

# **Data Manipulation Language**

# **Instructor Notes:**

8.1: Concept of Data Manipulation Language



# Data Manipulation Language

Data Manipulation Language (DML) is used to perform the following routines on database information:

- Retrieve
- Insert
- Modify

DML changes data in an object. If you insert a row into a table, that is DML.

All DML statements change data, and must be committed before the change becomes permanent.

8.1: Addition of Data into Tables

### INSERT

INSERT command:

- INSERT is a DML command. It is used to add rows to a table.
- In the simplest form of the command, the values for different columns in the row to be inserted have to be specified.
- Alternatively, the rows can be generated from some other tables by using a SQL query language command.

### **Addition of Data into Tables:**

### Requisites for using INSERT command:

- If values are specified for all columns in the order specified at creation, then col\_names could be omitted.
- Values should match "data type" of the respective columns.
- Number of values should match the number of column names mentioned.
- All columns declared as NOT NULL should be supplied with a value.
- Character strings should be enclosed in quotes.
- Date values should be enclosed in quotes.
- Values will insert one row at a time.
- Query will insert all the rows returned by the query.
- The table\_name can be a "table" or a "view". If table\_name is a "view", then the
  following restrictions apply:
  - 1. The "view" cannot have a GROUP BY, CONNECT BY, START WITH, DISTINCT, UNION, INTERSECT, or MINUS clause or a join.
  - 2. If the "view" has WITH CHECK OPTION clause, then a row, which will not be returned by the view, cannot be inserted.

```
Inserting Rows into a Table

Inserting by specifying values:

Example: To insert a new record in the DEPT table

INSERT INTO

table_name[(col_name1,col_name2,...)]

{VALUES (value1,value2,....) | query};

INSERT INTO Department_master
VALUES (10, 'Computer Science');
```

### **Inserting Rows into a Table:**

### Example:

Inserting a row in EMP table giving all values.

```
INSERT INTO student_master VALUES(1001,'Amit',10,'11-Jan-80','Chennai');
```

- ➤ 10 is a dept number which exists in DEPARTMENT\_MASTER table
- Inserting a row in STAFF\_MASTER table giving some values.

```
INSERT INTO staff_master (staff_code,staff_name,design_code,dept_code) VALUES(100001,'Arvind',102,30);
```

This row will be created if all the constraints like NOT NULL are satisfied.

8.1: Addition of Data into Tables

### Inserting Rows into a Table

Inserting rows in a table from another table using Subquery:

Example: The example given below assumes that a new\_emp\_table exists. You can use a subquery to insert rows from another table.

INSERT INTO new\_staff\_table SELECT \* FROM staff\_master WHERE staff\_master.hiredate > '01-jan-82';

8.1: Addition of Data into Tables

### Inserting Rows into a Table

Inserting by using "substitution variables":

Example: In the example given below, when the command is run, values are prompted every time.

INSERT INTO department\_master
VALUES (&dept\_code, '&dept\_name');
Enter a value for dept\_code : 20
Enter a value for dept\_name : Electricals

### **Inserting Rows into a Table:**

### Inserting by using "substitution variables":

- The problem with the INSERT statement is that it adds only "one row" to the table.
- However, by using "substitution variables" the speed of data input can be increased.
- Whenever a "substitution variable" is placed in a "value" field, the user will be prompted to enter the "actual value" when the command is executed.

8.2: Deletion of Data from Tables

### DELETE

The DELETE command is used to delete one or more rows from a table.

• The DELETE command removes all rows identified by the WHERE clause.

