

ASSIGNMENT-11

NAME – NAGA ISWARYA LAKSHMI BHOGADI

COMPANY – DXC TECHNOLOGY

BATCH – DXC-262-ANALYTICS-B12-AZURE

ROLLNO – DXC-262AB-1205

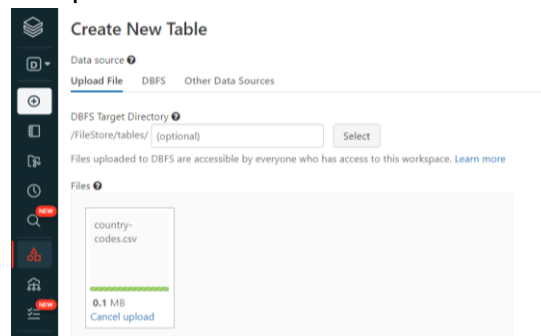
DATE OF SUBMISSION – 15th JUNE 2022

TRAINER NAME – MR. AJAY KUMAR

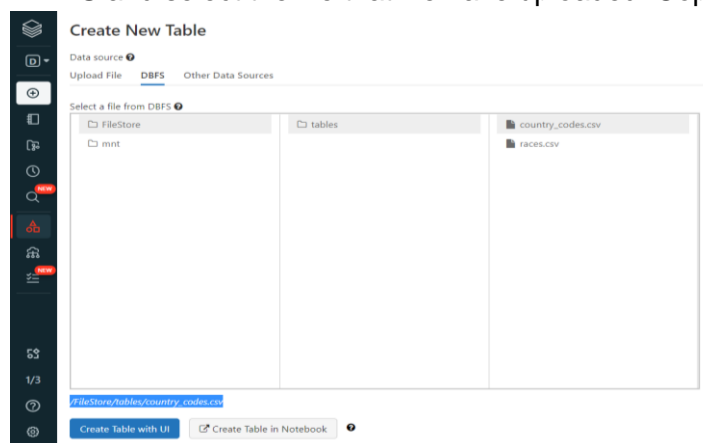
1. Using archive1.zip file - please ingest data into Databricks DBFS path & query the data, redesign columns accordingly using Dataframe commands - display with notebooks accordingly.

File Being used: country.csv

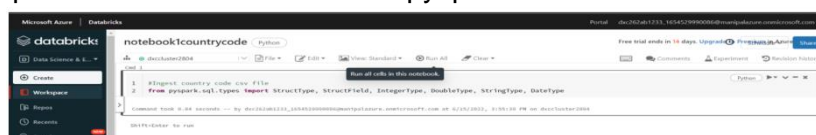
First, login to your Azure Portal and create a Databricks workspace. After, Open the Databricks workspace and create cluster. Now, create a notebook by clicking on the create Notebook option from the side panel. After creating the notebook, upload the data into the Databricks by dragging and dropping the required file.



Now, click on DBFS and select the file that we have uploaded. Copy the file path.



Import the required fields and features from pyspark.



Now, we need to ingest the schema

```
1 #include the schema
2 country_schema = StructType(fields=[StructField("marc",StringType(),True),
3                                         StructField("Capital",StringType(),True),
4                                         StructField("M49",IntegerType(),True),
5                                         StructField("Regioncode",IntegerType(),True),
6                                         ])
7
```

Command took 0.03 seconds -- by dxc262ab1233_165452999008@manipalazure.onmicrosoft.com at 6/15/2022, 4:04:19 PM on dxccluster2004

Shift+Enter to run

Then we need to create a data frame

```
> Command took 0.03 seconds -- by dxc262ab1233_165452999008@manipalazure.onmicrosoft.com at 6/15/2022, 4:04:19 PM on dxccluster2004
Cnd 3
1 #creating a data frame
2 country_df = spark.read \
3     .option("header",True) \
4     .schema(country_schema) \
5     .csv("/filestore/tables/country_codes.csv")
6
7 country_df: pyspark.sql.dataframe.DataFrame
8   marc: string
9   Capital: string
10  M49: integer
11  Regioncode: integer
12
13 Command took 0.14 seconds -- by dxc262ab1233_165452999008@manipalazure.onmicrosoft.com at 6/15/2022, 4:09:57 PM on dxccluster2004
14
15 Shift+Enter to run
```

