

Curso SQL

Nelson López Centeno

Curso SQL

- ☐ Consultas - SELECT
 - ☐ Combinar - JOIN
 - ☐ Anexar - UNION
- ☐ Subconsultas y CTE
- ☐ Funciones de ventana
- ☐ Modificar Datos - INSERT, UPDATE, DELETE
- ☐ Transacciones
- ☐ Modificar Estructura - CREATE, ALTER, DROP
 - ☐ Procedimientos almacenados
 - ☐ Triggers

Configuración del entorno de trabajo



MySQL

<https://dev.mysql.com/downloads/mysql/>



DBeaver

<https://dbeaver.io/download/>

SELECT

**SELECT DISTINCT
FROM
WHERE
GROUP BY
HAVING
ORDER BY
LIMIT**

Orden de ejecución

**FROM
WHERE
GROUP BY
HAVING
SELECT
DISTINCT
ORDER BY
LIMIT**

TIPOS DATOS

Numéricos

INT, SMALLINT, TINYINT, BIGINT

DECIMAL, NUMERIC

FLOAT, REAL

Cadenas Caracteres

CHAR

VARCHAR

TEXT

Fecha y Hora

DATE

TIME

DATETIME

Binario

BINARY

VARBINARY

BLOB

FUNCIONES DE TEXTO

UPPER, LOWER

LEFT, RIGHT

LENGTH

TRIM, LTRIM, RTRIM, BTRIM

SUBSTRING

CONCAT

FUNCIONES DE FECHA Y HORA

CURRENT_DATE

CURRENT_TIME

CURRENT_TIMESTAMP

DATEADD

DATEDIFF

DATEPART

FILTRADO - WHERE

=, <>, <, >, <=, >=

AND, OR, NOT

BETWEEN ... AND

IN (...)

LIKE

CASE

CASE exp

WHEN valor1 THEN res1

WHEN valor2 THEN res2

WHEN valor3 THEN res3

ELSE res4

END

CASE

WHEN cond1 THEN res1

WHEN cond2 THEN res2

WHEN cond3 THEN res3

ELSE res4

END

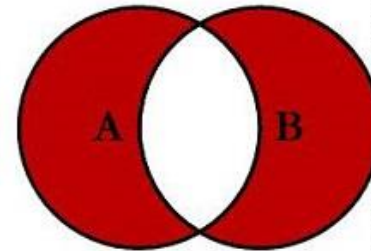
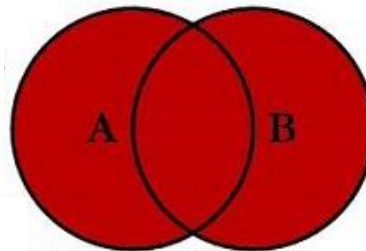
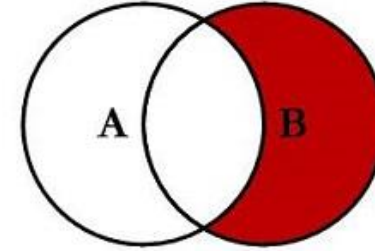
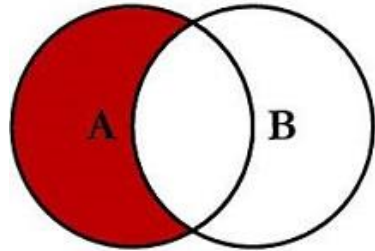
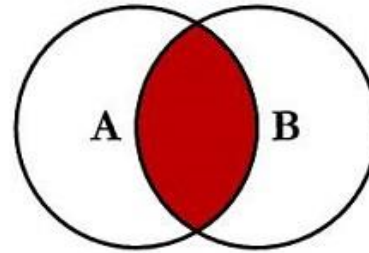
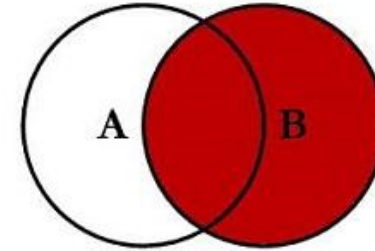
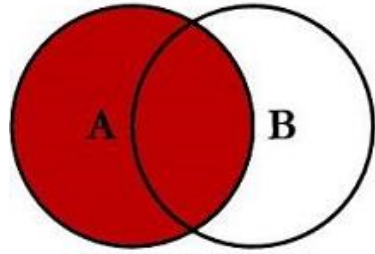
FUNCIONES DE AGREGACIÓN

