

A2. Consultas simples sobre una tabla.

Índice.

1.	Formato básico de una Consulta.....	2
2.	Filtros en una Consulta.....	3
2.1.	Expresiones para filtros.....	4
2.2.	Tipos de filtros.....	6
2.2.1.	Filtros con operador de pertenencia a conjuntos.....	7
2.2.2.	Filtros con operador de rango.....	8
2.2.3.	Filtros con test de valor nulo.....	9
2.2.4.	Filtros con test de patrón.....	10
2.2.5.	Filtros con límite de número de registros.....	11
2.2.6.	Filtros con expresiones regulares.....	12
3.	Ordenación de una Consulta.....	14

A2. Consultas simples sobre una tabla.

1. Formato básico de una Consulta.

El formato básico para hacer una Consulta en SQL es el siguiente:

```
SELECT * | { [DISTINCT] columna | expresión [alias], ... }  
FROM Tabla;
```

En donde:

- En la línea del SELECT:
 - Todos los campos (o atributos): *
 - Sólo aparecen valores NO duplicados : DISTINCT
 - Columnas o expresiones de cálculo : columna | expresión
 - Nuevo nombre de la columna resultado: ALIAS
- En la línea del FROM:
 - Tabla o tablas origen de los datos: Tabla

Para mostrar todos los datos de la tabla Personas:

```
SELECT * FROM Personas;
```

Para mostrar nombre y apellidos de la tabla Personas:

```
SELECT nombre, apellidos FROM Personas;
```

Para mostrar todos los nombres distintos de Personas:

```
SELECT DISTINCT nombre FROM Personas;
```

Para mostrar un encabezado en la Consulta:

```
SELECT DISTINCT nombre ALIAS "NOMBRES" FROM Personas;
```

A2. Consultas simples sobre una tabla.

2. Filtros de una Consulta.

Los **filtros** de las Consultas son las condiciones que cualquier gestor de la base de datos interpreta para realizar una selección de registros y mostrarlos como resultado de la Consulta.



En SQL la palabra clave para realizar filtros es la cláusula WHERE.

```
SELECT [DISTINCT] expresión_select [,expresión_select]  
      [FROM tabla]  
      [WHERE filtro]
```

A2. Consultas simples sobre una tabla.

2.1. Expresiones para filtros.

Un **filtro** es una expresión que indica la condición (o condiciones) que deben satisfacer todos los registros para ser seleccionados:

- Los filtros se construyen con expresiones.
- Una expresión es una combinación de operadores, operandos y funciones para producir un resultado.

```
SQL> SELECT APELLIDO,SALARIO FROM EMPL WHERE OFICIO='VENDEDOR' ;
```

APELLIDO	SALARIO
ARROYO	1500
SALA	1625
MARTIN	1600
TOVAR	1350

```
SQL>
```

Los elementos que forman parte de las expresiones son:

- Operandos → constantes, números enteros, números reales, cadenas de caracteres, fechas, variables u otras expresiones.
- Operadores aritméticos → +, -, *, /, %
- Operadores relacionales → >, <, <>, >=, <=, =
- Operadores lógicos → AND, OR, NOT
- Paréntesis → ()
- Funciones → cada sistema gestor de bases de datos incorpora su propio repertorio de funciones.

A2. Consultas simples sobre una tabla.

2.1. Expresiones para filtros.

Los elementos que forman parte de las expresiones son:

- Operandos → constantes, números enteros, números reales, cadenas de caracteres, fechas, variables u otras expresiones.
- Operadores aritméticos → +, -, *, /, %
- Operadores relacionales → >, <, <>, >=, <=, =
- Operadores lógicos → AND, OR, NOT
- Paréntesis → ()
- Funciones → cada sistema gestor de bases de datos incorpora su propio repertorio de funciones.

