

Ejemplos con multitable

COMBINACIONES O JOIN

Un join es una operación que relaciona dos o más tablas para obtener un resultado que incluya datos (campos y registros) de ambas; las tablas participantes se combinan según los campos comunes a ambas tablas.

Hay tres tipos esenciales de combinaciones:

1. combinaciones internas (inner join o join),
2. combinaciones externas
3. combinaciones cruzadas.

También es posible emplear varias combinaciones en una consulta "select", incluso puede combinarse una tabla consigo misma.

La combinación interna emplea "join", que es la forma abreviada de "inner join". Se emplea para obtener información de dos tablas y combinar dicha información en una salida.

Veamos un ejemplo

Una librería almacena la información de sus libros para la venta en dos tablas, "libros2" y "editoriales2". Recuperamos los datos de libros:

```
select * from libros2;
```

Vemos que en el campo "editorial" aparece el código, pero no sabemos el nombre de la editorial. Realizamos un join para obtener datos de ambas tablas (título, autor y nombre de la editorial):

```
select titulo, autor, nombre
from libros2
join editoriales2
on codigoeditorial=editoriales.codigo;
```

Mostramos el código del libro, título, autor y nombre de la editorial realizando un join y empleando alias:

```
select l.codigo,titulo,autor,nombre
from libros2 l
inner join editoriales2 e
on codigoeditorial=e.codigo;
```

Note que al listar el campo "codigo" especificamos a qué tabla pertenece; si no lo hacemos Oracle no sabrá si nos referimos al de la tabla "libros" o "editoriales". Los demás campos no tienen referencia a la tabla porque tienen nombres que no se repiten.

Realizamos la misma consulta anterior agregando un "where" para obtener solamente los libros de la editorial "Siglo XXI":

```
select l.codigo,titulo,autor,nombre
from libros2 l
join editoriales2 e
on codigoeditorial=e.codigo
where e.nombre='Siglo XXI';
```

COMBINACIONES EXTERNAS

Una combinación interna (join) encuentra registros de la primera tabla que se correspondan con los registros de la segunda, es decir, que cumplan la condición del "on" y si un valor de la primera tabla no se encuentra en la segunda tabla, el registro no aparece.

Si queremos saber qué registros de una tabla NO encuentran correspondencia en la otra, es decir, no existe valor coincidente en la segunda, necesitamos otro tipo de combinación, "outer join" (combinación externa).

Las combinaciones externas combinan registros de dos tablas que cumplen la condición, más los registros de la segunda tabla que no la cumplen; es decir, muestran todos los registros de las tablas relacionadas, aún cuando no haya valores coincidentes entre ellas.

Este tipo de combinación se emplea cuando se necesita una lista completa de los datos de una de las tablas y la información que cumple con la condición. Las combinaciones externas se realizan solamente entre 2 tablas.

Hay tres tipos de combinaciones externas: "left outer join", "right outer join" y "full outer join"; se pueden abreviar con "left join", "right join" y "full join" respectivamente.

Se emplea una **combinación externa izquierda** para mostrar todos los registros de la tabla de la izquierda. Si no encuentra coincidencia con la tabla de la derecha, el registro muestra los campos de la segunda tabla puestos a "null".

Ejemplo combinación externa izquierda

Una librería almacena la información de sus libros para la venta en dos tablas, "libros3" y "editoriales3".

Realizamos una combinación izquierda para obtener los títulos de los libros, incluyendo el nombre de la editorial:

```
select titulo,nombre
  from editoriales3 e
 left join libros3 l
    on codigoeditorial = e.codigo;
```

Las editoriales de las cuales no hay libros, es decir, cuyo código de editorial no está presente en "libros3" o tienen valor nulo, aparece en el resultado, pero con el valor "null" en el campo "titulo" (caso de "Siglo XXI", "Sudamericana" y "Norma").

Realizamos la misma consulta anterior pero cambiamos el orden de las tablas:

```
select titulo,nombre
  from libros3 l
 left join editoriales3 e
    on codigoeditorial = e.codigo;
```

El resultado mostrará el título del libro y el nombre de la editorial; los títulos cuyo código de editorial no está presente en "editoriales" o tienen valor nulo, aparecen en el resultado, pero con el valor "null" en el campo "nombre" (caso de los libros "El quijote..." y "Java...").

