

## **EXERCISE-2**

### **MANIPULATING DATA**

<b>NAME</b>	<b>PRATHESHA J</b>
<b>ROLL NO</b>	<b>241001172</b>
<b>DEPARTMENT</b>	<b>IT</b>

## **MANIPULATING DATA**

1. Create MY\_EMPLOYEE table with the following structure

The screenshot shows a database interface with the following details:

- Autocommit is checked.
- Rows are set to 10.
- Buttons for Save and Run are present.
- The SQL code is:

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

Below the code, there are tabs for Results, Explain, Describe, Saved SQL, and History. The Results tab is selected.

Output:

```
Table created.
```

Execution time:

```
0.15 seconds
```

2. Add the first and second rows data to MY\_EMPLOYEE table from the following sample

The screenshot shows a database interface with the following details:

- Autocommit is checked.
- Rows are set to 10.
- Buttons for Save and Run are present.
- The SQL code is:

```
INSERT ALL
INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary) VALUES (1, 'Patel', 'Ralph', '', 895)
INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary) VALUES (2, 'Dancs', 'Betty', '', 860)
INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary) VALUES (3, 'Biri', 'Ben', '', 1100)
INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary) VALUES (4, 'Newman', 'chad', '', 750)
INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary) VALUES (5, 'Ropebur', 'Audrey', '', 1550)
SELECT * FROM dual;
```

Below the code, there are tabs for Results, Explain, Describe, Saved SQL, and History. The Results tab is selected.

Output:

```
5 row(s) inserted.
```

Execution time:

```
0.03 seconds
```

3. Display the table with values.

The screenshot shows a MySQL query editor interface. At the top, there are buttons for 'Autocommit' (checked), 'Rows' (set to 10), and 'Save/Run'. Below the toolbar is a text input field containing the SQL query: 'SELECT \* FROM MY\_EMPLOYEE;'. Underneath the query, there is a navigation bar with tabs: 'Results' (which is selected and highlighted in blue), 'Explain', 'Describe', 'Saved SQL', and 'History'. The main area displays a table with the following data:

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	-	895
2	Dancs	Betty	-	860
3	Biri	Ben	-	1100
4	Newman	chad	-	750
5	Ropebur	Audrey	-	1550

Below the table, a message says '5 rows returned in 0.03 seconds' and there is a 'Download' link.

4. 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.

The screenshot shows a MySQL query editor interface. At the top, there are buttons for 'Autocommit' (checked), 'Rows' (set to 10), and 'Save/Run'. Below the toolbar is a text input field containing the SQL query: 'UPDATE MY\_EMPLOYEE SET Userid = SUBSTR(First\_name, 1, 1) || SUBSTR(Last\_name, 1, 7);'. Underneath the query, there is a navigation bar with tabs: 'Results' (selected), 'Explain', 'Describe', 'Saved SQL', and 'History'. The main area displays the message '5 row(s) updated.' followed by '0.00 seconds'.

Autocommit    Rows

```
SELECT * FROM MY_EMPLOYEE;
```

**Results Explain Describe Saved SQL History**

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	RPatel	895
2	Dancs	Betty	BDancs	860
3	Biri	Ben	BBiri	1100
4	Newman	chad	cNewman	750
5	Ropebur	Audrey	ARopebur	1550

5 rows returned in 0.00 seconds [Download](#)

5. Delete Betty dancs from MY\_EMPLOYEE table

Autocommit    Rows

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

**Results Explain Describe Saved SQL History**

1 row(s) deleted.

Autocommit    Rows

```
SELECT * FROM MY_EMPLOYEE;
```

**Results Explain Describe Saved SQL History**

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	RPatel	895
3	Biri	Ben	BBiri	1100
4	Newman	chad	cNewman	750
5	Ropebur	Audrey	ARopebur	1550

4 rows returned in 0.01 seconds [Download](#)

6. Empty the fourth row of the emp table.

Autocommit Rows 10 Save Run

```
UPDATE MY_EMPLOYEE SET Last_name = NULL, First_name = NULL, Userid = NULL,  
Salary = NULL WHERE ID = 4;
```

Results Explain Describe Saved SQL History

1 row(s) updated.

Autocommit Rows 10 Save Run

```
SELECT * FROM MY_EMPLOYEE;
```

Results Explain Describe Saved SQL History History

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	RPatel	895
3	Biri	Ben	BBiri	1100
4	-	-	-	-
5	Ropebur	Audrey	ARopebur	1550

4 rows returned in 0.00 seconds [Download](#)

7. Make the data additions permanent.

Autocommit Rows 10 Save Run

```
COMMIT;
```

Results Explain Describe Saved SQL History

Statement processed.

8. Change the last name of employee 3 to Drexler.

Autocommit    Rows

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

**Results** Explain Describe Saved SQL History

1 row(s) updated.

Autocommit    Rows

```
SELECT * FROM MY_EMPLOYEE;
```

**Results** Explain Describe Saved SQL History

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	RPatel	895
3	Drexler	Ben	BBiri	1100
4	-	-	-	-
5	Ropebur	Audrey	ARopebur	1550

4 rows returned in 0.00 seconds [Download](#)

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

Autocommit    Rows

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

**Results** Explain Describe Saved SQL History

1 row(s) updated.

Autocommit    Rows    Save  Run

```
SELECT * FROM MY_EMPLOYEE;
```

**Results** Explain Describe Saved SQL History

ID	LAST_NAME	FIRST_NAME	USERID	SALARY
1	Patel	Ralph	RPatel	1000
3	Drexler	Ben	BBiri	1100
4	-	-	-	-
5	Ropebur	Audrey	ARopebur	1550

4 rows returned in 0.01 seconds [Download](#)