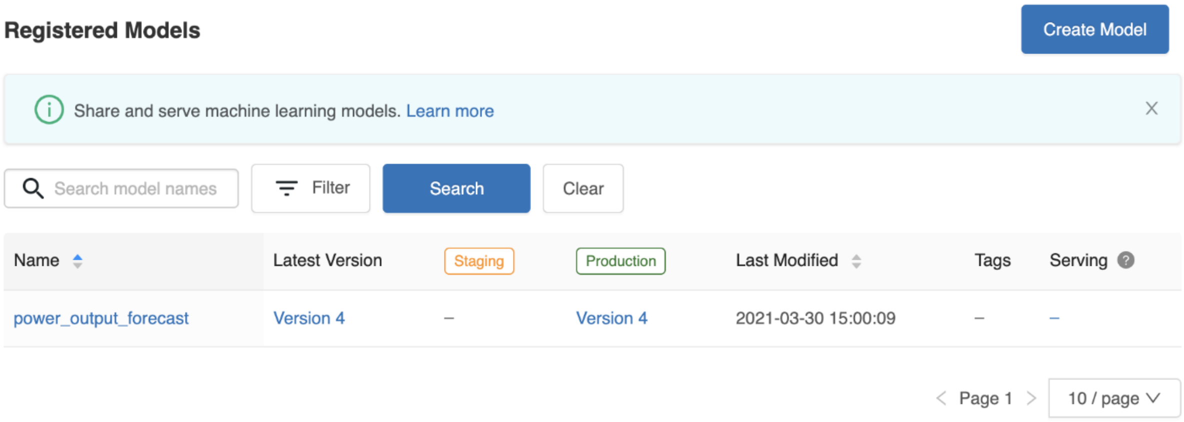
Cmd 8

```python
import hyperopt
print(hyperopt.space_eval(search_space, best_hyperparams))
```

```
{'C': 3.6002403259280142, 'kernel': 'rbf', 'type': 'svm'}
```

```
Command took 0.02 seconds -- by                          at 30/03/2021, 14:12:24 on dplearn
```

