



# INTRODUCCIÓN A SQL

CONSULTAS SENCILLAS A LA BASE DE DATOS



## SQL. La sentencia SELECT

---

- Se utiliza para realizar consultas sobre la base de datos
- Su formato básico es el siguiente:

*SELECT \* FROM nombredetabla;*



## SQL. La sentencia SELECT

---

- La primera palabra es SELECT e indica que se quiere realizar una consulta.
- \* quiere decir que se quieren recuperar todos los campos de la tabla
- FROM indica la(s) tabla(s) sobre la que se realiza la consulta
- A la derecha de FROM se escribe el nombre de la tabla sobre la que se realiza la consulta

# SQL. La sentencia SELECT

---

Datos Ventana Ayuda

rollback T Auto V440 public@visus

\*<V440> Script-35 X

```
select * from bancos b
```

bancos 1 X

select \* from bancos b | Enter a SQL expression to filter results (use Ctrl+Space)

Grilla	idbanco	codigo	nombre
1	1	7	BANCO DE GALICIA Y BUENOS AIRES S.A.
2	2	11	BANCO DE LA NACION ARGENTINA
3	3	14	BANCO DE LA PROVINCIA DE BUENOS AIRES
4	4	15	INDUSTRIAL AND COMMERCIAL BANK
5	5	16	CITIBANK N.A.
6	6	17	BANCO BBVA ARGENTINA S.A.
7	7	20	BANCO DE LA PROVINCIA DE CORDOBA
8	8	27	BANCO SUPERVIELLE S.A.
9	9	29	BANCO DE LA CIUDAD DE BUENOS AIRES
10	10	34	BANCO PATAGONIA S.A.





