DATABASE MANAGEMENT SYSTEMS

(Code 18CSC303J)

B.Tech (CSE) – 3rd year/6th Semester

Name: Akshat Gupta

Registration No: RA2011030030045

DEPARTMENT OF COMPUTER SCIENCE &

ENGINEERING FACULTY OF ENGINEERING &

TECHNOLOGY

SRM INSTITUTE OF SCIENCE & TECHNOLOGY,

DELHI NCR CAMPUS, MODINAGAR

SIKRI KALAN, DELHI MEERUT ROAD, DIST. – GHAZIABAD - 201204

www.srmup.in

Even Semester (2022-2023)

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

BONAFIDE CERTIFICATE

Registration no. RA2011030030045

Certified to be the bonafide record of work done by ***Akshat Gupta*** *of 6th semester 3nd year* ***B. TECH*** *degree course in* ***SRM INSTITUTE OF SCIENCE AND TECHNOLOGY,*** *NCR Campus of Department of Computer Science & Engineering, in DATABASE MANAGEMENT SYSTEMS, during the academic year 2022-23.*

Lab In Charge **Head of the Department**

Submitted for university examination held on__/___/___at SRMIST, NCR Campus.

Internal Examiner-I **Internal Examiner-II**

Index

S.No.	Table of Contents	Date of Experiment	Date of Submission	Faculty Sign
1	Lab 1: SQL Data Definition Language Commands on sample exercise			
2	Lab 2: SQL Data Manipulation Language Commands			
3	Lab 3: SQL DCL Commands and Transaction control commands to the sample exercises			
4	Lab 4: Inbuilt functions in SQL on sample			
5	Lab 5: Construct a ER Model for the application to be constructed to a database			
6	Lab 6: Nested Queries on sample exercise			
7	Lab 7: Join Queries on sample exercise			
8	Lab 8: Set operators & Views			
9	Lab9: PL/SQL Conditional and Iterative Statements			
10	Lab10: PL/SQL Procedures on sample exercise.			
11	Lab 11: PL/SQL Functions			
12	Lab 12: PL/SQL Cursors			
13	Lab 13: PL/SQL Exception Handling			
14	Lab 14: PL/SQL Trigger			
15	Lab 15: Frame and execute the appropriate PL/SQL Cursors and Exceptional Handling for the project			

Experiment No:-1

Aim:- SQL Data Definition LanguageCommands on sample exercise

SQL data definition:

Create command
Alter command
Drop
Truncate

<u>Theory</u>:-

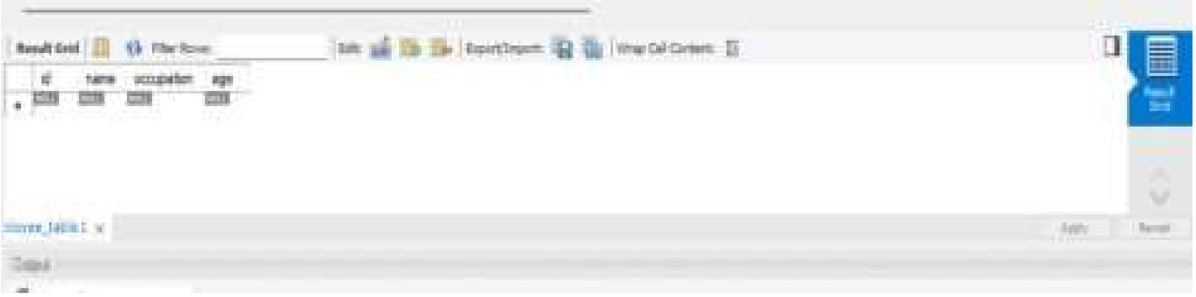
- ➢ DDL changes the structure of the table like creating a table, deleting a table,altering a table, etc.
- ➢ All the command of DDL are auto-committed that means it permanently saveall the changes in the database.

a. CREATE: It is used to create a new table in the database.

QUERY:-

```
CREATE TABLE employee_table(
   id int NOT NULLAUTO_INCREMENT,
   name varchar(45)  NOT
   NULL, occupation varchar(35)
   NOT NULL,age int NOT
   NULL,
   PRIMARY KEY (id)
);
```

OUTPUT:-

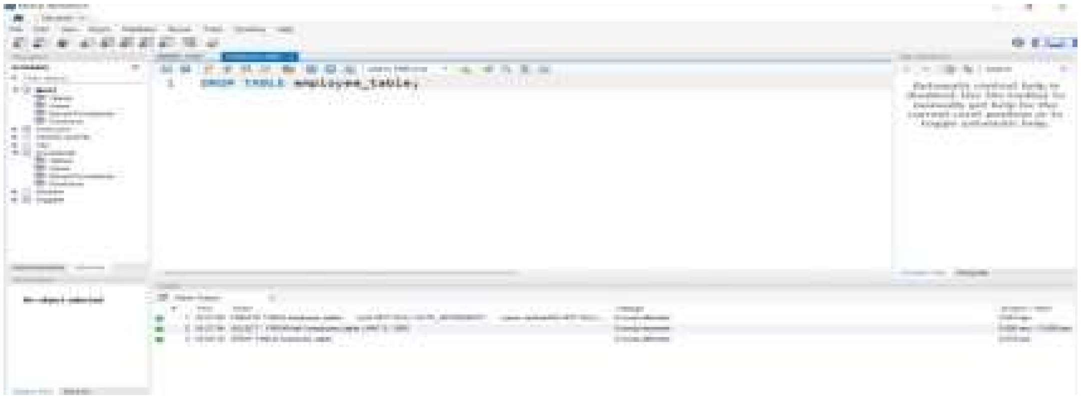


```
   id     name   occupation   age
 ● NULL   NULL   NULL         NULL
```

B .DROP: It is used to delete both the structure and record stored in the

table. QUERY:

```
DROP TABLE employee_table;
```

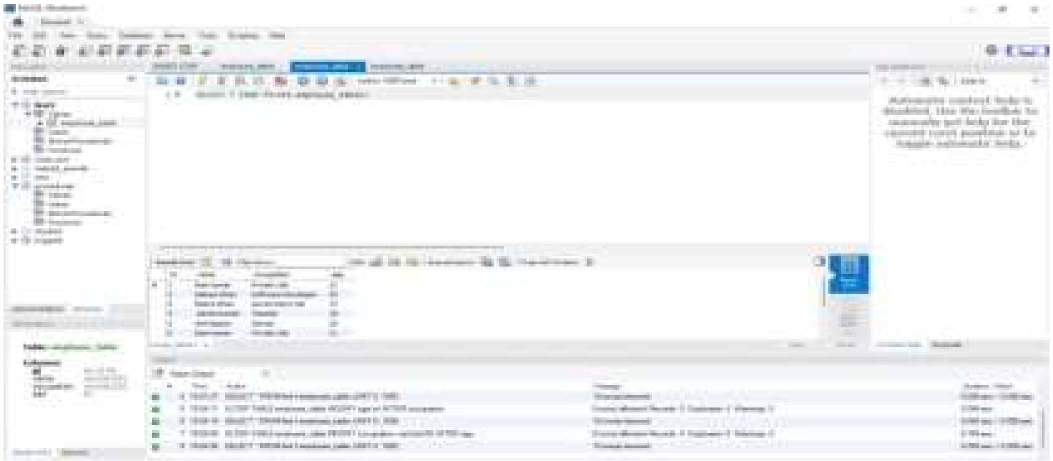


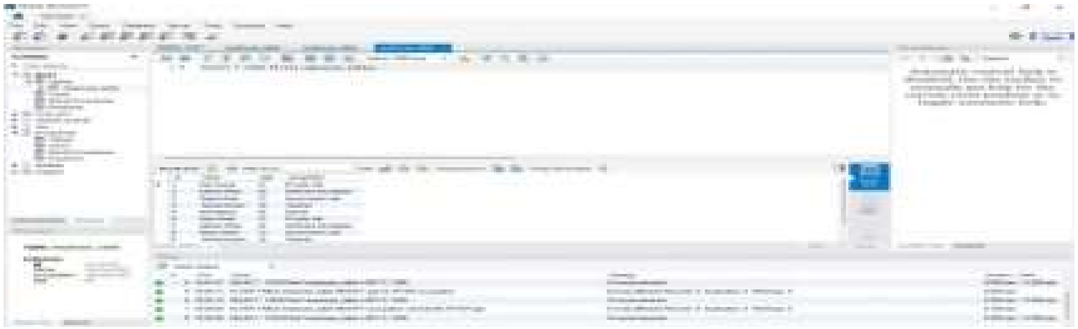
```
DROP TABLE employee_table;
```

ALTER : It is used to alter the structure of the database. This change could be either to modify the characteristics of an existing attribute or probably to add a newattribute.

