

Laboratorio IV – SQL

Lenguaje de Manipulación de Datos
(DML)

SELECT, JOIN, UNION

Ing. Gregorio Nicolás Tkachuk



Objetivos

- ▶ Que el estudiante logre:
 - Conocer y utilizar los comandos SELECT y JOIN.
 - Habilidad para utilizar las distintas variantes de los comandos SELECT, JOIN y UNION.

DML: Contenidos

- ▶ Consulta combinadas
- ▶ JOIN
- ▶ Combinación interna
 - INNER JOIN
- ▶ Combinación externa
 - LEFT OUTER JOIN
 - RIGHT OUTER JOIN
- ▶ UNION

Consultas Combinadas o Consultas MultiTablas

Habitualmente surge la necesidad de obtener información de la Base de Datos y esta se encuentra repartida en varias tablas, las consultas combinadas o multitabla permiten resolver este problema permitiendo obtener la información necesaria accediendo a varias tablas.

Las consultas combinadas pueden ser de dos tipos

- ▶ Combinación interna (INNER JOIN)
- ▶ Combinación externa (OUTER JOIN)

Consultas Combinadas, Introducción

- ▶ Selección de columnas específicas de varias tablas
 - La palabra clave JOIN especifica qué tablas se van a combinar y cómo
 - La palabra clave ON especifica la condición de combinación
- ▶ Consultas de dos o más tablas para producir un conjunto de resultados
 - Use claves principales y externas como condiciones de combinación
 - Para combinar tablas, utilice columnas comunes a las tablas especificadas

Consultas Combinadas – JOIN

- ▶ El formato modificado de la clausula FROM de la sentencia SELECT es la siguiente:
- ▶ Formato general:
- ▶ **FROM <tabla1> <operador_join> JOIN <tabla2> ON <condicion_join>**
- ▶ **<operador_join> = {INNER | LEFT [OUTER] | RIGHT [OUTER]}**
- ▶ Formato detallado:
- ▶ **FROM <tabla1> {INNER | LEFT [OUTER] | RIGHT [OUTER]} JOIN <tabla2> ON <tabla1.columna1> = <tabla2.columna2>**

Consultas Combinadas – JOIN

- ▶ **Combinación Interna:**

INNER JOIN: se muestran solo los registros en los cuales coinciden los campos de la unión (deben existir en las dos tablas)

- ▶ **Combinación Externa:**

LEFT OUTER JOIN: se muestran todos los registros de la tabla de la izquierda (tabla1); si hay valores coincidentes en los campos de la unión de la segunda tabla se muestran, caso contrario se dejan en blanco. La tabla a la izquierda de la palabra JOIN se denomina tabla preservada y la de la derecha no preservada.

RIGHT OUTER JOIN: se muestran todos los registros de la tabla de la derecha (tabla2); si hay valores coincidentes en los campos de la unión de la primera tabla se muestran, caso contrario se dejan en blanco. La tabla a la derecha de la palabra JOIN se denomina tabla preservada y la de la izquierda no preservada.

Ejemplo de combinaciones internas

```
SELECT clienteNombre, ventas.clienteId, cant
FROM cliente INNER JOIN ventas
ON cliente.clienteId = ventas.clienteId;
```

cliente

<i>clienteNombre</i>	<i>clienteId</i>
Adam Barr	1
Willy Turry	2
Eva Corets	3
Erin O'Melia	4

ventas

<i>clienteId</i>	<i>prodId</i>	<i>cant</i>
1	2	15
1	3	5
4	1	37
3	5	11
4	2	1003

Resultado

<i>clienteNombre</i>	<i>clienteId</i>	<i>cant</i>
Adam Barr	1	15
Adam Barr	1	5
Erin O'Melia	4	37
Eva Corets	3	11
Erin O'Melia	4	1003