We need to add the ingestion date

```
Cnd 5
1 #add ingestion date to the data frame
2 country_final_df = country_df.withColumn("ingestion_date",current_timestamp())
3
4 country_final_df: pyspark.sql.dataframe.DataFrame
5   marc: string
6   Capital: string
7   M49: integer
8   Regioncode: integer
9   ingestion_date: timestamp
10
11 Command took 0.04 seconds -- by dxc262ab1233_165452999008@manipalazure.onmicrosoft.com at 6/15/2022, 4:18:51 PM on dxccluster2004
12
13 Shift+Enter to run
```

Processed container in parquet format

```
Cnd 7
1 country_renamed_final_df.write.mode('overwrite').partitionBy('MARC').parquet('/mnt/formalaid/processed/country')
2
3 Cancel *Running command...
4 (1) Spark Jobs
```

To display the data we have been included.

```
1 display(spark.read.parquet("/mnt/formalaid/processed/country"))
```

(3) Spark Jobs

- Job 2 View (Stages: 1/7)
- Job 3 View (Stages: 1/7)
- Job 4 View (Stages: 1/7)
- Job 5 View (Stages: 1/7)
- Job 6 View (Stages: 1/7)

Table Data Profile

	CAPITAL	m49	Regioncode	INGESTION_DATE	MARC
1	1	null	null	2022-06-15T10:58:27.936+0000	PUR
2	7	null	null	2022-06-15T10:58:27.936+0000	KAZ
3	7	null	null	2022-06-15T10:58:27.936+0000	RUS
4	1	null	null	2022-06-15T10:58:27.936+0000	CAN
5	1	null	null	2022-06-15T10:58:27.936+0000	USA
6	95	null	null	2022-06-15T10:58:27.936+0000	MVA
7	11	null	null	2022-06-15T10:58:27.936+0000	NFD

Showing all 250 rows.

2. Using archive2.zip file - please ingest data into Databricks DBFS path & query the data, redesign columns accordingly using Dataframe commands - display with notebooks accordingly.

File Being Used: nces330.20.csv

Follow steps in 1 and create a new notebook by clicking on the create Notebook option from the side panel.

Create Notebook

Name

archive2Notebook

Default Language

Python

Cluster

clus1219

Cancel

Create

After creating the notebook, upload the data into the Databricks by dragging and dropping the required file.

Create New Table

Data source

Upload File

DBFS

Other Data Sources

DBFS Target Directory

/FileStore/tables/ (optional)

Select

Files

neces330_20.csv

0.2 MB

Remove file

File uploaded to /FileStore/tables/neces330_20.csv

Create Table with UI

Create Table in Notebook

Import required fields and features from pyspark

```

1 from pyspark.sql.types import StructType, StructField, IntegerType, StringType

```

Command took 0.04 seconds -- by dxc262ab1225_1654538816505@manpalazure.onmicrosoft.com at 6/15/2022, 5:24:11 PM on clus1219

Create a schema

```

1 races_schema = StructType(fields=[StructField("raceId",IntegerType(),False),
2                                     StructField("year",IntegerType(),True),
3                                     StructField("round",IntegerType(),True),
4                                     StructField("circuitId",IntegerType(),True),
5                                     StructField("name",StringType(),True),
6                                     StructField("date",DateType(),True),
7                                     StructField("time",StringType(),True),
8                                     StructField("url",StringType(),True),
9                                     ])

```

Command took 0.04 seconds -- by dxc262ab1225_1654538816505@manpalazure.onmicrosoft.com at 6/15/2022, 5:18:02 PM on clus1219

Ingest the data

```

1 races_df = spark.read \
2     .option("header", True) \
3     .schema(races_schema) \
4     .csv("/FileStore/tables/races.csv")

```

races_df: pyspark.sql.dataframe.DataFrame
 raceId: integer
 year: integer
 round: integer
 circuitId: integer
 name: string
 date: date
 time: string

Import col from sql functions

```

1 from pyspark.sql.functions import col,lit

```

Command took 0.03 seconds -- by dxc262ab1225_1654538816505@manpalazure.onmicrosoft.com at 6/15/2022, 5:34:02 PM on clus1219

```

1 neces330_20_selected_df = neces330_20_df.select(col('Year'),
2                                                  col('State'),col('Expense'))

```

neces330_20_selected_df: pyspark.sql.dataframe.DataFrame
 Year: integer
 State: string
 Expense: string

Command took 0.06 seconds -- by dxc262ab1225_1654538816505@manpalazure.onmicrosoft.com at 6/15/2022, 5:37:09 PM on clus1219

Displays the data we have been included.



