Database Programming with SQL
7-1: Oracle Equijoin and Cartesian Product
Practice Activities

Objectives

- Name the Oracle proprietary joins and their ANSI/ISO SQL: 1999 counterparts
- Describe the purpose of join conditions
- Construct and execute a SELECT statement that results in a Cartesian product
- Construct and execute SELECT statements to access data from more than one table using an equijoin
- Construct and execute SELECT statements that add search conditions using the AND operator
- Apply the rule for using column aliases in a join statement

Vocabulary

Identify the vocabulary word for each definition below.

Cartesian Product	Results from an invalid or omitted join condition; all combinations of rows are displayed
Equijoin	Values in a column in one table are equal to a value in another table; also called an inner join or simple join
Oracle Proprietary Join	Connection command exclusive to a specific company
Table Alias	Gives a table another name to simplify queries and improve performance
Join	Display data from two or more related tables

Try It / Solve It

1. Create a Cartesian product that displays the columns in the d_play_list_items and the d_track_listings in the DJs on Demand database.

```
SELECT *
FROM d_play_list_items, d_track_listings;
```

2. Correct the Cartesian product produced in question 1 by creating an equijoin using a common column.

```
SELECT *
FROM d_play_list_items p, d_track_listings t
WHERE p.song_id = t.song_id;
```

3. Write a query to display the title, type, description, and artist from the DJs on Demand database.

```
SELECT s.title, t.description, s.artist
FROM d_songs s
JOIN d_types t ON s.type_code = t.code;
```

4. Rewrite the query in question 3 to select only those titles with an ID of 47 or 48.

```
SELECT s.title, t.description, s.artist
FROM d_songs s
JOIN d_types t ON s.type_code = t.code
WHERE s.id IN (47, 48);
```

5. Write a query that extracts information from three tables in the DJs on Demand database, the d_clients table, the d_events table, and the d_job_assignments table.

```
SELECT c.first_name, c.last_name, e.name AS event_name, j.status, j.job_date
FROM d_clients c
JOIN d_events e ON c.client_number = e.client_number
JOIN d_job_assignments j ON e.id = j.event_id;
```

6. Create and execute an equijoin between DJs on Demand tables d_track_listings and d_cds. Return the song_id and the title only.

```
SELECT t.song_id, c.title
FROM d_track_listings t
JOIN d_cds c ON t.cd_number = c.cd_number;
```

SONG_ID	TITLE
47	Party Music for All Occasions
49	Party Music for All Occasions
45	Back to the Shire
46	Songs from My Childhood
48	Here Comes the Bride

7. Mark T for the statements that are true and F for the statements that are false.

F a. A join is a type of query that gets data from more than one table based on columns with the same name.

T b. To join tables using an equijoin, there must be a common column in both tables and that column is usually a primary key in one of the tables.

T c. A Cartesian product occurs because the query does not specify a WHERE clause.

F d. Table aliases are required to create a join condition.

T e. If a table alias is used for a table name in the FROM clause, it must be substituted for the table name throughout the SELECT statement.

F f. Table alias must be only one character in length.

T g. A simple join or inner join is the same as an equijoin.

8. What advantage does being able to combine data from multiple tables have for a business?

Combining data from multiple tables allows a business to gain more comprehensive insights by integrating different types of information. For example, joining customer data with sales data helps businesses analyze purchasing patterns, identify trends, and make informed decisions. This capability also enhances reporting and provides a holistic view of operations, customers, and performance.