Chapter 11: Managing and Serving Models with MLflow and MLeap



Registered Models

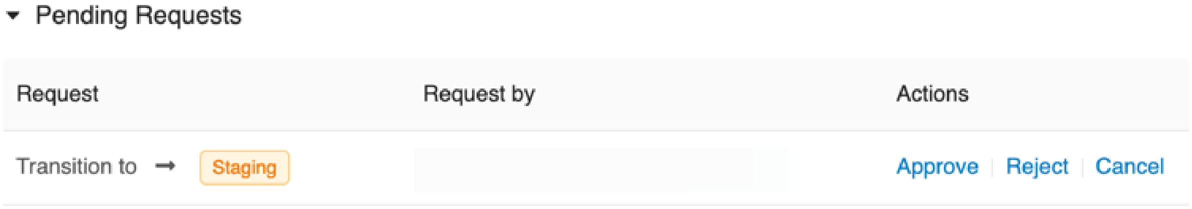
Create Model

ⓘ Share and serve machine learning models. Learn more ×

Search model names | Filter | Search | Clear

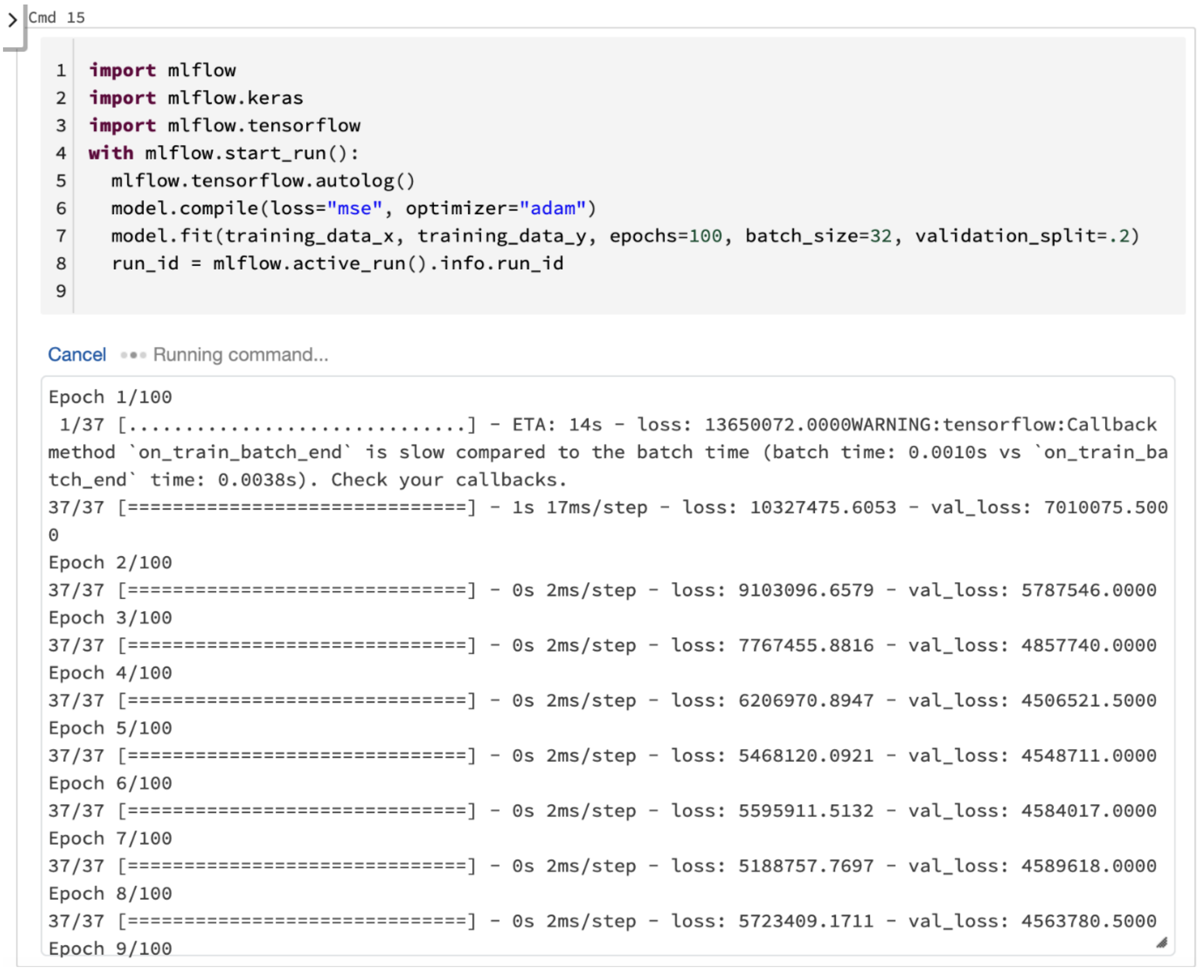
Name	Latest Version	Staging	Production	Last Modified	Tags	Serving
power_output_forecast	Version 4	–	Version 4	2021-03-30 15:00:09	–	–

< Page 1 > 10 / page ∨



▾ Pending Requests

Request	Request by	Actions
Transition to → Staging		Approve | Reject | Cancel

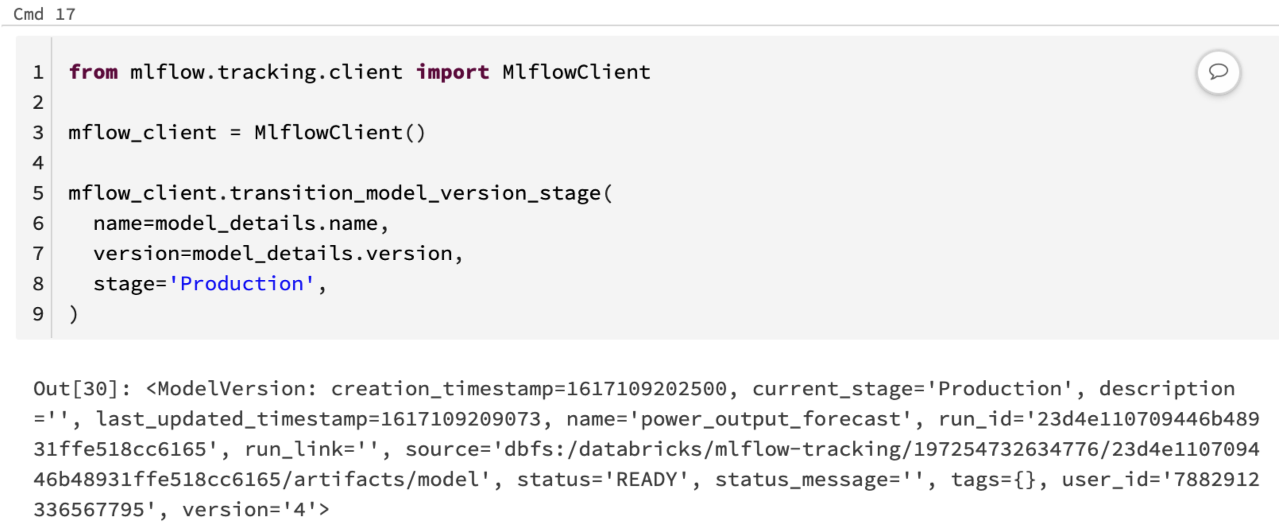


Cmd 15

```
import mlflow
import mlflow.keras
import mlflow.tensorflow
with mlflow.start_run():
  mlflow.tensorflow.autolog()
  model.compile(loss="mse", optimizer="adam")
  model.fit(training_data_x, training_data_y, epochs=100, batch_size=32, validation_split=.2)
  run_id = mlflow.active_run().info.run_id

```

Cancel ••• Running command...

