

# 8

## Manipulating Data

## Objectives

After completing this lesson, you should be able to do the following:

- Describe each data manipulation language (DML) statement
- Insert rows into a table
- Update rows in a table
- Delete rows from a table
- Control transactions

## INSERT Statement Syntax

- Add new rows to a table by using the INSERT statement:

```
INSERT INTO table [(column [, column...])]  
VALUES      (value [, value...]);
```

- With this syntax, only one row is inserted at a time.

## Inserting Rows with Null Values

- **Implicit method: Omit the column from the column list.**

```
INSERT INTO departments (department_id,  
                        department_name )  
VALUES (30, 'Purchasing');  
1 row created.
```

- **Explicit method: Specify the NULL keyword in the VALUES clause.**

```
INSERT INTO departments  
VALUES (100, 'Finance', NULL, NULL);  
1 row created.
```

## UPDATE Statement Syntax

- **Modify existing rows with the UPDATE statement:**

```
UPDATE      table
SET         column = value [, column = value, ...]
[WHERE     condition];
```

- **Update more than one row at a time (if required).**

## Updating Rows Based on Another Table

Use subqueries in UPDATE statements to update rows in a table based on values from another table:

```
UPDATE copy_emp
SET    department_id = (SELECT department_id
                        FROM employees
                        WHERE employee_id = 100)
WHERE  job_id        = (SELECT job_id
                        FROM employees
                        WHERE employee_id = 200);
1 row updated.
```

## DELETE Statement

You can remove existing rows from a table by using the DELETE statement:

```
DELETE [FROM] table  
[WHERE condition];
```

## Deleting Rows Based on Another Table

Use subqueries in DELETE statements to remove rows from a table based on values from another table:

```
DELETE FROM employees
WHERE department_id =
    (SELECT department_id
     FROM departments
     WHERE department_name
           LIKE '%Public%');
```

1 row deleted.



## TRUNCATE Statement

- Removes all rows from a table, leaving the table empty and the table structure intact
- Is a data definition language (DDL) statement rather than a DML statement; cannot easily be undone
- Syntax:

```
TRUNCATE TABLE table_name;
```

## Database Transactions

A database transaction consists of one of the following:

- **DML statements that constitute one consistent change to the data**
- **One DDL statement**
- **One data control language (DCL) statement**

# Controlling Transactions