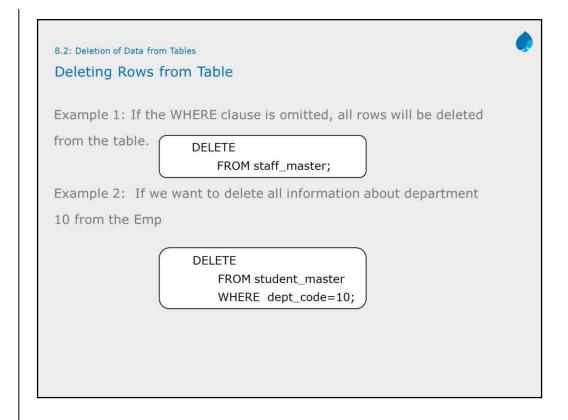
DELETE [FROM] {table\_name | alias }
 [WHERE condition];

### **Deletion of Data from Tables**

- The table name can be a "table" or a "view".
- The DELETE command is used to delete one or more rows from a table.
- The DELETE statement removes all rows identified by the WHERE clause.
  - > This is another DML, which means we can rollback the deleted data, and that to make our changes permanent.
- If WHERE clause is omitted, all rows from the table are removed. Else all rows which satisfy the condition are removed.
- FROM clause can be omitted without affecting the statement.

# **Data Manipulation Language**

# **Instructor Notes:**



### **Deletion of Data from Tables**

Example 3:

DELETE staff\_master WHERE staff\_name = 'Anil';

8.3: Modifying / Updating existing Data in a Table

### **UPDATE**

Use the UPDATE command to change single rows, groups of rows, or all rows in a table.

• In all data modification statements, you can change the data in only "one table at a time".

```
UPDATE table_name
SET col_name = value|
    col_name =
SELECT_statement_returning_single_value|
    (col_name,...) = SELECT_statement
[WHERE condition];
```

### **Modifying / Updating existing Data in a Table:**

- The table name can be a "table" or a "view".
- The "value" can be a value, an expression, or a query, which returns a single value.
- The UPDATE command provides automatic navigation to the data.
- Note: If the WHERE clause is omitted, all rows in the table will be updated by a
  value that is currently specified for the field. Else only those rows which satisfy the
  condition will be updated.

8.3: Modifying / Updating existing Data in a Table

# Updating Rows from Table

Example 1: To UPDATE the column "dname" of a row, where deptno is 10, give the following command:

UPDATE department\_master
SET dept\_name= 'Information Technology'
WHERE dept\_code=10;

8.3: Modifying / Updating existing Data in a Table

# Updating Rows from Table

Example 2: To UPDATE the subject marks details of a particular student, give the following command:

8.3: Modifying / Updating existing Data in a Table

# Using a Subquery to do an Update

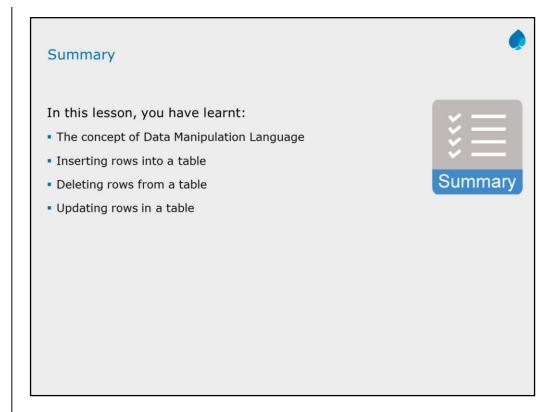
For making salary of "Anil" equal to that of staff member 100006, use the following command:

UPDATE staff\_master

SET staff\_sal = (SELECT staff\_sal FROM staff\_master

WHERE staff\_code = 100006)

WHERE staff\_name = 'Anil';



Answers to Review Questions

Question 1: True

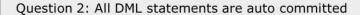
Question 2: False

Question 3: Option 2

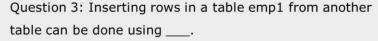
### **Review - Questions**

Question 1: Both TRUNCATE statement and DELETE without condition removes the entire date from a table

True/False



True/False



- Option 1: insert into emp1(t1) as select empno from emp
- Option 2: insert into emp1(t1) select empno from emp
- Option 3: insert into emp1(t1) as select \* from emp

