<u>Market Analysis in Banking Domain</u>

spark-shell

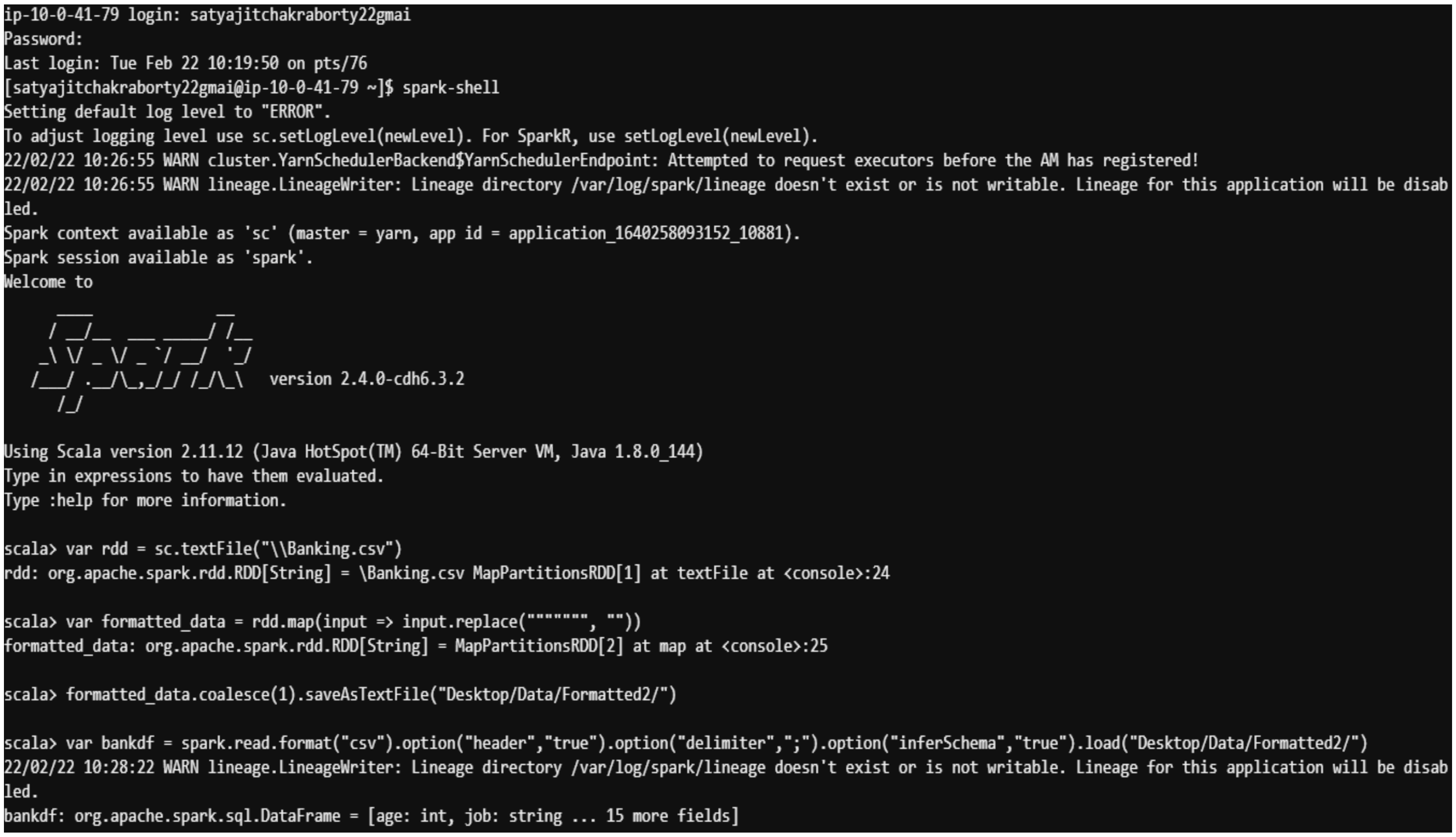
<u>Question1:</u> Load data and create Spark data frame

var rdd = sc.textFile("\\Banking.csv")

var formatted_data = rdd.map(input => input.replace("""""""", ""))

formatted_data.coalesce(1).saveAsTextFile("Desktop/Data/Formatted2/")

var bankdf =spark.read.format("csv").option("header","true").option("delimiter",";").option("inferSchema","true").load("Desktop/Data/Formatted2/")

