Date: 5/8/25 Task: 2 Generating design of traditional database model

Aim: To ~~study and~~ implement Data definition Language (DDL) commands in Relational Database management system (RDBMS) and DML commands in RDBMS.

I) DDL commands:- are used to design modify or delete the structure of database objects such as tables.

1) Create table - Create a new table in database.

Query :- create table students (Std id int, primary key, Std name varchar(50), rollno int, phno int);

create table employee (empid int, empname varchar(50));

Output:-

Table created(2).

2) Describe or Desc - Displays structure of a table

Query:

Desc students;

Output:

| Name | Null? | Type |
|---|---|---|
| St id | NOTNULL | NUMBER(38) |
| St.name | | VARCHAR 2(50) |
| roll no | | NUMBER(38) |
| phno | | NUMBER(38) |

3) ~~Alter~~ ALTER TABLE - used to add, delete or modify columns in existing table

Query:

alter table students add admission date;

alter table employee rename to employees;

Output : Table altered (2)

4) ~~Drop~~ DROP TABLE : Delete entire table structure and all its data.

Query:

drop table students;

output:

Table dropped.

II DML COMMANDS - Used to manage and manipulate data inside database tables.

1. INSERT INTO

Inserts new rows into a table

Query:

Insert all into employees (emp id, empname)
values (101, 'Rithees') into employees (emp id, empname)
values (102, 'Rithi') select * from dual;

output:

2 rows created.

2. UPDATE

Modifies existing data in a table

Query:

update employees set empname = 'Rithesh'
where emp id = 102;

Output: 1 row updated.

After update:

select * from employees.

output:

| emp id | empname |
|---|---|
| 101 | Rithees |
| 102 | Rithesh |

3) Select- Receives data from one or more tables select empname from employees;

Output:

| emp name |
|---|
| Rithees |
| Rithesh |

4) select with where clause
Retrieves specific records that satisfy condition.

Query: Select from * employees where empid = 101;

| emp-id | emp name |
|---|---|
| 101 | Rithees |

5) Delete.
Deletes one or more rows from table. Delete from employees where empid = 101;
Output: 1 row deleted

Output 1

| empid | empname |
|---|---|
| 102 | Rithesh |

| VEL TECH | |
|---|---|
| EX NO. | 2 |
| PERFORMANCE (5) | 5 |
| RESULT AND ANALYSIS (5) | 5 |
| VIVA VOCE (5) | 5 |
| RECORD (5) | - |
| TOTAL (20) | 15 |
| SIGN WITH DATE | 5/8/25 |

Result: Thus, the task to create, delete and alter the table are executed successfully.

19/8/25 TASK-2: DDL and DML Commands with Constraints

Aim: To implement DDL & DML commands with constraints.

DDL - (Data Definition Language) -> Create, Alter, Drop, Truncate, Rename)

DML - (Data Manipulation Language) - Insert, Update, Delete, Select.

Constraints -) Primary key, Foreign key, Not Null, Unique, Default.

1) DDL Commands:

Create table (slot_id int primary key, slot_type varchar(30), slot instructor varchar(30), join_date date, venue varchar(30));

OUTPUT: Table created.

Create table department (dept_id primary key, dept_name varchar(50), slot_id int foreign key (slot_id) references slot (slot_id)).

OUTPUT: Table created

Create table course (course_id int primary key, course_name varchar(50), credits_offered int, dept_id int, foreign key (dept_id) references department(dept_id)

OUTPUT: Table created.

Create table student (student_id int primary key, name varchar(50), email varchar(50) unique, dept_id, slot_id, academic_year, age.

foreign key (dept_id) references department (dept_id)

foreign key (slot_id) references slot (slot_id)

OUTPUT: Table created.

1.2 ALTER TABLE:

ALTER table Student add contact number

Output: Table Altered.

| Student id | name | email | Age | academic year | dept-id | Slot-id | Contact_number |
|---|---|---|---|---|---|---|---|
| 301 | Arun | arun@gmail com | 20 | 2 | 101 | 123 | NULL |
| 302 | Divya | divya@gmail com | 19 | 1 | 102 | 124 | NULL |

1.3 TRUNCATE TABLE:

Truncate table Student;

Output : Table Truncated.

1.4 DROP TABLE:

DROP table student;

Output: Table Dropped.

II DML Commands:

Insert:

Insert into Slot values ('1', 'Morning', 'Dr. Ravi', '2020-06-01', 'Hall A');

OUTPUT: 1 Row Inserted.

Insert into Slot values('2', 'Evening', 'Dr. Meena', '2019-07-01', 'Hall B');

OUTPUT: . Row Inserted.

Insert into Department values(101, 'Computer', 1);

Insert into Department values(102, 'Electronics', 2);

OUTPUT : Row Inserted.

Insert into course value(201, 'DBMS', 4 'Basic Sql', 101, 'Core')

Insert into Student value (301, 'Arun' 'arun@gmail.com', 20, 2, 121, 1);

Insert into Student values (302, 'Divya', 'divya@gmail.com', 19.1, 102, 2);

Output : 8 Rows Inserted.

UPDATE:

Update student set email:

output: Row updated.

'arun123@gmail.com' where student_id = 301;

| Std_id | Name | email | Age | Academic year | dept_id | slot_id | Phone number |
|---|---|---|---|---|---|---|---|
| 301 | Arun | arun123@gmail.com | 20 | 2 | 101 | 123 | NULL |
| 302 | Divya | divya@gmail.com | 19 | 1 | 102 | 124 | NULL |

DELETE:

Delete from student where student_id = 302;

output: 1 Row deleted.

| Std_id | Name | email | Age | Academic year | dept_id | slot_id | Phone number |
|---|---|---|---|---|---|---|---|
| 301 | Arun | arun123@gmail.com | 20 | 2 | 101 | 123 | NULL |

SELECT + WHERE:

Select name, email from student where age > 19;

output:

| name | email |
|---|---|
| Arun | arun@gmail.com. |

SELECT:

```
select * from student;
select name, student_id from student;
```

Output: (i)

| Std_id | Name | email | age | academic_year | dept_id | slot_id | phone_no |
|---|---|---|---|---|---|---|---|
| 301 | Arun | arun@gmail.com | 20 | 2022 | 101 | 123 | NULL |
| 302 | Divya | divya@gmail.com | 19 | 2021 | 102 | 124 | NULL |

(ii) output

| Name | Std_id |
|---|---|
| Arun | 301 |
| Divya | 302. |

| VEL TECH | |
|---|---|
| EX NO. | 2.1 |
| PERFORMANCE (5) | 5 |
| RESULT AND ANALYSIS (5) | 5 |
| VIVA VOCE (5) | 4 |
| RECORD (5) | — |
| TOTAL (20) | 14 |
| SIGN WITH DATE | |

Result:

Thus, the task to create DDL and DML commands created in task 1 entity in relational DBMS was been completed successfully.