Restringimos el resultado de una consulta considerando solamente los registros que encuentran coincidencia en la tabla de la derecha, es decir, cuyo valor de código está presente en "libros":

```
select titulo,nombre
  from editoriales3 e
 left join libros3 l
    on e.codigo=codigoeditorial
 where codigoeditorial is not null;
```

Mostramos las editoriales que NO están presentes en "libros", es decir, que NO encuentran coincidencia en la tabla de la derecha:

```
select nombre
  from editoriales3 e
 left join libros3 l
    on e.codigo=codigoeditorial
 where codigoeditorial is null;
```

Aparecen 3 editoriales.

Una **combinación externa derecha** ("right outer join" o "right join") opera del mismo modo sólo que la tabla derecha es la que localiza los registros en la tabla izquierda.

Ejemplo combinación externa derecha

Una librería almacena la información de sus libros para la venta en dos tablas, "libros4" y "editoriales4". Solicitamos el título y nombre de la editorial de los libros empleando un "right join":

```
select titulo,nombre as editorial
from libros4 l
right join editoriales4 e
on codigoeditorial = e.codigo;
```

Las editoriales de las cuales no hay libros, es decir, cuyo código de editorial no está presente en "libros" aparece en el resultado, pero con el valor "null" en el campo "titulo"; caso de la editorial "Siglo XXI".

Realizamos la misma consulta anterior agregando un "where" que restrinja el resultado considerando solamente los registros que encuentran coincidencia en la tabla izquierda:

```
select titulo,nombre as editorial
from libros4 l
right join editoriales4 e
on e.codigo=codigoeditorial
where codigoeditorial is not null;
```

Ya no aparece la editorial "Siglo XXI".

Mostramos las editoriales que NO están presentes en "libros" (que NO encuentran coincidencia en "editoriales"):

```
select nombre
from libros4 l
right join editoriales4 e
on e.codigo=codigoeditorial
where codigoeditorial is null;
```

Solamente aparece la editorial "Siglo XXI".

Una **combinación externa completa** ("full outer join" o "full join") retorna todos los registros de ambas tablas. Si un registro de una tabla izquierda no encuentra coincidencia en la tabla derecha, las columnas correspondientes a campos de la tabla derecha aparecen seteadas a "null", y si la tabla de la derecha no encuentra correspondencia en la tabla izquierda, los campos de esta última aparecen conteniendo "null".

Ejemplo combinación externa completa

Una librería almacena la información de sus libros para la venta en dos tablas, "libros5" y "editoriales5".

Realizamos una combinación externa completa para obtener todos los registros de ambas tablas, incluyendo los libros cuyo código de editorial no existe en la tabla "editoriales" y las editoriales de las cuales no hay correspondencia en "libros":

```
select titulo,nombre as editorial
from editoriales e
full join libros l
on codigoeditorial = e.codigo;
```

Note que el libro "El aleph" cuyo valor de "codigoeditorial" es null, muestra "null" en la columna "editorial" y las editoriales "Sudamericana" y "Siglo XXI" muestran "null" en el campo "titulo".

COMBINACIONES CRUZADAS

Las combinaciones cruzadas (cross join) muestran todas las combinaciones de todos los registros de las tablas combinadas. Para este tipo de join no se incluye una condición de enlace. Se genera el producto cartesiano en el que el número de filas del resultado es igual al número de registros de la primera tabla multiplicado por el número de registros de la segunda tabla, es decir, si hay 3 registros en una tabla y 4 en la otra, retorna 12 filas. No son muy utilizadas.

Ejemplo de combinaciones cruzadas

Un pequeño restaurante tiene almacenados los nombres y precios de sus comidas en una tabla llamada "comidas" y en

una tabla denominada "postres" los mismos datos de sus postres.

El restaurante quiere combinar los registros de ambas tablas para mostrar los distintos menús que ofrece. Lo hacemos usando un "cross join":

```
select c.nombre as "plato principal",
p.nombre as "postre"
from comidas c
cross join postres p;
```

La salida muestra cada plato combinado con cada uno de los postres. Se obtienen 8 registros.

En la siguiente combinación cruzada, agregamos una columna que calcula el precio total de cada menú:

```
select c.nombre as "plato principal",
p.nombre as "postre",
c.precio+p.precio as "total"
from comidas c
cross join postres p;
```

La salida muestra cada plato combinado con cada uno de los postres y el precio total de cada menú. Se obtienen 8 registros.

AUTOCOMBINACIONES

Combinar una tabla consigo misma. En una autocombinación se combina una tabla con una copia de si misma. Para ello debemos utilizar 2 alias para la tabla.

