

Módulo 2

Código: 484

Bases de datos

Técnico Superior en Desarrollo de
Aplicaciones Multiplataforma



Sesiones 20 a 24. Contenidos teóricos

INTRODUCCIÓN

En las próximas sesiones veremos la última parte (parte 4) de la unidad 6 (DQL). Esta parte se basa en el uso de múltiples consultas en una misma sentenci. En esta parte veremos:

- Combinaciones especiales.
- Subconsultas.
- Subconsultas en DML.

Sesiones 20 a 24. Contenidos teóricos

INTRODUCCIÓN

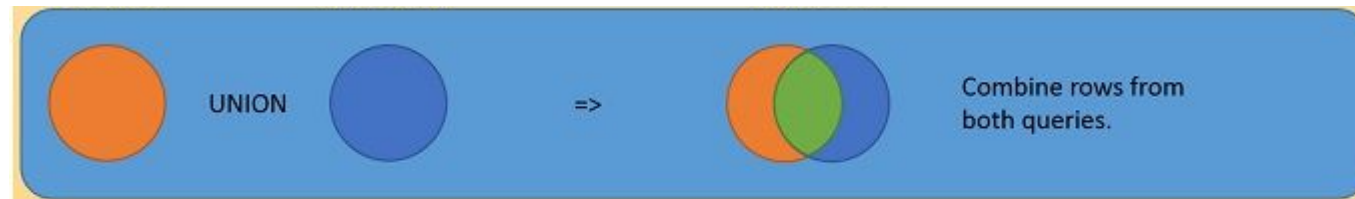
Empezaremos con las combinaciones especiales. Como su nombre indica permiten combinar el resultado de diferentes consultas SELECT. Las operaciones que nos permiten realizar este tipo de combinaciones son:

- UNION
- INTERSECT
- EXCEPT

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - UNION

- UNION permite añadir el resultado de un SELECT a otro SELECT. Crea una sola tabla con registros que estén presentes en cualquiera de las consultas. Si están repetidas sólo aparecen una vez.
- En el siguiente diagrama el círculo naranja representa las filas obtenidas al ejecutar el primer SELECT, el círculo azul el resultado de ejecutar el segundo SELECT. El resultado de la operación UNION entre los dos SELECT es la combinación de las filas de ambas SELECT.



- UNION ALL, igual que UNION pero muestra los datos duplicados
- Las SELECT deben unir el mismo tipo de columnas y en el mismo orden.

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - UNION

Ejemplo: Dadas las siguientes tablas A y B

TABLA A	TABLA B
10	0
10	10
20	10
20	20
50	

1. Calcular el resultado de A UNION B. Muestra los valores que están en cualquiera de las dos tablas sin repetir.
2. Calcular el resultado de A UNION ALL B. Muestra los valores que están en cualquiera de las dos tablas, incluido los repetidos.

A UNION B
0
10
20
50

A UNION ALL B
10
10
20
20
50
0
10
10
20

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - UNION

Ejemplo: Dadas las siguientes tablas A y B

TABLA A	TABLA B
10	0
10	10
20	10
20	20
50	

Como podemos ver A UNION B contiene todas las filas de la tabla A y la tabla B evitando duplicados.

Sin embargo A UNION ALL B contiene todas las filas de la tabla A y la tabla B incluyendo duplicados.

A UNION B
0
10
20
50

A UNION ALL B
10
10
20
20
50
0
10
10
20

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - UNION

Ejemplo: códigos de municipios donde hay clientes o vendedores.

```
SELECT municipio FROM cliente UNION SELECT municipio FROM vendedor;
```

La primera sentencia:

```
SELECT municipio FROM cliente
```

Obtiene los municipios de los clientes.

La segunda sentencia:

```
SELECT municipio FROM vendedor
```

Obtiene los municipios de los vendedores.

El resultado final de la operación UNION será los municipios de los clientes combinados con los municipios de los vendedores evitando duplicados.

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - UNION

Ejemplo: tickets del cliente con código 4 o del vendedor con código 3.

```
SELECT * FROM ticket WHERE cliente = 4 UNION  
SELECT * FROM ticket WHERE vendedor = 3;
```

