BIG DATA PROJECT CODE

Checkpoints

Checkpoint 1

Load the data into HDFS, Hive Managed table, Hive External table and Spark DataFrame.

➔ LOAD INTO HDFS

```
hdfs dfs -mkdir Project

hdfs dfs -put aadhar.csv Project
```

➔ CREATING HIVE MANAGED TABLE

```
drop database if exists project;

create database project;

use project;
```

➔ CREATING HIVE EXTERNAL TABLE

```
create external table if not exists aadhar_dataet(registrar string,private_agency string,state
string,district string,sub_district string,pincode string,gender string, age int,aadhar_generated
int,rejected int,email_id int,moblie_number int)

    > row format delimited fields terminated by ','

    > stored as textfile

    > location '/user/cloudera/Project'

        >
TBLPROPERTIES('serialization.null.format'='','skip.header.line.count'='1');
```

➔ CREATING SPARK DATAFRAME

```
val frdd=sc.textFile("/user/cloudera/Project/aadhar.csv")

val first=frdd.first()

val rdd=frdd.filter(row=>row!=first)

var
aadharrdd=rdd.map(x=>(x.split(",")(0),x.split(",")(1),x.split(",")(2),x.split(",")(3),x.split(",")(4),x.s
plit(",")(5),x.split(",")(6),x.split(",")(7).toInt,x.split(",")(8).toInt,x.split(",")(9).toInt,x.split(",")(10).
```

```
toInt,x.split(",")(11).toInt));

var
aadhardf=aadharrdd.toDF("registrar","private_agency","state","district","sub_district","pincod
e","gender","age","aadhar_generated","rejected","email_id","moblie_number");
```

1. Commit the screenshot of the view/result of the top 25 rows from each individual store (HDFS, Hive – Managed/External and Spark DataFrame).

```
insert overwrite local directory '/home/cloudera/Project/resultfile'
        > row format delimited fields terminated by ','
        > stored as textfile
        > select * from aadhar_dataet limit 25;
```

Checkpoint 2

2. Describe the schema.
```
describe aadhar_datamt;
```

3. Find the count and names of registrars in the table.
```
select count(distinct(registrar)) from aadhar_datamt;
select distinct(registrar) from aadhar_datamt;
```

4. Find the number of states, districts in each state and sub-districts in each district.
```
select count(distinct(state)) from aadhar_datamt;
select state,count(distinct(district)) from aadhar_datamt group by state;
select count(distinct(district)) from aadhar_datamt;
select district,count(distinct(sub_district)) from aadhar_datamt group by district;
```

5. Find the number of males and females in each state from the table.
```
create table  Male_count as select state,count(*) Number_of_Males from aadhar_datamt
where gender = 'M' group by state;
create table  Female_count as select state,count(*) Number_of_Females from
aadhar_datamt  where gender = 'F' group by state;
select t1.state,t1.Number_of_Females,t2.Number_of_Males  from Female_count t1 join
Male_count t2 on (t1.state=t2.state);
```

6. Find out the names of private agencies for each state.
```
select state,private_agency from aadhar_datamt limit 20;
```

7. Plot the number of private agencies for each state.

Checkpoint 3

8. Find top 3 states generating most number of Aadhaar cards?

   select state,sum(aadhar_generated) Number_of_aadhar from aadhar_datamt group by state sort by Number_of_aadhar desc limit 3;

9. Find top 3 private agencies generating the most number of Aadhar cards?

   -CONVERTING DATAFRAME TO TABLE
   aadhardf.registerTempTable("aadhar");
   val q9=sqlContext.sql("select private_agency,sum(aadhar_generated) Number_of_aadhar from aadhar group by private_agency sort by Number_of_aadhar desc limit 3");

10. Find the number of residents providing email, mobile number? (Hint: consider non-zero values.)
    val q10=sqlContext.sql("Select sum(email_id) Email_Id,sum(moblie_number) Mobile_Number from aadhar");

11. Find top 3 districts where enrolment numbers are maximum?
    val q11=sqlContext.sql("Select district,sum(aadhar_generated+rejected) Enrollments from aadhar group by district sort by Enrollments desc limit 3");

12. Find the no. of Aadhaar cards generated in each state?
    val q12=sqlContext.sql("Select state,sum(aadhar_generated) aadhar_gen from aadhar group by state");

Checkpoint 4

13. Create a data frame using the file and provide its summary.

    aadhardf.printSchema

14. Write a command to see the correlation between “age” and “mobile_number”? (Hint: Consider the percentage of people who have provided the mobile number out of the total applicants)
    val q14=sqlContext.sql("select corr(age,mobile_number) as Correlation from aadhar");

15. Find the number of unique pincodes in the data?
    val q15=sqlContext.sql("Select count(distinct(pincode)) from aadhar");

16. Find the number of Aadhaar registrations rejected in Uttar Pradesh and Maharashtra?

```
val q16=sqlContext.sql("Select sum(rejected) from aadhar where state like'Uttar Pradesh' or state like 'Maharashtra' ");
```

Checkpoint 5

On the given dataset, perform EDA and find:

17. The top 3 states where the percentage of Aadhaar cards being generated for males is the highest.

```
val q17=sqlContext.sql("Select state,round((sum(aadhar_generated)/sum(aadhar_generated+rejected))*100,2) Percentage_of_aadhar from aadhar where gender like 'M' group by state order by Percentage_of_aadhar desc limit 3");
```

18. In each of these 3 states, identify the top 3 districts where the percentage of Aadhaar cards

```
val q18=sqlContext.sql("Select state,district,round((sum(rejected)/sum(aadhar_generated+rejected))*100,2) Percentage_of_rejected from aadhar where gender like 'F' and state like 'Andaman and Nicobar Islands' or state like 'Lakshadweep' or state like 'Others' group by state,district order by Percentage_of_rejected desc");
```

19. The top 3 states where the percentage of Aadhaar cards being generated for females is the highest.

```
val q19=sqlContext.sql("Select state,round((sum(aadhar_generated)/sum(aadhar_generated+rejected))*100,2) Percentage_of_aadhar from aadhar where gender like 'F' group by state order by Percentage_of_aadhar desc limit 3");
```

20. In each of these 3 states, identify the top 3 districts where the percentage of Aadhaar cards being rejected for males is the highest.

```
val q20=sqlContext.sql("Select state,district,round((sum(rejected)/sum(aadhar_generated+rejected))*100,2) Percentage_of_rejected from aadhar where gender like 'M' and state like 'Dadra and Nagar Haveli' or state like 'Sikkim' or state like 'Others' group by state,district order by Percentage_of_rejected desc");
```

21. The summary of the acceptance percentage of all the Aadhaar cards applications by bucketing the age group into 10 buckets.

```
create table  aadhar_bucket(registrar string,private_agency string,state string,district string,sub_district string,pincode string,gender string, age int,aadhar_generated int,rejected
```

```
int,email_id int,moblie_number int) clustered by (age) into 10 buckets
        >  row format delimited fields terminated by ','
        > stored as textfile
        > TBLPROPERTIES('serialization.null.format'='','skip.header.line.count'='1');

Insert into aadhar_bucket select * from aadhar_datamt;

select round((sum(aadhar_generated)/sum(aadhar_generated+rejected))*100,2) from
aadhar_bucket;
```