

4

Visualización de Datos a partir de Varias Tablas

Objetivos

Al completar esta lección, debería ser capaz de hacer lo siguiente:

- **Escribir sentencias SELECT para acceder a los datos desde una o más tablas usando equality y non-equality joins (combinaciones por igualdad y por desigualdad).**
- **Visualizar datos que no se cumplirían normalmente con una condición de join usando outer joins (uniones externas).**
- **Combinar (Join) una tabla consigo misma.**

Obtención de Datos de Múltiples Tablas

EMP

EMPNO	ENAME	...	DEPTNO
7839	KING	...	10
7698	BLAKE	...	30
...			
7934	MILLER	...	10

DEPT

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON



EMPNO	DEPTNO	LOC
7839	10	NEW YORK
7698	30	CHICAGO
7782	10	NEW YORK
7566	20	DALLAS
7654	30	CHICAGO
7499	30	CHICAGO
...		
14 rows selected.		

¿Qué es un JOIN?

Use un join para consultar datos de más de una tabla.

```
SELECT      table.column, table.column  
FROM        table1, table2  
WHERE       table1.column1 = table2.column2;
```

- Escriba la condición de join en la cláusula WHERE.
- Preceda el nombre de la columna con el de la tabla, cuando el mismo nombre de columna, aparezca en más de una tabla.

Producto Cartesiano

- **Se establece un producto cartesiano cuando:**
 - Se omite una condición de join.
 - Se define una condición de join inválida.
 - Se combinan todas las filas de la primer tabla con todas las filas de la segunda.
- **Para evitar un producto cartesiano, se debe incluir siempre una condición de join válida en la cláusula WHERE.**

Generación de un Producto Cartesiano

EMP (14 registros)

EMPNO	ENAME	...	DEPTNO
7839	KING	...	10
7698	BLAKE	...	30
...			
7934	MILLER	...	10

DEPT (4 registros)

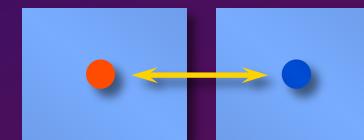
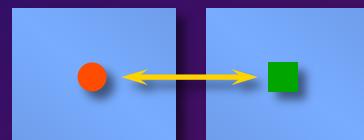
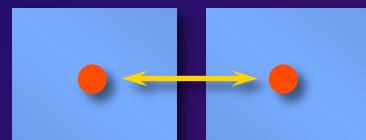
DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

“Producto
Cartesiano: →
 $14 \times 4 = 56$ rows”

ENAME	DNAME
KING	ACCOUNTING
BLAKE	ACCOUNTING
...	
KING	RESEARCH
BLAKE	RESEARCH
...	
56 rows selected.	

Tipos de Join

Equijoin Non-equijoin Outer join Self join



¿Qué es un Equijoin?

EMP

EMPNO	ENAME	DEPTNO
7839	KING	10
7698	BLAKE	30
7782	CLARK	10
7566	JONES	20
7654	MARTIN	30
7499	ALLEN	30
7844	TURNER	30
7900	JAMES	30
7521	WARD	30
7902	FORD	20
7369	SMITH	20
...		
14 rows selected.		

DEPT

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
30	SALES	CHICAGO
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
20	RESEARCH	DALLAS
20	RESEARCH	DALLAS
...		
14 rows selected.		

Primary key

Foreign key

Recuperación de Registros con Equijoins

```
SQL> SELECT      emp.empno, emp.ename, emp.deptno,  
2                  dept.deptno, dept.loc  
3  FROM        emp, dept  
4 WHERE      emp.deptno=dept.deptno;
```

EMPNO	ENAME	DEPTNO	DEPTNO	LOC
7839	KING	10	10	NEW YORK
7698	BLAKE	30	30	CHICAGO
7782	CLARK	10	10	NEW YORK
7566	JONES	20	20	DALLAS

...

14 rows selected.

Calificación de Nombres de Columna Ambiguos

- **Usar como prefijos los nombres de tablas para calificar los nombres de columnas que están en varias tablas.**
- **Mejorar el rendimiento usando como prefijos los nombres de tablas.**
- **Distinguir columnas que tienen nombres idénticos pero pertenecen a diferentes tablas usando alias de columnas.**

Condiciones de Búsqueda Adicionales Usando el Operador AND: Ejemplo

EMP

EMPNO	ENAME	DEPTNO
7839	KING	10
7698	BLAKE	30
7782	CLARK	10
7566	JONES	20
7654	MARTIN	30
7499	ALLEN	30
7844	TURNER	30
7900	JAMES	30
7521	WARD	30
7902	FORD	20
7369	SMITH	20
...		
14 rows selected.		

DEPT

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
30	SALES	CHICAGO
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
20	RESEARCH	DALLAS
20	RESEARCH	DALLAS
...		
14 rows selected.		