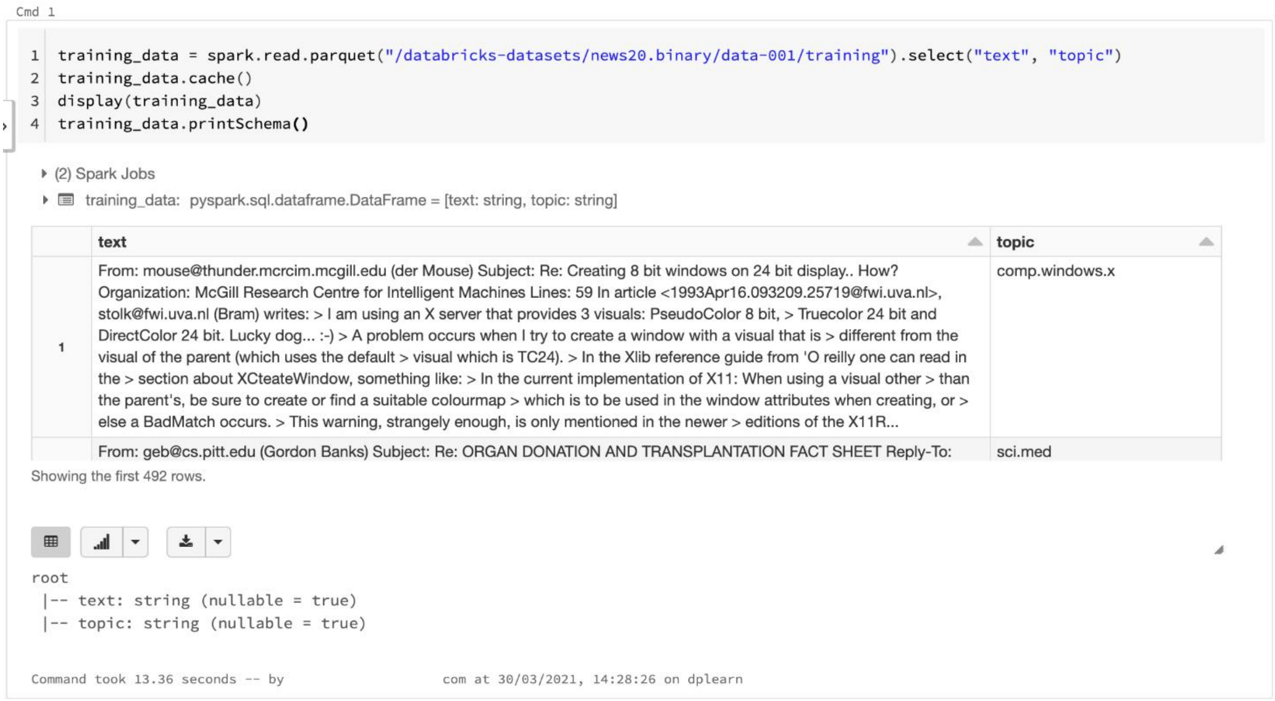
```
Epoch 1/100
 1/37 [..............................] - ETA: 14s - loss: 13650072.0000WARNING:tensorflow:Callback
method `on_train_batch_end` is slow compared to the batch time (batch time: 0.0010s vs `on_train_ba
tch_end` time: 0.0038s). Check your callbacks.
37/37 [==============================] - 1s 17ms/step - loss: 10327475.6053 - val_loss: 7010075.500
0
Epoch 2/100
37/37 [==============================] - 0s 2ms/step - loss: 9103096.6579 - val_loss: 5787546.0000
Epoch 3/100
37/37 [==============================] - 0s 2ms/step - loss: 7767455.8816 - val_loss: 4857740.0000
Epoch 4/100
37/37 [==============================] - 0s 2ms/step - loss: 6206970.8947 - val_loss: 4506521.5000
Epoch 5/100
37/37 [==============================] - 0s 2ms/step - loss: 5468120.0921 - val_loss: 4548711.0000
Epoch 6/100
37/37 [==============================] - 0s 2ms/step - loss: 5595911.5132 - val_loss: 4584017.0000
Epoch 7/100
37/37 [==============================] - 0s 2ms/step - loss: 5188757.7697 - val_loss: 4589618.0000
Epoch 8/100
37/37 [==============================] - 0s 2ms/step - loss: 5723409.1711 - val_loss: 4563780.5000
Epoch 9/100
```



