# Problem statement 01:

**Create Database:**

HIVE> CREATE DATABASE IF NOT EXISTS npn\_analytics\_rakesh COMMENT 'Holds assignaments data' WITH DBPROPERTIES('edited-by'='Rakesh');

Hive> use npn\_analytics\_rakesh;

**Create table:**

Hive > use npn\_analytics\_rakesh;

Hive > CREATE TABLE tenure\_vacation(

id INT,

tenure INT,

designation STRING,

salary BIGINT

);

A white background with black text

Description automatically generated

**Insert Data to Table:**

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1001,12,'Software Engineer',30000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1002,5,'Manager',50000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1003,2,'Test Engineer',30000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1004,1,'Sr Software Engineer',40000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1005,14,'Jr Software Engineer',20000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1006,6,'Analyst',20000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1007,2,'Software Engineer',20000);

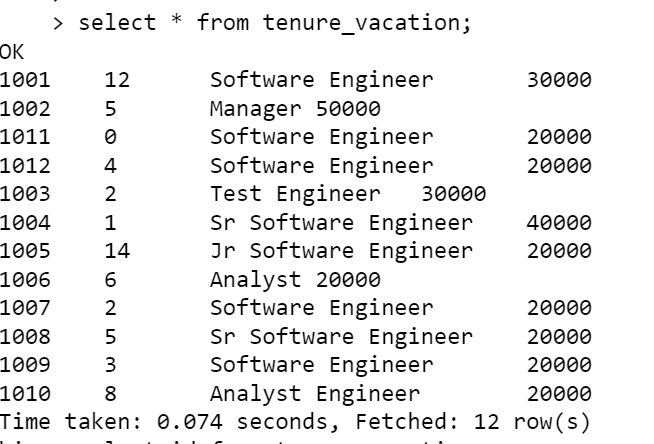
Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1008,5,'Sr Software Engineer',20000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1009,3,'Software Engineer',20000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1010,8,'Analyst Engineer',20000);

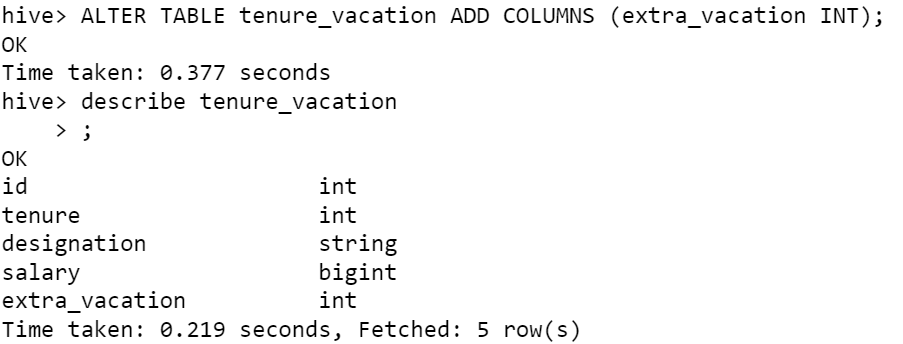
Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1011,0,'Software Engineer',20000);

Hive> insert into tenure\_vacation(id,tenure,designation,salary) values(1012,4,'Software Engineer',20000);



**Add New column:**

Hive> ALTER TABLE tenure\_vacation ADD COLUMNS (extra\_vacation INT);



**Update new column:**

**Method 1:** Using UPDATE

**Query:**

UPDATE tenure\_vacation SET extra\_vacation = VALUE ( CASE WHEN tenure < 2 THEN 20

WHEN extra\_vacation > 10 THEN 40

ELSE 30 END );

**Method 2:** Using another table. Crete another table with the new column added(tenure\_vacation\_1). Query using the insert.