QUERIES:

```
ALTERTABLE
employee_table MODIFY
occupation varchar(35)
AFTER age;
```

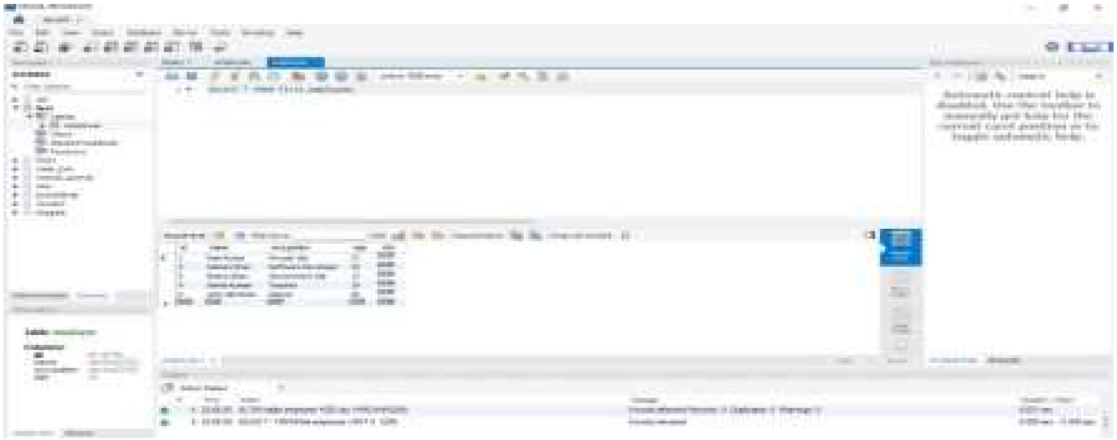




<u>ALTER TABLE-ADD COLUMN</u>

QUERIES:

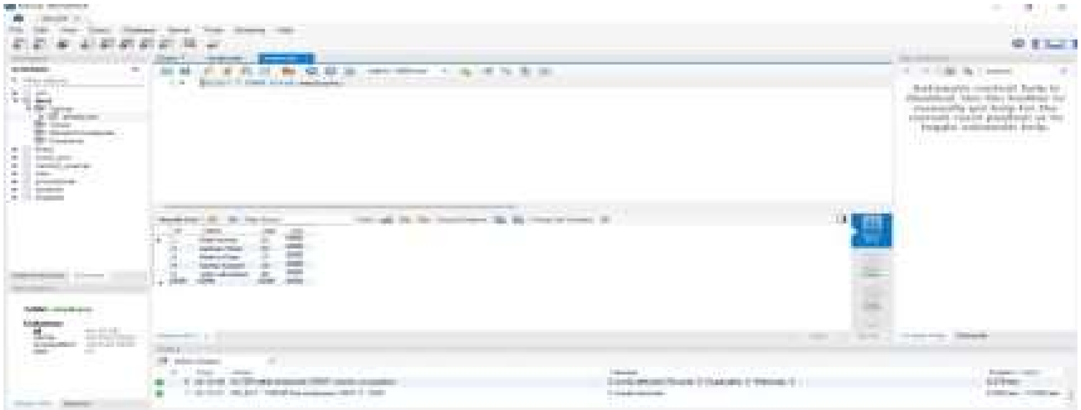
```
ALTER table
employee ADD city
VARCHAR(255);
```



ALTER TABLE-DROP COLUMN:

QUERIES:

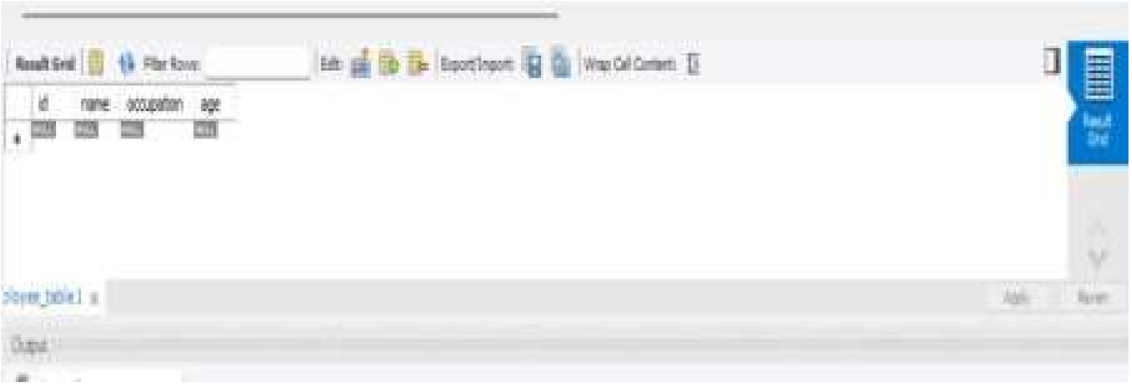
```
ALTER table
employee DROP
column occupation;
```



TRUNCATE: It is used to delete all the rows from the table and free the spacecontaining the table.

QUERIES:

```
TRUNCATE TABLE employee_table;
```



id	name	occupation	age
NULL	NULL	NULL	NULL

RESULT: The DDL commands in RDBMS are implemented successfully and output was verified.

Experiment No:-2

Aim:- Execute DML Commands

Theory:- DML commands are the most frequently used SQL commands and is usedto query and manipulate the existing database objects. Some of the commands are Insert, Select, Update, Delete.

Insert Command: This is used to add one or more rows to a table. The values are separated by commas and the data types char and date are enclosed in apostrophes.The values must be entered in the same order as they are defined.

Select Command: It is used to retrieve information from the table. It is generally referred to as querying the table. We can either display all columns in a table or onlyspecify column from the table.

Update Command: It is used to alter the column values in a table. A single columnmay be updated or more than one column could be updated.

Delete command: After inserting row in a table we can also delete them if required. The delete command consists of a from clause followed by an optional where clause.

PROCEDURE:-

1. Goto Start and click on programs.
2. Point to Oracle, then Ora81 and click on SQLPlus.
3. Then a pop-up menu will appear to authenticate you as user. Type 'scott' as username, 'tiger' as password and 'orcl' as the host.
4. Now SQLPlus window will get opened for working with various SQL commands in it.

SYNTAX AND DEFINITION:

DML commands:

1. Command: SELECT

Syntax:

```
SELECT[ALL | DISTINCT] select list
FROM table_name1[,…table_nameN]
[JOIN join_condition]
```

```
[WHERE search_condition]
```

2. Command: INSERT

Syntax:

```
INSERT INTO[[database_name]owner]{table_name|view_name}
[(column_list)]{[DEFAULT]VALUES|VALUES(value[…])|SELECT
Statement}
```

3. Command: UPDATE

Syntax:

```
UPDATE table_name
SET column_name=expression [,…n]
WHERE search_condition
```