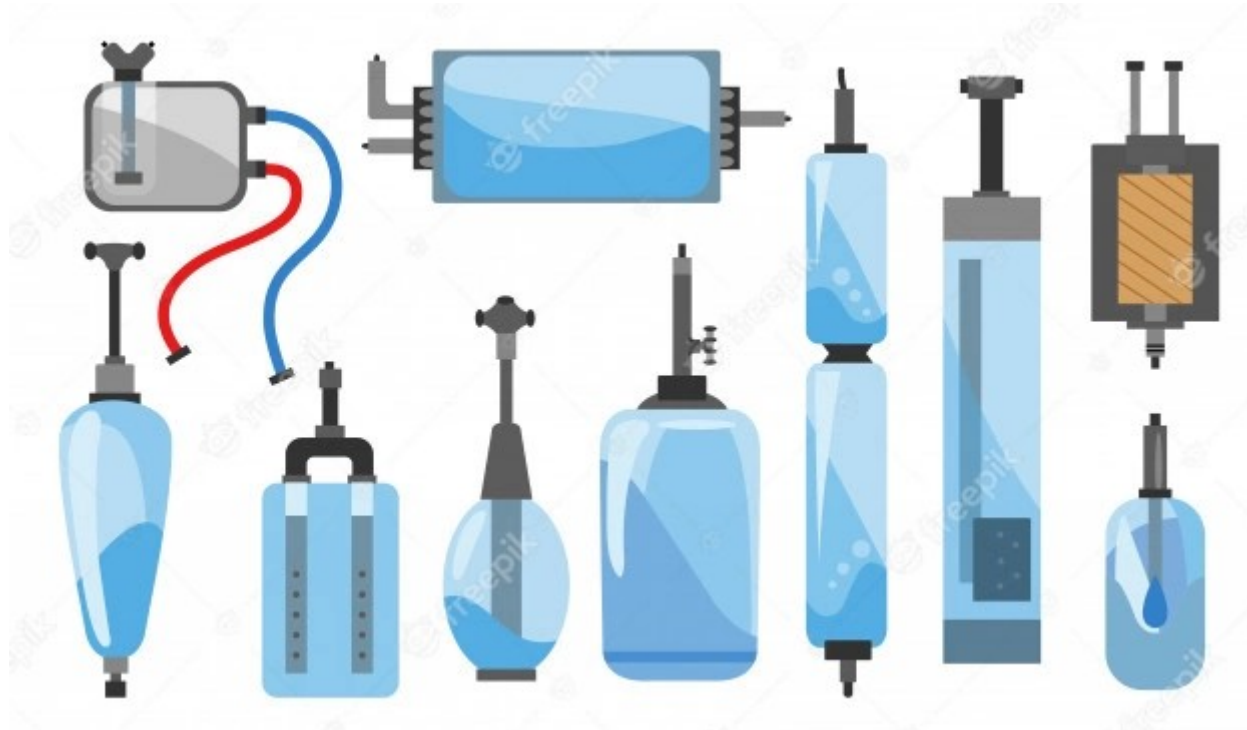
```
SELECT
    DATEPART(DAY,VMP.fechaPago) as fecha,
    ISNULL(SUM(VMP.totalPago),0) as total
FROM VentasMatriculasPagos VMP
WHERE VMP.estado = 1
    AND YEAR(VMP.fechaPago) = 2017--@anio
    AND MONTH(VMP.fechaPago) = 8--@mes
    AND VMP.idSede = 1--@idSede
```

A2. Consultas simples sobre una tabla.

2.2. Tipos de filtros.

Los filtros que se emplean en SQL pueden ser de los siguientes tipos:

- Filtros con operador de pertenencia a conjuntos.
- Filtros con operador de rango.
- Filtros con test de valor nulo.
- Filtros con test de patrón.
- Filtros con límite de número de registros.



A2. Consultas simples sobre una tabla.

2.2.1. Filtros con operador de pertenencia a conjuntos.

El operador asociado a la pertenencia a un conjunto es IN.

El operador IN permite comprobar si una columna tiene un valor igual a alguno de los pertenecientes al conjunto, que estarán incluidos entre paréntesis.

```
nombre_columna IN (valor1, valor2, ...)
```

Para seleccionar todos los datos de las personas cuya ciudad sea Pontevedra o Vigo:

```
SELECT *  
FROM persona  
WHERE ciudad IN ('Pontevedra','Vigo');
```

A2. Consultas simples sobre una tabla.

2.2.2. Filtros con operador de rango.

El operador asociado a la pertenencia a un rango es BETWEEN.

El operador BETWEEN permite seleccionar los registros que están incluidos en un rango.

```
nombre_columna BETWEEN valor1 AND valor2
```

Para seleccionar todos los datos de las personas cuya ciudad sea Pontevedra o Vigo:

```
SELECT *  
FROM persona  
WHERE altura BETWEEN 160 AND 180;
```


A2. Consultas simples sobre una tabla.

2.2.3. Filtros con test de valor nulo.

El operador asociado al valor nulo es IS.

