Databricks Interview Questions & Answers

https://www.youtube.com/c/techlake

databricks

1)What is Databricks Runtime?

2)What are the types of Databricks Runtimes?

3)How to share Notebook to other Developers in Workspace?

4)How to access one notebook variable into other notebooks?

5)How to call one notebook from another notebook?

6)How to exit a notebook with returning some output data?

7)How to create Internal & External tables in Databricks?

8)How to Access ADLS or Blob Storage in Databricks?

9)What are the types of Cluster Modes in Databricks?

10)What are the types of workloads we can use in Standard type Cluster?

11)Can I use both Python 2 and Python 3 notebooks on the same cluster?

12)What is pool? Why we use pool? How to create pool in Databricks?

13)How many ways we can create variables in Databricks?

14) What are the limitations in Jobs?

15) Can I use %pip in notebook for installing packages or libraries?

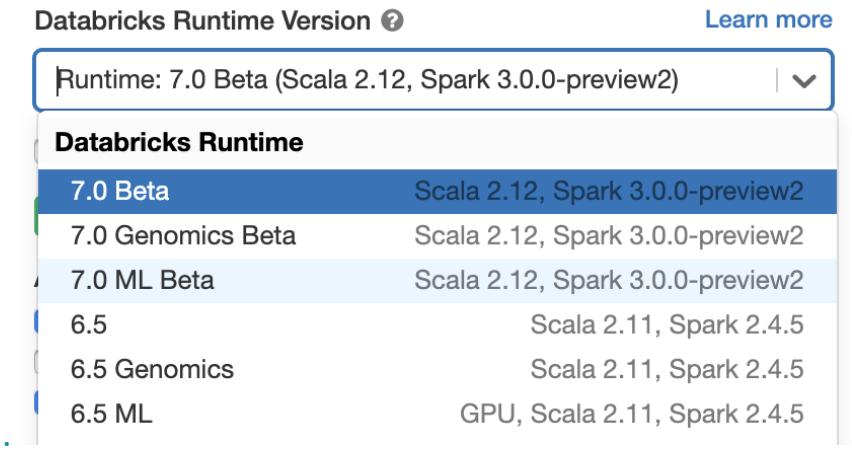
Databricks Interview Questions & Answers

https://www.youtube.com/c/techlake

1) What is Databricks Runtime?

The set of core components that run on the clusters managed by Databricks. Consists of the underlying Ubuntu OS, pre-installed languages and libraries (Java, Scala, Python, and R), Apache Spark, and various proprietary Databricks modules (e.g. DBIO, Databricks Serverless, etc.).

Azure Databricks offers several types of runtimes and several versions of those runtime types in the Databricks Runtime Version drop-down when you create or edit a cluster.



2) What are the types of Databricks Runtimes?

There are major 4 types of Databricks Runtimes.

a. Databricks Runtime for Standard
b. Databricks Runtime for Machine Learning
c. Databricks Runtime for Genomics
d. Databricks Light

Databricks Runtime for Standard

Databricks Runtime includes Apache Spark but also adds a number of components and updates that substantially improve the usability, performance, and security of big data analytics.

Databricks Runtime for Machine Learning

Databricks Interview Questions & Answers

https://www.youtube.com/c/techlake

Databricks Runtime ML is a variant of Databricks Runtime that adds multiple popular machine learning libraries, including TensorFlow, Keras, PyTorch, and XGBoost. ML also supports additional GPU supporting libraries clusters. **Graphics processing Units** Speeding up Machine Learning models. GPUs can drastically lower the cost because they support efficient parallel computation.

Databricks Runtime for Genomics

Databricks Runtime for Genomics is a variant of Databricks Runtime optimized for working with genomic and biomedical data.

Databricks Light

Databricks Light provides a runtime option for jobs that don't need the advanced performance, reliability, or autoscaling benefits provided by Databricks Runtime.

