

# SQL Programming

## Módulo 4 - Subconsultas

# SubConsultas

# SubConsultas

Una subconsulta es una consulta anidada en una instrucción SELECT, INSERT, UPDATE o DELETE, o bien en otra subconsulta. Las subconsultas se pueden utilizar en cualquier parte en la que se permita una expresión.

## Uso de Subconsulta en el SELECT

El siguiente ejemplo muestra el número del producto, el precio y el precio promedio de todos los productos.

### Sintaxis

```
SELECT ProductID, ListPrice, (SELECT AVG(ListPrice) FROM  
Production.Product)  
  
FROM Production.Product;
```



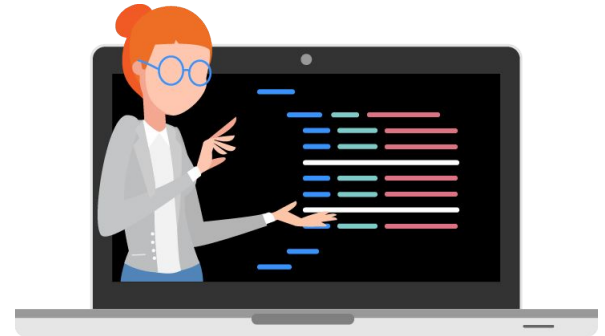
# SubConsultas

## Uso de Subconsulta en el FROM

El siguiente ejemplo muestra el número del producto, el precio y el precio promedio de venta para cada uno de los productos obtenido desde una subconsulta.

### Sintaxis

```
SELECT pp.ProductID, pp.ListPrice, x.promedio
FROM Production.Product pp
INNER JOIN (
    SELECT ProductID, avg(LineTotal) promedio
    FROM Sales.SalesOrderDetail
    GROUP BY ProductID
) x
ON pp.ProductID=x.ProductID;
```



# SubConsultas

## Uso de Subconsulta en el WHERE

El siguiente ejemplo muestra el número del producto y el precio para cada uno de los productos cuyo precio sea inferior al precio promedio de todos los productos.

### Sintaxis

```
SELECT ProductID, ListPrice  
  
FROM Production.Product  
WHERE ListPrice < ( SELECT AVG(LineTotal) FROM Sales.SalesOrderDetail  
);
```

# Restricciones en las Subconsultas

Las subconsultas están sujetas a las restricciones siguientes:

- Las subconsultas siempre van encerradas entre paréntesis
- La lista de selección de una subconsulta que se especifica con un operador de comparación sólo puede incluir un nombre de expresión o columna (excepto EXISTS e IN, que operan en SELECT \* o en una lista respectivamente).
- Si la cláusula WHERE de una consulta externa incluye un nombre de columna, debe ser compatible con una combinación con la columna indicada en la lista de selección de la subconsulta.
- Los tipos de datos ntext, text e image no están permitidos en las listas de selección de subconsultas.



## Restricciones en las Subconsultas

- Puesto que deben devolver un solo valor, las subconsultas que se especifican con un operador de comparación sin modificar no pueden incluir las cláusulas GROUP BY y HAVING.
- La palabra clave DISTINCT no se puede usar con subconsultas que incluyan GROUP BY.
- Sólo se puede especificar ORDER BY si se especifica también TOP.
- Una vista creada con una subconsulta no se puede actualizar.
- La select list de una subconsulta sólo puede retornar una comuna o una expresión. De todos modos, EXISTS puede operar con SELECT \*

# Subconsultas correlacionadas

Las subconsultas correlacionadas son aquellas que se evalúan por cada registro de la tabla referenciada en la consulta principal.

Esto significa que la ejecución de la subconsulta es repetida una vez por cada registro que se evalúa en la cláusula WHERE de la consulta principal.

El siguiente ejemplo muestra el producto más barato de cada subcategoría. Mostrar subcategoría, código de producto y el precio de lista más barato ordenado por subcategoría.

