**Ex.No:1** **DATA DEFINITION COMMANDS, DATA MANIPULATION COMMANDS FOR INSERTING, DELETING, UPDATING AND RETRIEVING TABLES AND TRANSACTION CONTROL STATEMENTS**

**AIM:**
To design and implement a database for manipulating & storing data items in MYSQL by using SQL commands.

**CREATE: (Syntax)**

**Database:**

Create database <database name>;

Use <database name>;

**Table:**

Create table <table name> (column_name1 datatype1 constraints, column_name2 datatype2 constraint....column_nameN datatypeN constraints);

**Note:** Constraints is optional

**DDL COMMANDS:**

**1)** Create
2) Alter
   - ✓ Add
   - ✓ Modify
   - ✓ Drop
3) Rename
4) 4) Drop

**ALTER**:

**ADD:**

Alter table<table name>add column_name1 datatype1 constraints;

**MODIFY**:

Alter table<table name>modify column_name1 datatype1;

**DROP:**

Alter table<table name>drop column_name;

**RENAME**:

Rename table <old table name>to<new table name>;

**DROP:**

Drop table<table name>;

**DML COMMANDS:**

1) Insert 3) Select
2) Update 4) Delete

**INSERT**:

Insert into<tablename>values(Value1,value2…….ValueN);

**SELECT:**

Select<column name>from<table name> where condition;

**UPDATE**:

Update<table name>set<column name>=values where condition;

**DELETE**:

Delete from<table name> where condition;

**TCL COMMANDS:**

1) Commit
2) Rollback
3) Savepoint

**COMMIT:**

Commit;

**ROLLBACK:**

Rollback to <savepoint>;

**SAVEPOINT:**

Savepoint <savepoint name>;

**PROBLEM STATEMENT:**

- A **branch** contains many **account**holders.
- A branch provides more than one **loan**.
- A loan can be availed by more than **customer**.
- A customer can get more than one loan.
- A customer can have more one account.
- An account can have more than one customer.

**1. TABLE FROM THE PROBLEM STATEMENT:**

1) Branch_m
2) Account_m
3) Loan_m
4) Customer_m

**Database Name: it**

```
mysql>create database it;
mysql>use it;
```

**Table Name: Branch_m**

```
mysql> create table branch_m(branch_name varchar(20) primary
key,branch_city varchar(20),asset int);

Query OK, 0 rows affected
mysql> desc branch_m;
```

**Table name: Customer_m**

```
mysql> create table customer_m(customer_id varchar(20) primary key,customer_name varchar(20),customer_street varchar(20),customer_city varchar(20));
Query OK, 0 rows affected

mysql> desc customer_m;
```

**Table name: Account_m**

```
mysql> create table account_m(account_no varchar(20) primary key,branch_name varchar(20),balance int,foreign key(branch_name) references branch(branch_name));
Query OK, 0 rows affected

mysql> desc account_m;
```

```
3 rows in set
```

**Table name: Loan_m**

```
mysql> create table loan_m(loan_no varchar(20) primary key,branch_name varchar(20),amount int,foreign key(branch_name) references branch(branch_name));
Query OK, 0 rows affected
mysql> desc loan_m;
```

**2.Alter the table branch_m by increasing the field width of branch city to 25.**

```
mysql> desc branch_m;
```

```
mysql> alter table branch_m modify branch_city varchar(25);

Query OK, 0 rows affected
Records: 0  Duplicates: 0  Warnings: 0
mysql> desc branch_m;
```

**3.Drop the primary key from loan_m**

```
mysql> desc loan_m;
```

```
mysql> alter table loan_m drop primary key;

Query OK, 0 rows affected
Records: 0  Duplicates: 0  Warnings: 0
mysql> desc loan_m;
```

**4.Alter the primary key to loan_m**

```
mysql> desc loan_m;
```

```
mysql> alter table loan_m add primary key(loan_no);
Query OK, 0 rows affected
Records: 0  Duplicates: 0  Warnings: 0
mysql> desc loan_m;
```

**5.Add new column to loan_m**

```
mysql> desc loan_m;
```

```
mysql> alter table loan_m add roi int;

Query OK, 0 rows affected
Records: 0  Duplicates: 0  Warnings: 0
mysql> desc loan_m;
```

**6.Drop the column from loan_m**

```
mysql> desc loan_m;
```

```
mysql> alter table loan_m drop roi;

Query OK, 0 rows affected
Records: 0  Duplicates: 0  Warnings: 0
mysql> desc loan_m;
```

**7. Rename the customer_m as customer_ma**

```
mysql> desc customer_m;
```

```
mysql> rename table customer_m to customer_ma;

Query OK, 0 rows affected

mysql> desc customer_m;
ERROR 1146 (42S02): Table 'lab.customer' doesn't exist

mysql> desc customer_ma;

+-----------------+------------+------+-----+---------+-------+
```