Databricks Light does not support:

- Delta Lake
- Autopilot features such as autoscaling
- Highly concurrent, all-purpose clusters
- Notebooks, dashboards, and collaboration features
- Connectors to various data sources and BI tools

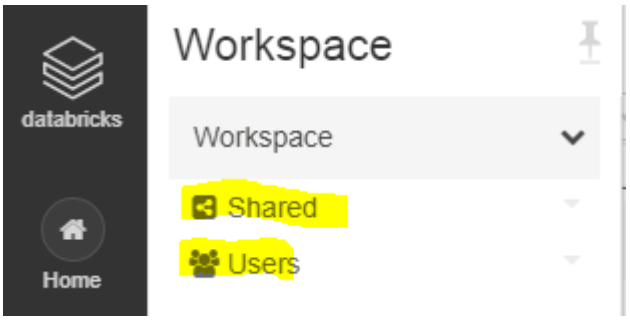
3) How to share Notebook to other Developers in Workspace?

There are two ways we can share notebooks to another developers.

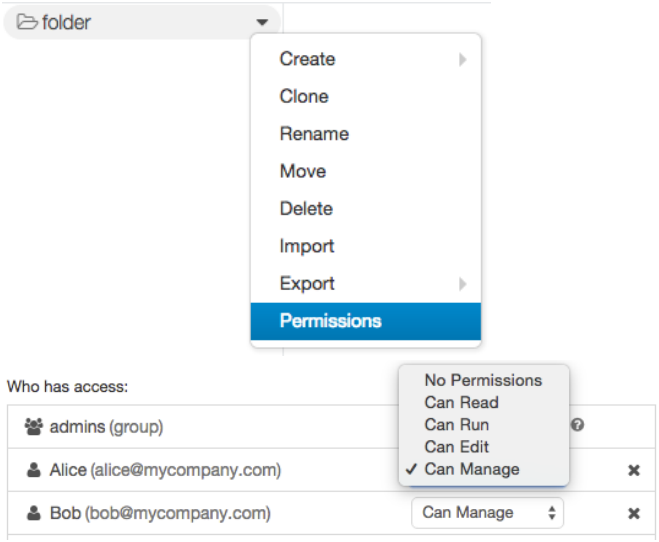
a) Copying notebook into from user folder to shared folder.

Databricks Interview Questions & Answers

https://www.youtube.com/c/techlake



b) Or Giving Access to other developers from current folder or user folder.



4) How to access one notebook variable into other notebooks?

If we run a one notebook into another notebook using %run we can use all functions, variables and imported libraries from callee notebook to caller notebook.

```
Python

%run /Users/path/to/notebookA
```

5) How to call one notebook from another notebook?

There are two ways we can call one notebook into another notebook.

1. %run notebook_name

   ```
   Python

   %run /Users/path/to/notebookA
   ```

   If we use the %run command. It will return from **calee notebooks** containing function and variable definitions. We can use those functions and variables in In **caller notebook.**
2. Dbutils.notebook.run(noteboo_name,timout_sec,aruguments_values)

You cannot use functions and variables. Only return value using arguments parameter.

```
returned_table = dbutils.notebook.run("notebook_exit_2", 60)
display(sqlContext.read.parquet(returned_table))
```

