BANKING PROJECT

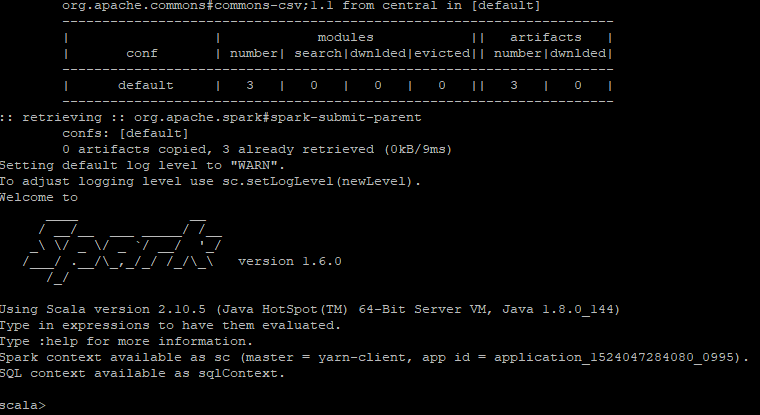
1. Load data and create Spark data frame

->a)first we have to intiate the spark shell

**Code:**

spark-shell --conf spark.ui.port=1234 --packages com.databricks:spark-csv\_2.11:1.5.0

**output:**

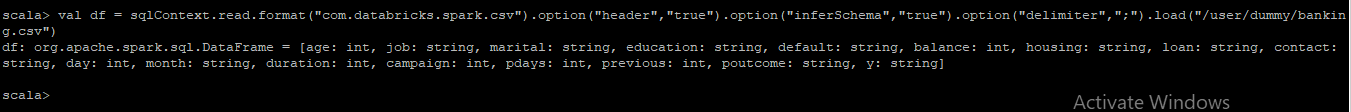


b)Loading the data and ceating spark dataframe

**CODE:**

**val df = sqlContext.read.format("com.databricks.spark.csv").option("header","true").option("inferSchema","true").option("delimiter",";").load("/user/dummy/banking.csv")**