Ejemplo de autocombinación

Un pequeño restaurante tiene almacenados los nombres, precios y rubro de sus comidas en una tabla llamada "comidas2".

Realizamos un "cross join":

```
select c1.nombre,
c2.nombre,
c1.precio+c2.precio as total
from comidas2 c1
cross join comidas2 c2;
```

Nota que aparecen filas duplicadas, por ejemplo, "raviolis" se combina con "ravioles" y la combinación "raviolis- flan" se repite como "flan- raviolis". Debemos especificar que combine el rubro "plato" con "postre":

```
select c1.nombre as "plato principal",
c2.nombre as postre,
c1.precio+c2.precio as total
from comidas2 c1
cross join comidas2 c2
where c1.rubro='plato' and
c2.rubro='postre';
```

La salida muestra cada plato combinado con cada postre, y una columna extra que calcula el total del menú.

También se puede realizar una autocombinación con "join":

```
select c1.nombre as "plato principal",
c2.nombre as postre,
c1.precio+c2.precio total
from comidas2 c1
join comidas2 c2
on c1.codigo<>c2.codigo
where c1.rubro='plato' and
```

```
c2.rubro='postre';
```

Para que no aparezcan filas duplicadas se agrega un "where".

COMBINACIONES Y FUNCIONES DE AGRUPAMIENTO

Podemos usar "group by" y las funciones de agrupamiento con combinaciones de tablas.

Una librería almacena la información de sus libros para la venta en dos tablas, "libros6" y "editoriales6".

Contamos la cantidad de libros de cada editorial consultando ambas tablas:

```
select nombre as editorial,
count(*) as cantidad
from editoriales6 e
join libros6 l
on codigoeditorial=e.codigo
group by e.nombre;
```

Nota que las editoriales que no tienen libros no aparecen en la salida porque empleamos un "join". Para que aparezcan todas las editoriales, tendríamos que usar un left join.

```
select nombre as editorial,
count(l.codigo) as "mayor precio"
from editoriales6 e
left join libros6 l
on codigoeditorial=e.codigo
group by nombre;
```

Buscamos el libro más costoso de cada editorial con un "left join":

```
select nombre as editorial,
max(precio) as "mayor precio"
from editoriales6 e
left join libros6 l
on codigoeditorial=e.codigo
group by nombre;
```

La sentencia anterior mostrará, para la editorial de la cual no haya libros, el valor "null" en la columna calculada

COMBINAR MÁS DE DOS TABLAS

Podemos hacer un "join" con más de dos tablas.

La librería almacena los datos de sus libros en tres tablas: libros7, editoriales7 y autores7. En la tabla "libros7" un campo "codigoautor" hace referencia al autor y un campo "codigoeditorial" referencia la editorial.

Para recuperar todos los datos de los libros empleamos la siguiente consulta:

```
select titulo,a.nombre,e.nombre
from autores7 a
join libros7 l
on codigoautor=a.codigo
join editoriales7 e
on codigoeditorial=e.codigo;
```

Analicemos la consulta anterior. Indicamos el nombre de la tabla luego del "from" ("autores7"), combinamos esa tabla con la tabla "libros7" especificando con "on" el campo por el cual se combinarán; luego debemos hacer coincidir los valores para el enlace con la tabla "editoriales7" enlazándolas por los campos correspondientes. Utilizamos alias para una sentencia más sencilla y comprensible.

Notar que especificamos a que tabla pertenecen los campos cuyo nombre se repiten en las tablas, esto es necesario para evitar confusiones y ambigüedades al momento de referenciar un campo. Los libros cuyo código de autor no se encuentra en "autores7" y cuya editorial no existe en "editoriales7", no aparecen

porque realizamos una combinación interna.

Notar que no aparecen los libros cuyo código de autor no se encuentra en "autores" (caso de "Java en 10 minutos" y "Matematica estas ahi") y cuya editorial no existe en "editoriales" (caso de "Matematica estas ahi" y "Java de la A a la Z"), esto es porque realizamos una combinación interna.

Podemos combinar varios tipos de join en una misma sentencia:

```
select titulo,a.nombre,e.nombre
from autores7 a
right join libros7 l
on codigoautor=a.codigo
left join editoriales7 e
on codigoeditorial=e.codigo;
```

En la consulta anterior solicitamos el título, autor y editorial de todos los libros que encuentren o no coincidencia con "autores" ("right join") y a ese resultado lo combinamos con "editoriales", encuentren o no coincidencia.