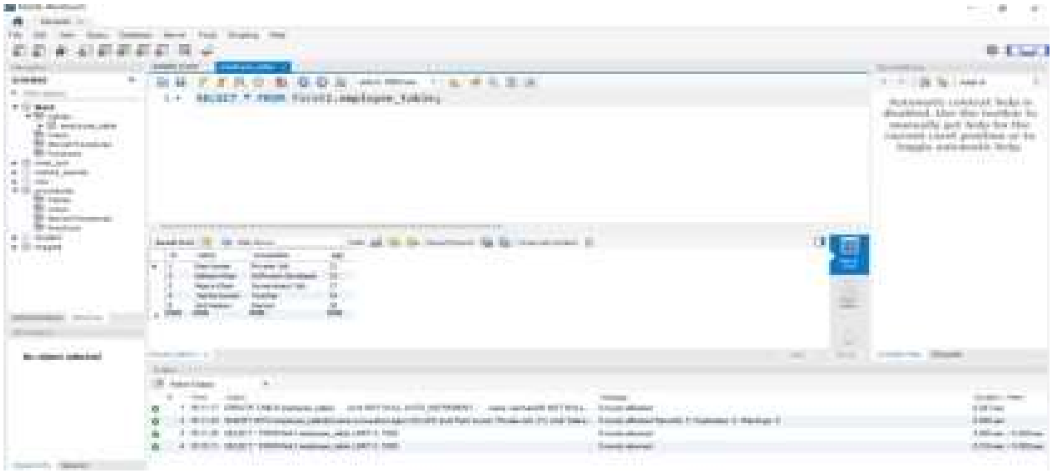
4. Command: DELETE

Syntax:

```
DELETE[FROM] table_name WHERE search_condition]
```

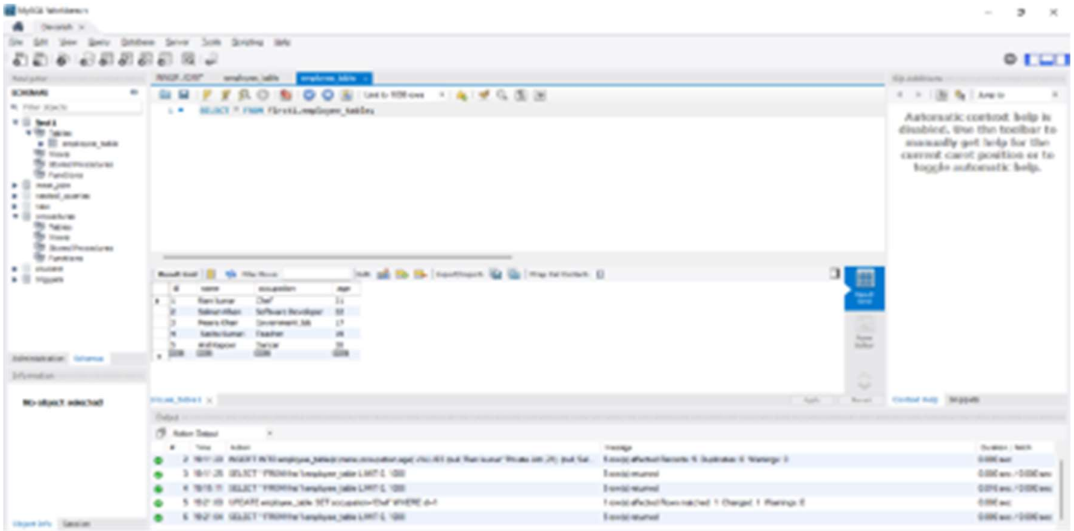
1. <u>INSERT Command:-</u>

INSERT INTO
employee_table(id,name,occupation,age)
VALUES
(null,'Ram kumar','Private Job',21),
(null,'Salman Khan','Software
Developer',22),(null,'Meera
Khan','Government Job',17), (null,'
Sarita Kumari','Teacher',19),
(null,'Anil Kapoor','Dancer',20);

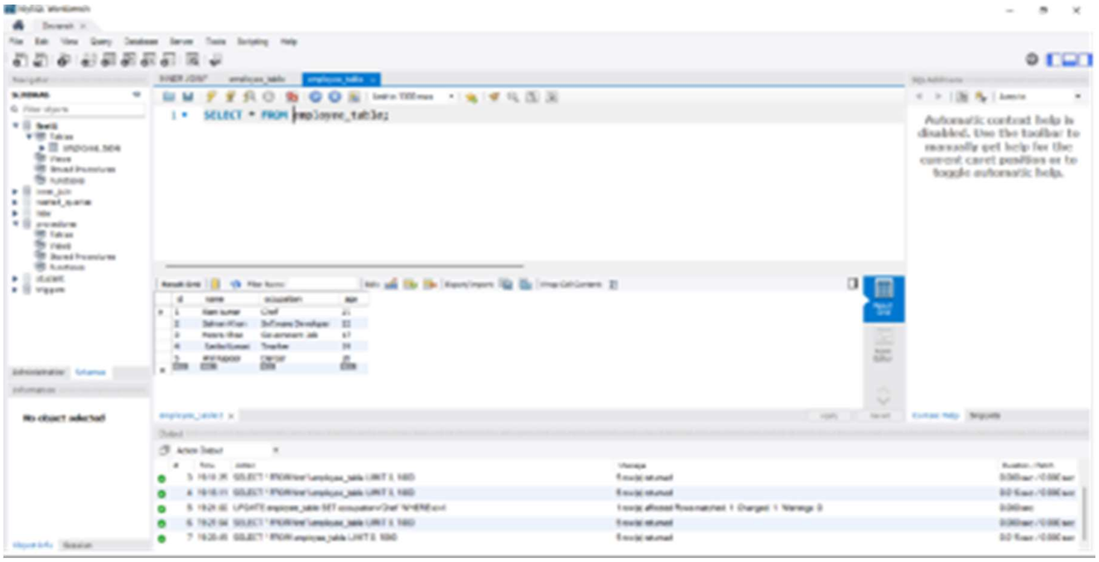


2. <u>Update Command:-</u>

UPDATE
employee_tableSET
occupation='Chef'
WHERE id=1;



3.<u>Select Command:-</u>



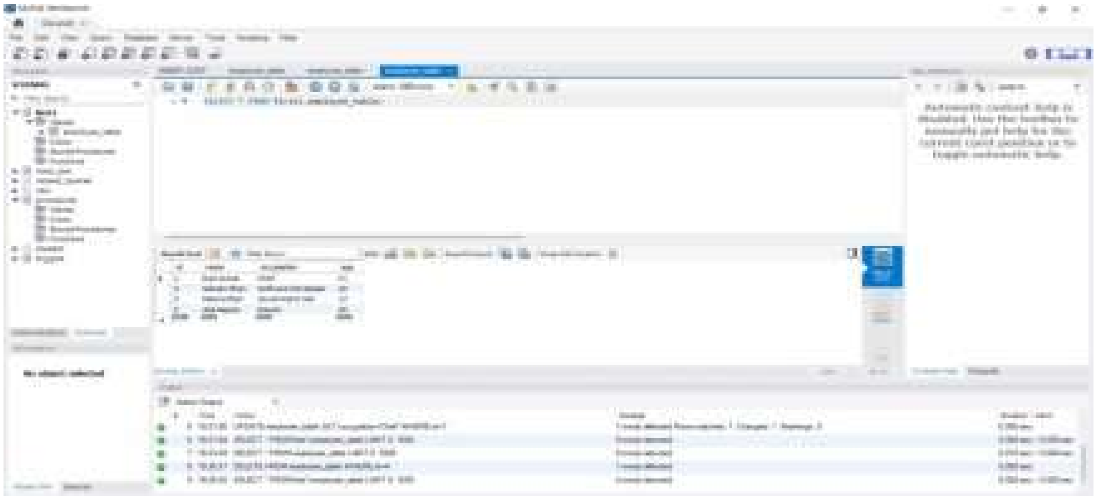
```
SELECT * FROM employee_table;
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

3. <u>Delete command:-</u>

DELETE FROM

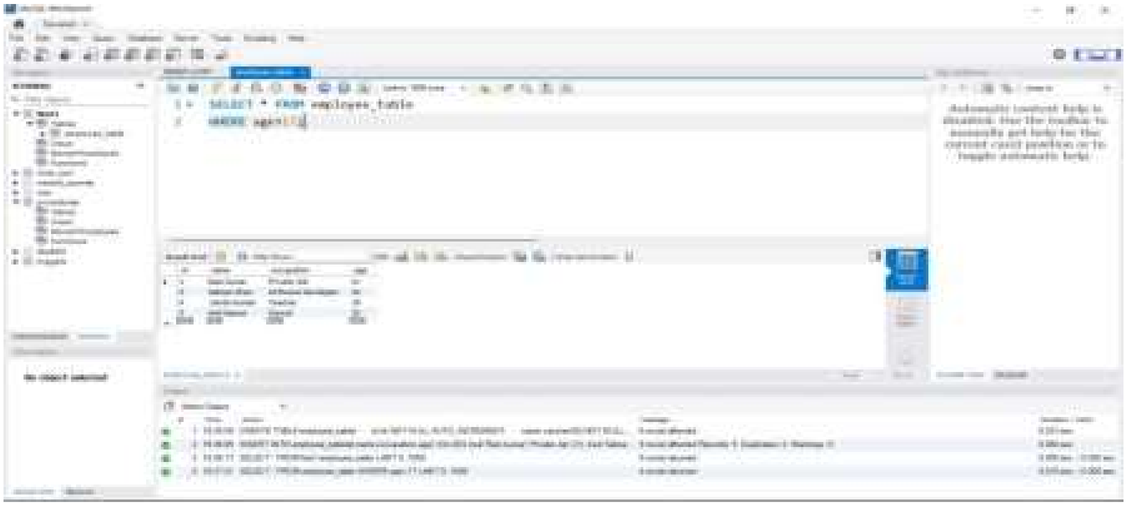
employee_tableWHERE
id=4;



To Implement Basic Select statements

1. <u>SELECT with WHERE Clause:-</u>

SELECT * FROM
employee_tableWHERE
age>17;



```
SELECT * FROM employee_table
WHERE age>17;
```

2. SELECT With ORDER BY:-

SELECT * FROM
employee_tableWHERE

age>17
ORDER BY name;

SELECT WITH DISTINCT:-

QUERIES:

SELECT DISTINCT age FROM employee_table;

To Execute Constraints in MySQL__**Theory:**__The constraint in MySQL is used to specify the rule that allows or restricts what values/data will be stored in the table. They provide a suitable method to ensure dataaccuracy and integrity inside the table. It also helps to limit the type of data that will be inserted inside the table. If any interruption occurs between the constraint and dataaction, the action is failed.

QUERIES:

```
CREATE TABLE employee(
id INT NOT NULL
UNIQUE,
name VARCHAR(50) NOT NULL,
age INT NOT NULL ,
CHECK(age>=18), gender
VARCHAR(10)NOT NULL,
city VARCHAR(10) NOT NULL DEFAULT 'Agra'
);
```