The screenshot shows a Databricks notebook interface. At the top, a code cell contains the command `display(nces330_20_selected_df)`. Below the code, the notebook displays a table with the following data:

	Year	State	Expense
1	2013	Alabama	Fees/Tuition
2	2013	Alabama	Room/Board
3	2013	Alabama	Fees/Tuition
4	2013	Alabama	Fees/Tuition
5	2013	Alabama	Room/Board
6	2013	Alabama	Fees/Tuition
7	2013	Alabama	Fees/Tuition

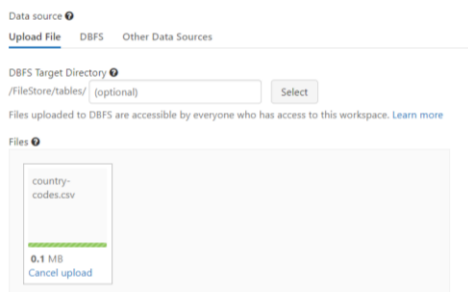
Below the table, it says "Truncated results, showing first 1000 rows. Click to re-execute with maximum result limits." At the bottom, a status bar indicates "Command took 0.39 seconds -- by dxc262ab1225_1654538016585@manipalazure.onmicrosoft.com at 6/15/2022, 5:39:23 PM on clus1219".

3. Using archive3.zip file - please ingest data into Databricks DBFS path & query the data, redesign columns accordingly using Dataframe commands - display with notebooks accordingly.

File Being used: final_data.csv

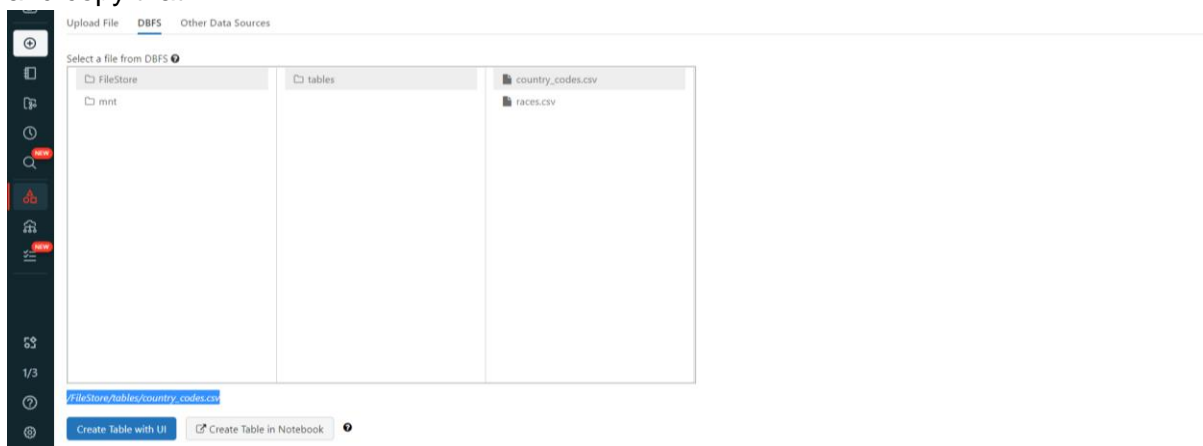
create a new notebook by clicking on the create Notebook option from the side panel. After creating the notebook, ingest the data into the Databricks by dragging and dropping the required file in the **drag & drop** region.

Create New Table



The screenshot shows the "Create New Table" interface. It has tabs for "Data source", "Upload File", "DBFS", and "Other Data Sources". The "Upload File" tab is selected. Below the tabs, there is a "DBFS Target Directory" field with the value `/FileStore/tables/` and a "Select" button. Below this, it says "Files uploaded to DBFS are accessible by everyone who has access to this workspace. Learn more". At the bottom, there is a "Files" section showing a file named "country_codes.csv" with a size of "0.1 MB" and a "Cancel upload" button.

Later, click on DBFS and select the file that you have dropped. This will give you the file path and copy that.



