Task-2

Generating Design of other Traditional database model

Aim: To generate design of other traditional database model and implement DDL Commands of SQL with sample

Data definition Language (DDL)

Definition: DDL commands are used to define modify or delete the database objects such as table

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

Describe (or) Desc:

displays the structure of a table (column names and datatypes)

Query:

```
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: (Adds fields in a table)

Query

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

Output:

"commands completed successfully".

DML Queries:

Insert:

Query:

sql

```
INSERT INTO customer (cust_ID, Cust_Name,
Phone_No, city, Amount_paid) VALUES ('John
Doe', '123-456-7890', 'New York', 100.00);
```

Output

1 row inserted to customer

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

Query

sql

```
SELECT * FROM customer;
```

Output:

Cust_ID	Cust_Name	Phone_NO	city	Amount_Paid
1	JohnDoe	123-456-7890	Newyork	100.00
2	Smith	987-654-321	Chicago	200.00
3	krish	555-123-4567	America	50.00

* UPDATE:

Query

sql

```
UPDATE Customers
SET Amount_Paid = 250.00
WHERE_cust_ID = 1;
```

Output

1 row updated

* DELETE

Query

sql

```
DELETE from customers
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	17/8

Result:

Therefore, DDL and DML commands using My SQL has been implemented successfully

12/8/25 Task-2b

Aim: To design and Implement a database for a database for a 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_amountpaid

Bill → price, Bid, CostName

Logic → Admin ID, password

mobile → Mobile_Name, mobile price, mobileID

3) Relationships

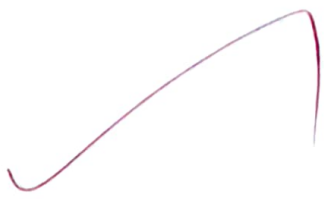
=> Customer_mobile → (many-to-many) A customer can Purchase multiple mobiles

=> Customer_Bill → (one-to-many) A customer can have Bills and bill is with one custom

=> Mobile_login → (one-to-many) A mobile is associated with one login can be multiple mobiles.

```
CREATE TABLE customer(
    Cust_ID VARCHAR(255) PRIMARY KEY,
    Cust_Name VARCHAR(255) NOT NULL,
    Cust_Phone_NO VARCHAR(20) NOT NULL,
    Cust_city VARCHAR(255) NOT NULL,
    Cust_amount_paid DECIMAL(10,2) NOT NULL
);
```

```
CREATE TABLE Bill (
    Bill_ID VARCHAR(255) 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,
    phone_ID VARCHAR(255) NOT NULL,
    FOREIGN KEY (Phone_ID) REFERENCES phone
        (phone_ID)
);
CREATE TABLE Admin (
    Login_ID VARCHAR(255) PRIMARY KEY,
    Adim_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 Bill
  reference Cust_Name in Customer
- Phone_ID in Mobile is a foreign key

VEL TECH	
EX NO.	2.1
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	3
RECORD (5)	
TOTAL (20)	13
IGN WITH DATE	

16/8/25

Result:

Thus the Design implement and a Database management system for the mobile phone hay been implemented successfully.