

EXERCISE-2

MANIPULATING DATA

OBJECTIVE

After, the completion of this exercise the students will be able to do the following

- Describe each DML statement
- Insert rows into tables
- Update rows into table
- Delete rows from table
- Control Transactions

A DML statement is executed when you:

- Add new rows to a table
- Modify existing rows
- Removing existing rows

A transaction consists of a collection of DML statements that form a logical unit of work.

To Add a New Row

INSERT Statement

Syntax

INSERT INTO table_name VALUES (column1 values, column2 values, ..., column values);

Example:

INSERT INTO department (70, 'Public relations', 100,1700);

Inserting rows with null values

Implicit Method: (Omit the column)

INSERT INTO department VALUES (30,'purchasing');

Explicit Method: (Specify NULL keyword)

INSERT INTO department VALUES (100,'finance', NULL, NULL);

Inserting Special Values

Example:

Using SYSDATE

INSERT INTO employees VALUES (113,'louis', 'popp', 'lpopp','5151244567',SYSDATE, 'ac_account', 6900, NULL, 205, 100);

Inserting Specific Date Values

Example:

```
INSERT INTO employees VALUES (114,'den','raphealy','drapheal','5151274561',
TO_DATE('feb 3,1999','mon, dd ,yyyy'), 'ac_account', 11000,100,30);
```

To Insert Multiple Rows

& is the placeholder for the variable value

Example:

```
INSERT INTO department VALUES (&dept_id, &dept_name, &location);
```

Copying Rows from another table

- Using Subquery

Example:

```
INSER INTO sales_reps(id, name, salary, commission_pct)
SELECT employee_id, Last_name, salary, commission_pct
FROM employees
WHERE job_id LIKE '%REP');
```

CHANGING DATA IN A TABLE

UPDATE Statement

Syntax1: (to update specific rows)

```
UPDATE table_name SET column=value WHERE condition;
```

Syntax 2: (To updae all rows)

```
UPDATE table_name SET column=value;
```

Updating columns with a subquery

```
UPDATE employees
SET job_id= (SELECT job_id
FROM employees
WHERE employee_id=205)
WHERE employee_id=114;
```

REMOVING A ROW FROM A TABLE

DELETE STATEMENT

Syntax

```
DELETE FROM table_name WHERE conditions;
```

Example:

```
DELETE FROM department WHERE dept_name='finance';
```

Find the Solution for the following:

1. Create MY_EMPLOYEE table with the following structure

NAME	NULL?	TYPE
ID	Not null	Number(4)
Last_name		Varchar(25)
First_name		Varchar(25)
Userid		Varchar(25)
Salary		Number(9,2)

2. Add the first and second rows data to MY_EMPLOYEE table from the following sample data.

ID	Last_name	First_name	Userid	salary
1	Patel	Ralph	rpatel	895
2	Dance	Betty	bdance	860
3	Biri	Ben	bbiri	1100
4	Newman	Chad	Cnewman	750
5	Ropebur	Audrey	Aropebur	1550

3. Display the table with values.

```
select * from MY_EMPLOYEE;
+----+-----+-----+-----+-----+
| ID | Last_name | First_name | Userid | salary |
+----+-----+-----+-----+-----+
| 1  | Patel     | Ralph      | rpatel | 895.00 |
| 2  | Dance    | Betty      | bdance | 860.00 |
| 3  | Biri     | Ben        | bbiri  | 1100.00|
| 4  | Newman   | Chad       | Cnewman| 750.00 |
| 5  | Ropebur  | Audrey    | Aropebur| 1550.00|
+----+-----+-----+-----+-----+
```

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.

```
INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary)
VALUES (4, 'Newman', 'chad', 'Cnewman', 750);
INSERT INTO MY_EMPLOYEE (ID, Last_name, First_name, Userid, Salary)
VALUES (5, 'Ropebur', 'Audrey', 'Aropebur', 1550);
```

5. Make the data additions permanent.

```
BEGIN TRANSACTION;
INSERT INTO MY_EMPLOYEE VALUES (3, "Biri", "Bet", "bbiri", 1100);
COMMIT;
```

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

```
update MY_EMPLOYEE set last_name = "Drexler"
```

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

update MY_EMPLOYEE set salary=1000 where
salary < 900;

8. Delete Betty dances from MY_EMPLOYEE table.

delete from MY_EMPLOYEE where ~~last~~-name = "dances";

9. Empty the fourth row of the emp table.

delete from MY_EMPLOYEE WHERE ID = 4;

Evaluation Procedure	Marks awarded
Query(5)	5
Execution (5)	5
Viva(5)	5
Total (15)	15
Faculty Signature	BPL