Cmd 17

```python
from mlflow.tracking.client import MlflowClient

mflow_client = MlflowClient()

mflow_client.transition_model_version_stage(
  name=model_details.name,
  version=model_details.version,
  stage='Production',
)
```

```
Out[30]: <ModelVersion: creation_timestamp=1617109202500, current_stage='Production', description
='', last_updated_timestamp=1617109209073, name='power_output_forecast', run_id='23d4e110709446b489
31ffe518cc6165', run_link='', source='dbfs:/databricks/mlflow-tracking/197254732634776/23d4e1107094
46b48931ffe518cc6165/artifacts/model', status='READY', status_message='', tags={}, user_id='7882912
336567795', version='4'>
```



Cmd 1

```python
training_data = spark.read.parquet("/databricks-datasets/news20.binary/data-001/training").select("text", "topic")
training_data.cache()
display(training_data)
training_data.printSchema()
```

▸ (2) Spark Jobs

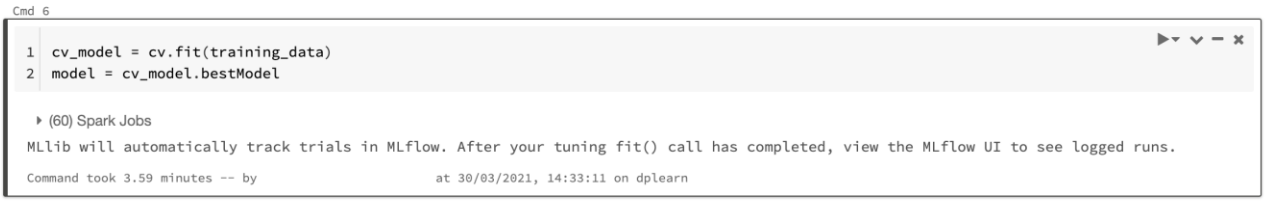
▸ training_data: pyspark.sql.dataframe.DataFrame = [text: string, topic: string]

	text	topic
1	From: mouse@thunder.mcrcim.mcgill.edu (der Mouse) Subject: Re: Creating 8 bit windows on 24 bit display.. How? Organization: McGill Research Centre for Intelligent Machines Lines: 59 In article <1993Apr16.093209.25719@fwi.uva.nl>, stolk@fwi.uva.nl (Bram) writes: > I am using an X server that provides 3 visuals: PseudoColor 8 bit, > Truecolor 24 bit and DirectColor 24 bit. Lucky dog... :-) > A problem occurs when I try to create a window with a visual that is > different from the visual of the parent (which uses the default > visual which is TC24). > In the Xlib reference guide from 'O reilly one can read in the > section about XCteateWindow, something like: > In the current implementation of X11: When using a visual other > than the parent's, be sure to create or find a suitable colourmap > which is to be used in the window attributes when creating, or > else a BadMatch occurs. > This warning, strangely enough, is only mentioned in the newer > editions of the X11R...	comp.windows.x
	From: geb@cs.pitt.edu (Gordon Banks) Subject: Re: ORGAN DONATION AND TRANSPLANTATION FACT SHEET Reply-To:	sci.med

Showing the first 492 rows.

```
root
 |-- text: string (nullable = true)
 |-- topic: string (nullable = true)
```