Import the required fields and features from pyspark.

```

1 from pyspark.sql.types import StructType, StructField, IntegerType, StringType

Command took 0.04 seconds -- by dxc262ab1225_1654530016505@manipalazure.onmicrosoft.com at 6/15/2022, 5:12:11 PM on clus1219

Cnd 2

1 final_data_schema = StructType(fields=[StructField("tweet_text",StringType(),False),
2                                     StructField("emotion_in_tweet_is_directed_at",StringType(),True),
3                                     StructField("is_there_an_emotion_directed_at_a_brand_or_product",StringType(),True),
4                                     ])

Command took 0.05 seconds -- by dxc262ab1225_1654530016505@manipalazure.onmicrosoft.com at 6/15/2022, 5:17:53 PM on clus1219

Cnd 3

1 final_data_df = spark.read \
2 .option("header" , True) \
3 .schema(final_data_schema) \
4 .csv("/FileStore/tables/final_data.csv")

Python ▶ ▼ ✕

final_data_df: pyspark.sql.dataframe.DataFrame
tweet_text: string
emotion_in_tweet_is_directed_at: string
is_there_an_emotion_directed_at_a_brand_or_product: string

Command took 0.19 seconds -- by dxc262ab1225_1654530016505@manipalazure.onmicrosoft.com at 6/15/2022, 5:51:51 PM on clus1219

Cnd 4

1 from pyspark.sql.functions import col,lit

Command took 0.03 seconds -- by dxc262ab1225_1654530016505@manipalazure.onmicrosoft.com at 6/15/2022, 5:34:02 PM on clus1219

Cnd 5

1 final_data_selected_df = final_data_df.select(col('tweet_text'),
2 col('emotion_in_tweet_is_directed_at').alias('emotion_towards'),col('is_there_an_emotion_directed_at_a_brand_or_product').alias('is_there_a_brand'))

Python ▶ ▼ ✕

final_data_selected_df: pyspark.sql.dataframe.DataFrame
tweet_text: string
emotion_towards: string
is_there_a_brand: string

Command took 0.06 seconds -- by dxc262ab1225_1654530016505@manipalazure.onmicrosoft.com at 6/15/2022, 5:54:25 PM on clus1219

Cnd 6

```

Display the data.

```

1 display(final_data_selected_df)

Python ▶ ▼ ✕

```

(1) Spark Jobs

Table Data Profile

	tweet_text	emotion_towards	is_there_a_brand
1	@wesley83 I have a 3G iPhone. After 3 hrs tweeting at #RISE_Austin, it was dead! I need to upgrade. Plugin stations at #SXSW.	iPhone	Negative emotion
2	@jessedee Know about @fludapp ? Awesome iPad/iPhone app that you'll likely appreciate for its design. Also, they're giving free Ts at #SXSW	iPad or iPhone App	Positive emotion
3	@swonderlin Can not wait for iPad 2 also. They should sale them down at #SXSW.	iPad	Positive emotion
4	@sxsw I hope this year's festival isn't as crashy as this year's iPhone app. #sxsw	iPad or iPhone App	Negative emotion
5	@sxstate great stuff on Fri #SXSW: Marissa Mayer (Google), Tim O'Reilly (tech books/conferences) & Matt Mullenweg (Wordpress)	Google	Positive emotion
6	@teachntech00 New iPad Apps For #SpeechTherapy And Communication Are Showcased At The #SXSW Conference http://ht.ly/49n4M #iear #edchat #asd	null	No emotion toward brand or product
7	null	null	No emotion toward brand or product

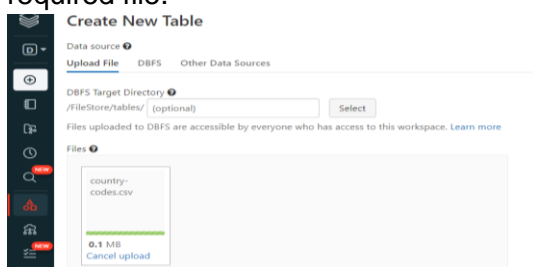
Truncated results, showing first 1000 rows.
Click to re-execute with maximum result limits.

Command took 0.56 seconds -- by dxc262ab1225_1654530016505@manipalazure.onmicrosoft.com at 6/15/2022, 5:55:03 PM on clus1219

