

Task No: 2 Generating Design of other traditional Database model.

Date: 05/03/25

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 object such as tables.

1. Create table:

Definition: used to create a new table in the database

Query :

SQL

create table structure employee(Emp-ID int, Emp-NAME varchar(50),
EMP-ph-no varchar(10), EMP-add varchar(100),
EMP-course varchar(50));

Output:

Table created.

2. Describe or desc.

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

Query:

SQL

DESC employee.

Output:

Name	Type
Emp-ID	Number(38)
Emp-NAME	VARCHAR(50)
EMP-ph-no	VARCHAR(10)
EMP-add	VARCHAR(100)
EMP-course	VARCHAR(50)

Drop table:

Defination: Deletes the entire table structure and all its data

query:

SQL

```
Drop table employee;
```

Output:

Table employee dropped successfully.

4. Alter table:

Defination: used to add, delete, or modify columns in an existing table.

query:

SQL

```
Alter table employee ADD salary;
```

Output:

column salary added to employee;

II DML COMMANDS (Data manipulation language)

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

1. Insert INTO

Defination: Inserts new rows into a table

query:

SQL

```
Insert into employee(310,'pallavi','1,2,3,4,5,6,7,8,9,8,'Anantapur,  
'Coding', 30,000);
```

```
Insert Into employee(340,'Radha','7993382491','Karnataka');
```

'Coding', '40,000');

Insert Info employee (210, 'Lakshmi', '3456789123', 'Tirupati', '50,000)

Output: 3 rows inserted into employee table.

Select:

Definition: retrieves data from one or more tables.

query:-

Sql

select * from employee;

Output

employee: ID	emp-name	emp-ph-no	emp-add	emp-salary
310	Pallavi	1234567898	Anantapur	30,000
340	Radha	9993382491	Karnataka	40,000
810	Lakshmi	3456789123	Tirupati	50,000

update:

Definition: Modifies existing data in a table.

Sql

update employee set employee-name = 'Radha'

where emp-name = 'Krishna'

Output:

1 row update.

After update:

Sql

select * from employee.

Emp-ID	emp-name	Emp-ph-no	Emp-add	Emp-salary.
310	Pallavi	1234567898	Anantapur	30000
340	Radha Krishna	7993382491	Karnataka	40000
210	Lakshmi	3456789126	Tirupati	50000

Delete

Definition: Deletes one or more rows from a table

Query:

Delete from employee where empID = 210

Output:

1 row deleted.

alter delete;

SQL

Select * from employee

Emp-ID	Emp-name	Emp-ph-no	Emp-add	Emp-salary.
340	Krishna	7993382491	Karnataka	40000
210	Lakshmi	3456789123	Tirupati	50000

5) Select with where Clause

Definition: Retrieves specific records that satisfy the condition

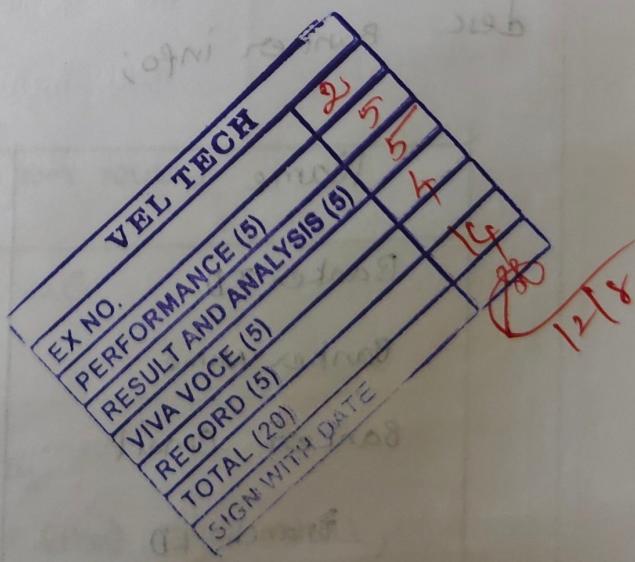
Query:

Select * from employee where Employee.emp-name = "krishna"

Output:

output:

Emp-ID	Emp-name	Emp-Ph-no	Emp-add	Emp-salary.
340	krishna	9993382491 (80) 9993382491 9993382491 (081) 9993382491 (081) 9993382491	Karnataka Bengaluru 223456789 123456789	40000
				10000
				15000
				20000
				25000
				30000
				35000
				40000
				45000
				50000
				55000
				60000
				65000
				70000
				75000
				80000
				85000
				90000
				95000
				100000



Result: The task to create, delete and alter the table are executed successfully.

DDL AND DML Commands with Constraints

Aim : implementation of DDL and DML commands with constraints.

DDL Commands

1) create table:

Def: used to create a new table in the database.