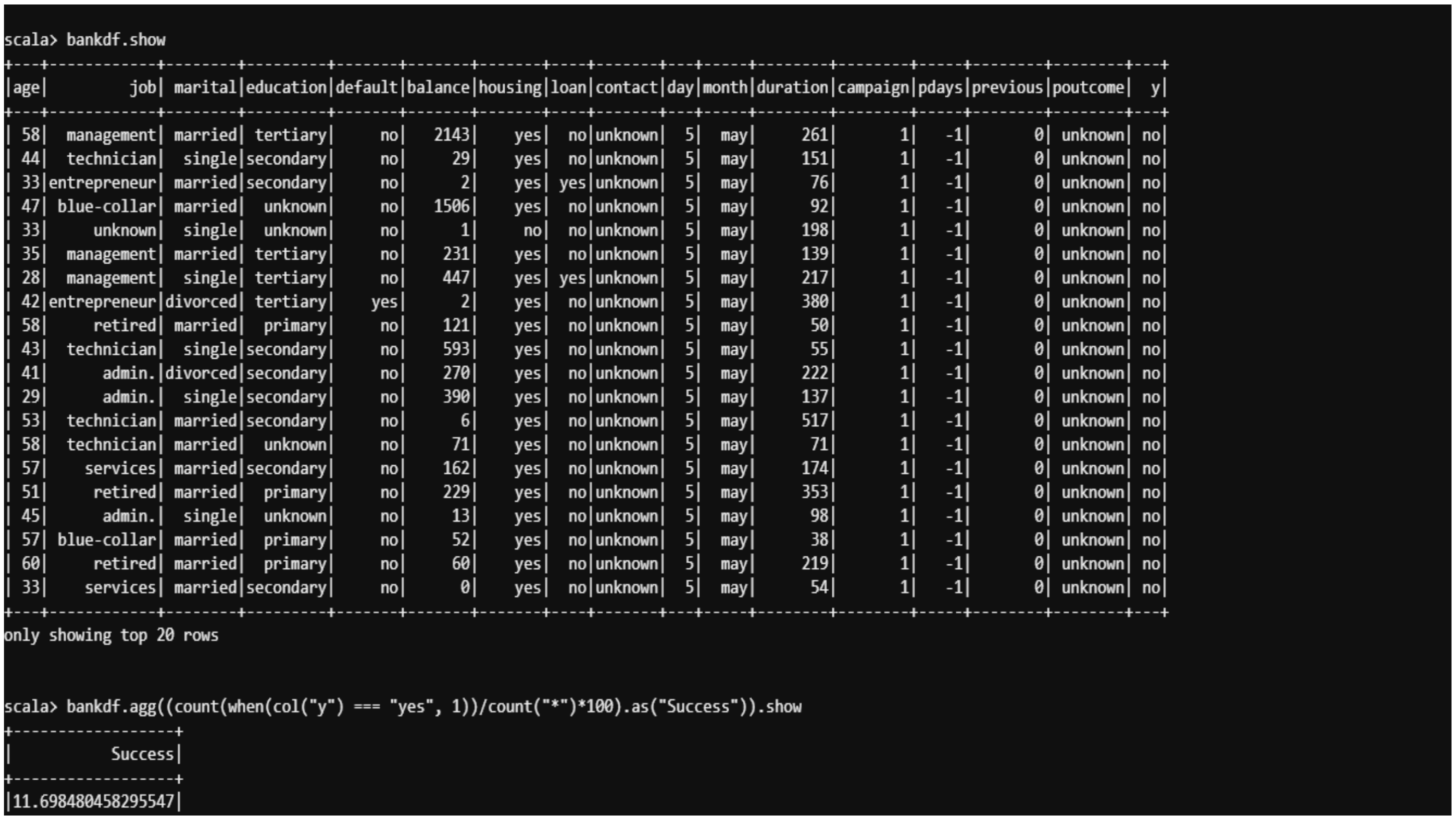


```
ip-10-0-41-79 login: satyajitchakraborty22gmai
Password:
Last login: Tue Feb 22 10:19:50 on pts/76
[satyajitchakraborty22gmai@ip-10-0-41-79 ~]$ spark-shell
Setting default log level to "ERROR".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/02/22 10:26:55 WARN cluster.YarnSchedulerBackend$YarnSchedulerEndpoint: Attempted to request executors before the AM has registered!
22/02/22 10:26:55 WARN lineage.LineageWriter: Lineage directory /var/log/spark/lineage doesn't exist or is not writable. Lineage for this application will be disab
led.
Spark context available as 'sc' (master = yarn, app id = application_1640258093152_10881).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.4.0-cdh6.3.2
      /_/

Using Scala version 2.11.12 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_144)
Type in expressions to have them evaluated.
Type :help for more information.

scala> var rdd = sc.textFile("\\Banking.csv")
rdd: org.apache.spark.rdd.RDD[String] = \Banking.csv MapPartitionsRDD[1] at textFile at <console>:24

scala> var formatted_data = rdd.map(input => input.replace("""""""", ""))
formatted_data: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[2] at map at <console>:25

scala> formatted_data.coalesce(1).saveAsTextFile("Desktop/Data/Formatted2/")

scala> var bankdf = spark.read.format("csv").option("header","true").option("delimiter",";").option("inferSchema","true").load("Desktop/Data/Formatted2/")
22/02/22 10:28:22 WARN lineage.LineageWriter: Lineage directory /var/log/spark/lineage doesn't exist or is not writable. Lineage for this application will be disab
led.
bankdf: org.apache.spark.sql.DataFrame = [age: int, job: string ... 15 more fields]
```