```
INSERT INTO
employee(id,name,age,gender)VALUES
(1,'Akshat',21,'M'),
(2,'Rahul',22,'M'),
(3,'Devansh',18,'M'),
(4,'Amit',19,'M'),
(5,'Akash',24,'M');
```

RESULT: The DML commands in RDBMS are implemented successfully and output is verified

Experiment No:-3

Aim:- SQL data control language and transaction control Language

DCL (Data Control Language) includes commands like GRANT and REVOKE, which are useful to give “rights & permissions.” Other permission controlsparameters of the database system.

Examples of DCL commands:-

Commands that come under DCL:-

- Grant
- Revoke

GRANT

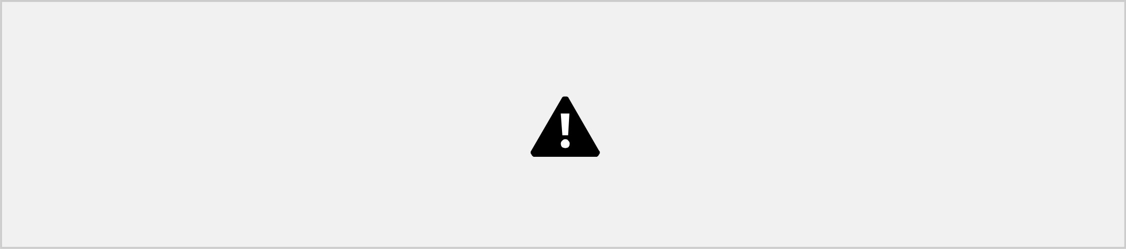
create user admin1 identified by password1;grant create session to

admin1;
grant connect to admin1;

REVOKE

revoke create session from
admin1;revoke connect from
admin1; revoke create table
from admin1;

GRANT



TCL Commands in SQL

➢ In SQL, TCL stands for Transaction control language. ➢ A single unit of work in a database is formed after the consecutive executionof commands is known as a transaction.

➢ There are certain commands present in SQL known as TCL commands thathelp the user manage the transactions that take place in a database.

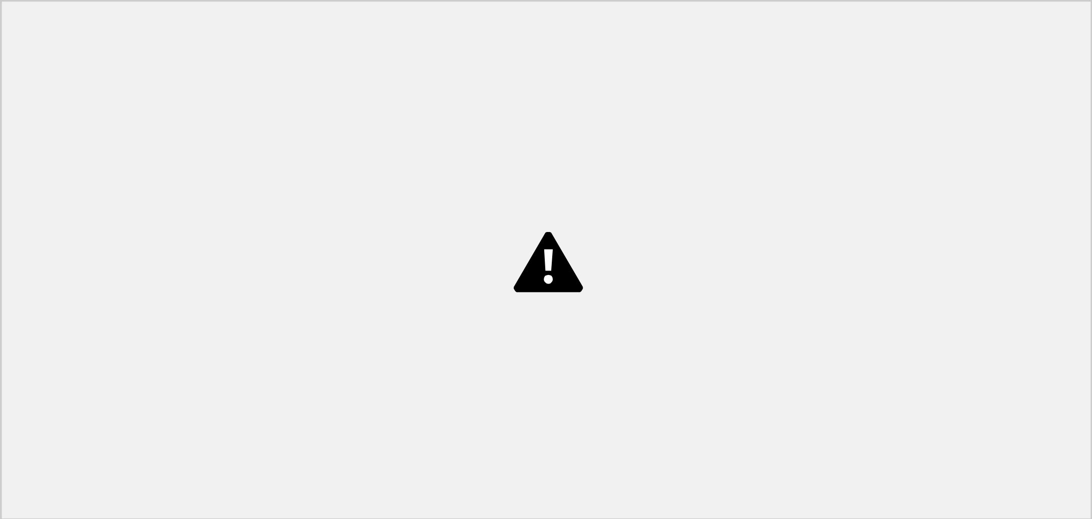
➢ COMMIT,ROLLBACK and SAVEPOINT are the most commonly used TCLcommands in SQL.

COMMIT:-

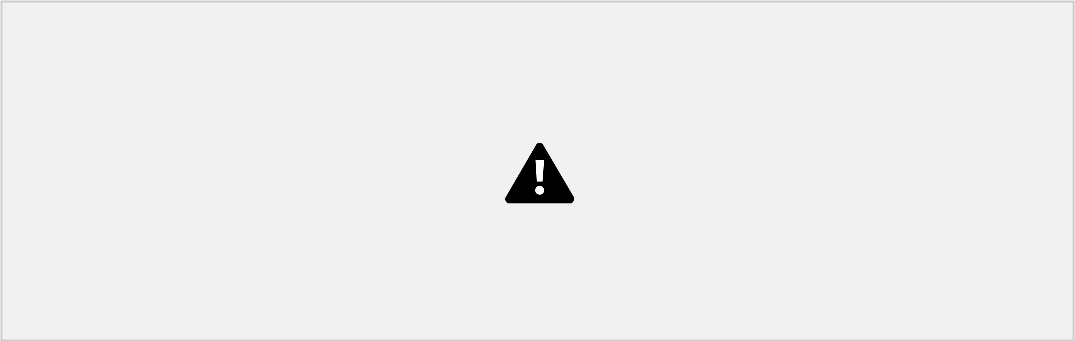
COMMIT command in SQL is used to save all the transaction-related changes permanently to the disk. Whenever DDL commands such as INSERT, UPDATE and DELETE are used, the changes made by these commands are permanent only after closing the current session. So before closing the session, one can easily roll back the changes made by the DDL commands. Hence, if we want the changes to be saved permanently to the disk without closing the session, we will use the commit command.

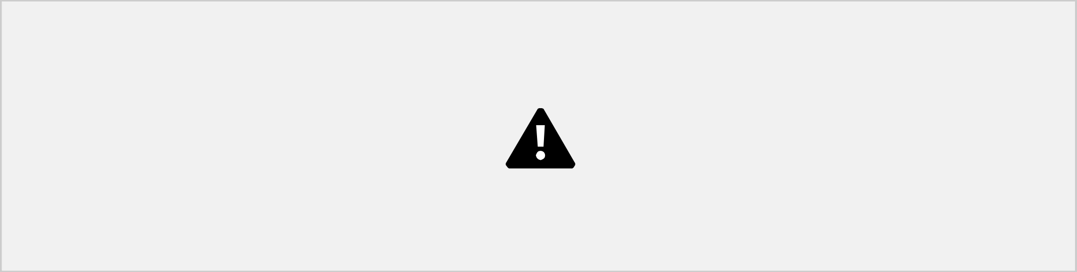
18

ROLLBACK: While carrying a transaction, we must create savepoints to save different parts of the transaction. According to the user's changing requirements, he/she can roll back the transaction to different savepoints. Consider a scenario: We have initiated a transaction followed by the table creation and record insertion into the table. After inserting records, we have created a savepoint INS. Then we executed a delete query, but later we thought that mistakenly we had removed the useful record. Therefore in such situations, we have an option of rolling back our transaction. In this case, we have to roll back our transaction using the ROLLBACK command to the savepoint INS, which we have created before executing the DELETE query.



SAVEPOINT: We can divide the database operations into parts. For example, we can consider all the insert related queries that we will execute consecutively as one part of the transaction and the delete command as the other part of the transaction. Using the SAVEPOINT command in SQL, we can save these different parts of the same transaction using different names. For example, we can save all the insert related queries with the savepoint named INS. To save all the insert related queries in one savepoint, we have to execute the SAVEPOINT query followed by the savepoint name after finishing the insert command execution.





RESULT: The DCL commands in RDBMS are implemented successfully and output is verified.

Experiment No:-4

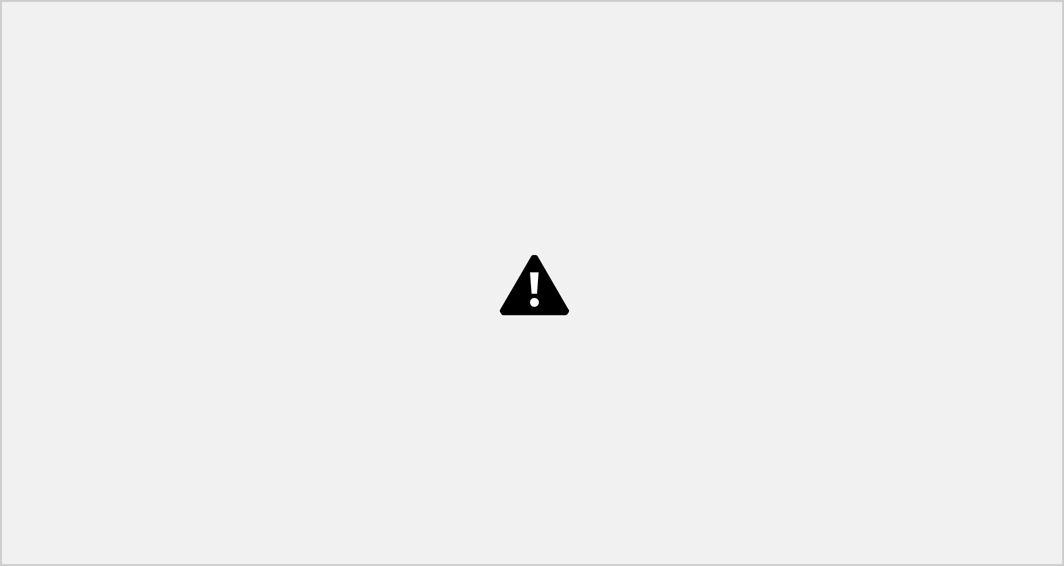
Aim:- To perform In-Built Functions In SQL.

To Execute String, Date,time function

Theory:- MySQL has many built-in functions.This reference contains string, numeric,date, and some advanced functions in MySQL.

1. **UPPER()**:-