Database Programming with SQL
7-2: Oracle Nonequijoins and Outer Joins
Practice Activities

Objectives

- Construct and execute a SELECT statement to access data from more than one table using a nonequijoin
- Create and execute a SELECT statement to access data from more than one table using an Oracle outer join

Try It / Solve It

1. Create a join based on the cost of the event between the DJs on Demand tables D_EVENTS and D_PACKAGES. Show the name of the event and the code for each event.

```
SELECT e.event_name, p.package_code
FROM D_EVENTS e
JOIN D_PACKAGES p ON e.cost = p.cost;
```

2. Using the Oracle database, create a query that returns the employee last name, salary, and job-grade level based on the salary. Select the salary between the lowest and highest salaries.

```
SELECT e.last_name, e.salary, j.grade_level
FROM EMPLOYEES e
JOIN JOB_GRADES j ON e.salary BETWEEN j.lowest_sal AND j.highest_sal;
```

3. What condition requires the creation of a nonequijoin?

```
A nonequijoin is needed when you compare columns that aren't directly equal but involve a range
(e.g., using BETWEEN, <, or > instead of =). An example would be comparing salary ranges between
two tables.
```

4. Rewrite the following nonequijoin statement using the logical condition operators (AND, OR, NOT): WHERE a.ranking BETWEEN g.lowest_rank AND g.highest_rank

```
WHERE a.ranking >= g.lowest_rank AND a.ranking <= g.highest_rank;
```

5. How do you know when to use a table alias and when not to use a table alias?

```
Use table aliases when you are dealing with multiple tables or need to disambiguate columns from
different tables that might have the same name.
```

6. What kind of join would you use if you wanted to find data between a range of numbers?

```
A nonequijoin is typically used when you want to join based on a range of numbers (like salary
ranges or age groups).
```

7. You need to produce a report for Global Fast Foods showing customers and orders. A customer must be included on the report even if the customer has had no orders.

```
SELECT c.first_name, c.last_name, o.order_number
FROM F_CUSTOMERS c
LEFT JOIN F_ORDERS o ON c.id = o.cust_id;
```

8. Create a query of the Oracle database that shows employee last names, department IDs, and department names. Include all employees even if they are not assigned to a department.

```
SELECT e.last_name, e.department_id, d.department_name
FROM EMPLOYEES e
LEFT JOIN DEPARTMENTS d ON e.department_id = d.department_id;
```

9. Modify the query in problem 8 to return all the department IDs even if no employees are assigned to them.

```
SELECT e.last_name, d.department_id, d.department_name
FROM DEPARTMENTS d
LEFT JOIN EMPLOYEES e ON e.department_id = d.department_id;
```

10. There are one or more errors in each of the following statements. Describe the errors and correct them.

    a. WHERE e.department_id(+) = d.department_id (+);

    ```
    WHERE e.department_id = d.department_id(+);
    ```

    b. SELECT e.employee id, e. last name, d. location id
    FROM employees, departments
    WHERE e.department_id = d.department_id(+);

    ```
    SELECT e.employee_id, e.last_name, d.location_id
    FROM EMPLOYEES e, DEPARTMENTS d
    WHERE e.department_id = d.department_id(+);
    ```

11. Create a query that will show all CD titles and song IDs in the DJs on Demand database even if there is no CD number in the track-listings table.

```
SELECT c.title, t.song_id
FROM D_CDS c
LEFT JOIN D_TRACK_LISTINGS t ON c.cd_number = t.cd_number;
```

12. How many times has someone asked you: “What do you want to be when you grow up?” For most of us, the first thing that comes to mind is something like business manager, engineer, teacher, game designer, doctor, scientist, computer programmer, or accountant -- all pretty much traditional career choices. Have you ever thought about working in an odd job or nontraditional career? There are people who are professional shoppers for busy executives, directors of zoos, recipe designers, insecticide chemists, golf-course designers, and turf managers. Picture yourself in a dream job or nontraditional career doing something that you think would be interesting, life fulfilling, and profitable.

    Use Internet resources to explore your idea. Write a brief description of the job to share with the class.

    A Video Game Designer is responsible for creating the concept, mechanics, and overall experience of a video game. They develop the storylines, character development, levels, and gameplay rules that shape how players interact with the game. This role involves both creative and technical skills, as designers must collaborate with artists, programmers, and sound engineers to bring their vision to life. They work on designing the game's narrative, user interface, and progression systems, often using specialized software to create prototypes and iterate on ideas. A successful video game designer balances creativity with user engagement, ensuring the game is fun, challenging, and immersive for players.