Uso de Alias de Tabla

Ayudan a simplificar las consultas

```
SQL> SELECT emp.empno, emp.ename, emp.deptno,  
2          dept.deptno, dept.loc  
3  FROM    emp, dept  
4 WHERE   emp.deptno=dept.deptno;
```

```
SQL> SELECT e.empno, e.ename, e.deptno,  
2          d.deptno, d.loc  
3  FROM    emp e, dept d  
4 WHERE   e.deptno=d.deptno;
```

Combinando Más de Dos Tablas

CUSTOMER		ORD		ITEM	
NAME	CUSTID	CUSTID	ORDID	ORDID	ITEMID
JOCKSPORTS	100	101	610		
TKB SPORT SHOP	101	102	611		
VOLLYRITE	102	104	612		
JUST TENNIS	103	106	601		
K+T SPORTS	105	102	602		
SHAPE UP	106	106			
WOMENS SPORTS	107	106			
...			
9 rows selected.		21 rows		64 rows selected.	

Non-Equijoins

EMP

EMPNO	ENAME	SAL
7839	KING	5000
7698	BLAKE	2850
7782	CLARK	2450
7566	JONES	2975
7654	MARTIN	1250
7499	ALLEN	1600
7844	TURNER	1500
7900	JAMES	950
...		
14 rows selected.		

SALGRADE

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999



“el salario en la tabla EMP, está entre el salario más bajo y el salario más alto de la tabla SALGRADE”

Recuperación de Registros con Non-Equijoins

```
SQL> SELECT e.ename, e.sal, s.grade  
2  FROM emp e, salgrade s  
3  WHERE e.sal  
4  BETWEEN s.losal AND s.hisal;
```

ENAME	SAL	GRADE
JAMES	950	1
SMITH	800	1
ADAMS	1100	1

...

14 rows selected.

Outer Joins

EMP

ENAME	DEPTNO
KING	10
BLAKE	30
CLARK	10
JONES	20
...	

DEPT

DEPTNO	DNAME
10	ACCOUNTING
30	SALES
10	ACCOUNTING
20	RESEARCH
...	
40	OPERATIONS



No hay empleados en el
departamento OPERATIONS

Outer Joins

- Usar Outer Join para ver las filas que no cumplen la condición de join.
- El operador de un Outer Join es el signo más (+).

```
SELECT table.column, table.column  
FROM   table1, table2  
WHERE  table1.column (+) = table2.column;
```

```
SELECT table.column, table.column  
FROM   table1, table2  
WHERE  table1.column = table2.column (+);
```

Uso de Outer Joins

```
SQL> SELECT      e.ename, d.deptno, d.dname  
  2  FROM        emp e, dept d  
  3  WHERE      e.deptno(+) = d.deptno  
  4  ORDER BY e.deptno;
```

ENAME	DEPTNO	DNAME
KING	10	ACCOUNTING
CLARK	10	ACCOUNTING
...		
		40 OPERATIONS
15 rows selected.		

Self Joins

EMP (WORKER)

EMPNO	ENAME	MGR
7839	KING	
7698	BLAKE	7839
7782	CLARK	7839
7566	JONES	7839
7654	MARTIN	7698
7499	ALLEN	7698

EMP (MANAGER)

EMPNO	ENAME
7839	KING
7839	KING
7839	KING
7698	BLAKE
7698	BLAKE



**“MGR en la tabla WORKER es igual a
EMPNO en la tabla MANAGER”**

Combinación de una Tabla Consigo Misma

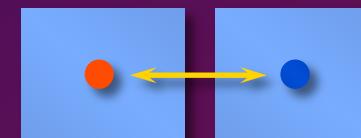
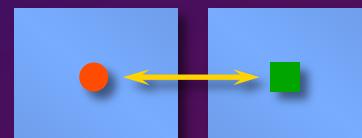
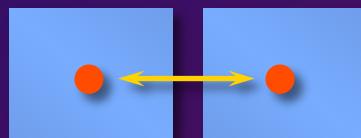
```
SQL> SELECT worker.ename || ' works for ' || manager.ename  
  2  FROM    emp worker, emp manager  
  3 WHERE   worker.mgr = manager.empno;
```

```
WORKER.ENAME || 'WORKSFOR' || MANAG  
-----  
BLAKE works for KING  
CLARK works for KING  
JONES works for KING  
MARTIN works for BLAKE  
...  
13 rows selected.
```

Resumen

```
SELECT      table.column, table.column  
FROM        table1, table2  
WHERE       table1.column1 = table2.column2;
```

Equijoin Non-equijoin Outer join Self join



Visión General de la Práctica

- Combinar tablas usando equijoin.
- Ejecutar outer y self joins.
- Agregar condiciones adicionales.