COUNT
SUM
MIN, MAX
AVG

**Estas funciones se utilizan
frecuentemente en
combinación con
GROUP BY y con
HAVING**

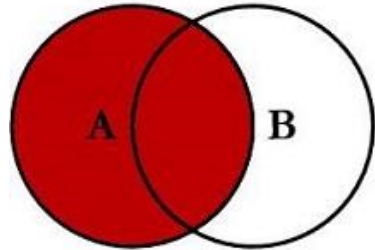
JOIN

SQL JOINS

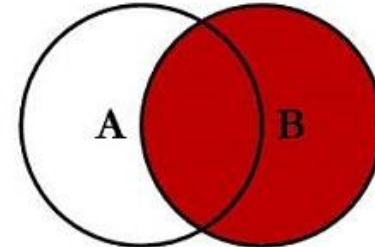


JOIN

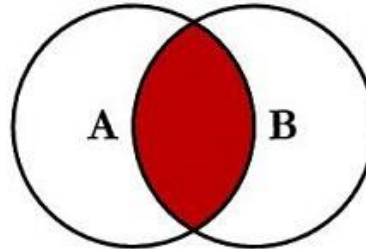
SQL JOINS



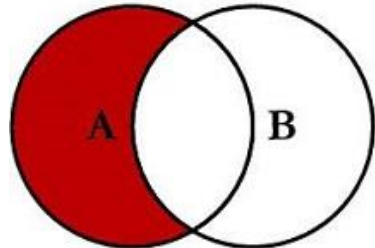
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key
```



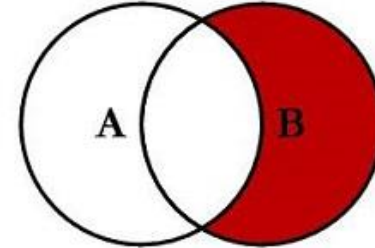
```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key
```



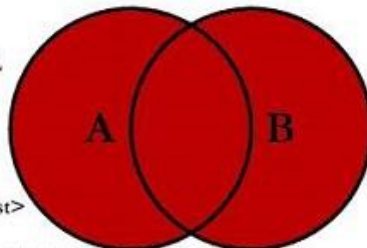
```
SELECT <select_list>  
FROM TableA A  
INNER JOIN TableB B  
ON A.Key = B.Key
```



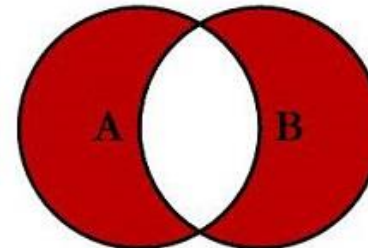
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key  
WHERE B.Key IS NULL
```



```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key  
WHERE A.Key IS NULL
```



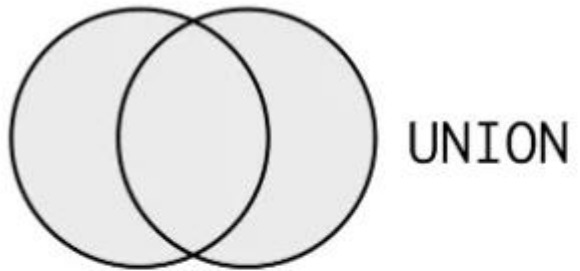
```
SELECT <select_list>  
FROM TableA A  
FULL OUTER JOIN TableB B  
ON A.Key = B.Key
```



```
SELECT <select_list>  
FROM TableA A  
FULL OUTER JOIN TableB B  
ON A.Key = B.Key  
WHERE A.Key IS NULL  
OR B.Key IS NULL
```

© C.L. Moffatt, 2008

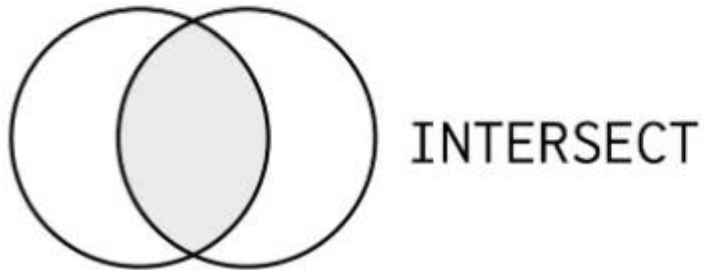
UNION



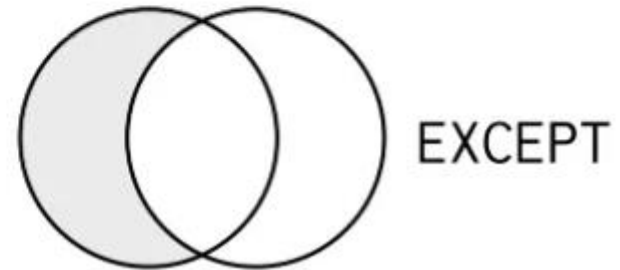
UNION



UNION ALL



INTERSECT



EXCEPT

SUBCONSULTA

Filtrar una consulta basándose en otra

```
SELECT c1
FROM t1
WHERE
    c2 > (
        SELECT AVG(c3)
        FROM t2
    )
```

```
SELECT c1
FROM t1
WHERE
    c2 IN (
        SELECT c3
        FROM t2
        WHERE c4 > 200
    )
```

SUBCONSULTA

Crear nuevas columnas a partir de agregaciones

```
SELECT  
    c1,  
    (  
        SELECT AVG(c3)  
        FROM t2  
    ) AS c2  
FROM t1
```

SUBCONSULTA

Consultas correlacionadas

```
SELECT
    c1,
    c2
FROM t1
WHERE c2 = (
    SELECT MAX(c3)
    FROM t2
    WHERE t1.c1 = t2.c4
)
```


CTE: Common Table Expression

Organizar consultas complejas

```
WITH cte1 AS (  
    SELECT  
        c1,  
        SUM(c2)  
    FROM t1  
    GROUP BY c1  
)
```

```
SELECT  
    ...  
FROM cte1  
...  
  
SELECT  
    ...  
FROM cte1
```

CTE: Common Table Expression

Consulta recursiva

```
WITH cte_recursiva AS (  
    SELECT  
        c1  
    FROM t1  
    INNER JOIN cte_recursiva  
    ...  
)
```

```
SELECT  
    ...  
FROM cte_recursiva  
...
```