bankdf.show

<u>Question2:</u> Give marketing success rate. (No. of people subscribed / total no. of entries)

bankdf.agg((count(when(col("y") === "yes", 1))/count("*")*100).as("Success")).show



```
scala> bankdf.show
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+--------+-----+--------+--------+---+
|age|         job| marital|education|default|balance|housing|loan|contact|day|month|duration|campaign|pdays|previous|poutcome|  y|
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+--------+-----+--------+--------+---+
| 58|  management| married| tertiary|     no|   2143|    yes|  no|unknown|  5|  may|     261|       1|   -1|       0| unknown| no|
| 44|  technician|  single|secondary|     no|     29|    yes|  no|unknown|  5|  may|     151|       1|   -1|       0| unknown| no|
| 33|entrepreneur| married|secondary|     no|      2|    yes| yes|unknown|  5|  may|      76|       1|   -1|       0| unknown| no|
| 47| blue-collar| married|  unknown|     no|   1506|    yes|  no|unknown|  5|  may|      92|       1|   -1|       0| unknown| no|
| 33|     unknown|  single|  unknown|     no|      1|     no|  no|unknown|  5|  may|     198|       1|   -1|       0| unknown| no|
| 35|  management| married| tertiary|     no|    231|    yes|  no|unknown|  5|  may|     139|       1|   -1|       0| unknown| no|
| 28|  management|  single| tertiary|     no|    447|    yes| yes|unknown|  5|  may|     217|       1|   -1|       0| unknown| no|
| 42|entrepreneur|divorced| tertiary|    yes|      2|    yes|  no|unknown|  5|  may|     380|       1|   -1|       0| unknown| no|
| 58|     retired| married|  primary|     no|    121|    yes|  no|unknown|  5|  may|      50|       1|   -1|       0| unknown| no|
| 43|  technician|  single|secondary|     no|    593|    yes|  no|unknown|  5|  may|      55|       1|   -1|       0| unknown| no|
| 41|      admin.|divorced|secondary|     no|    270|    yes|  no|unknown|  5|  may|     222|       1|   -1|       0| unknown| no|
| 29|      admin.|  single|secondary|     no|    390|    yes|  no|unknown|  5|  may|     137|       1|   -1|       0| unknown| no|
| 53|  technician| married|secondary|     no|      6|    yes|  no|unknown|  5|  may|     517|       1|   -1|       0| unknown| no|
| 58|  technician| married|  unknown|     no|     71|    yes|  no|unknown|  5|  may|      71|       1|   -1|       0| unknown| no|
| 57|    services| married|secondary|     no|    162|    yes|  no|unknown|  5|  may|     174|       1|   -1|       0| unknown| no|
| 51|     retired| married|  primary|     no|    229|    yes|  no|unknown|  5|  may|     353|       1|   -1|       0| unknown| no|
| 45|      admin.|  single|  unknown|     no|     13|    yes|  no|unknown|  5|  may|      98|       1|   -1|       0| unknown| no|
| 57| blue-collar| married|  primary|     no|     52|    yes|  no|unknown|  5|  may|      38|       1|   -1|       0| unknown| no|
| 60|     retired| married|  primary|     no|     60|    yes|  no|unknown|  5|  may|     219|       1|   -1|       0| unknown| no|
| 33|    services| married|secondary|     no|      0|    yes|  no|unknown|  5|  may|      54|       1|   -1|       0| unknown| no|
+---+------------+--------+---------+-------+-------+-------+----+-------+---+-----+--------+--------+-----+--------+--------+---+
only showing top 20 rows


scala> bankdf.agg((count(when(col("y") === "yes", 1))/count("*")*100).as("Success")).show
+------------------+
|           Success|
+------------------+
|11.698480458295547|
```