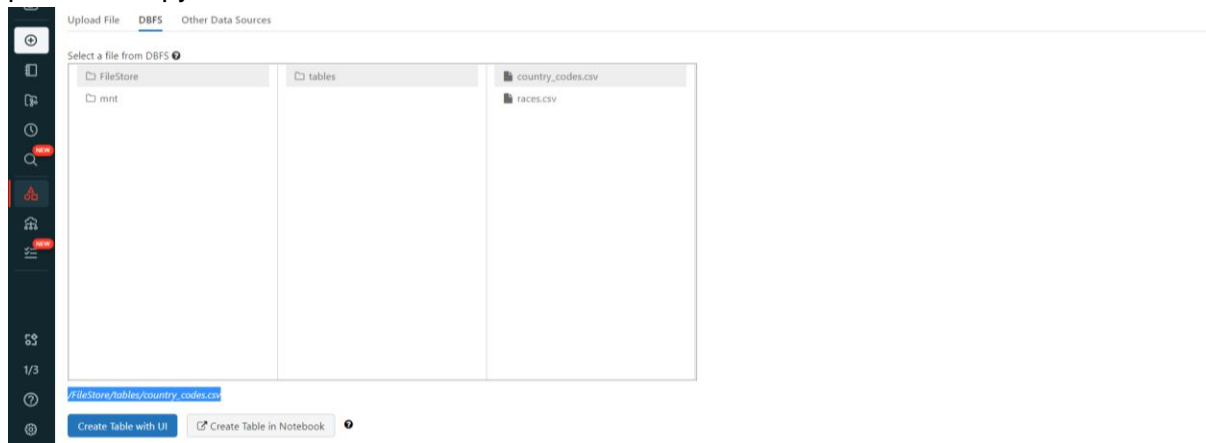
Cnd 7

4. Using archive4.zip file - please ingest data into Databricks DBFS path & query the data, redesign columns accordingly using Dataframe commands - display with notebooks accordingly.

create a notebook by clicking on the create Notebook option from the side panel. After creating the notebook, ingest the data into the Databricks by dragging and dropping the required file.



Later, click on DBFS and select the file that you have dropped. This will give you the file path and copy that.



Import the required fields and features from pyspark.

```
from pyspark.sql.types import StructType, StructField, IntegerType, StringType
```

```
from pyspark.sql.types import StructType, StructField, IntegerType, StringType
```

```
SEntFiN-v1_1_schema = StructType(fields=[StructField("S No.", IntegerType(), False),
                                             StructField("Title", StringType(), True),
                                             StructField("Decisions", StringType(), True),
                                             StructField("Words", IntegerType(), True),
```

```
)
```

```
SEntFiN-v1_1_df = spark.read \
    .option("header", True) \
    .schema(SEntFiN-v1_1_schema) \
    .csv("/FileStore/tables/ SEntFiN-v1_1.csv")
```

```
from pyspark.sql.functions import col,lit
```

```
SEntFiN-v1_1_selected_df = SEntFiN-v1_1_df.select(col('S No'),
                                                    col('Title'),col('Words'))
```

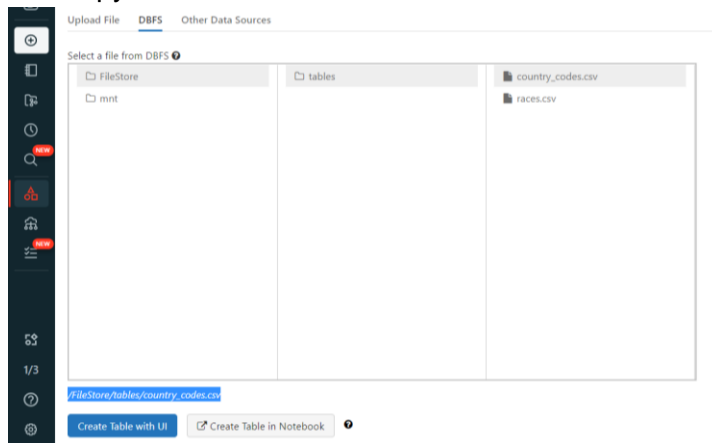
```
display(SEntFiN-v1_1_df)
```

5:Using archive5.zip file - please ingest data into Databricks DBFS path & query the data, redesign columns accordingly using Dataframe commands - display with notebooks accordingly.

create a notebook by clicking on the create Notebook option from the side panel.

After creating the notebook, ingest the data into the Databricks by dragging and dropping the required file.

Later, click on DBFS and select the file that you have dropped. This will give you the file path and copy that.



Import the required fields and features from pyspark.

```
from pyspark.sql.types import StructType, StructField, IntegerType, StringType, FloatType

cancer_death_rates_schema = StructType(fields=[StructField("Entity",StringType(),False),
                                                StructField("Code",StringType(),True),
                                                StructField("Year",IntegerType(),True),
                                                StructField("Deaths - Neoplasms - Sex: Both - Age: Age-standardized
(Rate)",FloatType(),True),
])
```

```
Cancer_death_rates_df = spark.read \
.option("header" , True) \
.schema(cancer_death_rates_schema) \
.csv("/FileStore/tables/ cancer_death_rates.csv")
```

