import findspark

findspark.init()

findspark.find()

from pyspark.sql import SparkSession

spark = SparkSession.builder.getOrCreate()

pf = spark.sql("select 'spark' as hello")

pf.show()

from pyspark.sql import \*

rf= spark.read.format("csv").load("C:/Users/ruhim/OneDrive/Pictures/Ruhh/BIG DATA/hiveproject/project2/Riders\_data.csv",header="true",inferSchema="true")

rf.show()

rf.describe()

rf.createOrReplaceTempView("rider")

1. Most searched pick-up point

> spark.sql("select distinct(source) from rider").show()

2. Rider name and contact details with rating>=4

> spark.sql("select Riders\_Name, Age,Rid\_cont from rider where Rating\_cus>=4").show()

3. Rider’s name who received the payment through Uber Wallet

> spark.sql("select Riders\_Name from rider where Payment\_mode='Uber wallet'").show()

4.Rider details whose customers do not have trusted contacts

> spark.sql("select Riders\_Name, Rid\_cont from rider where Trusted\_contact='NO'").show()

5. Rider details with travel miles more than 10

> spark.sql("select Riders\_Name, Miles from rider where Miles>10").show()

6. Ride category with more than 10 miles

> spark.sql("select Ride\_category, Miles from rider where Miles>10").show()

7. Pick up and drop points and ride category with estimated costing more than 100

> spark.sql("select Source, Destination,Ride\_category from rider where Est\_Costing>100").show()

8. Rider details with Arrived ride status

> spark.sql("select Riders\_Name, Rid\_cont from rider where Ride\_Status='Arrived'").show()

9. Rider details with cancelled ride status

> spark.sql("select Riders\_Name, Rid\_cont from rider where Ride\_Status='Cancelled'").show()

10. Rider details whose vaccination is done

> spark.sql("select Riders\_Name, Rid\_cont, Vaccine\_Cus from rider where Vaccine\_Cus='YES'").show()

11. Rider details whose age is greater than 45 and vaccination is not done

> spark.sql("select Riders\_Name, Rid\_cont, Vaccine\_Cus, Age from rider where Vaccine\_Cus='NO' and Age>45").show()

12. Start time and end time of the trip with miles less than 10

> spark.sql("select Start\_time, End\_time from rider where Miles<10").show()

13. Loaction ID of the pick up point whose final cost is greater than 100

> spark.sql("select Location\_Id from rider where Final\_cost>100").show()

14. Number of vaccinated people out of records

> spark.sql("select count(Riders\_Name) from rider where Vaccine\_Cus='YES'").show()

15. Rider and passenger details who received and paid through cash respectively

> spark.sql("select Riders\_Name, Name from rider where Payment\_mode='cash'").show()

16. Loaction ID and estimated cost of the trip with the price<100

> spark.sql("select Location\_Id,Est\_Costing from rider where Est\_Costing<100").show()