Task No:- 2	Generating Design of other Traditional Database model.
Date:- 09/08/25	

Aim:- Implementation of DDL and DML commands of SQL with suitable examples.

DDL commands (Data Definition Language)

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 the database.

Query:

```
Sql
Create table Student(Std_id int, Std_name Varchar(50),
  Std_ph_no varchar(10), Std_add varchar(100),
  Std_course Varchar(50));
```

Output:-

Table created.

2. Describe or desc

Definition: Displays the structure of a table (column names and datatypes).

Query:

```
Sql
DESC student.
```

Output:

Name	TYPE
STD_ID	NUMBER(38)
STD_NAME	VARCHAR2(50)
STD_PH_NO	VARCHAR2(10)
STD_ADD	VARCHAR2(100)
STD_COURSE	VARCHAR2(50)

3. Drop Table

definition: Deletes the entire table structure and all its data

Query:

```
sql
Drop Table Student ;
```

Output:-

Table Student dropped successfully.

4. Alter Table

Definition: Used to add, delete, or modify columns in an existing table.

Query:

```
sql
Alter table student Add course_fee int;
```

Output:-

Table Altered.

DML Commands (Data Manipulation language)

Definition: DML commands are used to manage and manipulate data ~~inside~~ database tables.

1. Insert into

Definition:- Inserts new rows into table.

Query:

```
sql
insert into student (200, 'John', '12345689', 'chennai',
                     'Semiconductors', 10000);
Insert into Student (300, 'Sita', '3874986', 'Vijayawada',
                     'Electronics', 20000);
insert into Student (500, 'Ram', '9869891', 'Bangalore',
                     'maths', 90008);
```

output:-

3 rows created

After altering the table:

Name	Type
STD-ID	NUMBER(38)
STD-NAME	VARCHAR2 (50)
STD-PH-NO	VARCHAR2 (10)
STD-ADD	VARCHAR2 (100)
STD-COURSE	VARCHAR2 (50)
COURSE-FEE	VARCHAR2 (38) NUMBER

2. Select

Definition: Retrieves data from one or more tables

Query:

Sql

Select * from student;

output:

STD_ID	STD_NAME	STD_PH_NO	STD_ADDRESS	STD_COURSE	COURSE_FEE
200	John	12345678	chennai	Semi-Conductors	10000
800	Sita	3874968	Banglore	Electronics	20000
500	Ram	9849861	Vijawada	maths	90000

3. Update

Definition: modifies existing data in a table

Query:

Sql

Update student set std_name = 'lakshman' where

Std_id = 200;

output:

1 row updated.

After update

select * from student;

STD_ID	STD_NAME	STD_PH_NO	STD_ADD	STD_COURSE	COURSE_FEE
200	lakshman	12345678	chennai	Semiconductors	10000
800	Sita	3874968	Banglore	Electronics	20000
500	Ram	9849861	Vijayawada	maths	90000

4. Delete Select

Defition : Retrieves specific records that satisfy the condition.

Query :

Sql

Select * from Student where Std-id = 200

Output:

STD-ID	STD-NAME	STD-PH-NO	STD-ADD	STD-COURSE	COURSE-FEE
200	Lakshman	12345678	chennai	semiConductor	10000

5. Delete

Definition: Deletes one or more raws from a table

Query:

Sql

Delete from student where Std-id = 200.

output:

1 raw deleted

output:- After delete :-

Select from * student

STD-ID	STD-NAME	STD-PH-NO	STD-ADD	STD-COURSE	COURSE-FEE
300	Sita	3874968	Banglore	Electronics	20000
500	Ram	9849861	Vijaywada	maths	90000

VEL TECH	
EX NO.	2
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	—
TOTAL (20)	15
IGN WITH DATE	9/8/25

Result :- The task to ran DDL and DML commands executed successfully.

Task No:- 2.1
Date:- 12/08/25

DDL and DML commands with constraints

Aim:- Implementation of DDL and DML commands with constraints.

DDL Commands

1.1 Create table

Definition:-
Used to create a new table in the database

Query:-
Sql
```
Create table customer(
    customerID primary key,
    name varchar(100) Not Null,
    address VARCHAR(200),
    phone_number
);

Create table customerCreditCard(
  creditCardNumber VARCHAR(20) PRIMARY KEY,
  expiry_date DATE NOTNULL,
  FOREIGN KEY (customerID) REFERENCES Customer(CustomerID
);

Create table Branch(
    branchID INT PRIMARY KEY,
    branchName VARCHAR(100) NOT NULL,
    location VARCHAR(100),
    ifsc_code VARCHAR(20) UNIQUE
);

Create table BankerInfo(
    bankerID INT PRIMARY KEY,
    bankerName VARCHAR(100) NOT NULL,
    bankerEmail VARCHAR(100) UNIQUE,
  FOREIGN KEY (branchID)REFERENCES Branch(BranchID)
);
```

desc customer;

