



Arquitectura y Administración de bases de datos con SQL 2021

Héctor Manuel Garduño Castañeda

Diciembre, 2021



Contenido

IN

BETWEEN

LIKE



El comando **IN** se ocupa para reducir el uso de **OR** cuando hay múltiples de estos últimos; es compatible con **SELECT**, **INSERT**, **UPDATE** o **DELETE**.

```
SELECT <<nombre de la columna>>  
FROM <<nombre de la tabla>>  
WHERE <<nombre de otra columna>> IN ('Valor 1', 'Valor 2',...);
```



Práctica

De la tabla *customer*, seleccionar aquellos registros cuya ciudad es Philadelphia o Seattle; seleccionar aquellos registros cuya ciudad es Philadelphia o Seattle, y que además pertenezcan al segmento de corporate.

```
SELECT * FROM customer WHERE city IN ('Philadelphia', 'Seattle');
```

```
SELECT * FROM customer WHERE city IN ('Philadelphia', 'Seattle')  
AND segment IN ('Corporate');
```



La condición **BETWEEN** se utiliza para filtrar mediante valores en un rango. Es compatible con **SELECT**, **INSERT**, **UPDATE** o **DELETE**.

```
SELECT <<nombre de la columna>>  
FROM <<nombre de la tabla>>  
WHERE <<nombre de otra columna>> BETWEEN 'Valor 1' AND 'Valor  
2' ;
```



Práctica

De la tabla *customer*, seleccionar aquellos registros cuya edad está entre 20 y 30, incluidos ambos; seleccionar aquellos registros cuya edad no esté entre 20 y 30, incluidos ambos; de la tabla *sales* seleccionar aquellos registros en los que el envío se realizó entre el 1 de abril de 2015 y el 1 de abril de 2016, ambos inclusive.

```
SELECT * FROM customer WHERE age BETWEEN 20 AND 30;
```

```
SELECT * FROM customer WHERE age NOT BETWEEN 20  
AND 30;
```

```
SELECT * FROM sales WHERE ship_date BETWEEN '2015-04-01'  
AND '2016-04-01';
```



La sentencia **LIKE** se utiliza para filtrar mediante valores de coincidencia por patrones mediante *comodines*. Es compatible con **SELECT**, **INSERT**, **UPDATE** o **DELETE**.

SELECT <<nombre de la columna>>
FROM <<nombre de la tabla>>
WHERE <<nombre de otra columna>> **LIKE** {patrón} ;
donde el patrón consiste en un comodín.



Los comodines que se utilizan son %, que permite que *entre sus extremos haya cualquier cantidad de caracteres, incluyendo espacios*, y -, que permite que *entre sus extremos haya un solo caracter*.

Por ejemplo:

A% significa que busque textos que comiencen con A, como 'ABC' o 'ABCDE'.

%A significa que busque textos que terminen con A, como 'BCA' o 'BCDE A'.

A%B significa que busque textos que comiencen con A y terminen con B, como 'AC HB' o 'AB'.

AB_C significa que busque textos que comiencen con AB, luego siga un único caracter, y luego una C, como 'AB C' o 'ABDC'.



Práctica

De la tabla de clientes, muestra los registros cuyo nombre comience con J; muestra los registros cuyo nombre incluya Nelson; muestra los registros cuyo primer nombre tenga exactamente 4 caracteres; muestra las ciudades, sin repetición, donde el nombre de la ciudad no comience son S.

```
SELECT * FROM customer WHERE customer_name LIKE 'J%';
```

```
SELECT * FROM customer WHERE customer_name LIKE  
'%Nelson%';
```

```
SELECT * FROM customer WHERE customer_name LIKE '____ %';
```

```
SELECT DISTINCT city FROM customer WHERE city NOT  
LIKE 'S%';
```

NOTA. Si deseas buscar algo que incluya el símbolo porcentaje, la búsqueda se hace añadiendo un slash. Por ejemplo

```
SELECT * FROM customer WHERE customer_name LIKE 'G\%';
```

