ORACLE® ACADEMY

# Database Programming with SQL

**3-1**
**Logical Comparisons and Precedence Rules**

# Objectives

This lesson covers the following objectives:

- Evaluate logical comparisons to restrict the rows returned based on two or more conditions

- Apply the rules of precedence to determine the order in which expressions are evaluated and calculated

# Purpose

- Not too many things in life depend on just one condition.

- For instance, if you want to go to college, you probably need good grades and the tuition money to pay for it.

- If you have extra money, you could either save it or spend it.

- If you want to go to a movie, you may not want to go this weekend and you may not want to sit in the first 10 rows of the theater.

# Purpose

- In SQL, it is often desirable to be able to restrict the rows returned by a query based on two or more conditions.

- As the manager of a fast food business, you may need to know the names of your staff who are either cooks or order takers.

- You don't need or want the entire staff list, you just want a subset of it.

- Conditional operators such as AND, NOT, and OR make these types of requests easy to do.

# Logical Conditions

- Logical conditions combine the result of two component conditions to produce a single result based on them.

- For example, to attend a rock concert, you need to buy a ticket  AND have transportation to get there.

- If both conditions are met, you can go to the concert.

- What if you can't get transportation, can you go?

# Logical Conditions

- Another logical condition combines two component conditions with OR.

- All employees will receive a raise either by having a perfect attendance record OR by meeting their monthly sales quota.

- If an employee meets either of these two conditions, the employee gets a raise.

# Logical Operators

- A logical operator combines the results of two or more conditions to produce a single result.

- A result is returned ONLY IF the overall result of the condition is true.

- AND  --  Returns TRUE if both conditions are true.

- OR  --  Returns TRUE if either condition is true.

- NOT  --  Returns TRUE if the condition is false.

# AND Operator

- In the query below, the results returned will be rows that satisfy BOTH conditions specified in the WHERE clause.

```
SELECT last_name, department_id, salary
FROM employees
WHERE department_id > 50 AND salary > 12000;
```

| LAST_NAME | DEPARTMENT_ID | SALARY |
|-----------|---------------|--------|
| King | 90 | 24000 |
| Kochhar | 90 | 17000 |
| De Haan | 90 | 17000 |

ORACLE® ACADEMY

# AND Operator

- Another example of using AND in the where clause.

```
SELECT last_name, hire_date, job_id
FROM employees
WHERE hire_date > '01/jan/1998' AND job_id LIKE 'SA%';
```

| LAST_NAME | HIRE_DATE | JOB_ID |
|-----------|-----------|--------|
| Zlotkey | 29/Jan/2000 | SA_MAN |
| Taylor | 24/Mar/1998 | SA_REP |
| Grant | 24/May/1999 | SA_REP |

# OR Operator

- If the WHERE clause uses the OR condition, the results returned from a query will be rows that satisfy either one of the OR conditions.

- In other words, all rows returned have a location_id of 2500 OR they have a manager_id equal to 124.

```
SELECT department_name, manager_id, location_id
FROM departments
WHERE location_id = 2500 OR manager_id=124;
```

| DEPARTMENT_NAME | MANAGER_ID | LOCATION_ID |
|---|---|---|
| Shipping | 124 | 1500 |
| Sales | 149 | 2500 |

# NOT Operator

- The NOT operator will return rows that do NOT satisfy the condition in the WHERE clause.

```
SELECT department_name, location_id
FROM departments
WHERE location_id NOT IN (1700,1800);
```

| DEPARTMENT_NAME | LOCATION_ID |
|-----------------|-------------|
| Shipping | 1500 |
| IT | 1400 |
| Sales | 2500 |

# Rules of Precedence or What Happens First?

- Consider the following SELECT statement.

- In what order are the expressions evaluated and calculated?

```
SELECT last_name||' '||salary*1.05
As "Employee Raise"
FROM employees
WHERE department_id IN(50,80)
AND first_name LIKE 'C%'
OR last_name LIKE '%s%';
```

- Luckily, when things get this complicated, SQL has a few basic rules that are easy to follow.

