

12/09/2023

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 samples

DATA DEFINITION LANGUAGE (DDL) :

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

1) Create Table:

Definition: Used to create a new table in database Query SQL :

```
CREATE TABLE Mobile phone(
    Customer ID INT,
    Name VARCHAR(50),
    Brand VARCHAR(30),
    Amount INT
);
```

Table created

Output : Tables mobile phone and Customer.

2) DESCRIBE or DESC:

Definition: Displays the structure of table columns, names and databases.

Query:

SQL .

DESC mobile phone

Output :

ID	INT
Mobile	VARCHAR(50)
BRAND	VARCHAR(30)
AMOUNT	INT

3) DROP Table : (Deletes the table)

Query : DROP TABLE mobile phone

Output : Table mobile phone successfully deleted

4) Alter table (Adds Add in a table)

Query : Alter table mobile phone ADD

 modify Model - Name) VARCHAR(100)

II DML Queries:

* INSERT INTO : (insert new rows in table)

Query :

→ INSERT INTO mobile phone (ID, Mobile, Brand, Amount)
values (1, 'iphone', 'Apple', 1,00000);

Output:

1 row inserted to mobile phone.

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

Query :

SQL

SELECT * FROM mobilephone;

Output:

ID	Mobile	Brand	Amount
1	Realme	Narzo	30,000
2	Redmi	Poco	15,000
3	vivo	Iqoo	25,000

* Update : (modifies Existing data)

Query :

→ update mobile phone SET & ID = 2 WHERE
Amount = 30,000.

1 row updated.

* Delete : (Delete one or more rows from a table)

Query : 1 row deleted;

* Select : (Retrieves specific record that satisfy the
conditions)

Query :

SELECT * FROM mobile phones WHERE
 = 'Redmi';

Output:

ID	Mobile	Brand	Amount
2	Redmi	Poco	15,000

19|08|2025.

Task-2b

Aim: To design and implement a database for a mobile phone purchase and Billing Management System, that manages information about Customer, Bill, Logic, Mobile.

1) Identify Entities

- customer
 - Bill
 - logic
 - mobile

2) Identify Attributes

customer → cust-name, cust-ID, cust-phone No, 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 purpose multiple mobiles.
 - Customer-Bill → (one-to-many) A customer can have bills and bill is with one customer.
 - Mobile-login → (one-to-many) A mobile is associated with one login can be multiple mobiles.

```
CREATE TABLE customers(
```

cust-ID VARCHAR(255) PRIMARY KEY,
cust-Name VARCHAR(255) NOT NULL,
cust-phone-no VARCHAR(20) NOT NULL,
email VARCHAR(255) NOT NULL.

1

CREATED TABLE BillC

TABLE BILL
BILL_ID VARCHAR (255) PRIMARY KEY !

PRICE DECIMAL (10,2), NOT NULL

PRICE DECIMAL (10,2)
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.

Admin_ID VARCHAR(255) NOT NULL;

password VARCHAR(255) NOT NULL,

);

Constraints:

1) Primary Keys:

- Login_ID in login
- Cust_ID in customer
- Bid and 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.	21
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	3
RECORD (5)	
TOTAL (20)	13
SIGN WITH DATE	8/8/25

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