Government College of Engineering, Jalgaon
(An Autonomous Institute of Government of Maharashtra)

Name : **Semester :** V **PRN :**
Class : T. Y. B.Tech Computer **Academic Year :** 2024-25 **Subject :** CO307U DBMS Lab
Course Teacher : Mr. Vinit Kakde
Date of Performance : **Date of Completion :**

Practical no. 2

Aim : Design and develop SQL DDL statements which demonstrate the use of SQL objects such as table, view, index, synonym.

Theory:

Introduction to SQL :

1. SQL stands for Structured Query Language.
2. SQL lets you access and manipulate database.
3. SQL is and ANSI standards.

· **Data Definition Language (DDL):**

· **DDL commands are used to define and manage database structures, such as tables, indexes, and schemas.**

- **Examples include:**
  - **CREATE: Creates a new table, index, or database.**
  - **ALTER: Modifies an existing database object, such as a table or column.**
  - **DROP: Deletes an existing database object, such as a table or database.**
  - **TRUNCATE: Removes all records from a table without deleting the table itself.**

· **Data Manipulation Language (DML):**

· **DML commands are used to manipulate the data stored in database objects.**

- **Examples include:**
  - **SELECT: Retrieves data from one or more tables.**
  - **INSERT: Adds new records into a table.**
  - **UPDATE: Modifies existing records within a table.**
  - **DELETE: Removes existing records from a table.**

· **Data Control Language (DCL):**

· **DCL commands deal with the rights, permissions, and other controls of the database system.**

- **Examples include:**
  - **GRANT: Provides users with access privileges.**
  - **REVOKE: Removes access privileges from users.**

. Transaction Control Language (TCL):

· **TCL commands manage the changes made by DML statements and ensure data integrity.**

- **Examples include:**
  - **COMMIT: Saves all changes made in the current transaction.**
  - **ROLLBACK: Reverts all changes made in the current transaction.**
  - **SAVEPOINT: Sets a point within a transaction to which you can later roll back.**
  - **SET TRANSACTION: Sets the properties of the current transaction.**

Queries and Output:

Aim: Create a table called Employee with the following structure.

Name	**Type**
Empno	Number
Ename	Varchar2(10)
Job	Varchar2(10)
Mgr	Number
Sal	Number

Solution:

1. **Create table:**
   Syntax:
   CREATE TABLE table_name (column1 datatype , column2 datatype , column3 datatype , ...);
   Query:
   mysql>create table employee(empno number, ename varchar2(10),job varchar2(10),mgr number ,sal number);

```
mysql> desc employee;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| empno | int         | YES  |     | NULL    |       |
| ename | varchar(10) | YES  |     | NULL    |       |
| job   | varchar(10) | YES  |     | NULL    |       |
| mgr   | int         | YES  |     | NULL    |       |
| sal   | int         | YES  |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

2. **Add a column commission with domain to the Employee table.**

Syntax:

Alter table table_name add(column_name type);

Query:

mysql>alter table employee add(commission number);

```
mysql> desc employee;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| empno      | int         | YES  |     | NULL    |       |
| ename      | varchar(10) | YES  |     | NULL    |       |
| job        | varchar(10) | YES  |     | NULL    |       |
| mgr        | int         | YES  |     | NULL    |       |
| sal        | int         | YES  |     | NULL    |       |
| commission | int         | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
6 rows in set (0.00 sec)
```

3. **Insert any five records into the table.**

Syntax:

insert into table_name

(column1,column2, ...columnN)

values

(value1,value2,..... ,valueN);

Query:

mysql>insert into employee values(101,'abhi','manager',1234,10000,'70');

mysql>insert into employee values(102,'rohith','analyst',2345,9000,'65');

mysql>insert into employee values(103,'david','analyst',3456,9000,'65');

mysql>insert into employee values(104,'rahul','clerk',4567,7000,'55');

mysql>insert into employee values(105,'pramod','salesman',5678,5000,'50');

```
mysql> select * from employee;
+-------+--------+----------+------+-------+------------+
| empno | ename  | job      | mgr  | sal   | commission |
+-------+--------+----------+------+-------+------------+
|   101 | abhi   | manager  | 1234 | 10000 |         70 |
|   102 | Rohit  | analyst  | 2345 |  9000 |         65 |
|   103 | david  | analyst  | 3456 |  9000 |         65 |
|   104 | rahul  | clerk    | 4567 |  7000 |         55 |
|   105 | pramod | salesman | 5678 |  5000 |         50 |
+-------+--------+----------+------+-------+------------+
5 rows in set (0.00 sec)
```

4. **Update the column details of job.**

   Syntax:

   update table_name set column1=value1,column2=value2, ... columnN=valueN;

   Query:

   mysql>update employee set job='trainee' where empno=103;

   mysql>select * from employee;