Uso de combinaciones externas

```
SELECT clienteNombre, ventas.clienteId, cant  
FROM cliente LEFT OUTER JOIN ventas  
ON cliente.clienteId = ventas.clienteId;
```

cliente

<i>clienteNombre</i>	<i>clienteId</i>
Adam Barr	1
Willy Turry	2
Eva Corets	3
Erin O'Melia	4

ventas

<i>clienteId</i>	<i>prodId</i>	<i>cant</i>
1	2	15
1	3	5
4	1	37
3	5	11
4	2	1003

Resultado

<i>clienteNombre</i>	<i>clienteId</i>	<i>cant</i>
Adam Barr	1	15
Adam Barr	1	5
Erin O'Melia	4	37
Eva Corets	3	11
Erin O'Melia	4	1003
Willy Turry	NULL	NULL

SQL – JOIN – Tablas de ejemplo

Docente

Docenteld (PK)	DocenteNombre	DocenteDptold (FK)
1	PEREZ, MARTIN	1
2	PAZ, MONICA	
3	GARCIA, DIEGO	2
4	LOPEZ, MAGALI	1

Departamento

Departamentold (PK)	Departamento Nombre
1	Informatica
2	Matemática
3	Agrimensura
4	Electrónica

SQL – INNER JOIN – Ejemplos

```
SELECT *  
FROM Docente INNER JOIN Departamento  
ON DocenteDptold = Departamentold;
```

/* Esta consulta muestra todos los campos de las dos tablas pero solo los registros para los cuales se cumple estrictamente la condición.

No aparecen PAZ, MONICA, porque no tiene departamento asignado, ni los departamentos AGRIMENSURA y ELECTRONICA.

*/

SQL – LEFT OUTER JOIN – Ejemplos

```
SELECT *  
FROM Docente LEFT OUTER JOIN Departamento  
ON DocenteDptold = Departamentold;
```

/* Esta consulta muestra todos los registros de la tabla de la izquierda del JOIN o sea de Docente y solo los Departamentos para los cuales se cumple la condición. Los demás registros de la tabla departamento se muestran en blanco

Aparece PAZ, MONICA, pero no los departamentos AGRIMENSURA y ELECTRONICA.

*/

SQL – JOIN – Tablas de ejemplo

Docente

Docenteld (PK)	DocenteNombre	DocenteDptold (FK)
1	PEREZ, MARTIN	1
2	PAZ, MONICA	
3	GARCIA, DIEGO	2
4	LOPEZ, MAGALI	1

Departamento

Departamentold (PK)	Departamento Nombre
1	Informatica
2	Matemática
3	Agrimensura
4	Electrónica

SQL – RIGHT OUTER JOIN – Ejemplos

```
SELECT *  
FROM Docente RIGHT OUTER JOIN Departamento  
ON DocenteDptold = Departamentold;
```

/* Esta consulta muestra todos los registros de la tabla de la derecha del JOIN o sea de Departamento y solo los Docentes para los cuales se cumple la condición. Los demás registros de la tabla Docente se muestran en blanco.

Aparecen los departamentos AGRIMENSURA y ELECTRONICA, pero no la docente PAZ, MONICA.

*/

Combinación de varios conjuntos de resultados – UNION

- ▶ **UNION** se usa para combinar los resultados de varias sentencias Select en un único conjunto de resultados.
- ▶ Formato general:
- ▶ **SELECT ...**
- ▶ **UNION [ALL]**
- ▶ **SELECT ...**
 - [UNION [ALL]
 - SELECT ...]

UNION

- ▶ El operador UNION permite crear un conjunto de resultados único a partir de varias consultas
- ▶ Cada consulta debe tener:
 - El número y el orden de las columnas deben ser idénticos en todas las consultas.
 - Los tipos de datos deben ser compatibles.
- ▶ Si se utiliza UNION sin ninguna cláusula, no se incluirán los valores repetidos en el conjunto de resultado.
- ▶ Si se incluye la cláusula [ALL] se incluirán los valores repetidos en el conjunto de resultado.

UNION, Ejemplo

Ejemplo:

```
SELECT (nombre + ' ' + apellido) AS nombre, ciudad,  
codigoPostal
```

```
FROM empleados
```

```
UNION
```

```
SELECT clienteNombre, ciudad, codigoPostal  
FROM clientes
```

/* Este ejemplo muestra la salida del primer select mas la salida
del segundo select, no muestra valores repetidos.

*/

¿ Preguntas ?