Command took 13.36 seconds -- by com at 30/03/2021, 14:28:26 on dplearn



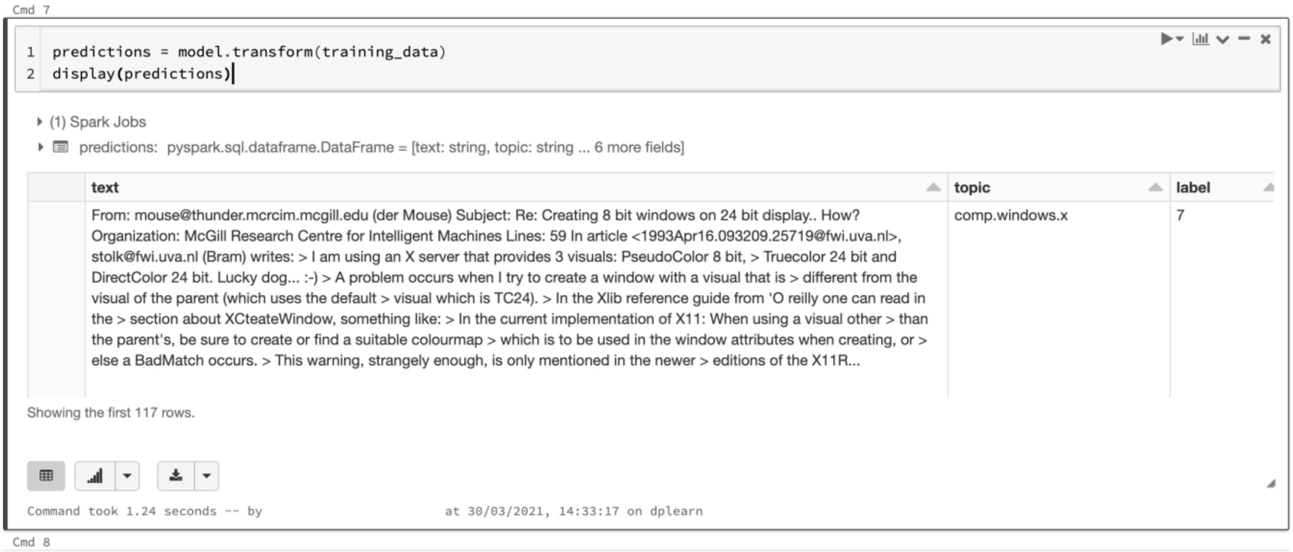
Cmd 6

```python
cv_model = cv.fit(training_data)
model = cv_model.bestModel
```

▸ (60) Spark Jobs

```
MLlib will automatically track trials in MLflow. After your tuning fit() call has completed, view the MLflow UI to see logged runs.
```

Command took 3.59 minutes -- by at 30/03/2021, 14:33:11 on dplearn



Cmd 7

```
predictions = model.transform(training_data)
display(predictions)
```

(1) Spark Jobs

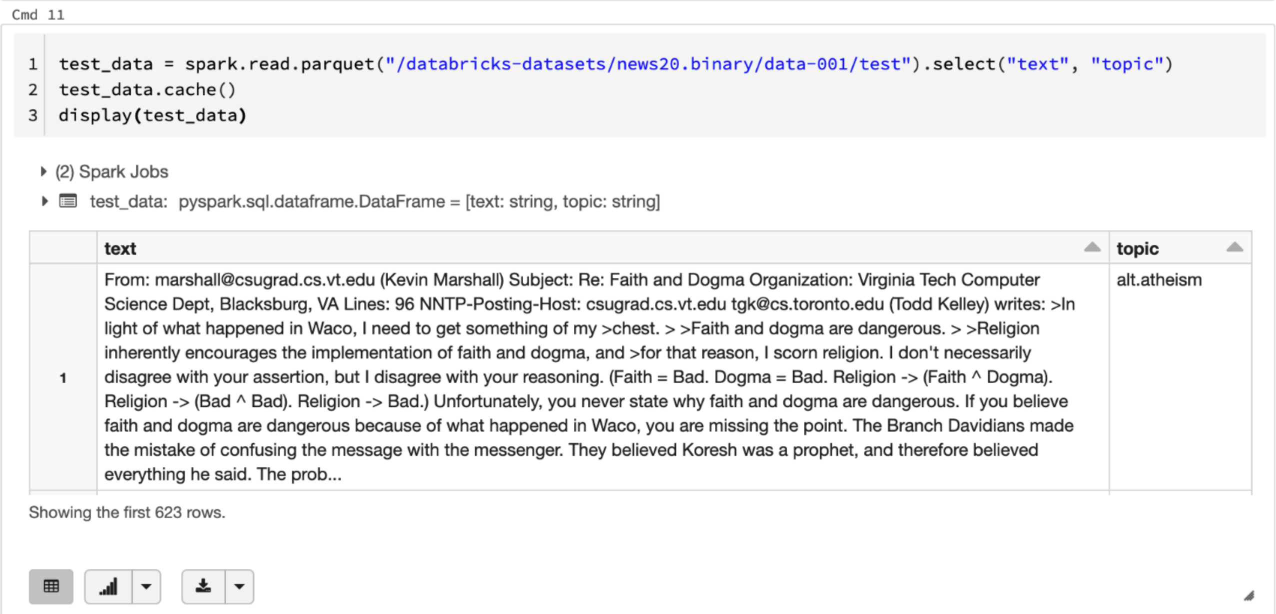
predictions: pyspark.sql.dataframe.DataFrame = [text: string, topic: string ... 6 more fields]

	text	topic	label
	From: mouse@thunder.mcrcim.mcgill.edu (der Mouse) Subject: Re: Creating 8 bit windows on 24 bit display.. How? Organization: McGill Research Centre for Intelligent Machines Lines: 59 In article <1993Apr16.093209.25719@fwi.uva.nl>, stolk@fwi.uva.nl (Bram) writes: > I am using an X server that provides 3 visuals: PseudoColor 8 bit, > Truecolor 24 bit and DirectColor 24 bit. Lucky dog... :-) > A problem occurs when I try to create a window with a visual that is > different from the visual of the parent (which uses the default > visual which is TC24). > In the Xlib reference guide from 'O reilly one can read in the > section about XCteateWindow, something like: > In the current implementation of X11: When using a visual other > than the parent's, be sure to create or find a suitable colourmap > which is to be used in the window attributes when creating, or > else a BadMatch occurs. > This warning, strangely enough, is only mentioned in the newer > editions of the X11R...	comp.windows.x	7