# Rules of Precedence or What Happens First?

- Notice that the AND operator is evaluated before the OR operator.

- This means that for the example on the previous slide, if either of the conditions in the AND statement are not met, then the OR operator is used to select the rows.

- This is an important concept to remember.

| ORDER | OPERATORS |
|-------|-----------|
| 1 | Arithmetic + - * / |
| 2 | Concatenation \|\| |
| 3 | Comparison <, <=, >, >=, <> |
| 4 | IS (NOT) NULL, LIKE, (NOT) IN |
| 5 | (NOT) BETWEEN |
| 6 | NOT |
| 7 | AND |
| 8 | OR |

# Rules of Precedence or What Happens First?

- First, the AND condition is evaluated, so all employees working in dept 80 or 50, AND who have a first name starting with "C" are returned.

- The OR clause is then evaluated and returns employees whose last name contains "s".

```
SELECT last_name||' '||salary*1.05
  As "Employee Raise",
  department_id, first_name
FROM employees
WHERE department_id IN(50,80)
AND first_name LIKE 'C%'
OR last_name LIKE '%s%';
```

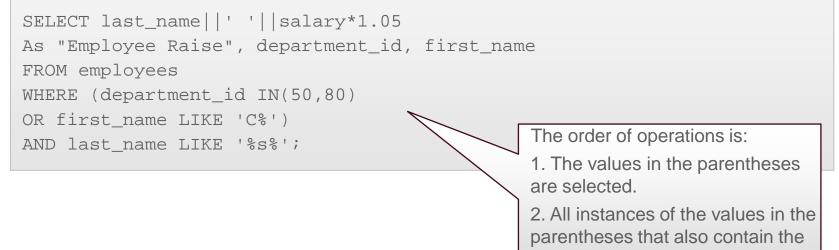| Employee Raise | DEPARTMENT_ID | FIRST_NAME |
|---|---|---|
| Higgins 12600 | 110 | Shelley |
| Mourgos 6090 | 50 | Kevin |
| Rajs 3675 | 50 | Trenna |
| Davies 3255 | 50 | Curtis |
| Matos 2730 | 50 | Randall |
| Vargas 2625 | 50 | Peter |
| Ernst 6300 | 60 | Bruce |
| Hartstein 13650 | 20 | Michael |

# Rules of Precedence or What Happens First?

- In this example, the order of the OR and AND have been reversed from the previous slide.
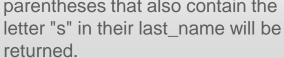
```
SELECT last_name||' '||salary*1.05
As "Employee Raise", department_id, first_name
FROM employees
WHERE department_id IN(50,80)
OR first_name LIKE 'C%'
AND last_name LIKE '%s%';
```

The order of operations is:

1. first_name starts with a "C" AND last_name contains an "s". Both these conditions must be met to be returned.

2. Any instance of employees in department 50 and 80 will be returned.

# Rules of Precedence or What Happens First?

- Adding parenthesis changes the way the Where clause is evaluated, and the rows returned.

```
SELECT last_name||' '||salary*1.05
As "Employee Raise", department_id, first_name
FROM employees
WHERE (department_id IN(50,80)
OR first_name LIKE 'C%')
AND last_name LIKE '%s%';
```

The order of operations is:

1. The values in the parentheses are selected.

2. All instances of the values in the parentheses that also contain the letter "s" in their last_name will be returned.

# Terminology

Key terms used in this lesson included:

- AND

- OR

- NOT

- Precedence Rules

# Summary

In this lesson, you should have learned how to:

- Evaluate logical comparisons to restrict the rows returned based on two or more conditions

- Apply the rules of precedence to determine the order in which expressions are evaluated and calculated

ORACLE® **ACADEMY**