```
SELECT id, UPPER(name) AS
Name ,percentageFROM employee;
```



2. **<u>LOWER():-</u>**

SELECT id, LOWER(name) AS
Name ,percentageFROM employee;

3. **<u>CHARACTER_LENGTH()</u>:-**
<u>QUERY:</u>-

SELECT id,NAME, character_length(name)AS
"CHARACTERLENGTH" ,percentage
FROM employee;

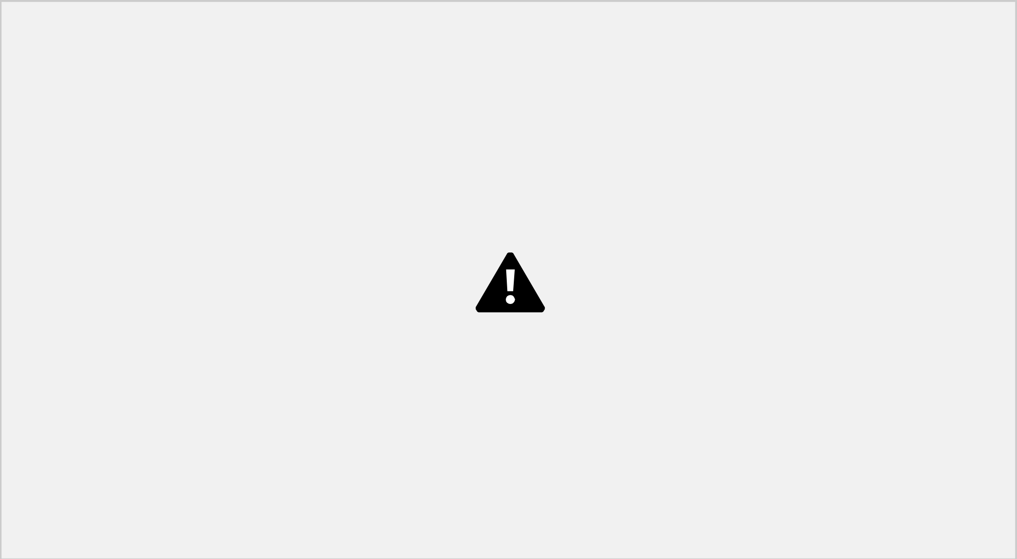
CONCAT():-

QUERY:-
SELECT id,NAME, CONCAT(name ,percentage) AS CONCATFROM employee;

DATE AND TIME FUNCTIONS:-

1.CURRENT_DATE():-

QUERY:-
SELECT CURRENT_DATE();



2. NOW():-

<u>QUERY:</u>-

SELECT NOW();

3.DATE():-<u>QUERY</u>:-

SELECT DATE("2022-04-24 23:48:15");

TIME FUNCTIONS:-

1.Current_time():-

QUERY:-

SELECT current_time();

2 Current_timestamp():-

QUERY:-

SELECT current_timestamp();

3. localtime():-

QUERY:-

SELECT localtime();

To Execute Aggregate Functions.: SQL aggregation function is used to perform the calculations on multiple rows of a single column of a table. It returns a single value.It is also used to summarize the data.

1. **<u>COUNT FUNCTION:</u>** COUNT function is used to Count the number of rows in a database table. It can workon both numeric and non-numeric data types.

QUERY:

SELECT COUNT(name) FROM employee_table;

2. <u>SUM Function:</u> Sum function is used to calculate the sum of all selected columns. It works on numeric fields only.

<u>QUERY:-</u>

SELECT SUM(percentage) FROM employee_table;

3. <u>AVG function:</u> The AVG function is used to calculate the average value of the numeric type.

<u>QUERY:-</u>

SELECT AVG(percentage) FROM employee_table;

4. <u>MAX Function:</u> MAX function is used to find the maximum value of a certain column. This function determines the largest value of all selected values of a column.

<u>QUERY:-</u>

SELECT MAX(percentage) FROM employee;

5. MIN Function: MIN function is used to find the minimum value of a certain column. This function determines the smallest value of all selected values of a column.

QUERY:-

SELECT MIN(percentage) FROM employee_table;

RESULT: The inbuilt function commands in RDBMS are implemented successfully and output is verified.

Experiment No:-5

Aim:-Construct a e- r model for the application to be constructedto a

Database: E-R model for Property Management Database Project

E-R model for Water Supply Management System

E-R model for Home renting system database

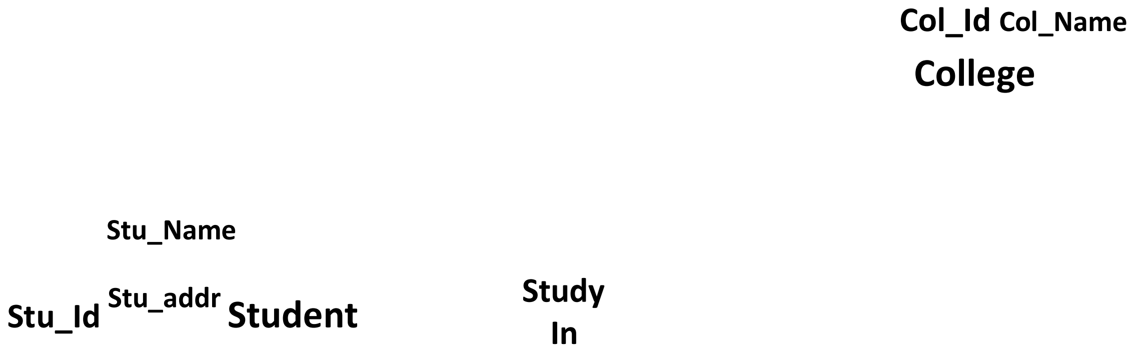
E-R model for Complaint management system database

E-R model for Employee performance review systemdatabase

E-R model for Employee track and report systemdatabase

Theory:-

ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.It develops a conceptual design for the database. It alsodevelops a very simple and easy to design view of data.In ER modeling, the database structure is portrayed as a diagram called an entity relationship diagram.



Experiment No:-6

Aim:- To Execute Subqueries

Theory:-A subquery in MySQL is a query, which is nested into another SQL query and embedded with SELECT, INSERT, UPDATE or DELETE statement along with the various operators. We can also nest the subquery with another subquery. Asubquery is known as the inner query, and the query that contains subquery is known as the outer query. The inner query executed first gives the result to the outer query, and then the main/outer query will be performed. MySQL allows us to use subquery anywhere, but it must be closed within parenthesis. All subquery forms and operations supported by the SQL standard will be supported in MySQL also.

QUERY:-

```
CREATE TABLE
department(dept_id INT
PRIMARY KEY,dept_name
VARCHAR(50)
);
```

```
INSERT INTO
departmentVALUES
(1,'H-R'),
(2,'Finance'),
(3,'Accounts'),
(4,'Administration')
,(5,'Counselling');


CREATE TABLE employee(
emp_id INT PRIMARY
KEY,name
VARCHAR(500),
gender
VARCHAR(50),age
INT,
salary INT,
dept_id INT,
FOREIGN KEY(dept_id)
REFERENCES
department(dept_id)
);
```

```
INSERT INTO
employeeVALUES
(1,'Ali','M',23,24000,3),
(2,'Anup','M',24,25000,4),
(3,'Akshay','M',22,22000,1),
(4,'Akshat','M',21,65000,2),
(5,'Rahul','M',23, 22000,4);
```

```
-- THIS IS NESTED QUERY--
SELECT * from employee
WHERE dept_id =(SELECT dept_id FROM departmentWHERE dept_name='H-R');
```

Subqueries:-

CREATING TABLE FOR SUBQUERY

```
CREATE TABLE
 department(id INT
 primary key,
 name varchar(100) NOT
 NULL,gender varchar(50)
 NOT NULL,city
 varchar(20) NOT NULL,
 salary int NOT NULL
 );

INSERT INTO
department(id,name,gender,city,salary)
VALUES
(1,'Ram Kumar','M','Rajasthan',12000),
(2,'Neeraj Singh','M','MP',15000),
(3,'Devansh Sharma','M','Delhi',30000),
(4,'Rahul Kalia','M','UP',40000),
(5,'Akshat Jain','M','UP',50000);
```

QUERY 1:-

```
SELECT * from department where id
IN(SELECT id from department where salary>12000);
```

QUERY 1:-

```
SELECT * from department where
salary>(SELECT AVG(salary)
from department);
```

Find the name and salary of the employee with maximum salary:-

CREATING TABLE

```
CREATE TABLE
 department(id INT,
 name
 varchar(100),
 gender
 varchar(50),
 city
 varchar(20),
 salary int
 );
```