El operador IS permite verificar si un campo es nulo (o no nulo).

```
nombre_columna IS [NOT] NULL
```

Para seleccionar todos los datos de las personas cuya altura se desconozca:

```
SELECT *  
FROM persona  
WHERE altura IS NULL;
```

A2. Consultas simples sobre una tabla.

2.2.4. Filtros con test de patrón.

El operador asociado a los test de patrón es LIKE.

El operador LIKE permite seleccionar los registros a partir de un patrón.

```
nombre_columna LIKE patrón
```

Los filtros asociados a los test de patrón son de dos tipos:

- % busca coincidencias de cualquier número de caracteres (incluso de cero caracteres)
- _ busca coincidencias de, exactamente, un carácter

Para seleccionar todos las personas cuyo nombre contenga el texto 'mas':

```
SELECT *  
FROM persona  
WHERE nombre LIKE '%mas%';
```

A2. Consultas simples sobre una tabla.

2.2.5. Filtros con límite de número de registros.

El operador asociado al límite en el número de registros es LIMIT.

El operador LIKE permite limitar el número de registros que se devuelve en cada consulta.

```
[LIMIT [desplazamiento,] filas]
```

desplazamiento

a partir de qué fila se empieza a contar el desplazamiento

filas

número de filas que se devolverán

Para seleccionar todos los datos de la primera persona de la lista:

```
SELECT *  
FROM persona LIMIT 1;
```

A2. Consultas simples sobre una tabla.

2.2.6. Filtros con expresiones regulares de patrones.

El operador asociado a las expresiones regulares de patrones es REGEXP (o RLIKE).

El operador REGEXP (o RLIKE) permite seleccionar los registros a partir de expresiones regulares.

```
nombre_columna REGEXP expresión_regul
```

Una expresión regular es una poderosa herramienta que sirve para comprobar si una cadena de texto se ajusta a un determinado patrón.

La expresión regular estará contenido entre comillas simples ', o doble "".

Para seleccionar todos los datos de de una persona cuyo nombre cumpla una condición:

```
SELECT *  
FROM persona  
WHERE nombre REGEXP expresión_regul;
```

A2. Consultas simples sobre una tabla.

2.2.6. Filtros con expresiones regulares de patrones.

Los caracteres especiales que puede contener la expresión regular son:

- **Comienzo y fin de cadena**

- Indicador de comienzo de cadena: **^**
- Indicador de fin de cadena: **\$**

```
select * from persona where nombre REGEXP '^H';
```

```
select * from persona where nombre REGEXP 'z$';
```

- **Coincidencia con caracteres**

- De cualquier tipo: **.**
- De alguno de la secuencia: **|**

```
select * from persona where nombre REGEXP '^.';
```

```
select * from persona where nombre REGEXP '^H|^A';
```

- **Indicadores de repetición de caracteres**

- Si NO aparece o lo hace una: **?**
- Si aparece una o más veces: **+**
- Si NO aparece o lo hace más veces: *****
- Si aparece en un rango: **{min,max}**
- Si los valores aparecen en un rango: **(caracteres){min,max}**

```
select * from persona where nombre REGEXP 'a?';
```

```
select * from persona where nombre REGEXP 'a+';
```

```
select * from persona where nombre REGEXP 'a*';
```

```
select * from persona where nombre REGEXP 'a{2,4}';
```

```
select * from persona where nombre REGEXP '(a,e){2,4}';
```

- **Coincidencia de secuencia de caracteres**

- Con un carácter de un conjunto: **[]**
- Con la totalidad de caracteres: **()**

```
select * from persona where nombre REGEXP '[a,b]';
```

```
select * from persona where nombre REGEXP '(María,Luis)';
```

A2. Consultas simples sobre una tabla.

3. Ordenación de una Consulta.

La ordenación de los datos mostrados en una Consulta se realiza a través de la cláusula ORDER BY de la sentencia SELECT.

Esta cláusula permite ordenar el conjunto de resultados de forma ascendente o descendente, en función de una o más columnas. Si no se indica nada, se ordenará de forma ascendente.

La columna por la que se ordenará se indicará por su nombre, una expresión o por la posición numérica en el campo.

```
SELECT [DISTINCT] expresión_select [,expresión_select]  
[FROM tabla]  
[WHERE filtro]  
[ORDER BY {nombre_columna | expresión | posición} [ASC | DESC] ], ...
```