<u>Question 2a:</u> Give marketing failure rate

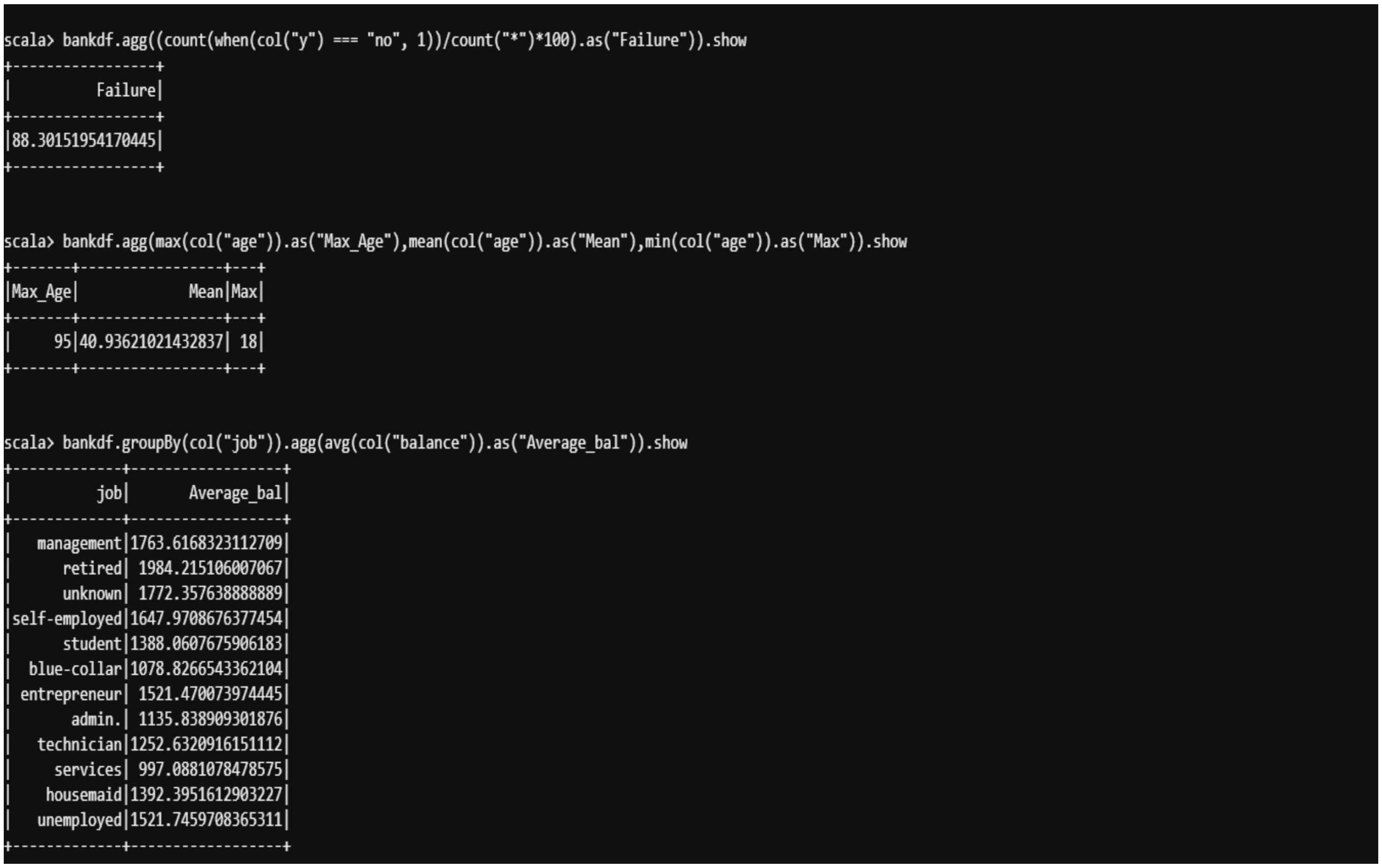
bankdf.agg((count(when(col("y") === "no", 1))/count("*")*100).as("Failure")).show