```
INSERT INTO
department(id,name,gender,city,salary)
VALUES
(1,'Ram Kumar','M','Rajasthan',12000),
(2,'Neeraj Singh','M','MP',15000),
(3,'Devansh Sharma','M','Delhi',30000),
(4,'Rahul Kalia','M','UP',40000),
(5,'Akshat Jain','M','UP',50000);
```

<u>QUERY:-</u>

```
SELECT * from department where id
IN(SELECT max(salary) from
department);
```

Find the count of employees for each job so that at least two of the employees had salary greater than 10000:-

<u>QUERY:-</u>

```
SELECT count(name) AS COUNT from department
where name=(select count(salary)>10000 from department)
```

RESULT: The program for nested queries in RDBMS are implemented successfully and output is verified.

Experiment No:-7

Aim:-To Execute Joins in MySQL

Theory: MySQL JOINS are used with SELECT statement. It is used to retrieve data from multiple tables. It is performed whenever you need to fetch records from two or more tables.

There are three types of MySQL joins:

MySQL INNER JOIN (or sometimes called simple join)
MySQL LEFT OUTER JOIN (or sometimes called LEFT JOIN)
MySQL RIGHT OUTER JOIN (or sometimes called RIGHT JOIN)

MySQL Inner JOIN (Simple Join): The MySQL INNER JOIN is used to return all rows from multiple tables where thejoin condition is satisfied. It is the most common type of join.

QUERY:-

```
CREATE TABLE teacher(t_id int primary key,
name varchar(50) not null, qualification varchar(50) not null,salary int not
null );
```

```
INSERT INTO teacher VALUES (1,'Akshay','MCS',12000),
(2,'Amit','MBA',14000),
(3,'Aditya','MSC',13000),
(4,'Akshat','BSIT',15000),
(5,'Rahul','MPHIL',16000);

CREATE TABLE student(s_id int primary key,
name varchar(50) not null,class int not null,
t_id int not null
);

INSERT INTO studentVALUES (1,'Noman',11,2),
(2,'Asghar',12,4),
(3,'Furqan',10,2),
(4,'Khurram',11,1),
(5,'Asad',12,5),
(6,'Anees',10,1),
(7,'Khalid',11,2);
```

```
-- INNER JOIN QUERY--

SELECT t.t_id,t.name,t.qualification,s.name,s.classFROM teacher
t INNER JOIN student s ON t.t_id= s.t_id ORDER BY
t_id,t.name;
```

MySQL Left Outer Join:The LEFT OUTER JOIN returns all rows from the left hand table specified in the ON condition and only those rows from the other table where the join condition is fulfilled.

QUERY:-

```
CREATE TABLE city(
cid INT NOT NULL AUTO_INCREMENT, cityname VARCHAR(50) NOT NULL,
PRIMARY KEY(cid)
);

INSERT INTO city(cityname)VALUES
('Agra'),
('Delhi'),
('Bhopal'),
('Jaipur'),
('Noida');

CREATE TABLE personal(id INT NOT NULL,
name VARCHAR(50) NOT NULL,
percentage INT NOT NULL,age INT NOT NULL,
gender VARCHAR(1) NOT NULL,city INT NOT NULL,
PRIMARY KEY(id),
FOREIGN KEY(city) REFERENCES City(cid)
);

INSERT INTO
personal(id,name,percentage,age,gender,city)VALUES (1,'Ram
Kumar',45,19,"M",1),
(2,'Sarita Kumari',55,22,"M",2),
```

```
(3,'Salman Khan',62,20,"M",1),
(4,'Juhi Chawla',41,18,"M",3),
(5,'Anil Kaapoor',74,22,"M",1),
(6,'John Abraham',64,21,"M",2),
(7,'Shahid Kapoor',52,20,"M",1);

SELECT * FROM personalLEFT JOIN city
ON personal.city=city.cid;
```

MySQL Right Outer Join: The MySQL Right Outer Join returns all rows from the RIGHT-hand table specifiedin the ON condition and only those rows from the other table where he join condition is fulfilled.

SQL FULL OUTER JOIN Keyword:The FULL OUTER JOIN keyword returns all records when there is a match in left(table1) or right (table2) table records.

QUERY:-

```
SELECT * FROM personalLEFT JOIN city
ON personal.city=city.cid UNIONSELECT * FROM personal RIGHT JOIN
city ON personal.city=city.cid;
```

SQL CROSS JOIN Keyword: The CROSS JOIN keyword returns all records from both tables (table1 and table2).

QUERY:

```
SELECT * FROM studentCROSS JOIN City;
```

RESULT: The program for join queries in RDBMS are implemented successfully and output is verified.

Experiment No:-8

Aim:- To execute Set Operators and Views in Mysql

Theory:- SQL supports few Set operations which can be performed on the table data. These are used to get meaningful results from data stored in the table, under different special conditions.

SET OPERATORS:-

Union

Union all

Intersect

Minus

```
CREATING TABLES FOR SET OPERATORS(UNION and UNION
ALL):- CREATE TABLE student1(id INT,
name VARCHAR(255),
age INT
);

INSERT INTO student1(id,name,age)VALUES
(1,'Devansh Sharma',21),
(2,'Rahul Kalia',26),
(3,'Akshat Jain',34);

CREATE TABLE students(id INT,
name VARCHAR(255),
age INT
);

INSERT INTO students(id,name,age)VALUES
(1,'Devansh Sharma',21),
(2,'Sarita Kumari',26),
(3,'Rohan Dubey',34);
```

Union:-Query:-

```
SELECT *from student1UNION
SELECT * from students
```

Union All:-

```
SELECT *from student1UNION ALL
SELECT * from students
```

INTERSECT:-Query:-

FOR CREATING TABLE

```
CREATE TABLE tab1 ( Id INT PRIMARYKEY
);
INSERT INTO tab1 VALUES (1), (2), (3), (4);

CREATE TABLE tab2 (id INT PRIMARY KEY
);
INSERT INTO tab2 VALUES (3), (4), (5), (6);
```

FOR EXECUTION

```
SELECT DISTINCT Id FROM tab1INNER JOIN tab2 USING (Id);
```

MINUS:-Query:

FOR CREATING TABLE

```
CREATE TABLE t1 (
id INT PRIMARY KEY
```

```
);
CREATE TABLE t2 (
```

```
id INT PRIMARY KEY
);

INSERT INTO t1 VALUES (1),(2),(3);INSERT INTO t2 VALUES (2),(3),(4);
```

FOR EXECUTION

```
SELECT
id FROM
t1
LEFT JOIN
t2 USING (id)WHERE
t2.id IS NULL;
```

VIEWS:-

Theory:-

In SQL, a view is a virtual table based on the result-set of an SQL statement.

A view contains rows and columns, just like a real table. The fields in a view arefields from one or more real tables in the database.

You can add SQL statements and functions to a view and present the data as if thedata were coming from one single table.

A view is created with the CREATE VIEW statement.

QUERY:-

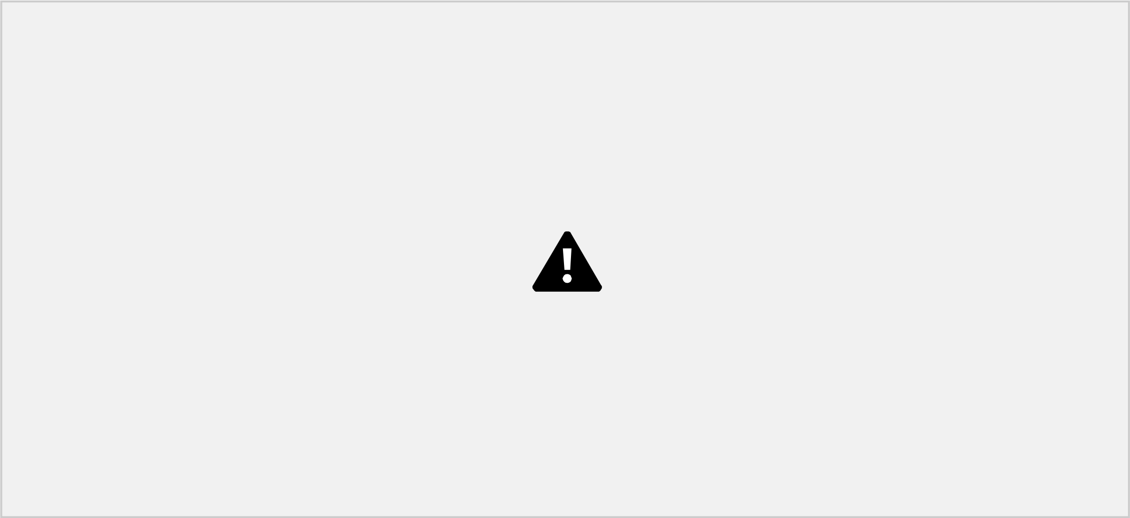
```
CREATE VIEW student_dataAS
SELECT s.id,s.name,c.cityFROM student s
INNER JOIN City c
```