**8)a) Drop customer_ma**

```
mysql> desc customer_ma;
```

```
mysql> drop table customer_ma;

Query OK, 0 rows affected

mysql> desc customer_ma;
ERROR 1146 (42S02): Table 'lab.customer1' doesn't exist
```

**8)b) Rename the column loanamount to amount from loan table.**

```
mysql> create table loan_m(loan_no int primary key,branch_name
varchar(20),loanamount int);
Query OK, 0 rows affected (0.20 sec)

mysql> desc loan_m;
```

```
mysql> alter table loan_m change column loanamount amount int;
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc loan_m;
```

**9. INSERTRECORDS IN ALL THE FOUR CREATED TABLES:**

**Insert the values given below.(Branch Table)**

| BRANCH_NAME | BRANCH_CITY | ASSETS |
|---|---|---|
| Perryridge | Rye | 50000 |
| Downtown | Stamford | 100000 |
| Brighton | Paloalto | 25000 |
| Redwood | Harrison | 150000 |
| Mianus | Pitsfield | 450000 |
| Roundhill | Princeton | 150000 |

```
mysql> desc branch_m;

mysql> insert into branch_m values('perryridge','rye',50000);
Query OK, 1 row affected
```

**Insert the values given below. (Loan Table)**

| LOAN | BRANCH_NAME | AMOUNT |
|---|---|---|
| 1_11 | Roundhill | 900 |
| 1_14 | Downtown | 1500 |
| 1_15 | Perryridge | 1500 |
| 1_16 | Perryridge | 1300 |
| 1_17 | Downtown | 1000 |
| 1_23 | Redwood | 2000 |
| 1_93 | Mianus | 500 |
| l_102 | Mianus | Null |

```
-----+-----+---------+-------+
mysql> desc loan_m;
 +-------------+-------------+-
-----+-----+---------+-------+
| Field        | Type         |
Null | Key | Default | Extra |
 +-------------+-------------+-
```

```
| loan_no     | varchar(20) | NO   | PRI | NULL    |       |
| branch_name | varchar(20) | YES  | MUL | NULL    |       |
| amount      | int(11)     | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set
mysql> insert into loan_m values('l_23','redwood',2000);
Query OK, 1 row affected
```

**Insert the values given below. (Customer Table)**

| CUSTOMER_ID | CUSTOMER_NAME | CUSTOMER_STREET | CUSTOMER_CITY |
|---|---|---|---|
| c_01 | Smith | north | rye |
| c_02 | Turner | putnam | stamford |
| c_03 | Johnson | alma | paloalto |
| c_04 | Curry | north | rye |
| c_05 | Jones | main | harrisdon |
| c_06 | Adoms | spring | pittsfield |
| c_07 | Lindsay | park | pittsfeild |
| c_08 | Hayes | main | harrison |
| c_09 | Williams | nassu | princeton |

```
mysql> desc customer_m;
```

```
4 rows in set
mysql>insert into customer_m
values('c_01','smith','north','rye');
Query OK, 1 row affected
```

**Insert the values given below. (Account Table)**

| ACCOUNT_NO | BRANCH_NAME | BALANCE |
|---|---|---|
| 019_28_3746 | Perryridge | 1500 |
| 182_73_6091 | Downtown | 1800 |
| 192_83_7465 | Brighton | 500 |
| 321_12_3123 | Redwood | 2300 |
| 336_96_9999 | Mianus | 500 |
| 963_96_3963 | Roundhill | 500 |

| | | |
|---|---|---|
| 376_66_9999 | Mianus | 900 |
| 963_96_3964 | Mianus | 1300 |

```
mysql>desc account_m;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
```

```
mysql>insert into account_mvalues(019_28_3746,'perryridge',1500);
Query ok,1 row affected.
```

**10. Find the names of all branches in loan relation.**

```
mysql> select branch_name from loan_m;
```

```
7 rows in set
```

**11. Find the names of all branches in loan relation eliminateduplicate**.

```
mysql> select distinct branch_name from loan_m;
```

```
5 rows in set
```

**12.Updatethe customer city stamford to rye in customerrelation.**

```
mysql>   update   customer_m   set   customer_city='rye'   where
customer_city='stamford';

Query OK, 2 rows affected
Rows matched: 2  Changed: 2  Warnings: 0

mysql>commit;
```

**13.Showthe effect of savepoint and roll back command using delete query with example.**

```
mysql> start transaction;
Query OK, 0 rows affected
mysql> select * from loan_m;
```

```
8 rows in set
mysql> savepoint s1;
savepoint created.
mysql> delete from loan_m;
Query OK, 8 rows affected
mysql> select * from loan_m;
Empty set
mysql> rollback to s1;
Query OK, 0 rows affected
mysql> select * from loan_m;
```

```
8 rows in set
mysql>commit;
```

**RESULT:**

**INFERENCE:**

1. What are the disadvantages of file processing system?
2. Explain the basic structure of a relational database with an example.
3. What do you mean by weak entity set?
4. Give example for one to one and one to many relationships.
5. What is the need of normalization?