Showing the first 117 rows.

```
Command took 1.24 seconds -- by                     at 30/03/2021, 14:33:17 on dplearn
```

Cmd 8



Cmd 11

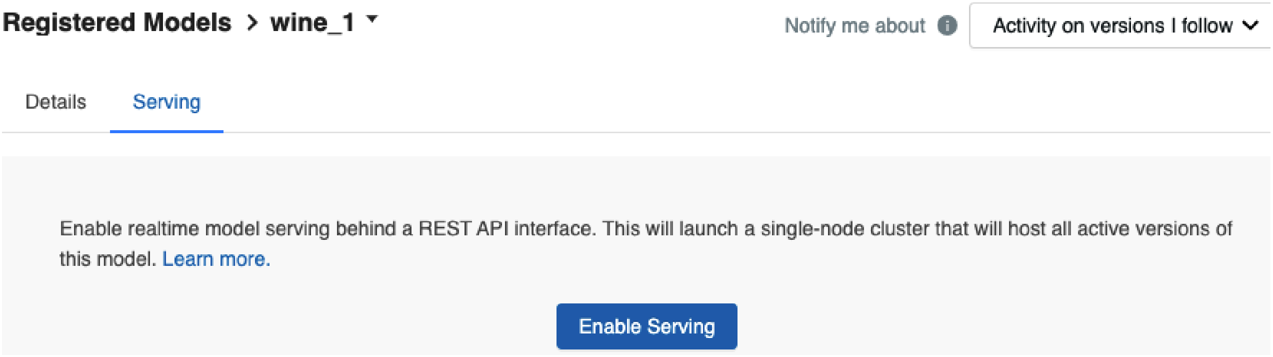
```
test_data = spark.read.parquet("/databricks-datasets/news20.binary/data-001/test").select("text", "topic")
test_data.cache()
display(test_data)
```

(2) Spark Jobs

test_data: pyspark.sql.dataframe.DataFrame = [text: string, topic: string]

	text	topic
1	From: marshall@csugrad.cs.vt.edu (Kevin Marshall) Subject: Re: Faith and Dogma Organization: Virginia Tech Computer Science Dept, Blacksburg, VA Lines: 96 NNTP-Posting-Host: csugrad.cs.vt.edu tgk@cs.toronto.edu (Todd Kelley) writes: >In light of what happened in Waco, I need to get something of my >chest. > >Faith and dogma are dangerous. > >Religion inherently encourages the implementation of faith and dogma, and >for that reason, I scorn religion. I don't necessarily disagree with your assertion, but I disagree with your reasoning. (Faith = Bad. Dogma = Bad. Religion -> (Faith ^ Dogma). Religion -> (Bad ^ Bad). Religion -> Bad.) Unfortunately, you never state why faith and dogma are dangerous. If you believe faith and dogma are dangerous because of what happened in Waco, you are missing the point. The Branch Davidians made the mistake of confusing the message with the messenger. They believed Koresh was a prophet, and therefore believed everything he said. The prob...	alt.atheism