**Query:**

INSERT OVERWRITE TABLE tenure\_vacation

SELECT

id, tenure, designation, salary,

case WHEN tenure < 2 THEN 20

WHEN tenure >10 THEN 40

ELSE 30

END extra\_vacation

FROM tenure\_vacation\_1;

A white screen with black text

Description automatically generated

# Problem Statement-2:

**Query:**

CREATE TABLE temperature(

Date\_ STRING,

State STRING,

Temperature ARRAY<DOUBLE>

)ROW FORMAT DELIMITED FIELDS TERMINATED BY '|'

COLLECTION ITEMS TERMINATED BY ','

LINES TERMINATED BY '\n'

STORED AS TextFile;

A screenshot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

**Query:**

LOAD DATA LOCAL INPATH '/home/march8lab23/vasagirir/temperature.txt' into table temperature;

A screenshot of a computer

Description automatically generated

**Query:**

SELECT state, SORT\_ARRAY(temperature)[SIZE(temperature)-1] AS max\_temperature FROM temperature;

A white paper with black text

Description automatically generated

# Problem statement-3:

**Table creation Query:**

CREATE TABLE student\_marks(

Name STRING,

Marks MAP<STRING,INT>

)ROW FORMAT DELIMITED FIELDS TERMINATED BY '|'

COLLECTION ITEMS TERMINATED BY ','

MAP KEYS TERMINATED BY ':'

LINES TERMINATED BY '\n'

STORED AS TextFile;

**Data Load Query:**

LOAD DATA LOCAL INPATH '/home/march8lab23/vasagirir/student\_marks.txt' into table student\_marks;

A screenshot of a computer

Description automatically generated

1. **Display NAME who have scored more than 90 in subject Maths subject**

**Query:**

SELECT Name, Marks["Maths"] from student\_marks WHERE Marks["Maths"] > 90;