<u>Question3:</u> Maximum, Mean, and Minimum age of average targeted customer

bankdf.agg(max(col("age")).as("Max_Age"),mean(col("age")).as("Mean"),min(col("age")).as("Max")).show

<u>Question4:</u> Check quality of customers by checking average balance, median balance of customers

bankdf.groupBy(col("job")).agg(avg(col("balance")).as("Average_bal")).show

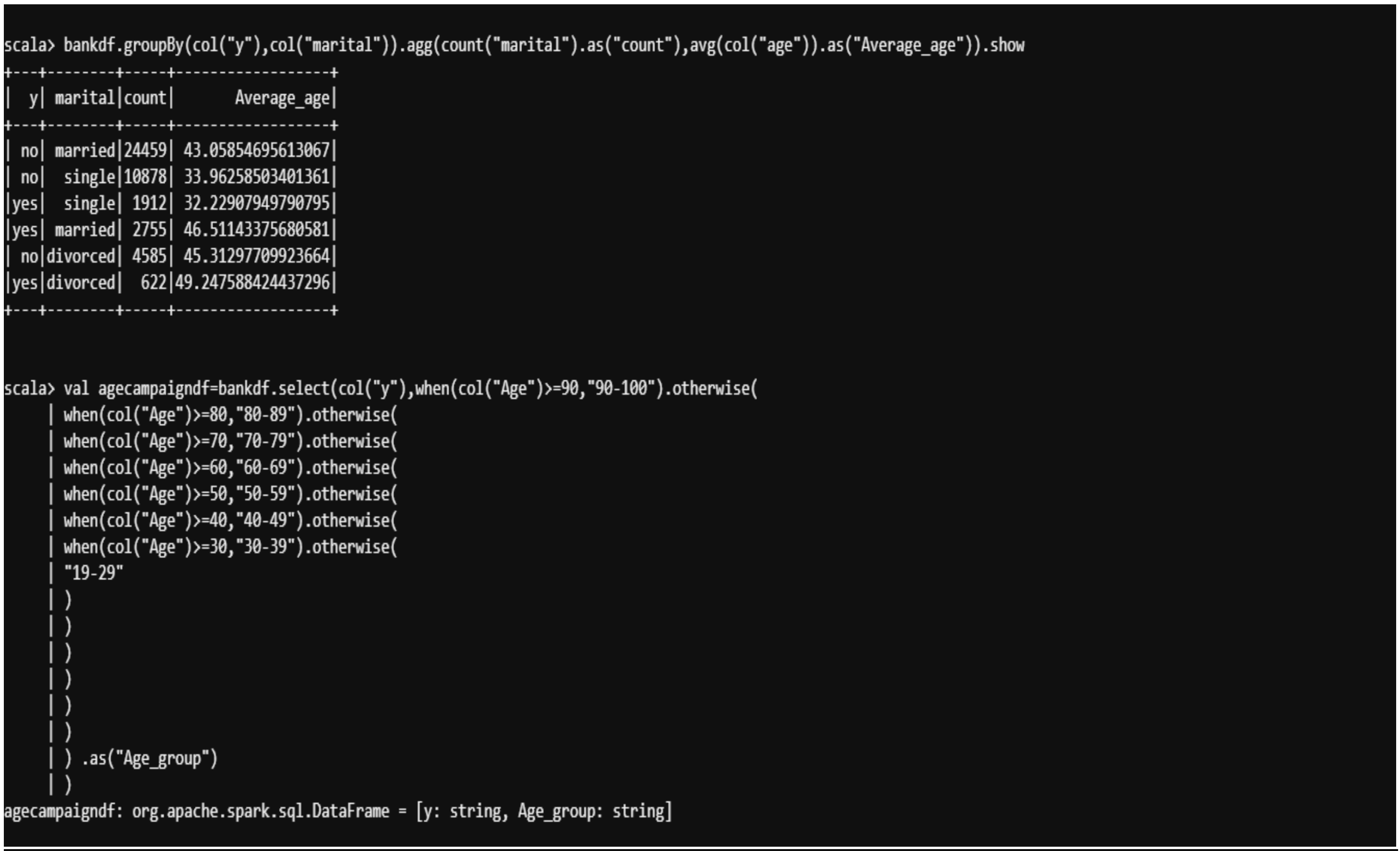

```
scala> bankdf.agg((count(when(col("y") === "no", 1))/count("*")*100).as("Failure")).show
+-----------------+
|          Failure|
+-----------------+
|88.30151954170445|
+-----------------+


scala> bankdf.agg(max(col("age")).as("Max_Age"),mean(col("age")).as("Mean"),min(col("age")).as("Max")).show
+-------+-----------------+---+
|Max_Age|             Mean|Max|
+-------+-----------------+---+
|     95|40.93621021432837| 18|
+-------+-----------------+---+


scala> bankdf.groupBy(col("job")).agg(avg(col("balance")).as("Average_bal")).show
+-------------+------------------+
|          job|       Average_bal|
+-------------+------------------+
|   management|1763.6168323112709|
|      retired| 1984.215106007067|
|      unknown| 1772.357638888889|
|self-employed|1647.9708676377454|
|      student|1388.0607675906183|
|  blue-collar|1078.8266543362104|
| entrepreneur| 1521.470073974445|
|       admin.| 1135.838909301876|
|   technician|1252.6320916151112|
|     services| 997.0881078478575|
|    housemaid|1392.3951612903227|
|   unemployed|1521.7459708365311|
+-------------+------------------+
```

