Assignment – Day 17

-Sarthak Niranjan Kulkarni (Maverick)

- [sarthakkul2311@gmail.com](mailto:sarthakkul2311@gmail.com) - (+91) 93256 02791

**28/11/2024 (Thursday)**

**Practice of Loading Data:-**

1. **"Load and Display Loan Table Data"**

🡪 # data =spark.read.table("samples.nyctaxi.trips")

datatable =spark.read.table("hive\_metastore.default.loan")

datatable.display()

A screenshot of a computer

Description automatically generated

1. **"Create RDDs and Load Delta Tables"**

🡪 # to create rdds and  dataframe

from pyspark import SparkContext

from pyspark.sql import SparkSession

# Initialize SparkContext and SparkSession

sc = SparkContext.getOrCreate()

spark = SparkSession.builder.appName('pyspark first program').getOrCreate()

data = spark.read.format("delta").load("dbfs:/databricks-datasets/nyctaxi-with-zipcodes/subsampled")

datatable = spark.read.format("delta").load("dbfs:/user/hive/warehouse/loan")

data.display()

datatable.display()

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Summary of Loading Data: -**

In the first code block, I used PySpark to create a Spark session, which is essential for processing data in Databricks. I then loaded the loan data stored in a Delta format table from the Databricks File System (DBFS) into a DataFrame using spark.read.format("delta"). Delta format offers several advantages such as ACID transactions and time travel, making it a reliable choice for working with large datasets in Databricks. After loading the data, I displayed it to visually inspect the information, which allows me to quickly understand the structure of the dataset.

In the second code block, I accessed two tables from the Databricks metastore using spark.table(). This method allows me to easily query tables that have already been registered in the metastore, which is a centralized place to manage metadata for structured data. The first table, loan\_table, was loaded from the default schema (hive\_metastore.default), while the second table, trips\_table, came from the samples.nyctaxi schema. By displaying both tables, I can examine the content and start analyzing them for insights. These two tables represent two different kinds of data: financial data in the loan\_table and transportation data in the trips\_table.

This entire process showcases the simplicity and flexibility of working with various data formats (like Delta) and managing data in Databricks using PySpark, which is a powerful tool for big data analysis. With this setup, I can perform various analyses, transformations, and queries on the data to derive meaningful insights.