Date: 12/8/25 TASK-2.1

Genarating design of other traditional DataBase model

Aim: To generate design of other traditional database model and implement DDL commonds of SQL with samples

Data definition language: (DDL)

Definition DDL commands are used to define modify (or) delete the structure of data base objects such as tables

DDL Queries:

SQL

```
CREATE TABLE customer(
    cust-ID INT PRIMARY KEY
    cust-Name VARCHAR(100),
    Phone-No VARCHAR(20),
    city VARCHAR(50),
    Amount_Paid DECIMAL(10,2)
);
```

output: Table created successfully

2. **DESCRIBE (or) DESC**

Definition: : Displays the resu structure of a table (column names and data type).

Query

SQL

```
DESC customer;
```

output

cust - ID	int
cust - Name	varchar (100)
Phone - No	varchar (20)
city	varchar (50)
Amount Paid	decimal (10,2)

3. DROP TABLE

Query.

```
DROP TABLE customer;
```

output

Table dropped successfully.

4. ALTER TABLE (Add fields in a table)

Query

```
ALTER TABLE customer
ADD COLUMN Email VARCHAR (100);
```

output

"commands completed successfully."

II

DML Queries:

Query:

SQL

```
INSERT INTO customer (cust-ID, cust-Name, Phone-No,
city, Amount-Paid) VALUES 1 'John Doe', 123-456-
7890' 'New York 10,00);
```

output

1 row inserted to customer

* SELECT: (Retrieves data from one or more tables)

query

```
sql
    SELECT * FROM customer;
```

output

Cust-ID	cust-Name	Phone-No	city	Amount-Paid
1.	John Doe	123-456-7890	New York	100.00
2	Smith	987-654-321	chicago	200.00
3.	krish	555-123-4567	Ameria	50.00

* UPDATE

query

```
SQL
    UPDATE customer
    SET Amount-Paid = 250.00
     WHERE cust-ID = 1;
```

output

1 row updated

* DELETE

query

```
sql
    DELETE FROM customer
     WHERE cust-ID = 2
```

output

1 row deleted

VEL TECH	
EX NO.	2
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	0
RECORD (5)	—
TOTAL (20)	10
SIGN WITH DATE	

10
17/5

Result:

Therefore DDL and DML commands using my SQL has been implemented successfully.

Task-22

Date: 2/8/25

Aim: To design and Implement a database for mobile Phone Purchase and Billing management system that manages information about customer, Bill, logic, mobile.

Steps:

1. Identify Entities
   - customer
   - Bill
   - logic
   - mobile
2. Identify Attributes
   customer → cust-Name, cust-ID, cust-PhoneNo, cust-city, cust-amount Paid
   Bill → Price, Bid, cust Name
   logic → Admin ID, Password.
   mobile → mobile-Name, mobile Price, mobile ID
3. Relationships
   - customer-mobile → (many-to-many) A customer can Purchases multiple mobiles
   - customer-Bill → (one-to-many) A customer can have bills and bill is with one costomer
   - Mobile-login → (one-to-many) A mobile is associated with one login be multiple mobiles.

```
CREATE TABLE customers(
cust_ID VARCHAR (255) PRIMARY KEY,
cust_Name VARCHAR (225) NOT NULL
cust_Phone_no VARCHAR (20) NOT NULL,
cust_city VARCHAR (225) NOT NULL,
cust_amount_Paid DECIMAL (10,2) NOT NULL
);
CREATE TABLE Bill (
Bill_ID VARCHAR (225) PRIMARY KEY,
Price DECIMAL (10,2) NOT NULL,
cust_Name VARCHAR (255) NOT NULL,
FOREIGN KEY (cust_Name) REFERENCES
        customer (cust_Name)
);
CREATE TABLE MOBILE(
mobile_ID VARCHAR (255) PRIMARY KEY
mobile_Name VARCHAR (255) NOT NULL,
mobile_Price DECIMAL (10,2) NOT NULL,
mPhone_ID VARCHAR (255) NOT NULL,
FOREIGN KEY (Phone_ID) REFERENCE Phone
                        (Phone_ID)
);
CREATE TABLE Admin(
    login_ID VARCHAR (255) PRIMARY KEY
    Admin_ID VARCHAR (255) NOT NULL
    Password VARCHAR (255) NOT NULL,
);
```

Constraints

1. Primary keys:
   - Login_ID in login
   - cust_ID in customer
   - Bid in Bill
   - Phone_ID in mobile

2. foreign keys:
   cust_Name in customer
   Phone_ID in mobile is a forgeign.

VEL TECH	
EX NO.	21
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	3
RECORD (5)	
TOTAL (20)	13
SIGN WITH DATE	

26/8

Result: thus the Implement a database for mobile Phone Purchase and Billing management system is successfully completed