<u>Questions: 5, 6, 7</u>

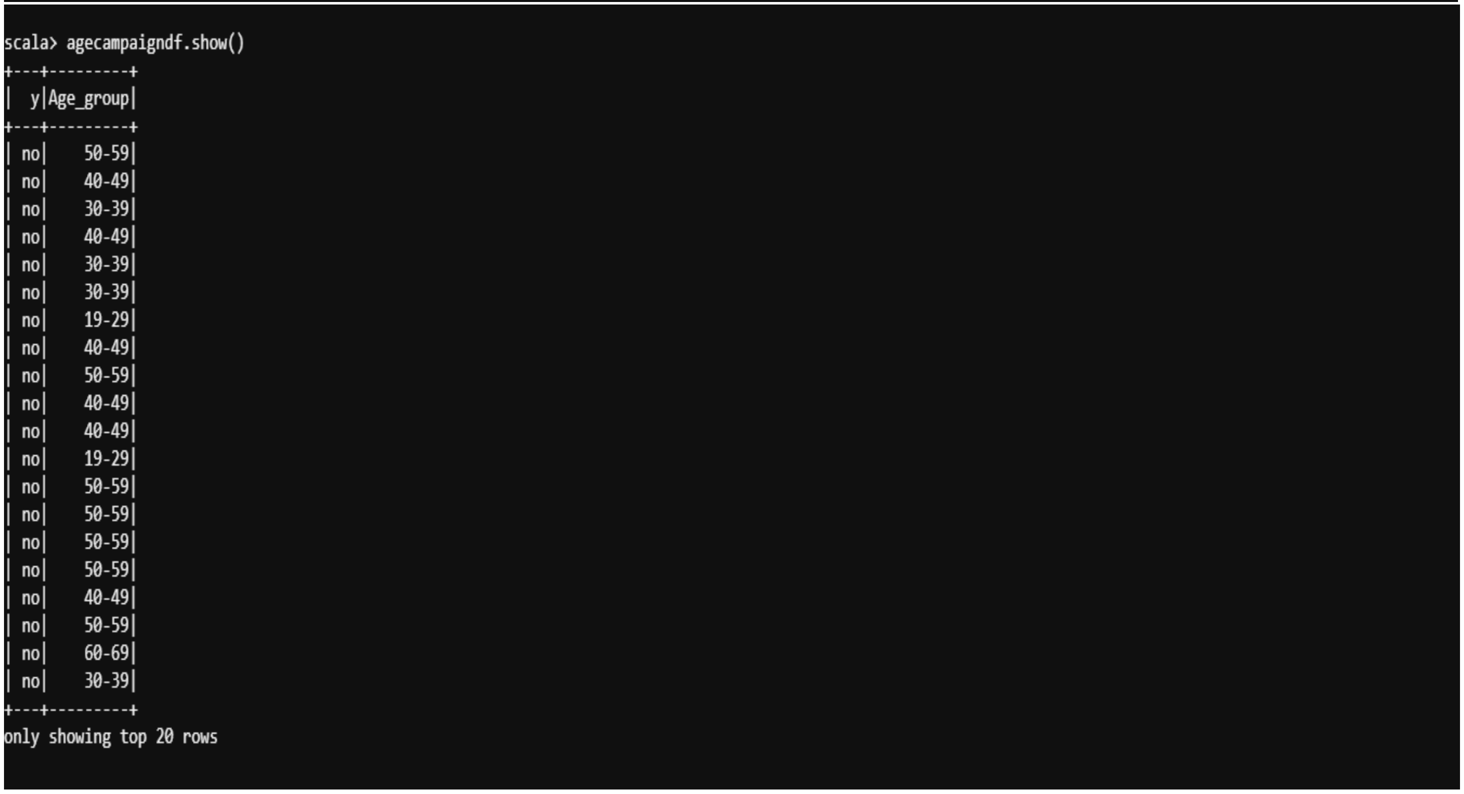
bankdf.groupBy(col("y"),col("marital")).agg(count("marital").as("count"),avg(col("age")).as("Average_age")).show