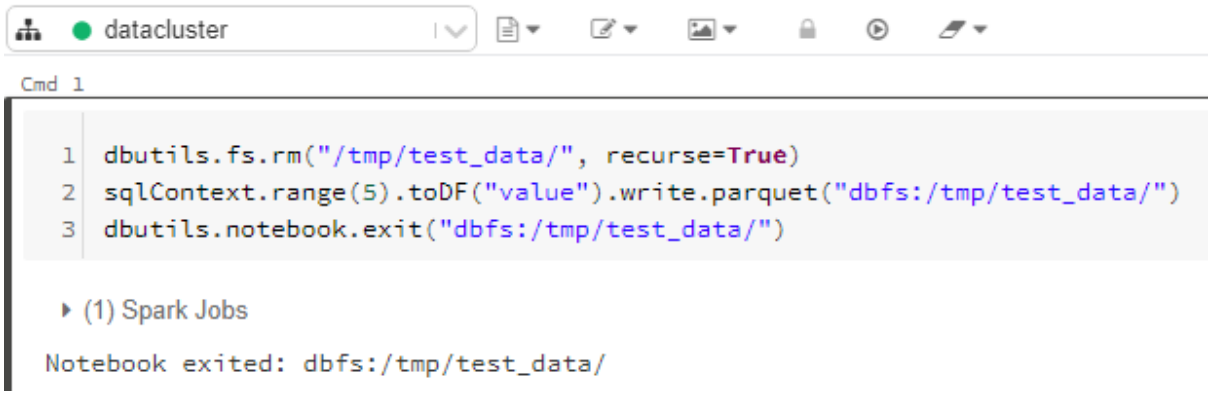
6) How to exit a notebook with returning some output data?

dbutils.notebook.exit (value: String): void -> This method lets you exit a notebook with a value.

In this example we have two notebooks. 1 for running exit and passing input value and another notebook for running 1st notebook using dbutlis.notebook.run() method. And storing into variable.

1st notebook

notebook_exit_2 (Python)



2nd notebook.

```
returned_table = dbutils.notebook.run("notebook_exit_2", 60)
display(sqlContext.read.parquet(returned_table))
```

7) How to create Internal & External tables in Databricks?

A Databricks database is a collection of tables. A Databricks table is a collection of structured data. You can cache, filter, and perform any operations supported by Apache Spark DataFrames on Databricks tables. You can query tables with Spark APIs and Spark SQL.

External Table.

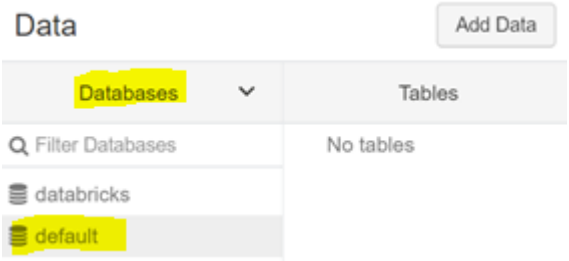
The table uses the custom directory specified with LOCATION. Queries on the table access existing data previously stored in the directory. When an EXTERNAL table is dropped, its data is not deleted from the file system. This flag is implied if LOCATION is specified.

```
CREATE EXTERNAL TABLE IF NOT EXISTS my_table (name STRING, age INT)
  COMMENT 'This table is created with existing data'
  LOCATION 'spark-warehouse/tables/my_existing_table'
```

Internal or Managed Table

Create a managed table using the definition/metadata of an existing table or view. The created table always uses its own directory in the default warehouse location.

```
CREATE TABLE boxes
  (width INT, length INT, height INT)
  USING PARQUET
  OPTIONS ('compression'='snappy')
```



8) How to Access ADLS or Blob Storage in Databricks?

We can Mount Azure Blob storage containers to DBFS and we can access through DBFS mount point.

You can mount a Blob storage container or a folder inside a container to Databricks File System (DBFS) using dbutils.fs.mount. The mount is a pointer to a Blob storage container, so the data is never synced locally.

```
dbutils.fs.mount(
  source = "wasbs://<container-name>@<storage-account-name>.blob.core.windows.net",
  mount_point = "/mnt/<mount-name>",
  extra_configs = {"<conf-key>":dbutils.secrets.get(scope = "<scope-name>", key = "<key-name>")})
```

Access files in your container as if they were local files, for example:

```
# python
df = spark.read.text("/mnt/<mount-name>/...")
df = spark.read.text("dbfs:/<mount-name>/...")
```

Once an account access key or a SAS is set up in your notebook, you can use standard Spark and Databricks APIs to read from the storage account

Set up an account access key:

```
spark.conf.set(
  "fs.azure.account.key.<storage-account-name>.blob.core.windows.net",
  "<storage-account-access-key>")
```

Set up a SAS for a container:

```
spark.conf.set(
  "fs.azure.sas.<container-name>.<storage-account-name>.blob.core.windows.net",
  "<complete-query-string-of-sas-for-the-container>")
```

9) What are the types of Cluster Modes in Databricks?

a) **Standard clusters**
b) **High concurrency clusters**

10) What are the types of workloads we can use in Standard type Cluster?