```
ON s.id= c.cid;

SELECT * FROM student_data
```

IF we rename the existing view then we use RENAME Command:-RENAME TABLE
student_data
TO new_data;

```
SELECT * FROM new_data
```



IF we DELETE the VIEW then we use DROP VIEW:-

<u>QUERY:</u>-

```
DROP VIEW new_data;
```

ALTER IN VIEWSQUERY:-

```
ALTER VIEW dataAS
SELECT * FROM student_tableINNER JOIN City
ON student_table.city=City.cidWHERE age>22;
SELECT *from data;
```

RESULT: The program for Set Operators & Views in RDBMS is implemented successfully and output is verified.

Experiment No:-9

AIM – PL/SQL conditional and iterative statements.

Theory:- A program that supports a user interface may run a main loop that waits for, and then processes, user keystrokes (this doesn’t apply to stored programs, however).Many mathematical algorithms can be implemented only by loops in computer programs.When processing a file, a program may loop through each record in the file and perform computations.A database program may loop through the rows returned by a SELECT statement.

QUERY:

a)Conditional Statement

FOR CREATING PROCEDURE

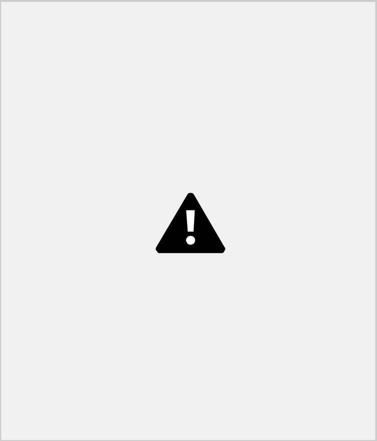
```
DELIMITER $$
CREATE
PROCEDURE
GetCustomerLevel(
   IN pCustomerNumber INT,
   OUT pCustomerLevel
VARCHAR(20))BEGIN
   DECLARE credit
DECIMAL(10,2)DEFAULT 0;
SELECT
   creditLimit
   INTO credit
   FROM
   customers
   WHERE
customerNumber =
pCustomerNumber;
IF credit > 50000 THEN
      SET pCustomerLevel =
   'PLATINUM';END IF;
END$$
DELIMIT
ER ;
```

FOR EXECUTION

```
SELECT
   customerNu
   mber,
   creditLimit
```

```
FROM
 customers
WHERE
   creditLimit >
```

```
50000ORDER
BY
   creditLimit DESC;
```



Iterative Statement

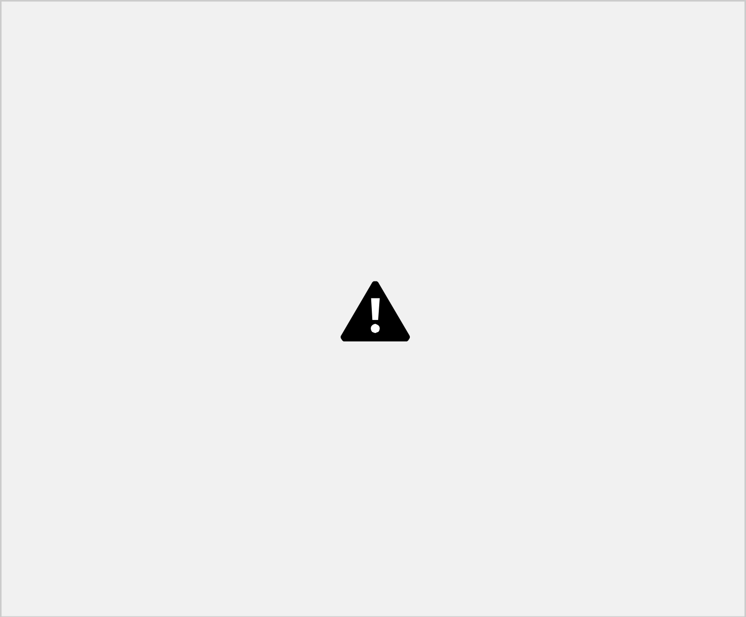
```
CREATE TABLE calendars(
id INT AUTO_INCREMENT,
fulldate DATE UNIQUE, day TINYINT NOT NULL,
month TINYINT NOT NULL,
quarter TINYINT NOT NULL,year INT NOT NULL, PRIMARY
KEY(id) );
```

FOR CREATING PROCEDURE

```
DELIMITER $$
CREATE PROCEDURE InsertCalendar(dtDATE)
BEGIN
INSERT INTO calendars(fulldate,
day, month, quarter,year
) VALUES(
dt,
EXTRACT(DAY FROM dt), EXTRACT(MONTH FROM dt), EXTRACT(QUARTER
FROM dt),EXTRACT(YEAR FROM dt)
);
END$$ DELIMITER ;

DELIMITER $$
CREATE PROCEDURE LoadCalendars(startDate DATE,
day INT
) BEGIN
DECLARE counter INT DEFAULT 1;
DECLARE dt DATE DEFAULT
startDate;
WHILE counter <= day DO CALL InsertCalendar(dt); SET counter = counter + 1;SET dt
= DATE_ADD(dt,INTERVAL 1 day);END WHILE;
END$$ DELIMITER ;
```

FOR EXECUTION: CALL LoadCalendars('2019-01-01',31);



RESULT: The program for PL/SQL conditional and iterative statements in RDBMS is implemented successfully and output is verified.

Experiment No:-10

Aim:- To Execute Procedure in MySql

Theory:- A procedure (often called a stored procedure) is a collection of pre- compiled SQL statements stored inside the database. It is a subroutine or a subprogram in the regular computing language. A procedure always contains a name, parameter lists, and SQL statements. We can invoke the procedures by using triggers, other procedures and applications such as Java, Python, PHP, etc. It was first introduced in MySQL version 5. Presently, it can be supported by almost all relationaldatabase systems.

If we consider the enterprise application, we always need to perform specific tasks such as database cleanup, processing payroll, and many more on the database regularly. Such tasks involve multiple SQL statements for executing each task. This process might easy if we group these tasks into a single task. We can fulfill this requirement in MySQL by creating a stored procedure in our database.

Query:-

```
create table film(rating int ,name varchar(20),release_date int); insert into film
values(4,'tomandJerry',90);
insert into film values(5,'harrypotter',21);insert into film values(2,'jamesBond',85);insert into
```

```
film values(3,'jumanji',22);

DELIMITER //
CREATE PROCEDURE sp_GetMovies()BEGIN
select rating,name,release_date from film;END //

DELIMITER ;

CALL sp_GetMovies();
```

RESULT: The program for PL/SQL Procedures on sample exercise in RDBMS is implemented successfully and output is verified.

Experiment No:-11

AIM - PL/SQL functions

Theory:- A function can be used as a part of SQL expression i.e. we can use them with select/ update/merge commands. One most important characteristic of a functionis that unlike procedures, it must return a value.

QUERY

CREATING TABLE FOR FUNCTION

```
CREATE TABLE
employee(emp_id INT,
fname
varchar(50), lname
varchar(50),
start_date date
);
INSERT INTO
employee(emp_id,fname,lname,start_date
)VALUES
(1,'Michael','Smith','2001-06-22'),
(2,'Susan', 'Barker','2002-09-12'),
(3,'Robert','Tvler','2000-02-09'),
(4,'Susan','Hawthorne','2002-04-24');
```

CREATING FUNCTION

```
DELIMITER //
CREATE FUNCTION no_of_years(date1 date) RETURNS int
DETERMINISTICBEGIN
 DECLARE date2 DATE; Select current_date()into date2; RETURN year(date2)-year(date1);
 END //
DELIMITER ;
```

CALLING FUNCTION

```
Select emp_id, fname, lname, no_of_years(start_date) as 'years' from employee;
```

OUTPUT:-

RESULT: The program for PL/SQL PL/SQL Functions in RDBMS is implemented successfully and output is verified.

Experiment No:-12

Aim:- To Execute Cursors in MySql

Theory:- A cursor allows you to iterate a set of rows returned by a query and processeach row individually. MySQL cursor is read-only, non-scrollable and asensitive. Read-only: you cannot update data in the underlying table through the cursor.

QUERY:-