<u>Question8:</u> Do feature engineering for the bank and find the right age effect on the campaign

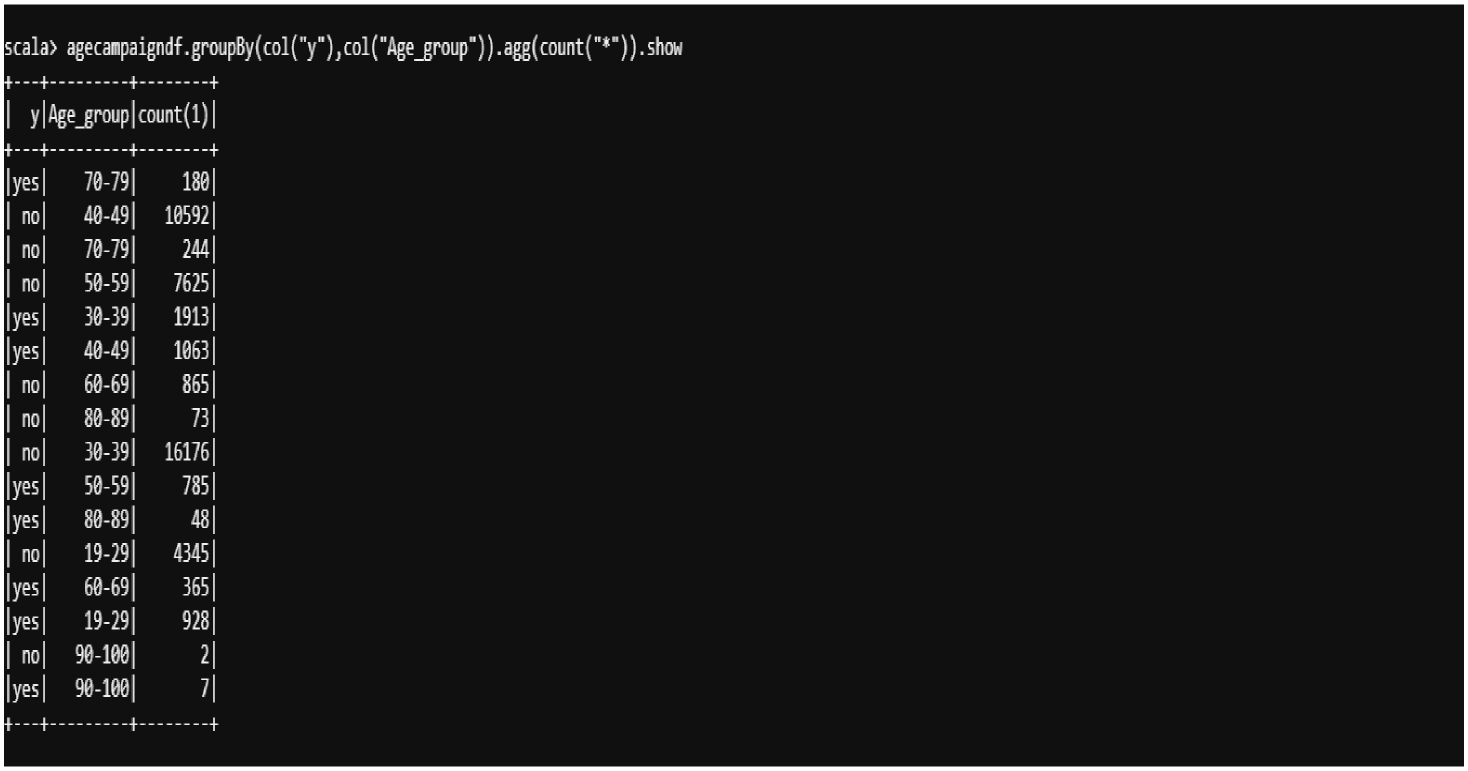
```
val agecampaigndf=bankdf.select(col("y"),when(col("Age")>=90,"90-100").otherwise(
when(col("Age")>=80,"80-89").otherwise(
when(col("Age")>=70,"70-79").otherwise(
when(col("Age")>=60,"60-69").otherwise(
when(col("Age")>=50,"50-59").otherwise(
when(col("Age")>=40,"40-49").otherwise(
when(col("Age")>=30,"30-39").otherwise(
"19-29"
)
)
)
)
)
)
) .as("Age_group")
)
agecampaigndf.show()
agecampaigndf.groupBy(col("y"),col("Age_group")).agg(count("*")).show
```



```
scala> bankdf.groupBy(col("y"),col("marital")).agg(count("marital").as("count"),avg(col("age")).as("Average_age")).show
+---+--------+-----+------------------+
|  y| marital|count|       Average_age|
+---+--------+-----+------------------+
| no| married|24459| 43.05854695613067|
| no|  single|10878| 33.96258503401361|
|yes|  single| 1912| 32.22907949790795|
|yes| married| 2755| 46.51143375680581|
| no|divorced| 4585| 45.31297709923664|
|yes|divorced|  622|49.247588424437296|
+---+--------+-----+------------------+


scala> val agecampaigndf=bankdf.select(col("y"),when(col("Age")>=90,"90-100").otherwise(
     | when(col("Age")>=80,"80-89").otherwise(
     | when(col("Age")>=70,"70-79").otherwise(
     | when(col("Age")>=60,"60-69").otherwise(