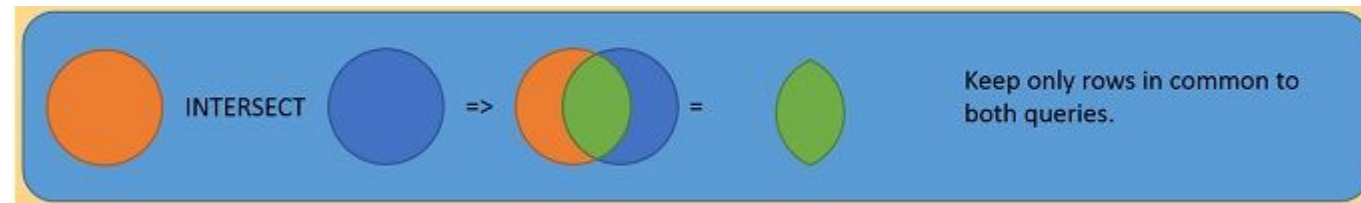
Se podría realizar con una única SELECT:

```
SELECT * FROM ticket WHERE cliente = 4 OR vendedor = 3;
```


Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - INTERSECT

- **INTERSECT** permite unir dos consultas SELECT de modo que el resultado serán las filas que estén presentes en ambas consultas



- **INTERSECT ALL** Si una misma fila aparece m veces en la primera sentencia y n veces en la segunda, en el resultado esta fila aparecerá $\text{MIN}(m,n)$ veces. Es decir, aparecerá el número de veces que se repite cada fila en las diferentes tablas.

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - INTERSECT

Ejemplo: Dadas las siguientes tablas A y B

TABLA A	TABLA B
10	0
10	10
20	10
20	20
50	

1. Calcular el resultado de A INTERSECT B. Muestra los valores que están en las dos tablas evitando duplicados.
2. Calcular el resultado de A INTERSECT ALL B. Muestra los valores que están en cualquiera de las dos tablas, tantas veces como se repitan.

A INTERSECT B
10
20

A INTERSECT ALL B
10
10
20

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - INTERSECT

Otro ejemplo: Dadas las siguientes tablas A y B

TABLA A	TABLA B
10	0
20	10
20	10
50	20

1. Calcular el resultado de A INTERSECT B. Muestra los valores que están en las dos tablas evitando duplicados.
2. Calcular el resultado de A INTERSECT ALL B. Muestra los valores que están en cualquiera de las dos tablas, tantas veces como se repitan.

A INTERSECT B
10
20

A INTERSECT ALL B
10
20

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - INTERSECT

Ejemplo: código de aquellos tickets que tienen al menos una línea en la que se venda más de 5 artículos y su vendedor sea el número 3.

```
SELECT ticket FROM linea_ticket WHERE cant > 5 INTERSECT  
SELECT codigo FROM ticket WHERE vendedor = 3;
```

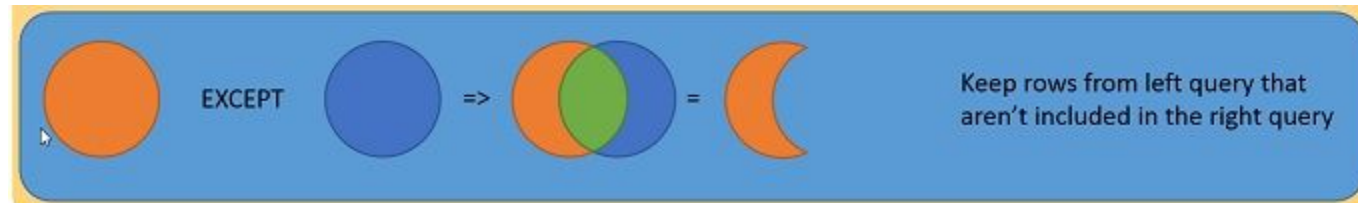
También podríamos realizar la consulta con un INNER JOIN:

```
SELECT DISTINCT t.codigo  
FROM ticket t JOIN linea_ticket lt ON lt.ticket = t.codigo  
WHERE cant > 5 AND vendedor = 3;
```

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - EXCEPT

EXCEPT mostrará los registros de la primera SELECT que no estén presentes en la segunda.



Ejemplo: Dadas las anteriores tablas A y B, calcular el resultado de A EXCEPT B.

TABLA A	TABLA B	A EXCEPT B
10	0	50
10	10	
20	10	
20	20	
50		

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES - EXCEPT

Ejemplo: Códigos de municipios donde no hay clientes.

```
SELECT codigo FROM municipio EXCEPT  
SELECT municipio FROM cliente;
```

Sesiones 20 a 24. Contenidos teóricos

COMBINACIONES ESPECIALES

Recuerda. Cuando se realiza una operación algebraica de dos consultas, es obligatorio que coincida el número y tipo de las columnas devueltas por cada una de las consultas.

Ejemplo incorrecto. No coincide ni el número ni el tipo de las columnas devueltas por ambas consultas.

```
SELECT codigo, provincia FROM municipio EXCEPT  
SELECT nombre, municipio, codigo FROM cliente;
```