SQL:

```
create table customer (
```

Customer ID Primary Key,

name varchar(100) NOT NULL,

address varchar(200),

```
);
```

```
create table customer creditcard (
```

creditcard number VARCHAR(20) primary key

Expiry Date KEY DATE, NOT NULL,

FOREIGN KEY (Customer ID) REFERENCES CUSTOMER (Customer ID)

```
);
```

```
create table Branch (
```

Branch ID int primary key,

branch Name VARCHAR(100) NOT NULL,

location VARCHAR(100),

FSC-Code VARCHAR(20) UNIQUE

```
);
```

```
create table Banker into (
```

banker ID INT PRIMARY KEY,

branch banker Name varchar(100) NOT NULL,

banker Email VARCHAR(100) UNIQUE

FOREIGN KEY (Branch ID) REFERENCES Branch (Branch ID)

```
);
```

desc customers;

Name	Type
Customer ID	Number (38)
Name	VARCHAR(100)
Address	VARCHAR(100)

desc customer credit card;

Name	Type
Credit - card number	VARCHAR(20)
Expiry - date	Date
Customer ID	Number (38)

desc Branch;

Name	Type
Branch ID	Number (38)
Branch name	VARCHAR(100)
Location	VARCHAR(100)
ifsc - code	VARCHAR(20)

desc Banker info;

Name	Type :-
Banker ID	Number (38)
Banker Name	VARCHAR(100)
Banker Email	VARCHAR(100)
Branch ID	Number (38)

Create table loan
loan number INT PRIMARY KEY,
amount INT
FOREIGN key (customer ID) REFERENCES customer (Customer ID),
FOREIGN key (Branch ID) REFERENCES Branch (branch ID);
};

Create table Account
account Number INT PRIMARY KEY
balance INT
category VARCHAR (50)
FOREIGN key (Customer ID) REFERENCES customer (Customer ID),
FOREIGN key (branch ID) REFERENCES branch (branch ID);
};

1.2 Alter table:

Alter table customer add ph-no VARCHAR (10);

1.3 Truncate table:

Truncate table loan;

1.4 Rename table:

Rename Table customer to customer.

2. DML Commands:-

2.1 Insert data:

insert into customer (Customer ID, Name, address, ph.no)

values (238, 'Ram', 'chennai', '9834567891');

insert into customer credit card (credit card number,
Expiry - date)

values (C83294562P 6234, 12-MAR-2030);

insert into Branch (branch ID, branch Name, location, ifsc -
code)

values (4590, 'chennai branch', 'chennai', 98254567 031);

insert into Banker info (banker ID, banker name, banker email))

values (4896, 'Chandy', 'chandu1@gmail.com), (7897, nandhu, nandhu2@gmail.com)

desc loans;

28-3-21 3:20

Name	Type
loan-number	Number(38)
Amount	Number(38)
customer ID	Number(38)
Branch ID	Number(38)

Name	Type
Account-number	Number(38)
Balance	Number(38)
category	VARCHAR 2(50)
customer ID	Number(38)
Branch ID	Number(38)

1.2 desc customer;

Name	Null	Type
Customer ID	Number(38)	
Name	NOT NULL	VARCHAR 2(100)
Address	NOT NULL	VARCHAR 2(100)
ph-no		VARCHAR 2(100)

1.4 Rename table.

2.1 Table renamed.

Insert customer

Customer ID	Name	Address	Ph-no
238	Ram	Chennai	834567891

Insert - credit card number

credit card number	expiry date	Customer-ID
832992586834	12-mar-2030	238

Branch-ID

Branch ID	branch Name	Location	ifsc-code
4590	Chennai branch	chennai	897545981

Insert - Banker info

banker ID	Banker name	Banker email	Branch ID
7896	Chandu	chandu@ gmail.com	4590

Insert - loan

loan numbers	amount	customer ID	Branch - ID
8996	50000	238	4590

Insert - account numbers

account number	balance	category
5985423108	10000	savings

After update the table

Customer ID	Name	Address	ph-no
238	Vinay	Chennai	8345678910

After deleting the table

Banker ID	Banker name	Banker email	branch ID
7897	Nandu	nandu72@gmail.com	4590

Name	ph-no.
Ram	8345678910

Value(8996, 5)

insert into loan(loan number, amount)

Values(8996, 50000);

insert into Account(account number, balance, category)

Values(58985423108, 100000, 'Savings');

2.2 update Data:

update customer set Name = 'vinay' where customer ID = 238;

2.3 Delete Data:

Delete from Banker info where banker ID = 7896;

24. select data:

Select name, ph-no from customers;

VELTECH	
EX No.	VELTECH
EX No.	21
PERIOD	5
PERIOD	6
PERIOD	2
PERIOD	12
PERIOD	18
PERIOD	29

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