     | when(col("Age")>=50,"50-59").otherwise(
     | when(col("Age")>=40,"40-49").otherwise(
     | when(col("Age")>=30,"30-39").otherwise(
     | "19-29"
     | )
     | )
     | )
     | )
     | )
     | )
     | ) .as("Age_group")
     | )
agecampaigndf: org.apache.spark.sql.DataFrame = [y: string, Age_group: string]
```



```
scala> agecampaigndf.show()
+---+---------+
|  y|Age_group|
+---+---------+
| no|    50-59|
| no|    40-49|
| no|    30-39|
| no|    40-49|
| no|    30-39|
| no|    30-39|
| no|    19-29|
| no|    40-49|
| no|    50-59|
| no|    40-49|
| no|    40-49|
| no|    19-29|
| no|    50-59|
| no|    50-59|
| no|    50-59|
| no|    50-59|
| no|    40-49|
| no|    50-59|
| no|    60-69|
| no|    30-39|
+---+---------+
only showing top 20 rows
```



```
scala> agecampaigndf.groupBy(col("y"),col("Age_group")).agg(count("*")).show
+---+---------+--------+
|  y|Age_group|count(1)|
+---+---------+--------+
|yes|    70-79|     180|
| no|    40-49|   10592|
| no|    70-79|     244|
| no|    50-59|    7625|
|yes|    30-39|    1913|
|yes|    40-49|    1063|
| no|    60-69|     865|
| no|    80-89|      73|
| no|    30-39|   16176|
|yes|    50-59|     785|
|yes|    80-89|      48|
| no|    19-29|    4345|
|yes|    60-69|     365|
|yes|    19-29|     928|
| no|   90-100|       2|
|yes|   90-100|       7|
+---+---------+--------+
```