

Taskno: 2

Date: 5/8/25

Generating Design of other Traditional DataBase Model.

Aim: To Implementation of DDL commands and DML Command Using SQL

1. CREATE: Def: Used to create a new tables in the database

Query: SQL > CREATE TABLE Employees (

ID INT PRIMARY KEY,

Name VARCHAR (50),

Age INT ,

Department VARCHAR (50)

Output: Table created successfully.

CREATE TABLE Department (

Dept ID INT,

Location VARCHAR (50)

);

Output:

Tables Employees and Department Created Successfully

2. Describe or DESC :

Def: Displays the structure of a table (column names and data types).

Query:

SQL

DESC Employee;

Output:

Name	Type
ID	INT
Name	VARCHAR(50)
Age	INT
Department	VARCHAR(50)

3. Drop table:

Def: Deletes the entire table structure and all its data.

Query:

Sql

```
DROPTABLE Employee;
```

Output:

Table Employee dropped successfully.

4. ALTER TABLE

Def: Used to add, delete or modify columns in an existing tables.

Query:

Sql: ALTER TABLE Employees ADD joiningSalary INT(10, 2);

Output:

Column joiningSalary added to Employees.

II. DML commands : (Data Manipulation Language):

1. Insert INT:

Def: Inserts new rows into a table

Query:

Sql: INSERT INTO employees (id, name, age, department)
VALUES (101, 'John Doe', '18', 'Finance');

INSERT INTO employees (id, name, age, department)
VALUES (102, 'Vinoth', '19', 'Manager');

INSERT INTO employees (id, name, age, department)
VALUES (103, 'Jax', '18', 'Worker');

Output:

3 rows inserted into employees table

2. SELECT:

Def: Retrives data from one or more tables

Query:

Sql

SELECT * FROM Collage;

output:

ID	Name	Age	department
101	John Doe	18	Finance
102	Vinoth	19	manager
103	Jax	19	Worker

3. UPDATE:

Def: Modifies existing data into a table

Query:

Sql:

UPDATE Collage Employees SET Age = 20 WHERE ID = 102;

Output:

1 row updated

After update:

Sql: SELECT * FROM employees;

ID	Name	Age	department
101	John Doe	18	Finance
102	Vinoth	20	manager
103	Jax	19	Worker

4 Delete : Def: Delete one ore more rows from a table:

Query: Sql: DELETE FROM Employees WHERE id = 103

Output:

1 row deleted

After delete

Sql: SELECT * from employees;

Output:

ID	Name	Age	department
101	John Doe	18	Finance
102	Vinoth	20	manager.

5. SELECT WITH WHERE clause :

Def: Return Specific records that satisfy

Query: sql

SELECT * FROM employees WHERE id = 101

Output:

ID	Name	Age	department
101	John Doe	18	Finance.

Result!

VEL TECH	
EX NO.	2
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	4
VIVA VOCE (5)	3
RECORD (5)	4
TOTAL (20)	14
SIGN WITH DATE	20/10/15

The task of ~~creating, deleting and altering~~ DDL and DML Commands in the database of the table are executed successfully.

Task: 2.1
Date: 5/8/25

DDL and DML Commands with Constraints

Aim:

To execute DDL and DML commands with constraints for the database

1. DDL Commands for E-Commerce Database management System:

1.1. Create Table:

Create Table User7(

User ID INT Primary Key,
Phone Varchar (15),
Name Varchar (100) Not Null,
Email Varchar (100) Unique Not Null,
Password Varchar (100) Not Null,
Address Varchar (100);
);

Create Table products(

Product ID int primary key,
Name Varchar (50) Not Null,
Description Varchar (50),
Image URL Varchar (50),
Price INT NOT NULL,
Stock int not null,
Qty int,
Discount int,
Category ID INT,
foreign key (Category ID) Reference Category
Category
);

Create Table Category(

Category ID Int Primary Key,
Category Name Varchar (50) Not null;

~~Create Table Orders(~~

Order ID Int Primary Key,
User ID Int,
Order Date DATE,
Total Amount Int,
Status Varchar (50),
foreign key (Order ID) Reference Order
(Order ID)

);

Create Table Order Details(

Order Details ID int primary key,
Order ID Int,
Product ID Int,
Quantity Int Not Null,
Price At Purchase int not null,

foreign key (orderID) Reference Orders (orderID),
foreign key (productID) References Product (productID);

Create Table Review (

ReviewID int Primary key.

User ID int,

product ID int,

Review Date DATE,

Ratings int check (rating between (and 5),

Comment Varchar (50)

foreign key (Product ID) References product (productID)

foreign key (User ID) References User (User ID);

Create Table payments (

payment ID int primary key,

Order ID int,

payment Date DATE,

Payment status Varchar (50),

foreign key (Order ID) References order (orderID);

1.2 ALTER Table :

Alter table Reviews ADD Rev Description Varchar (50);

1.3 Truncate Table :

Truncate Table Category;

Result: All rows removed from category table and structure remains

1.4 Rename Table :

Rename Table Category To categories;

2. DML commands for E-commerce datatype management System

2.1 INSERT Data:

Insert into User Values (111, '9574637810', 'sanjai', 'sanjai@gmail.com', 'password#\$', 'chennai');

Insert into products Values (711, 'cricket ball', 'Branded Cricket bat', 'bat.jpg', 2500, 10, 1, 3, 1);

Insert into Categories Values (1, 'sports items');

Insert into order Values (999, 111, '2025-08-19')

, 2500, 'pending'),

Insert into order Values (1999, 711, 1, 2500),

Insert into Review Values (101, 111, 711, '2025-

08-20', 4, 'wood');

product A worth ₹6, product B is at its good cost and afford to buy it');

Insert into payments Values (1000, 999, '2025-08-19' 'complete');

2.2 UPDATE DATA:

UPDATE User Set Email = 'Sanjai718@gmail.com' where User ID = 111;

After update :

Query :

Sql >

Select * from USER;

User ID	Phone	Name	Email	Password	Address
111	7574637810	Sanjai	Sanjai718@gmail.com	password#7	Chennai

2.3 DELETE Data

Delete from Order Details where Order Details ID = 1000;

After deletion :

Query :

Sql >

Select * from Order Details.

VEL TECH					
EX No.					2.2
PERFORMANCE (S)					6
RESULT AND ANALYSIS (S)					5
VIVA VOCE (S)					1
RECORD (S)					-
TOTAL (20)					11
GEN WRITING DATE					

orderDetails ID	Order ID	product ID	Quantity	price At purchase
11	999	711	1	2500

2.4 SELECT Data:

Used to retrieve information

Query: select * from product;

Product ID	Name	Description	Image URL	Price	Stock	Qty
711	Cricket ball	Branded Decathlon bat	bat.jpg	2500	10	1

Result: Thus the DDL and DML Commands for E-commerce Database Management System was executed successfully