Step1:

# Load Loan\_data.csv from local into hdfs directory;

* hdfs dfs -mkdir /user/srikarthik/loan\_project
* hdfs dfs -moveFromLocal /home/srikarthik/ Loan\_data.csv /user/srikarthik/loan\_project

Step2: Create external table and load data into it.

create external table loan(

id int, member\_id int, loan\_amnt int, funded\_amnt int, funded\_amnt\_inv double,

term string, int\_rate string, installment double, grade string, sub\_grade string, emp\_title string,

emp\_length string, home\_ownership string, annual\_inc double, is\_inc\_v string, issue\_d string,

loan\_status string, pymnt\_plan string, url string, desc string, purpose string,

title string, zip\_code string, addr\_state string, dti double, delinq\_2yrs int, earliest\_cr\_line string,

inq\_last\_6mths int, mths\_since\_last\_delinq int, mths\_since\_last\_record int, open\_acc int,

pub\_rec int, revol\_bal int, revol\_util string, total\_acc int, initial\_list\_status string,

out\_prncp double, out\_prncp\_inv double, total\_pymnt double, total\_pymnt\_inv double,

total\_rec\_prncp double, total\_rec\_int double, total\_rec\_late\_fee double, recoveries double,

collection\_recovery\_fee double, last\_pymnt\_d string, last\_pymnt\_amnt double, next\_pymnt\_d string,

last\_credit\_pull\_d string, collections\_12\_mths\_ex\_med int, mths\_since\_last\_major\_derog int,

policy\_code int)

Row format delimited fields terminated by ',' LOCATION "/user/srikarthik/loan\_project" tblproperties('skip.header.line.count'='1');

Step3:

//Q1: Find the list of people with grade “B” who have taken loan.

SELECT id, member\_id FROM loan WHERE grade = 'B';

result= 62605 row(s) fetched , sample result as below:

10119623 11971241

10127816 11979581

10129454 11981072

10129477 11981093

10129506 11981122

10139658 11991209

10149342 12000897

10149488 12001033

10149566 12001108

10159611 12011228

//Q2: Find the list of people having interest more than 1000

Formula used to calculate interest = % interest rate/100/12 \* loan amount

SELECT member\_id, ROUND((CAST(SUBSTRING(int\_rate,1,5) as float)/100/12 \* loan\_amnt),2) AS interest FROM loan WHERE cast(substring(int\_rate,1,5) as float)/100/12 \* loan\_amnt >1000;

result= Fetched: 0 row(s), Seems every ones monthly interest is less than 1000.

//Q3: Find the list of people having loan amount more than 1000.

SELECT id, member\_id, loan\_amnt FROM loan WHERE loan\_amnt >1000 order by id;

result = Fetched: 187591 row(s), sample result as below:

58524 149512 7200

356706 362135 2400

380041 407046 16000

438264 527690 32875

442319 537198 24000

448846 551068 20000

475622 602177 3600

476326 603349 15000

497951 638479 8400

506836 653549 15000

//Q4:Get the highest loan amount given to grade users (A-G).

SELECT MAX (loan\_amnt), Grade FROM loan WHERE grade BETWEEN 'A' AND 'G' GROUP BY grade;

result:

35000 A

35000 B

35000 C

35000 D

35000 E

35000 F

35000 G

Time taken: 24.038 seconds, Fetched: 7 row(s)

//Q5:Highest loan amount given in that year with that Employee id and Employees annual income.

SELECT member\_id, annual\_inc, loan\_amnt FROM loan WHERE loan\_amnt IN (SELECT MAX(loan\_amnt) FROM loan);

result= Fetched: 5921 row(s), sample result as below:

maximum loan amount = 35000

11951022 93500.0 35000

11941016 90000.0 35000

12020134 101000.0 35000

11960883 93000.0 35000

11970925 425000.0 35000

11970970 100000.0 35000

11980585 140000.0 35000

11990959 94507.0 35000

11920905 180000.0 35000

12010312 120000.0 35000

//Q6:Get the total number of loans with loan id and load amount which are having loan status as Late.

SELECT id, loan\_amnt, COUNT(\*) AS total\_loans FROM loan WHERE loan\_status like 'Late%' GROUP BY id, loan\_amnt;

Note: since loan id is unique, when above query is executed, it shows count as 1 for each record. Hence I presume the question shall be either of two:

1. the total number of loans and total loan amount which are having loan status as Late. Based on this presumption:

SELECT COUNT(\*) AS total\_loans, sum(loan\_amnt) as total\_loan\_amount FROM loan WHERE loan\_status like 'Late%';

Result:

4195 62750700

1. the total number of loans per loan amount which are having loan status as Late. Based on this presumption:

SELECT COUNT(\*) AS total\_loans, loan\_amnt as total\_loan\_amount FROM loan WHERE loan\_status LIKE 'Late%'GROUP BY loan\_amnt order by loan\_amnt desc;

Result: Fetched: 518 row(s), sample result shown as below:

159 35000

2 34475

1 34000

5 33950

2 33600

4 33425

3 33000

6 32875

1 32650

5 32350

//Q7: Average loan interest rate with 60-month term and 36-month term.

SELECT term, concat(ROUND(AVG(cast(substring(int\_rate,1,5) as float)),2),'%') AS average\_loan\_interest FROM loan GROUP BY term;

result:

36 months 14.7%

60 months 18.22%