

GENERATING DESIGN OF OTHER TRADITIONAL AL DATABASE MODEL

Aim:- To generate design of other traditional database model and implement DDL commands of SQL with examples

DATA DEFINITION LANGUAGE (DDL) :-

Definitions of DDL commands are used to define, modify or delete the structure of database objects such as tables.

i) 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

DESCRIBE OR DESCRIBE

6

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

Output:-

SQCL

DESC mobile phone.

Output:-

ID	INT
MOBILE	VARCHAR(50)
BRAND	VARCHAR(50)
AMOUNT	INT

DROP TABLE:- (Deletes the table)

Query:- DROP TABLE mobile phone.

Output :- Table mobile phone successfully deleted

ALTER TABLE: (Add/Added in a table)

Query:- ALTER TABLE mobile phone ADD
model - NAME VARCHAR(100).

DML QUERIES:

Insert Into: (Putst new rows in table)

Query:

→ ~~INSERT INTO mobile phone (ID, mobile, brand, amount) values (1, 'iPhone', 'Apple', 100000);~~

Output :-

1 row inserted to mobile phone

* SELECT : CREATURES DATA FROM ONE OR MORE TABLE

Query :-

SQ1

SELECT * FROM MOBILEPHONE;

Output :-

ID	MOBILE	BRAND	AMOUNT
1.	REALME	NOKIA	30,000
2.	REDMI	POLY	15,000
3.	VIVO	IQOO	25,000

* Update : (modify 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 :

SELECT * FROM MOBILEPHONE WHERE
MOBILE = "Redmi"

Output :-

8

ID	MOBILE	BRAND	AMOUNT
3.	REDMI	POCO	15,000

VEL TECH	
EX NO.	2
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	0
RECORD (5)	1
TOTAL (20)	10

ON WITH DATE

8/8

19/8

RESULT:- Thus, the DDL and DML commands using my SQL have been implemented successfully.

19/08/2025

Aim:- To design and implement a database for a mobile phone purchase and Billing management system that manages information about customer Bill, login, mobile.

1. Identify entities

- Customer
- Bill
- Login
- Mobile

2. Identify attributes

Customer → cust - name, cust ID, cust - phone no,
cust - city, cust - amount paid.

Bill → Price, Bill ID, cust name.

Login → admin ID, password.

Mobile → mobile ID, mobile name, mobile price, mobile ID.

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 with the customer.
- 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 (20) NOT NULL,
 cust - city VARCHAR(225) NOT NULL,
 cust - amount - paid DECIMAL (10,12) 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 (

Admin - ID VARCHAR(255) PRIMARY KEY,

Admin - Name VARCHAR(255) NOT NULL

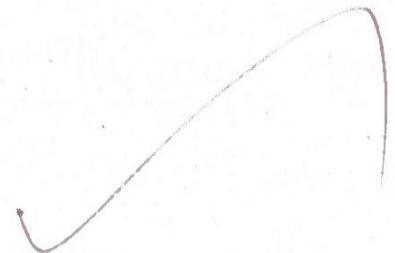
Admin - IP VARCHAR(255) NOT NULL

Passwd

);

2. Foreign keys

- VIT - Name in Bill
reference (VIT - Name in WIT) in
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
SIGN WITH DATE	<i>[Signature]</i>

26/8

RESULT:- Thus the design and implement and a database management system for the mobile phone has been implemented successfully.