Name	NULL	TYPE
CUSTOMERID	NOT NULL	NUMBER(38)
NAME	NOT NULL	VARCHAR2(100)
ADDRESS		VARCHAR2(200)

desc CustomerCreditCard;

Name	TYPE	NULL
CREDIT CARD NUMBER	VARCHAR2(20)	NOT NULL
EXPIRY-DATE	DATE	NOT NULL
CUSTOMERID	NUMBER(38)	

desc Branch;

Name	NULL	TYPE
BRANCH_ID	NOT NULL	NUMBER(38)
BRANCH NAME	NOT NULL	VARCHAR2(100)
LOCATION		VARCHAR2(100)
IFSC-CODE		VARCHAR2(20)

desc Bankerinfo;

Name	Null	Type
BANKERID	Not Null	NUMBER(38)
BANKER NAME	Not Null	VARCHAR2(100)
BANKEREMAIL		VARCHAR2(100)
BRANCHID		NUMBER(38)

```
Create table loan(
    loanNumber INT PRIMARY KEY,
    amount  INT ,
    FOREIGN KEY (customerID) REFERENCES Customer(customerID),
    FOREIGN KEY (branchID) REFERENCES Branch(branchID)
);
Create table Account (
    accountNumber INT PRIMARY KEY
    balance  INT ,
    category VARCHAR(50),
    FOREIGN KEY (customerID) REFERENCES Customer(customerID),
    FOREIGN KEY (branchID) REFERENCES Branch(branchID)
);
```

1.2 Alter Table

```
Alter table customer add Ph_NO VARCHAR(10);
```

1.3 Truncate Table

```
Truncate table loan
```

Result: All rows are removed from loan table, structure remains.

1.4 Rename Table

```
Rename Table customer to Customers.
```

2. DML commands

2.1 Insert data

```
insert into Customers (customerID, name, address, Ph-no)
Values (238, 'Ram', 'chennai', '83456789');

insert into customerCreditCard (creditCardNumber, expiry_date)
Values ('8329 9258 6234', 12-MAR-2030);
```

desc loan;

Name	Null	Type
LOAN_NUMBER	NOT NULL	NUMBER(38)
AMOUNT		NUMBER(38)
CUSTOMER_ID		NUMBER(38)
BRANCH_ID		NUMBER(38)

desc Account;

Name	Null	Type
ACCOUNT_NUMBER	NOT NULL	NUMBER(38)
BALANCE		NUMBER(38)
CATEGORY		VARCHAR2(50)
CUSTOMER_ID		NUMBER(38)
BRANCH_ID		NUMBER(38)

After altering the table:

1.2 desc customer;

Name	Null	Type
CUSTOMERID	NOT NULL	NUMBER(38)
NAME	NOT NULL	VARCHAR2(100)
ADDRESS		VARCHAR2(100)
PH_NO		VARCHAR2(10)

1.4 Rename Table

Table renamed.

insert into Branch(branchID, branchName, location, ifsc_code) values (4590, 'chennaibranch', 'chennai', '8925 4596 0311');

insert into Bankerinfo(bankerID, bankerName, bankerFmail) values (7896, 'chandu', 'chandu41@gmail.com'); (7897, 'nandhu', 'nandhu72@gMail.com');

insert into loan(loan number, amount) values (8996, 50000);

insert into Account (accountNumber, Balance, Category) values (5985423108, 100000, 'Savings');

2.2 Update Data

Update Customers set Name = 'Vinay' where customerID = 238;

Result: Name is updated to Vinay.

2.3 Delete data

Delete from Bankerinfo where bankerID = 7896;

2.4 Select Data

Select name, Ph-no From customers;

Name	Ph-no
Ram	8345678 9

VEL TECH	
EX NO.	2.2
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	15
SIGN WITH DATE	17/8/2

Result :- Thus implementation of DDL and DML commands with constraints executed successfully.

customer_ID	Name	Address	Ph_no
238	Ram	chennai	83456789

creditCard number	expiry_date	customer_ID
8329 92586234	12-MAR-2030	238

BranchID	branchName	location	IFSC_code
4590	chennaiBranch	chennai	8925459031l

BankerID	bankerName	bankerEmail	BranchID
7896	Chandu	chandu4@gmail.com	4590
7897	nandhu	nandhu72@gmail.com	4590

loan number	amount	customer_ID	Branch_ID
8996	50000	238	4590

account number	Balance	category
5985423108	10000	savings

After updating the table:

CustomerID	Name	Address	Ph_no
238	Vinay	chennai	83456789

After deleting the table:

BankerID	BankerName	bankerEmail	BranchID
7897	nandhu	nandhu72@gmail.com	4590