Project Deployment and Execution in Databricks

Instructions for deploying and executing the Spark jobs in a Databricks cluster

**1) Pre-requisites**-

Before proceeding with the deployment and execution of the Spark jobs, make

sure you have the following:

• A Databricks account with appropriate permissions to create and run jobs.

• A Databricks cluster with the appropriate runtime version and configuration.

• Access to the source data and the location where the Delta table will be

created.

**2)** **Deployment-**

Create a Databricks cluster: In the Databricks workspace, go to the Clusters tab and create

a new cluster. Choose the appropriate configuration for your use case.

**3) Execution-**

Follow these steps to execute the Spark jobs in your Databricks cluster:

a) Upload the code:

Upload the code files to the workspace. You can either use the UI to upload the files or use

the dbutils.fs command in a notebook.

b) Install necessary libraries: If you're using any external libraries in your code, make sure to

install them on the cluster.

You can do this by going to the Libraries tab and installing the libraries.

To Execute Spark job in standalone mode:

1. Copy paste the code to file.py

2. Execute: spark-submit file.py

**4) Job Scheduling**

To schedule the Spark jobs to run at specific times or intervals, you can create a job in Databricks.

Follow these steps to create a job:

* Open the Databricks workspace and navigate to the "Jobs" tab.
* Click the "Create Job" button and fill in the job details, such as the name, schedule, and cluster configuration.
* Select the .py file that contains the Spark jobs.
* In the "Parameters" section, specify any parameters or arguments that need to be passed to the Spark jobs.
* Save the job.
* Once the job is created, it will run at the scheduled times or intervals and execute the Spark jobs.
* You can monitor the job status and view the job logs in the Databricks UI.