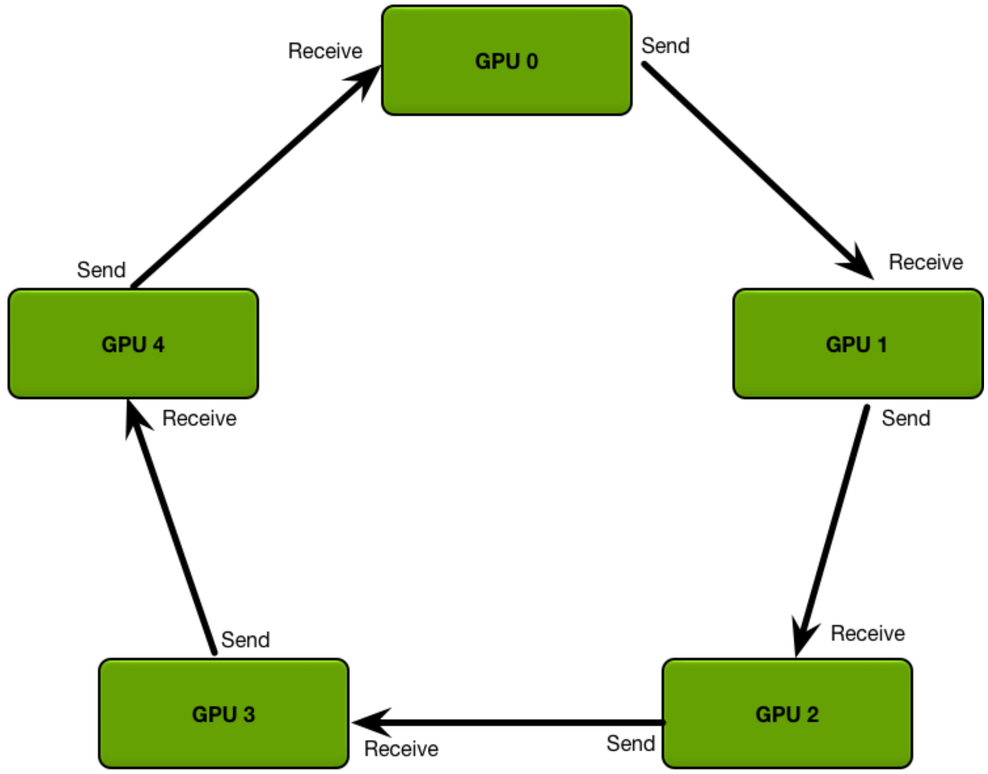
Showing the first 623 rows.

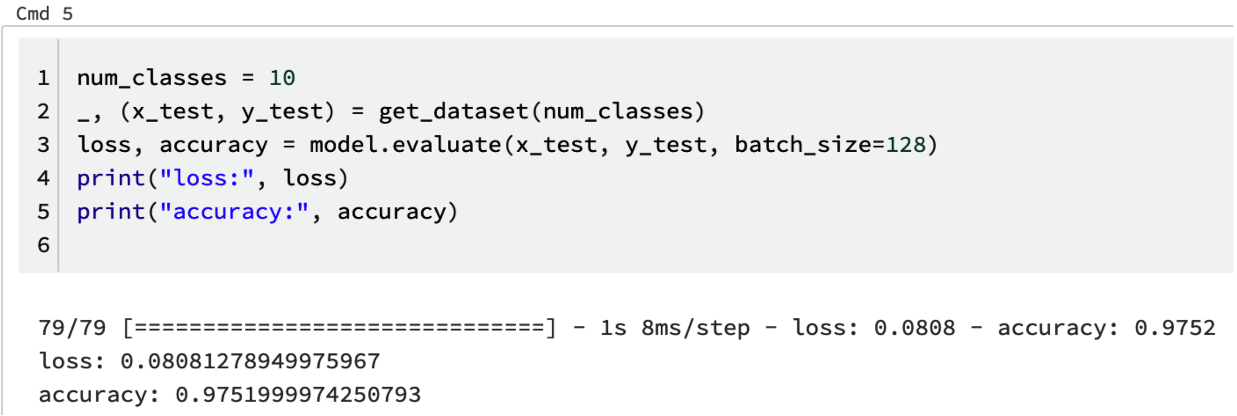


Registered Models > wine_1

Enable realtime model serving behind a REST API interface. This will launch a single-node cluster that will host all active versions of this model. Learn more.

Chapter 12: Distributed Deep Learning in Azure Databricks

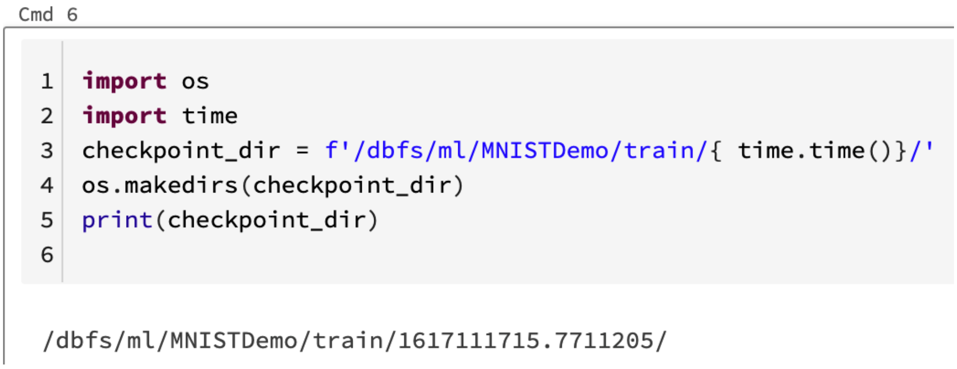




Cmd 5

```
num_classes = 10
_, (x_test, y_test) = get_dataset(num_classes)
loss, accuracy = model.evaluate(x_test, y_test, batch_size=128)
print("loss:", loss)
print("accuracy:", accuracy)