```
+-------+--------+----------+------+-------+------------+
| empno | ename  | job      | mgr  | sal   | commission |
+-------+--------+----------+------+-------+------------+
|   101 | abhi   | manager  | 1234 | 10000 |         70 |
|   102 | Rohit  | analyst  | 2345 |  9000 |         65 |
|   103 | david  | trainee  | 3456 |  9000 |         65 |
|   104 | rahul  | clerk    | 4567 |  7000 |         55 |
|   105 | pramod | salesman | 5678 |  5000 |         50 |
+-------+--------+----------+------+-------+------------+
5 rows in set (0.00 sec)
```

5. **Rename the column of the Employee table using alter command.**

   Syntax:

   alter table table_name rename to new_table_name;

   Query:

   mysql>alter table employee rename column mgr to manager_no;

```
mysql> desc employee;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| empno      | int         | YES  |     | NULL    |       |
| ename      | varchar(10) | YES  |     | NULL    |       |
| job        | varchar(10) | YES  |     | NULL    |       |
| manager_no | int         | YES  |     | NULL    |       |
| sal        | int         | YES  |     | NULL    |       |
| commission | int         | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
6 rows in set (0.00 sec)
```

6. **Delete the employee whose Empno is 105;**

   Syntax:

   delete from table_name where(condition);

   Query:

mysql>delete employee where empno=105;

```
mysql> select * from employee;
+-------+-------+---------+------------+-------+------------+
| empno | ename | job     | manager_no | sal   | commission |
+-------+-------+---------+------------+-------+------------+
|   101 | abhi  | manager |       1234 | 10000 |         70 |
|   102 | Rohit | analyst |       2345 |  9000 |         65 |
|   103 | david | trainee |       3456 |  9000 |         65 |
|   104 | rahul | clerk   |       4567 |  7000 |         55 |
+-------+-------+---------+------------+-------+------------+
4 rows in set (0.00 sec)
```

7. **alter table_name drop (column_name);**

   Syntax:

   Query:

   mysql>alter table employee drop(commission);

```
mysql> select * from employee;
+-------+-------+---------+------------+-------+
| empno | ename | job     | manager_no | sal   |
+-------+-------+---------+------------+-------+
|   101 | abhi  | manager |       1234 | 10000 |
|   102 | Rohit | analyst |       2345 |  9000 |
|   103 | david | trainee |       3456 |  9000 |
|   104 | rahul | clerk   |       4567 |  7000 |
+-------+-------+---------+------------+-------+
4 rows in set (0.00 sec)
```

8. **List the records of employee table grouped by EMPNO.**

   Syntax:

   select column1,aggregate_function(column2)from table_name group by column1;

   Query:

   mysql>select empno,count(ename) from employee group by empno;

```
+-------+--------------+
| empno | count(ename) |
+-------+--------------+
|   101 |            1 |
|   102 |            1 |
|   103 |            1 |
|   104 |            1 |
+-------+--------------+
4 rows in set (0.00 sec)
```

Conclusion:

In this practical we learned about how many languages are there in SQL (4 types) also we learned about DDL languages in which we practically implemented queries on postgresql.

We created table students and altered its column name, its data type, deleted column and added also. We created index on table and also deleted it. At last we fired truncate query to delete all rows in table and at last we fired drop table query to delete table so that out table schema is deleted from database.

Questions:

1) **Explain the use of DDL language :**

   DDL is a abbreviation of Data Defination Language. It is used to create and modify the structure of database objects in database.

2) **Explain the use of DML language :**

   DML is abbreviation of Data Manipulation Language. It is used to retrieve, store, modify, delete, insert and data in database.

3) **Explain the use of DCL language :**

   DCL is an abbreviation of Data Control Language. It is used to create rules, permissions, and referential integrity as well it is used to control access to database by securing it.

4) **Explain the use of TCL language :**

   TCL is an abbreviation of Transactional Control Language. It is used to manage different transactions occurring within a database.

5) **What is the difference between DROP TABLE and TRUNCATE commands of DDL language?**

   DROP TABLE command deletes the whole table whereas TRUNCATE deletes only rows present in the table.

Name & Sign of Course Teacher

Mr. Vinit Kakde