There are three types of workloads we can use in Standard type cluster. Those are

a. DATA ANALYTICS
b. DATA ENGINEERING
c. DATA ENGINEERING LIGHT

FEATURE	DATA ANALYTICS	DATA ENGINEERING	DATA ENGINEERING LIGHT
Apache Spark on Databricks platform	■	■	■
Job scheduling with libraries	■	■	■
Job scheduling with Notebooks	■	■	
Autopilot clusters	■	■	
Databricks Runtime for ML	■	■	
MLflow on Databricks Preview	■	■	
Databricks Delta	■	■	
Interactive clusters	■		
Notebooks and collaboration	■		
Ecosystem integrations	■		

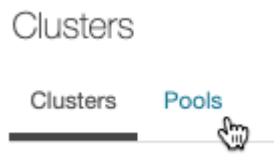
11) Can I use both Python 2 and Python 3 notebooks on the same cluster?

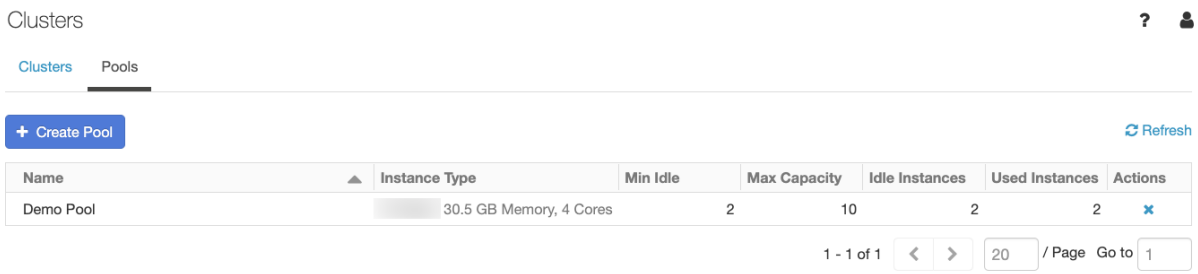
No. In Single Cluster you can use only one Python2 or Python 3. You cannot use both Python2 and python3 on same databricks cluster.

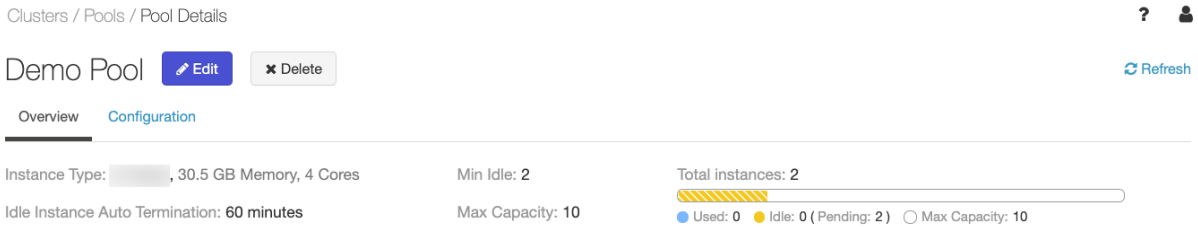
12) What is pool? Why we use pool? How to create pool in Databricks?

Pool is used to reduce cluster start time while auto scaling, you can attach a cluster to a predefined **pool of idle instances**. When attached to a pool, a cluster allocates its driver and worker nodes from the pool. If the pool does not have sufficient idle resources to accommodate the cluster's request, the pool expands by allocating new instances from the

instance provider. When an attached cluster is terminated, the instances it used are returned to the pool and can be reused by a different cluster.