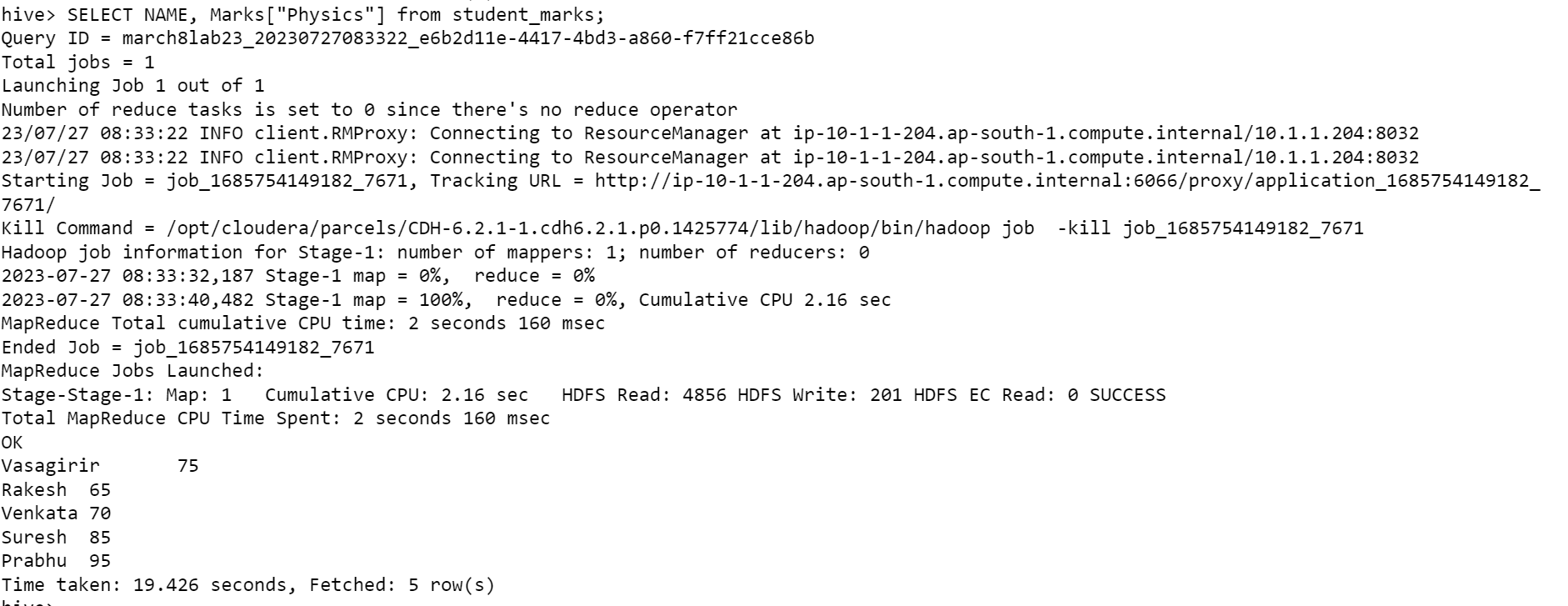
A close-up of a white page

Description automatically generated

1. **Display NAME and <Marks Scored in Physics subject>**

**Query:**

SELECT NAME, Marks["Physics"] from student\_marks;



1. **Display NAME, and <Maximum-Subject-Marks>**

**Query:**

SELECT NAME,

CASE WHEN SORT\_ARRAY(MAP\_VALUES(marks))[size(marks)-1]=marks["Maths"] THEN "Maths"

WHEN SORT\_ARRAY(MAP\_VALUES(marks))[size(marks)-1]=marks["Physics"] THEN "Physics"

ELSE "CHEMISTRY" END,

SORT\_ARRAY(MAP\_VALUES(marks))[size(marks)-1] as Max\_mark

from student\_marks;

A close-up of a white page

Description automatically generated

1. **Display NAME and <Average Marks Scored>**

**Query:**

SELECT A.NAME, ROUND(SUM(A.IND\_MARKS)/3.00,2) AS Avg\_Marks\_Scored FROM (SELECT NAME, IND\_MARKS FROM student\_marks LATERAL VIEW EXPLODE(MAP\_VALUES(marks)) dummy AS IND\_MARKS) A GROUP BY A.NAME;

A close-up of a computer screen

Description automatically generated

1. **Display NAME and <Percentage of marks>**

**Query:**

SELECT A.NAME, ROUND((SUM(A.IND\_MARKS)/300.00)\*100,2) AS Marks\_Percentage FROM (SELECT NAME, IND\_MARKS FROM student\_marks LATERAL VIEW EXPLODE(MAP\_VALUES(marks)) dummy AS IND\_MARKS) A GROUP BY A.NAME;

A white text with black text

Description automatically generated

# Problem Statement-4:

A screen shot of a computer

Description automatically generated

**Query:**

CREATE TABLE student\_info(

Name STRING,

Marks MAP<STRING,INT>,

Address STRUCT<doorNo:INT,Location:STRING, Pincode:INT>

)ROW FORMAT DELIMITED FIELDS TERMINATED BY '|'

COLLECTION ITEMS TERMINATED BY ','

MAP KEYS TERMINATED BY ':'

LINES TERMINATED BY '\n'

STORED AS TextFile;

A computer code with black text

Description automatically generated

**DATA LOAD:**

**Query:**

LOAD DATA LOCAL INPATH '/home/march8lab23/vasagirir/student\_info.txt' into table student\_info;

In Local:

A computer code on a white background

Description automatically generated

In hive:

A screen shot of a computer

Description automatically generated

**Write a query for the below mentioned tasks**

1. **Display all “NAME” who is located in Banashankari.**

**Query:**

SELECT Name FROM student\_info WHERE address.location = 'Banashankari';

A close-up of a letter

Description automatically generated

1. **Calculate the total count who is staying in pin code 560001.**

**Query:**

SELECT COUNT(\*) FROM student\_info WHERE address.pincode=560001;

A close-up of a document

Description automatically generated