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

SQL Statements

SELECT INSERT Data manipulation language (DML) UPDATE DELETE MERGE CREATE ALTER DROP Data definition language (DDL) RENAME TRUNCATE COMMENT GRANT Data control language (DCL) REVOKE COMMIT Transaction control ROLLBACK SAVEPOINT

Lesson Agenda

- Adding new rows in a table
 - INSERT statement
- Changing data in a table
 - UPDATE statement
- Removing rows from a table:
 - DELETE statement
 - TRUNCATE statement
- Database transactions control using COMMIT, ROLLBACK, and SAVEPOINT
- Read consistency
- FOR UPDATE clause in a SELECT statement

Data Manipulation Language

- A DML statement is executed when you:
 - Add new rows to a table
 - Modify existing rows in a table
 - Remove existing rows from a table
- A transaction consists of a collection of DML statements that form a logical unit of work.

Adding a New Row to a Table

New 70 Public Relations 100 1700 **DEPARTMENTS** row DEPARTMENT_ID 2 DEPARTMENT_NAME 2 MANAGER_ID 2 LOCATION_ID 1 10 Administration 200 1700 Insert new row 2 20 Marketing 201 1800 into the 3 50 Shipping 124 1500 4 60 IT 103 1400 DEPARTMENTS table. 5 80 Sales 149 2500 6 90 Executive 1700 100 110 Accounting 1700 7 205 190 Contracting 1700 8 (null) DEPARTMENT_ID DEPARTMENT_NAME MANAGER_ID 3 LOCATION_ID 1700 1 70 Public Relations 100 2 10 Administration 200 1700 1800 3 20 Marketing 201 4 50 Shipping 124 1500 5 60 IT 103 1400 6 80 Sales 149 2500 7 90 Executive 100 1700 8 110 Accounting 205 1700

1700

(null)

190 Contracting

9

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 New Rows

- Insert a new row containing values for each column.
- List values in the default order of the columns in the table.
- Optionally, list the columns in the INSERT clause.

Enclose character and date values within single quotation marks.

Inserting Rows with Null Values

Implicit method: Omit the column from the column list.

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

```
INSERT INTO departments
VALUES (100, 'Finance', NULL, NULL);

1 rows inserted
```

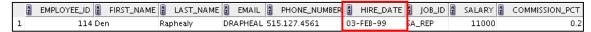
Inserting Special Values

The SYSDATE function records the current date and time.

Inserting Specific Date and Time Values

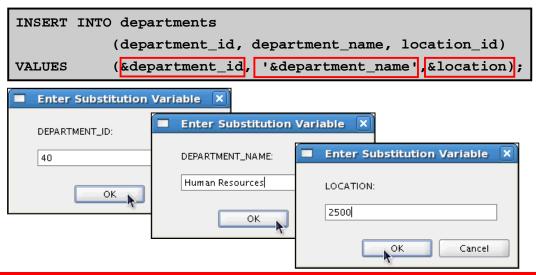
Add a new employee.

Verify your addition.



Creating a Script

- Use the & substitution in a SQL statement to prompt for values.
- & is a placeholder for the variable value.



Copying Rows from Another Table



Write your INSERT statement with a subquery:

```
INSERT INTO sales_reps(id, name, salary, commission_pct)

SELECT employee_id, last_name, salary, commission_pct
FROM employees
WHERE job_id LIKE '%REP%';

4 rows inserted
```

- Do not use the VALUES clause.
- Match the number of columns in the INSERT clause to those in the subquery.
- Inserts all the rows returned by the subquery in the table, sales_reps.

Lesson Agenda

- Adding new rows in a table
 - INSERT statement
- Changing data in a table
 - UPDATE statement
- Removing rows from a table:
 - DELETE statement
 - TRUNCATE statement
- Database transactions control using COMMIT, ROLLBACK, and SAVEPOINT
- Read consistency
- FOR UPDATE clause in a SELECT statement



Changing Data in a Table

EMPLOYEES

A	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	SALARY	MANAGER_ID	② COMMISSION_PCT	DEPARTMENT_ID
	100	Steven	King	24000	(null)	(null)	90
	101	Neena	Kochhar	17000	100	(null)	90
	102	Lex	De Haan	17000	100	(null)	90
	103	Alexander	Hunold	9000	102	(null)	60
	104	Bruce	Ernst	6000	103	(null)	60
	107	Diana	Lorentz	4200	103	(null)	60
	124	Kevin	Mourgos	5800	100	(null)	50

Update rows in the EMPLOYEES table:

A	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	2 SALARY	MANAGER_ID	2 COMMISSION_PCT	DEPARTMENT_ID
	100	Steven	King	24000	(null)	(null)	90
	101	Neena	Kochhar	17000	100	(null)	90
	102	Lex	De Haan	17000	100	(null)	90
	103	Alexander	Hunold	9000	102	(null)	80
	104	Bruce	Ernst	6000	103	(null)	80
	107	Diana	Lorentz	4200	103	(null)	80
	124	Kevin	Mourgos	5800	100	(null)	50

UPDATE Statement Syntax

Modify existing values in a table 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 in a Table

 Values for a specific row or rows are modified if you specify the WHERE clause:

```
UPDATE employees
SET department_id = 50
WHERE employee_id = 113;
1 rows updated
```

 Values for all the rows in the table are modified if you omit the WHERE clause:

```
UPDATE copy_emp
SET department_id = 110;
22 rows updated
```

Specify SET column_name = NULL to update a column value to NULL.



Updating Two Columns with a Subquery

Update employee 113's job and salary to match those of employee 205.

```
UPDATE
         employees
         job id
                   (SELECT
                             job_id
SET
                     FROM
                           employees
                           employee_id = 205),
                     WHERE
         salary
                    SELECT salary
                     FROM
                           employees
                     WHERE employee_id = 205)
                            113;
WHERE
         employee id
l rows updated
```

Updating Rows Based on Another Table

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