## Sintaxis

```
SELECT p1.ProductSubcategoryID, p1.ProductID,  
p1.ListPrice  
FROM Production.Product p1  
WHERE ListPrice =(  
                SELECT MIN (ListPrice)  
                FROM Production.Product p2  
                WHERE p2.ProductSubcategoryID =  
p1.ProductSubcategoryID  
                )  
ORDER BY p1.ProductSubcategoryID;
```



# IN

Determina si un valor especificado coincide con algún valor de una subconsulta o una lista.

La cláusula IN especifica un rango de búsqueda. Este operador devuelve aquellos registros cuyo campo indicado coincide con alguno de los en una lista

## Uso de IN y NOT IN

La siguiente subconsulta obtiene las personas que son vendedores.

## Sintaxis

```
SELECT FirstName, MiddleName, LastName  
  
FROM Person.Person  
WHERE BusinessEntityID IN (  
    SELECT BusinessEntityID  
    FROM Sales.SalesPerson  
);
```

# IN

La siguiente subconsulta obtiene las personas que son vendedores.

## Sintaxis

```
SELECT FirstName, MiddleName, LastName  
FROM Person.Person  
WHERE BusinessEntityID NOT IN (  
    SELECT BusinessEntityID  
    FROM Sales.SalesPerson  
);
```



# EXISTS

Especifica una subconsulta para probar la existencia de filas.

Cuando se introduce una subconsulta con la palabra clave EXISTS, esta funciona como un test de existencia. La cláusula WHERE de la consulta exterior comprueba por la existencia de las filas retornadas por la subconsulta. La subconsulta en realidad no produce ningún dato, solo retorna un valor de TRUE o FALSE.

## Uso de EXISTS y NOT EXISTS

La siguiente subconsulta obtiene las personas que son vendedores.

### Sintaxis

```
SELECT FirstName, LastName
FROM Person.Person p
WHERE EXISTS (
    SELECT BusinessEntityID
    FROM Sales.SalesPerson s
    WHERE p.BusinessEntityID=s.BusinessEntityID
);
```

# EXISTS

La siguiente subconsulta obtiene las personas que no son vendedores.

## Sintaxis

```
SELECT FirstName, LastName  
  
FROM Person.Person p  
WHERE NOT EXISTS (  
    SELECT BusinessEntityID  
    FROM Sales.SalesPerson s  
    WHERE p.BusinessEntityID=s.BusinessEntityID  
);
```



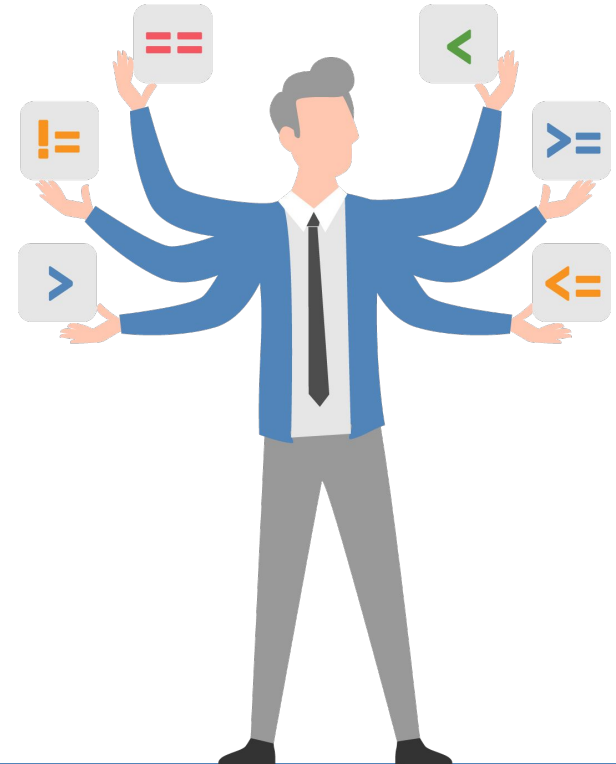
# SOME | ANY

Compara un valor escalar con un conjunto de valores de una sola columna. SOME y ANY son equivalentes.

## Argumentos

{ = | <> | != | > | >= | !=> | < | <= | !=< }

Es cualquier operador de comparación válido.