**OUTPUT:**

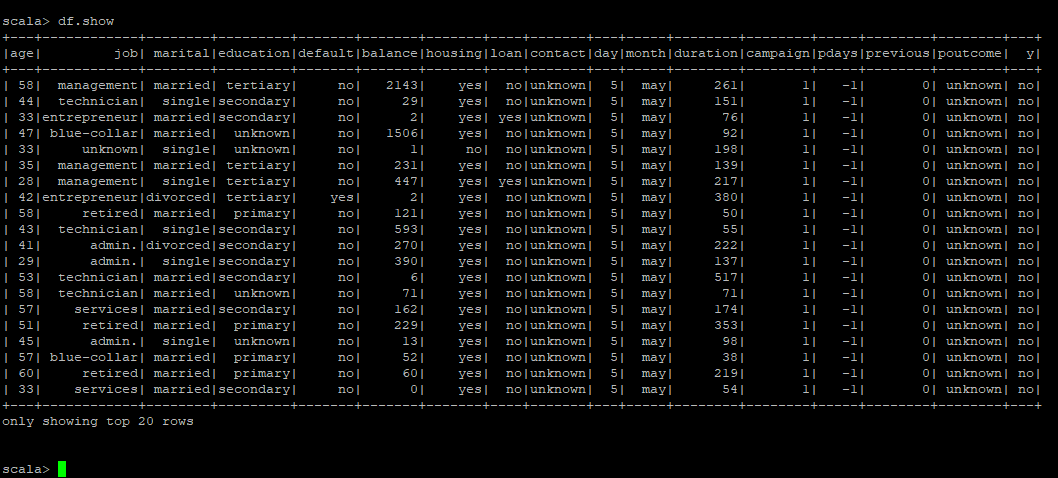


->now to view the dataframe

**Code:**

df.show

**OUTPUT:**



1. Give marketing success rate. (No. of people subscribed / total no. of entries)

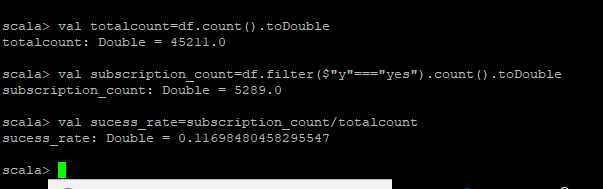
**Code:**

val totalcount=df.count().toDouble

val subscription\_count=df.filter($"y"==="yes").count().toDouble

val sucess\_rate=subscription\_count/totalcount

**OUTPUT:**



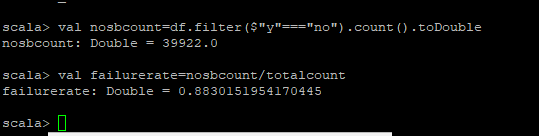
2a) Give marketing failure rate

**Code:**

val nosbcount=df.filter($"y"==="no").count().toDouble

val failurerate=nosbcount/totalcount

**output:**

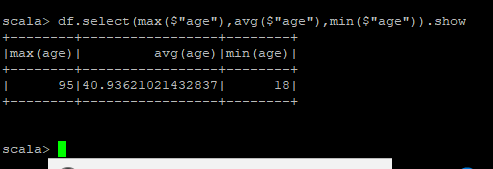


3) Maximum, Mean, and Minimum age of average targeted customer

**Code:**

df.select(max($"age"),avg($"age"),min($"age")).show

**OUTPUT:**



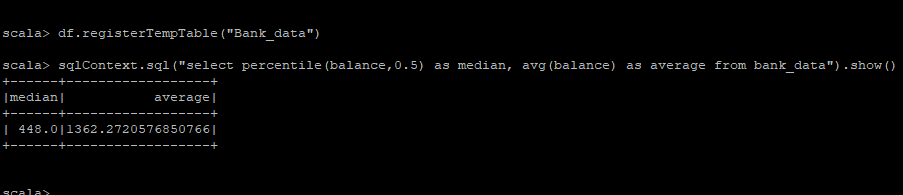
4)Check quality of customers by checking average balance, median balance of customers

**Code:**

df.registerTempTable("Bank\_data")

sqlContext.sql("select percentile(balance,0.5) as median, avg(balance) as average from bank\_data").show()

**OUTPUT:**

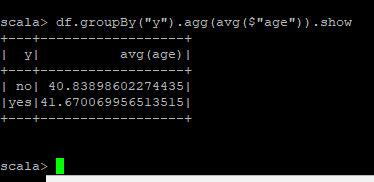


5. Check if age matters in marketing subscription for deposit

**Code:**

df.groupBy("y").agg(avg($"age")).show

**OUTPUT:**

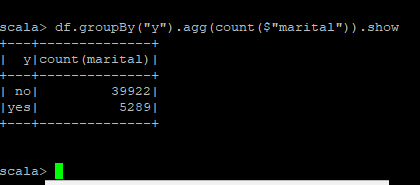


6. Check if marital status mattered for subscription to deposit.

**CODE:**

df.groupBy("y").agg(count($"marital")).show

**output:**

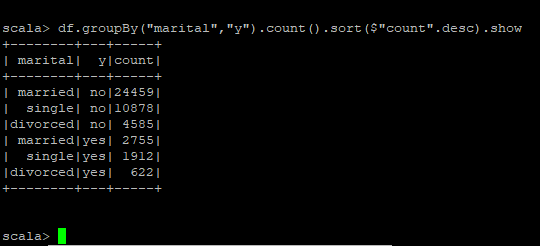


7. Check if age and marital status together mattered for subscription to deposit scheme

**CODE:**

df.groupBy("marital","y").count().sort($"count".desc).show

**OUTPUT:**



8. Do feature engineering for column—age and find right age effect on campaign

**Code:**

df.groupBy("age","y").count().sort($"count".desc).show

output:

