



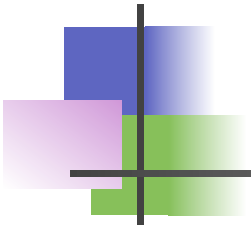
**Departamento de
Ciencias de la Computación y Tecnologías de Información
Universidad del Bío-Bío
Sede Chillán**

Bases de Datos

Lenguaje SQL

M^a Angélica Caro Gutiérrez

<http://www.face.ubiobio.cl/~mcaro/>
mcaro@ubiobio.cl



Lenguaje SQL

- Introducción
- Conceptos básicos
- ➔ ■ Consultas básicas en SQL
- Consultas complejas en SQL
- Vistas (tablas virtuales) en SQL



Consultas Básicas en SQL

- SELECT: Consulta para recuperar datos de la BD
- La sintaxis de la orden SELECT consta básicamente de las cláusulas SELECT y FROM como obligatorias y de otras varias cláusulas opcionales:

<cláusula SELECT> <cláusula FROM>

[<cláusula WHERE>]

[<cláusula GROUP BY> [<cláusula HAVING>]]

[<cláusula ORDER BY>]



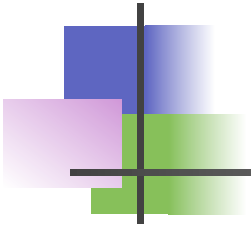
Consultas Básicas en SQL

- <Cláusula SELECT>

- Permite indicar los datos que queremos seleccionar

SELECT [ALL | DISTINCT | UNIQUE]
<lista-de selecciones>

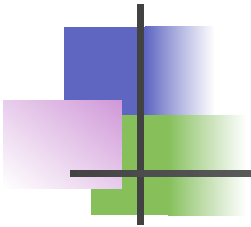
- ***DISTINCT***: Elimina las filas duplicadas en el resultado de la consulta.
- ***UNIQUE*** es igual que DISTINCT.
- ***<lista-de selecciones>***: Lista de nombre de columnas o expresiones separadas por comas.



Consultas Básicas en SQL

- <Cláusula FROM>
 - Permite indicar las tablas que contienen los datos.

FROM {<nombre de tabla> [alias de tabla] }



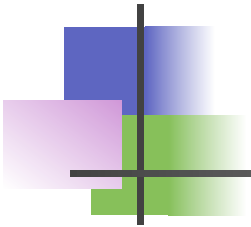
Consultas Básicas en SQL

- <Cláusula WHERE>

- Sirve para indicar la condición que deben cumplir las filas resultantes.

WHERE <condición>

- Una condición está formada por una o varias expresiones condicionales conectadas por los operadores lógicos AND, OR y NOT.



Consultas Básicas en SQL

- La expresión condicional de la cláusula WHERE tiene una de las formas siguientes (1/2):

<expresión1> <operador relacional> <expresión2>

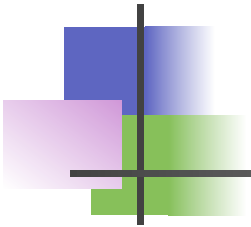
Verifica si las dos expresiones satisfacen la comparación.

<expresión1> [NOT] BETWEEN <expresión2> AND <expresión3>

Verifica si la expresión1 tiene un valor comprendido entre los valores de la expresión2 y la expresión3.

<expresión> [NOT] IN (<lista-de valores>)

Verifica si la expresión tiene un valor de los indicados en la lista de valores.



Consultas Básicas en SQL

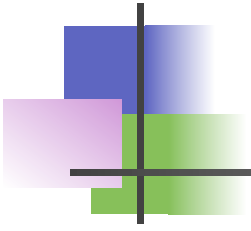
- La expresión condicional de la cláusula WHERE tiene una de las formas siguientes (2/2):

<nombre de columna> [NOT] LIKE "<string>" [ESCAPE "<carácter de escape>"]

Verifica si el valor de la columna se adapta al patrón de búsqueda (string). Se admiten caracteres comodín ("%" representa cero o más caracteres, "_" representa un único carácter). El carácter de escape '\' permite referirse a los caracteres comodín como caracteres y no como comodines.

<nombre de columna> IS [NOT] NULL

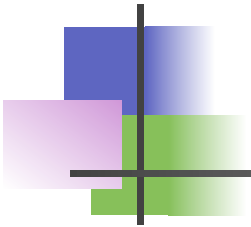
Verifica si el valor de la columna es nulo.



Consultas Básicas en SQL

- Ejemplos:
 - Libro (codigo, autor, titulo, editor, clase),
 - Recuperar el autor y titulo del libro cuyo codigo es 'CB1020'.

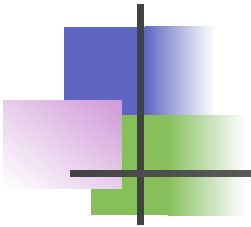
```
SELECT AUTOR, TITULO  
FROM LIBRO  
WHERE CODIGO='CB1020';
```



Consultas Básicas en SQL

- Ejemplos:
 - Libro (codigo, autor, titulo, editor, clase),
 - Recuperar el autor y titulo del libro cuyo editor es 'Alfaguara' y de clase 3.

```
SELECT CODIGO, TITULO  
FROM LIBRO  
WHERE EDITOR='ALFAGUARA' AND CLASE=3;
```



Consultas Básicas en SQL

- Ejemplos:
 - Libro (codigo, autor, titulo, editor, clase),
 - Recuperar el editor de todos los libros de clase 3.

```
SELECT EDITOR  
FROM LIBRO  
WHERE CLASE=3;
```

¿=?

```
SELECT DISTINCT EDITOR  
FROM LIBRO  
WHERE CLASE=3;
```



Consultas Básicas en SQL

- Ejemplos:
 - Libro (codigo, autor, titulo, editor, clase),
 - Recuperar todos los datos de los libros de clase 3.

```
SELECT CODIGO, AUTOR, TITULO, EDITOR, CLASE  
FROM LIBRO  
WHERE CLASE=3;
```

ó

```
SELECT *  
FROM LIBRO  
WHERE CLASE=3;
```



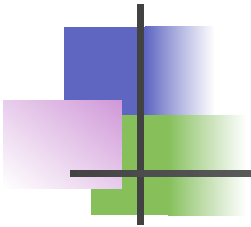
Consultas Básicas en SQL

- Ejemplos: **Uso de LIKE**
 - Libro (codigo, autor, titulo, editor, clase)
 - Usuario (carnet, nombre, direccion),
 - Prestamo (codigo, carnet, fecha_inicio, fecha_fin)
 - Recuperar los datos de los libros cuya editorial contenga el string 'AL'.

SELECT * FROM LIBRO WHERE EDITOR LIKE '%AL%';

SELECT * FROM LIBRO WHERE EDITOR LIKE '%L%';

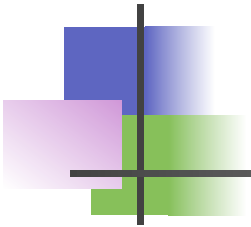
SELECT * FROM LIBRO WHERE EDITOR LIKE '_L%';



Consultas Básicas en SQL

■ Ejercicios:

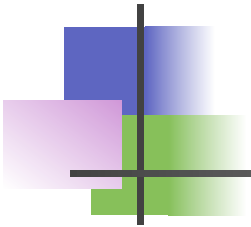
- Recuperar todos los datos de los empleados de sexo masculino.
- Obtener los nombres y dni de los empleados que ganan menos de 40.000 y que pertenecen al departamento 4.
- Recuperar los nombres de todos los hijos de los empleados.
- Obtener los nombres y apellidos de los empleados cuyo nombre comience con la letra "A".
- Obtener los departamentos y nombres de sus proyectos que contienen la palabra "producto" en cualquier posición de su nombre.
- Obtener los dni de los directores cuyo nombre de departamento tiene como segunda letra una "d".
- Obtener el nombre completo de los empleados que no tienen un supervisor.



Consultas Básicas en SQL

- Ejemplos:
 - Libro (codigo, autor, titulo, editor, clase)
 - Clase (clave, tiempo_prestamo)
- Recuperar los títulos de los libros cuyo tiempo de préstamo es mayor o igual a 4 días.