## SOME | ANY

### Subconsultas que son usadas con operadores de comparación

Los operadores de comparación que introduce una subconsulta se pueden modificar con las palabras claves ALL y ANY.

Las Subconsultas introducidas con un operador de comparación modificado retorna una lista de cero o más valores y pueden incluir una cláusula GROUP BY o HAVING. Estas consultas pueden tener la cláusula EXISTS.

Las palabras clave ALL y ANY comparan un valor escalar con un conjunto de valores simples. La palabra clave ALL indica que se debe aplicar a todos los valores, mientras que ANY indica que al menos a un valor.

Si se usa como ejemplo el operador de comparación > (mayor), entonces >ALL significa mayor que cualquier valor. Es decir, significa mayor que el valor máximo. Por ejemplo, >ALL (1, 2, 3) significa mayor que 3. >ANY significa mayor que por lo menos un valor; es decir, mayor que el mínimo. Entonces, >ANY (1, 2, 3) significa mayor que 1.

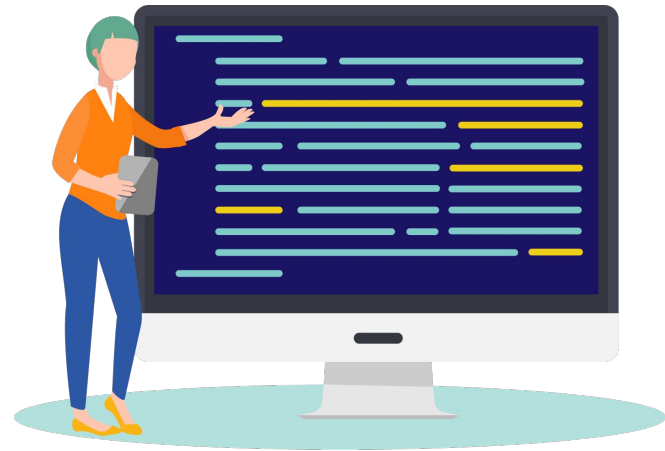
**Importante:** =ANY es equivalente a EXISTS

## SOME | ANY

La siguiente subconsulta obtiene los productos cuyo precio de lista sea mayor al promedio.

### Sintaxis

```
SELECT Name, ListPrice
FROM Production.Product
WHERE ListPrice > ANY (
    SELECT AVG(ListPrice) Promedio
    FROM Production.Product
);
```



# ALL

Compara un valor escalar con un conjunto de valores de una sola columna. Para que una fila de una subconsulta con >ALL satisfaga la condición especificada en la consulta externa, el valor de la columna que presenta la subconsulta debe ser mayor que cada valor de la lista de los valores devueltos por la subconsulta. De forma parecida, >ANY significa que, para que una fila satisfaga la condición especificada en la consulta externa, el valor de la columna que presenta la subconsulta debe ser mayor que, como mínimo, uno de los valores de la lista devuelta por la subconsulta.

## Argumentos

{ = | <> | != | > | >= | !> | < | <= | !< }

Es un operador de comparación.

**Importante:** <>ALL es equivalente a NOT EXISTS

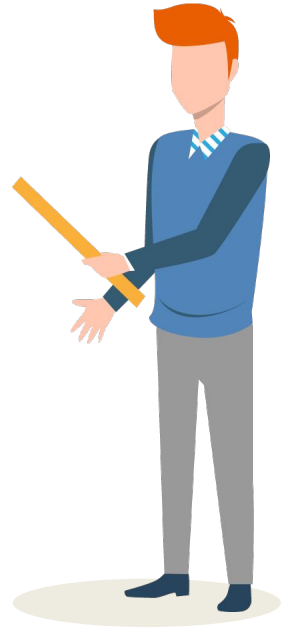


# ALL

La siguiente subconsulta obtiene los productos cuyo precio de lista sea distinto al promedio.

## Sintaxis

```
SELECT Name, ListPrice
FROM Production.Product
WHERE ListPrice <> ALL (
    SELECT AVG(ListPrice) Promedio
    FROM Production.Product
);
```



# ¡Muchas gracias!

¡Sigamos trabajando!