```
CREATE TABLE GetVatsaCursor(
C_ID INT PRIMARY KEY AUTO_INCREMENT,
c_name VARCHAR(50), c_address VARCHAR(200)); CREATE TABLE Vbackupdata(C_ID
INT,
c_name VARCHAR(50), c_address VARCHAR(200));
INSERT INTO GetVatsaCursor(c_name, c_address) VALUES('Test', '132,
VatsaColony'), ('Admin', '133, Vatsa Colony'),
('Vatsa', '134, Vatsa Colony'),
('Onkar', '135, Vatsa Colony'),
('Rohit', '136, Vatsa Colony'),
('Simran', '137, Vatsa Colony'), ('Jashmin', '138, Vatsa Colony'), ('Anamika', '139, Vatsa
Colony'),('Radhika', '140, Vatsa Colony'); SELECT * FROM GetVatsaCursor;SELECT *
FROM Vbackupdata; delimiter //
CREATE PROCEDURE firstCurs()BEGIN
DECLARE d INT DEFAULT 0;DECLARE c_id INT;
DECLARE c_name, c_address VARCHAR(20);
DECLARE Get_cur CURSOR FOR SELECT * FROM
GetVatsaCursor;DECLARE CONTINUE HANDLER FOR SQLSTATE '02000'
SET d = 1;
DECLARE CONTINUE HANDLER FOR SQLSTATE '23000'SET d =
1; OPEN Get_cur;lbl: LOOP
IF d = 1 THEN
LEAVE lbl;
        END IF;
        IF NOT d = 1 THEN
        FETCH Get_cur INTO c_id, c_name, c_address;
        INSERT INTO Vbackupdata VALUES(c_id,
        c_name, c_address);END IF;
        END LOOP;
        CLOSE
        Get_cur;
        END;
        CALL firstCurs();
        SELECT * FROM Vbackupdata
```

RESULT: The program for **PL/SQL Cursors** in RDBMS is implemented successfully and output is verified.

Experiment No:-13

Aim- PL/SQL exception handling.

Theory:-When an error occurs inside a stored procedure, it is important to handle it appropriately, such as continuing or exiting the current code block's execution, and issuing a meaningful error message.MySQL provides an easy way to define handlersthat handle from general conditions such as warnings or exceptions to specific conditions e.g., specific error codes.

QUERY:-

```
DROP PROCEDURE IF EXISTS InsertSupplierProduct;
DELIMITER $$
CREATE PROCEDURE InsertSupplierProduct( IN
  inSupplierId INT,
   IN inProductId INT
)
BEGIN

   -- exit if the duplicate key occurs

   DECLARE EXIT HANDLER FOR 1062 SELECT

'Duplicate keys error encountered' Message; DECLARE
EXIT HANDLER FOR SQLEXCEPTION

SELECT 'SQLException encountered' Message;
DECLARE EXIT HANDLER FOR SQLSTATE '23000'
SELECT 'SQLSTATE 23000' ErrorCode;

   -- insert a new row into the SupplierProducts

   INSERT INTO SupplierProducts(supplierId,productId)
   VALUES(inSupplierId,inProductId);
   -- return the products supplied by the supplier idSELECT
   COUNT(*)

                  FROM SupplierProducts

                  WHERE supplierId =
                  inSupplierId;END$$
                  DELIMITER ;
```

<u>OUTPUT</u>

RESULT: The program for **PL/SQL EXCEPTION handling** in RDBMS is implemented successfully and output is verified.

Experiment No:-14

Aim:- To Execute Triggers in MYsql

Theory:- A trigger in MySQL is a set of SQL statements that reside in a system catalog. It is a special type of stored procedure that is invoked automatically inresponse to an event. Each trigger is associated with a table, which is activated on any DML statement such as INSERT, UPDATE, or DELETE.

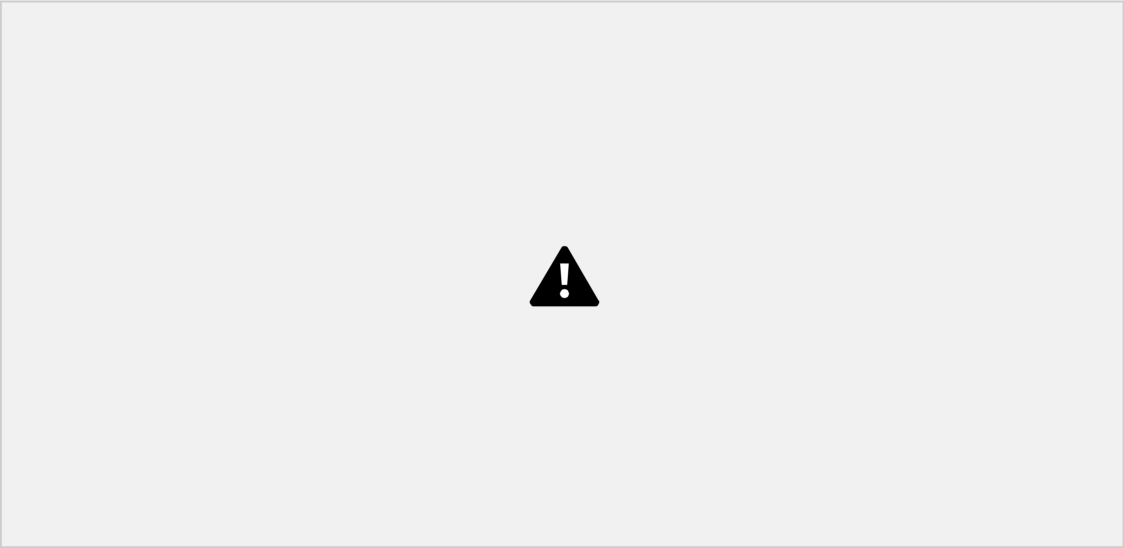
A trigger is called a special procedure because it cannot be called directly like a storedprocedure. The main difference between the trigger and procedure is that a trigger is called automatically when a data modification event is made against a table. In contrast, a stored procedure must be called explicitly.

QUERY:-

```
CREATE TABLE emp(
id INT PRIMARY KEY
AUTO_INCREMENT,name
VARCHAR(50),
age INT
);
CREATE TABLE emp_audit(
id INT PRIMARY KEY AUTO_INCREMENT,
audit_description VARCHAR(500)
);
DELIMITER //
CREATE TRIGGER
tr_AfterInsetEmpAFTER
INSERT
ON emp
FOR EACH
ROW
BEGIN
INSERT INTO emp_audit

VALUES(null,concat('newrow',date_format(now(),'%d-%m-%y
%h:%i:%s %p')));END//
DELIMITER
; INSERT
INTO emp
VALUES
```

```
(null,'Akash',22),
(null,'Devansh',18),
(null,'Akshat',21),
(null,'Rahul',24);
```



RESULT: The program for **PL/SQL trigger** in RDBMS is implemented successfully and output is verified.

Experiment No:-15

AIM –Frame and execute all queries for a project: Home renting system database

INTRODUCTION:-The Home Rental System is Searching in Based on the Apartment House for rent in metropolitan cities. The Home Rental System is Basedon the Owners and the Customers. The Owner is updated on the Apartment details,and rent details. The Customer is details about the Room space, Room rent and the Address Details also.

PROBLEM STATEMENT

There is no properly allocate home and the system is not easily arranges according to their user interest. And also the home rental management system almost is done through the manual system.

The administrative system doesn't have the facility to make home rental management system through online and the most time the work done through illegal intermediate personwithout awareness of the administrative and this make more complex and more cost to find home for the customer. This leads to customer in to more trouble, cost, dishonest and time wastage.

The problem found in the current system:

more tedious.

The system needs more human power.

OBJECTIVE

The main objective of the system is to develop online home rental managementsystem for wolkite city

Specificobjectives

In order to attain the general objective, the following are the list of specific objectives:

- To facilitate home record keeping for who wants home and for administrativemanagement system. •Prepare an online home rental system for the home finders •To reduce the travel costs and other unnecessary expenses of the buyer. • Reduce the role of the broker, thus, providing protection from frauds related topapers.

MODULES

- Tenant
- Owner
- Contract
- Payment

OPERATION:

First will be the login page where the User will login using their ID andPassword. Next the user will enter the details like the requirements of the house he needsand contact details

The other ebd user who wants to rent their house can also give their details He can also search the houses available in a particular area.

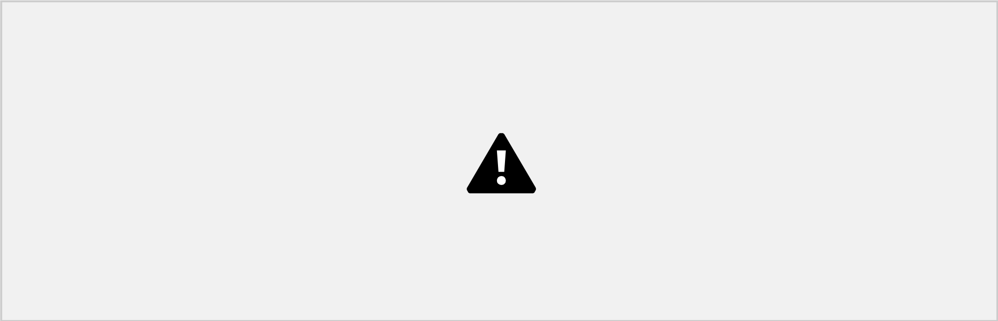
Whoever has the required spcifications of the house can contact using thedetails provided.

There is also a chatting option if they want to communicate with each other.There is also an option of video calling.

After providing the details in the website, a contract will be generated whichthey can download.

OUTPUT:-

a) Cursor Output



b) Exception Handling

<u>RESULT</u>: The program for Frame and Execute PL/SQL Cursor & Exceptional Handling in RDBMS is implemented successfully and output is verified.