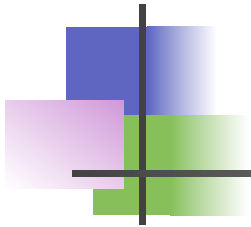
```
SELECT TITULO  
FROM LIBRO, CLASE  
WHERE CLAVE=CLASE AND  
        TIEMPO_PRESTAMO >= 4;
```



Consultas Básicas en SQL

- Ejemplos:
 - Libro (codigo, autor, titulo, editor, clase)
 - Clase (clave, tiempo_prestamo)
- Recuperar los títulos de los libros cuyo tiempo de préstamo es mayor o igual a 4 días.

```
SELECT TITULO  
FROM LIBRO, CLASE  
WHERE CLAVE=CLASE AND  
        TIEMPO_PRESTAMO >= 4;
```

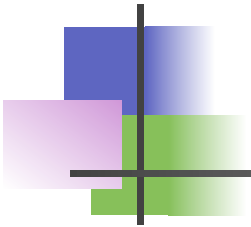



Consultas Básicas en SQL

- Ejemplos: **Uso de Alias**
 - Libro (codigo, autor, titulo, editor, clase)
 - Clase (clave, tiempo_prestamo)
- Recuperar los titulos de los libros cuyo tiempo de prestamo es mayor o igual a 4 días.

```
SELECT A.TITULO FROM LIBRO AS A, CLASE AS B
WHERE B.CLAVE=A.CLASE AND
        B.TIEMPO_PRESTAMO >= 4;
```

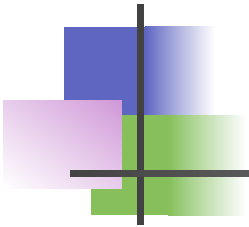
- Los alias son útiles cuando hay nombres de atributos repetidos.



Consultas Básicas en SQL

- Ejemplos:
 - Libro (codigo, autor, titulo, editor, clase)
 - Usuario (carnet, nombre, direccion),
 - Prestamo (codigo, carnet, fecha_inicio, fecha_fin)
 - Recuperar los datos de los usuarios que tienen prestados libros de la clase 1.

```
SELECT B.CARNET, B.NOMBRE, B.DIRECCION  
FROM LIBRO A, USUARIO B, PRESTAMO C  
WHERE A.CODIGO=C.CODIGO AND  
        B.CARNET=C.CARNET AND A.CLASE=1;
```



Consultas Básicas en SQL

- Renombrar los atributos de la relación:

```
SELECT NOMBRE AS NOMBRE_SOCIO  
FROM USUARIO  
WHERE CARNET ='11111111';
```

- Consultas sin cláusula **where**:

```
SELECT CARNET FROM USUARIO;
```

```
SELECT * FROM PRESTAMOS;
```

```
SELECT * FROM LIBROS, CLASE;
```



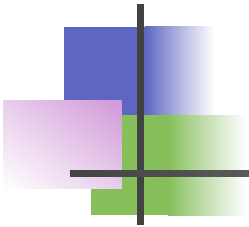
Consultas Básicas en SQL

- Tablas como conjuntos
 - SQL suele tratar una tabla no como un conjunto sino como un multiconjunto
 - Pueden aparecer tuplas repetidas en una tabla y en el resultado de una consulta
 - Algunas razones para mantener las tuplas repetidas:
 - La eliminación de tuplas es una operación muy costosa
 - El usuario quiere ver las tuplas repetidas
 - Funciones de agregación (veremos esto más adelante)



Consultas Básicas en SQL

- Tablas como conjuntos
 - **DISTINCT** – Eliminación de tuplas repetidas
 - **ALL** – Se conservan las tuplas repetidas
- Ejemplo:
 - **SELECT ALL CODIGO**
FROM LIBRO
 - **SELECT DISTINCT CARNET**
FROM PRESTAMO

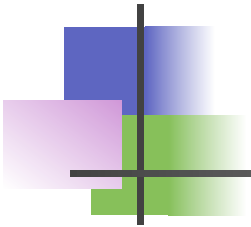


Consultas Básicas en SQL

- Operador **BETWEEN**

```
SELECT CODIGO, CARNET  
FROM PRESTAMO  
WHERE (FECHA_INICIO BETWEEN '1-3-2011' AND '20-3-2011');
```

```
SELECT TITULO  
FROM LIBRO, CLASE  
WHERE CLASE = CLAVE AND  
      (TIEMPO_PRESTAMO BETWEEN 3 AND 6);
```



Consultas Básicas en SQL

- Ordenación: **ORDER BY**
 - Obtener una lista de los nombres de usuarios y títulos de los libros se han prestado durante el mes de marzo, ordenados por título y por nombre de usuario

```
SELECT TITULO, NOMBRE  
FROM LIBRO A, USUARIO B, PRESTAMO C  
WHERE A.CODIGO=C.CODIGO AND  
        B.CARNET=C.CARNET AND (FECHA_INICIO  
        BETWEEN '1-3-2011' AND '31-3-2011')  
ORDER BY TITULO, NOMBRE;
```