```

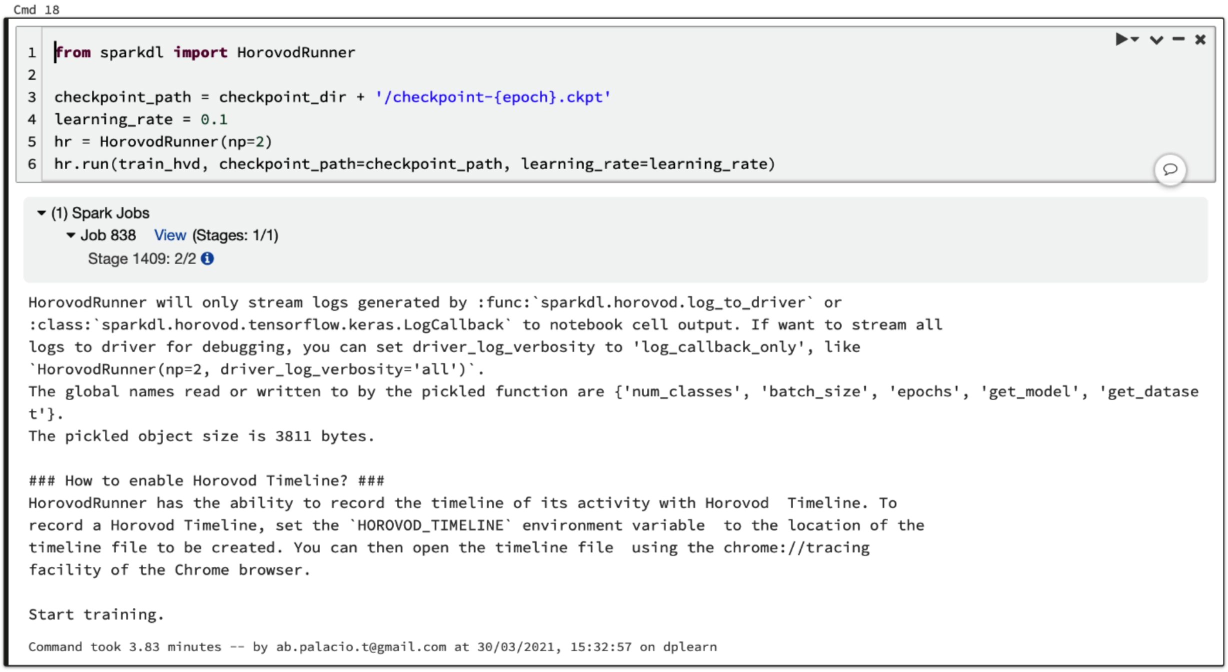
```
79/79 [==============================] - 1s 8ms/step - loss: 0.0808 - accuracy: 0.9752
loss: 0.08081278949975967
accuracy: 0.9751999974250793
```



Cmd 6

```
import os
import time
checkpoint_dir = f'/dbfs/ml/MNISTDemo/train/{ time.time()}/'
os.makedirs(checkpoint_dir)
print(checkpoint_dir)

```

```
/dbfs/ml/MNISTDemo/train/1617111715.7711205/
```



Cmd 18

```python
from sparkdl import HorovodRunner

checkpoint_path = checkpoint_dir + '/checkpoint-{epoch}.ckpt'
learning_rate = 0.1
hr = HorovodRunner(np=2)
hr.run(train_hvd, checkpoint_path=checkpoint_path, learning_rate=learning_rate)
```

```
HorovodRunner will only stream logs generated by :func:`sparkdl.horovod.log_to_driver` or
:class:`sparkdl.horovod.tensorflow.keras.LogCallback` to notebook cell output. If want to stream all
logs to driver for debugging, you can set driver_log_verbosity to 'log_callback_only', like
`HorovodRunner(np=2, driver_log_verbosity='all')`.
The global names read or written to by the pickled function are {'num_classes', 'batch_size', 'epochs', 'get_model', 'get_datase
t'}.
The pickled object size is 3811 bytes.

### How to enable Horovod Timeline? ###
HorovodRunner has the ability to record the timeline of its activity with Horovod  Timeline. To
record a Horovod Timeline, set the `HOROVOD_TIMELINE` environment variable  to the location of the
timeline file to be created. You can then open the timeline file  using the chrome://tracing
facility of the Chrome browser.

Start training.
```

Command took 3.83 minutes -- by ab.palacio.t@gmail.com at 30/03/2021, 15:32:57 on dplearn