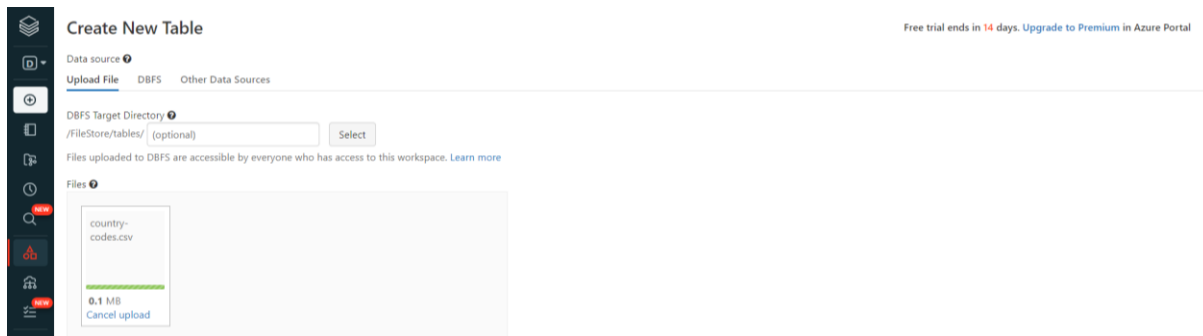
```
from pyspark.sql.functions import col,lit
```

```
cancer_death_rates_selected_df = cancer_death_rates_df.select(col(' Entity'),
                                                             col(' Year'),col(' Deaths - Neoplasms - Sex: Both - Age: Age-
standardized (Rate)').alias('Deaths'))
```

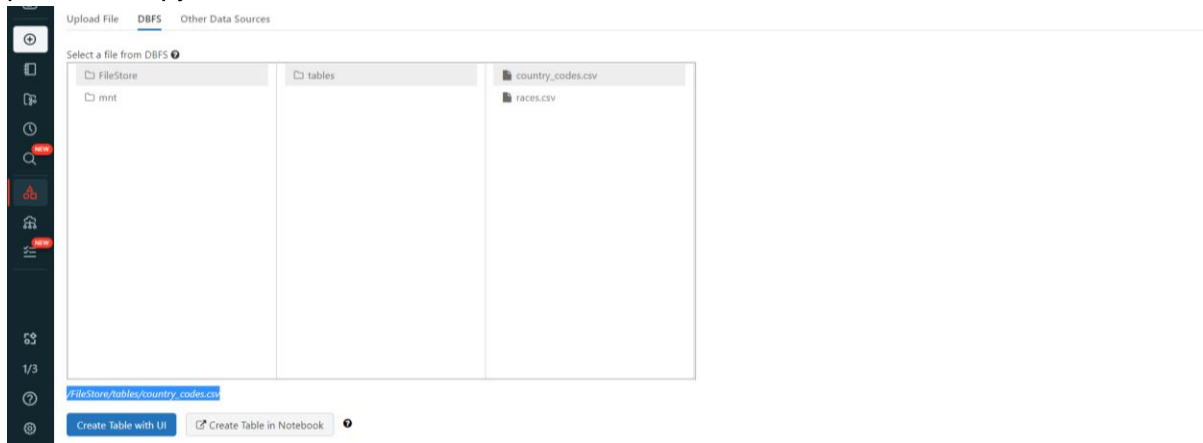
```
display(cancer_death_rates_df)
```

6. Using archive6.zip file - please ingest data into Databricks DBFS path & query the data, redesign columns accordingly using Dataframe commands - display with notebooks accordingly.

create a notebook by clicking on the create Notebook option from the side panel. After creating the notebook, ingest the data into the Databricks by dragging and dropping the required file.



Later, click on DBFS and select the file that you have dropped. This will give you the file path and copy that.



Import the required fields and features from pyspark.

```
from pyspark.sql.types import StructType, StructField, IntegerType, StringType, FloatType
```

```
inflation_gdp_schema = StructType(fields=[StructField("Country ",StringType(),False),
                                             StructField("Country Code ",StringType(),True),
                                             StructField("Year ",IntegerType(),True),
                                             StructField("Inflation ",FloatType(),True),
```

```
])
```

```
inflation_gdp_df = spark.read \
    .option("header" , True) \
    .schema(inflation_gdp_schema) \
    .csv("/FileStore/tables/ inflation_gdp.csv")
```

```
from pyspark.sql.functions import col,lit
```

```
inflation_gdp_selected_df = inflation_gdp_df.select(col('S No'),
                                                    col('Title'),col('Words'))
```

```
display(inflation_gdp_df)
```