Title: Creation of Base Table and DML Operations

Author: S. Santhosh Kumar

AIM:

To perform various DML operations on a base table using SQL.

ALGORITHM:

STEP-1: Start

STEP-2: Create a base Table

```
CREATE TABLE MY_EMPLOYEE (
    ID NUMBER(4) NOT NULL,
    Last_name VARCHAR2(25),
    First_name VARCHAR2(25),
    Userid VARCHAR2(25),
    Salary NUMBER(9,2)
);
```

STEP-3: Describe the Table structure

```
DESC MY_EMPLOYEE;
```

STEP-4: Add new rows to the Table using INSERT statement

```
INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary)
VALUES (1, 'Patel', 'Ralph', 'rpatel', 895);
INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary)
VALUES (2, 'Dancs', 'Betty', 'bdancs', 860);
```

STEP-5: Display the table with values

```
SELECT * FROM MY_EMPLOYEE;
```

STEP-6: Populate the next two rows of data from the sample data

Concatenate the first letter of the first_name with the first seven characters of the last_name to produce Userid:

```
INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid,
Salary)

VALUES (3, 'Biri', 'Ben', SUBSTR('Ben', 1, 1) ||
SUBSTR('Biri', 1, 7), 1100);

INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid,
Salary)

VALUES (4, 'Newman', 'Chad', SUBSTR('Chad', 1, 1) ||
SUBSTR('Newman', 1, 7), 750);
```

STEP-7: Delete Betty Dancs from MY_EMPLOYEE table

```
DELETE FROM MY_EMPLOYEE WHERE First_name = 'Betty' AND
Last_name = 'Dancs';
```

STEP-8: Empty the fourth row of the MY_EMPLOYEE table

```
DELETE FROM MY_EMPLOYEE WHERE ID = 4;
```

STEP-9: Make the data additions permanent

```
COMMIT;
```

STEP-10: Change the last name of employee 3 to Drexler

```
UPDATE MY_EMPLOYEE SET Last_name = 'Drexler' WHERE ID = 3;
```

STEP-11: Change the salary to 1000 for all the employees with a salary less than 900

```
UPDATE MY_EMPLOYEE SET Salary = 1000 WHERE Salary < 900;
```

STEP-12: **`Exit`**

EXERCISE

1. Create MY_EMPLOYEE table with the following structure:

Command:

```
CREATE TABLE MY_EMPLOYEE (
   ID NUMBER(4) NOT NULL,
   Last_name VARCHAR2(25),
   First_name VARCHAR2(25),
   Userid VARCHAR2(25),
   Salary NUMBER(9,2)
);
```

Output:

Table created.

2. Describe the table structure:

Command:

```
DESC MY_EMPLOYEE;
```

Output:

Name	Null	Type
ID	NOT NULL	NUMBER(4)
Last_name		VARCHAR2(25)
First_name		VARCHAR2(25)
Userid		VARCHAR2(25)
Salary		NUMBER(9,2)

3. Add the first and second rows of data to MY_EMPLOYEE table:

Command:

```
INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary)
VALUES (1, 'Patel', 'Ralph', 'rpatel', 895);

INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary)
VALUES (2, 'Dancs', 'Betty', 'bdancs', 860);
```

Output:

1 row inserted.

1 row inserted.

4. Display the table with values:

Command:

```
SELECT * FROM MY_EMPLOYEE;
```

Output:

ID	Last_name	First_name	Userid	Salary
1	Patel	Ralph	rpatel	895
2	Dancs	Betty	bdancs	860

5. Populate the next two rows of data from the sample data. Concatenate the first letter of the first_name with the first seven characters of the last_name to produce Userid:

Command:

```
INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary)
VALUES (3, 'Biri', 'Ben', SUBSTR('Ben', 1, 1) || SUBSTR('Biri', 1, 7), 1100);

INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary)
VALUES (4, 'Newman', 'Chad', SUBSTR('Chad', 1, 1) || SUBSTR('Newman', 1, 7), 750);
```

Output:

1 row inserted.

1 row inserted.

6. Delete Betty Dancs from MY_EMPLOYEE table:

Command:

```
DELETE FROM MY_EMPLOYEE WHERE First_name = 'Betty' AND Last_name =
'Dancs';
```

Output:

1 row deleted.

7. Empty the fourth row of the MY_EMPLOYEE table:

Command:

```
DELETE FROM MY_EMPLOYEE WHERE ID = 4;
```

Output:

1 row deleted.

8. Make the data additions permanent:

Command:

COMMIT;

Output:

Commit complete.

9. Change the last name of employee 3 to Drexler:

Command:

UPDATE MY_EMPLOYEE SET Last_name = 'Drexler' WHERE ID = 3;

Output:

1 row updated.

10. Change the salary to 1000 for all the employees with a salary less than 900:

Command:

UPDATE MY_EMPLOYEE SET Salary = 1000 WHERE Salary < 900;

Output:

2 rows updated.

Final Data in MY_EMPLOYEE table:

Command:

SELECT * FROM MY_EMPLOYEE;

Output:

ID	Last_name	First_name	Userid	Salary
1	Patel	Ralph	rpatel	1000
3	Drexler	Ben	bbiri	1100
5	Ropebur	Audrey	aropebur	1550