13) How many ways we can create variables in Databricks?

There are different ways we can create variables in Databricks.

One method is creating variable and assigning values and calling that notebook into another notebook using **%run**

```
Python

%run /Users/path/to/notebookA
```

Another method is using dbutils.widgets method.

Use dbutils.widgets.text() or dbutils.widgets.dropdown() to create a widget parameter or variable

and dbutils.widgets.get() to get its bound value.

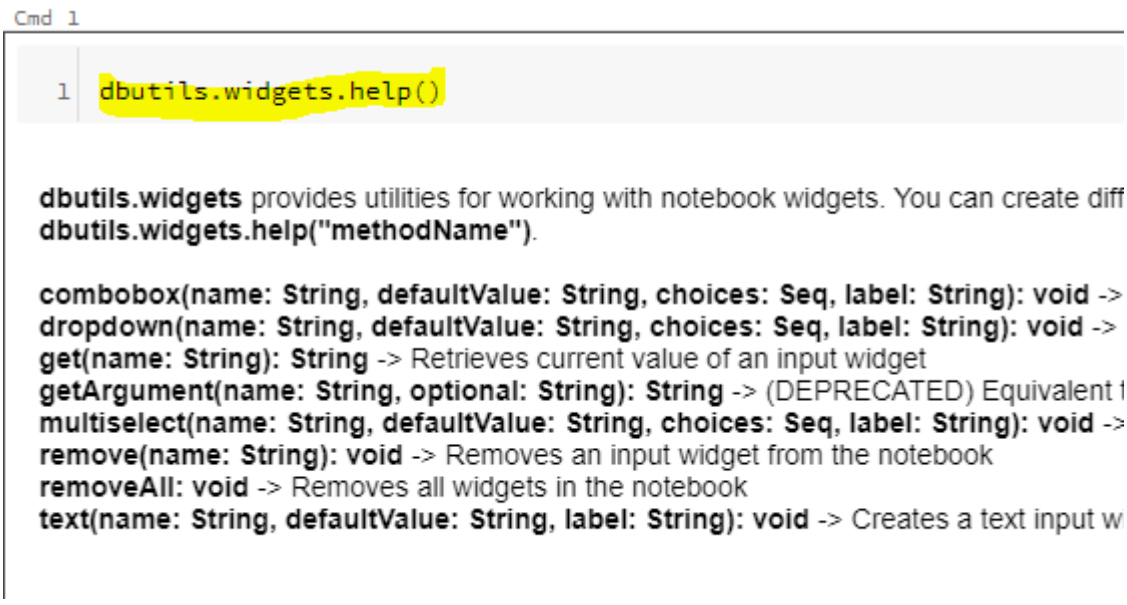
Databricks Interview Questions & Answers

https://www.youtube.com/c/techlake

If we type .help() it will show available methods in dbutils.widgets

Like **creating** text, **dropdown**, **combobox** variables and getting values using GET method.

Removing all variables using **dbutils.widgets.removeAll()**



Databricks Interview Questions & Answers

https://www.youtube.com/c/techlake

14) What are the limitations in Jobs?

A. The number of jobs is limited to 1000.
B. A workspace is limited to 150 concurrent (running) job runs.
C. A workspace is limited to 1000 active (running and pending) job runs.

15) Can I use %pip in notebook for installing packages or libraries?

Yes. We can use.

Creating a file for list of commands to be executed

```
dbutils.fs.put("dbfs:/home/myScripts/fast.ai", "conda install -c pytorch -c fastai fastai -y", True)
```

Installing using created file

```
%pip install -r /dbfs/home/myScripts/fast.ai
```

Installing using %pip

```
%pip install matplotlib
```

Uninstalling using %pip

```
%pip uninstall -y matplotlib
```

We can use conda also same as like %pip

```
%conda install matplotlib
```

Import the file to another notebook using Conda env update.

```
%conda env update -f /dbfs/myenv.yml
```

List the Python environment of a notebook

```
%conda list
```