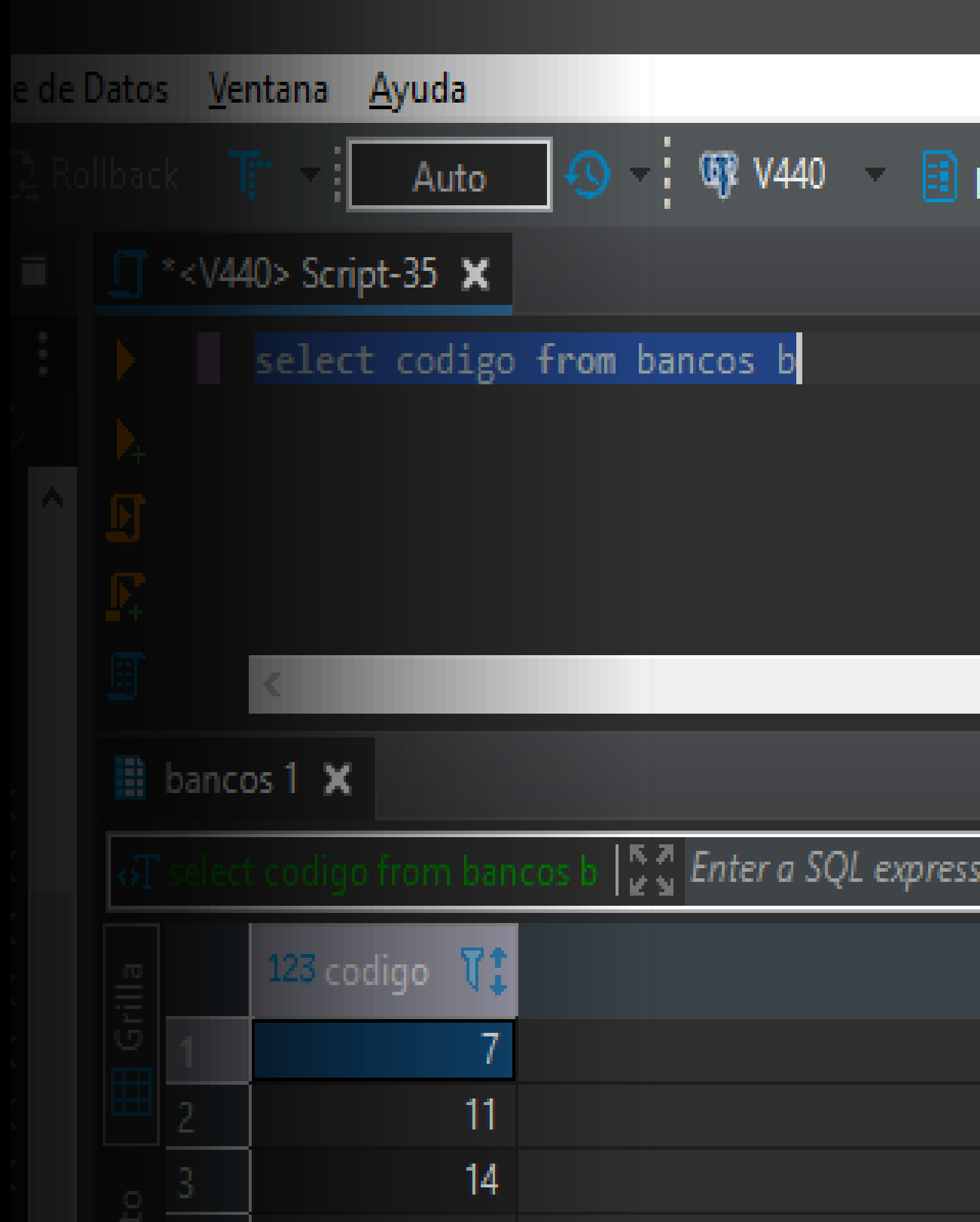
## SQL. Selección de columnas

---

- Con \* a la derecha del SELECT se visualizan todas las columnas de la tabla
- Si queremos consultar ciertas columnas, escribiremos sus nombres separados por comas.
- `SELECT campo1,campo2,campo3... FROM nombretabla;`

## SQL. Ejemplos

- Obtener el código de todos los bancos:
- SELECT codigo FROM bancos;



## SQL. Ejemplos

- Obtener el apellido y la situación fiscal de todas las personas
- `SELECT apellido_persona, situacion_fiscal FROM personas;`

The screenshot shows a database management tool interface. At the top, there is a menu bar with 'e Datos', 'Ventana', and 'Ayuda'. Below the menu bar, there is a toolbar with 'Rollback', a text editor icon, a dropdown menu showing 'Auto', a circular arrow icon, a dropdown menu showing 'V440', and a document icon labeled 'publ'. The main area displays a script editor with a tab labeled '\*<V440> Script-35'. The script contains the following SQL query:

```
select apellido_persona, situacion_fiscal
from personas p
```

Below the script editor, there is a table viewer with a tab labeled 'personas 1'. The table viewer shows the results of the query in a grid format. The first column is labeled 'Grilla' and contains the numbers 1 and 2. The second column is labeled 'apellido\_persona' and contains the values 'Zapata lcart' and 'Martínez'. The third column is labeled 'situacion\_fiscal' and contains the values 'Responsable Monotribu' and 'IVA Responsable Inscript'.

Grilla	apellido_persona	situacion_fiscal
1	Zapata lcart	Responsable Monotribu
2	Martínez	IVA Responsable Inscript

## SQL. Alias de tablas

- Se puede asociar un nuevo nombre a la tabla utilizando alias.

- `select u.descripcion, u.numero`

`from ubicaciones u`

- A la tabla ***ubicaciones*** se le asigna un nuevo nombre (*u*).

The screenshot shows a database management tool interface. At the top, there is a menu bar with 'Datos', 'Ventana', and 'Ayuda'. Below the menu bar, there is a toolbar with 'Rollback', a dropdown menu showing 'Auto', a circular arrow icon, a dropdown menu showing 'V440', and a 'pub' icon. The main window has a tab labeled '\*<V440> Script-35'. The SQL editor contains the query: `select u.descripcion, u.numero from ubicaciones u`. Below the editor, there is a tab labeled 'ubicaciones 1'. The results are displayed in a table with the following data:

	ABC descripcion	ABC numero
1	Estantería	498AA
2	Estantería	584Yes
3	Pallet	987

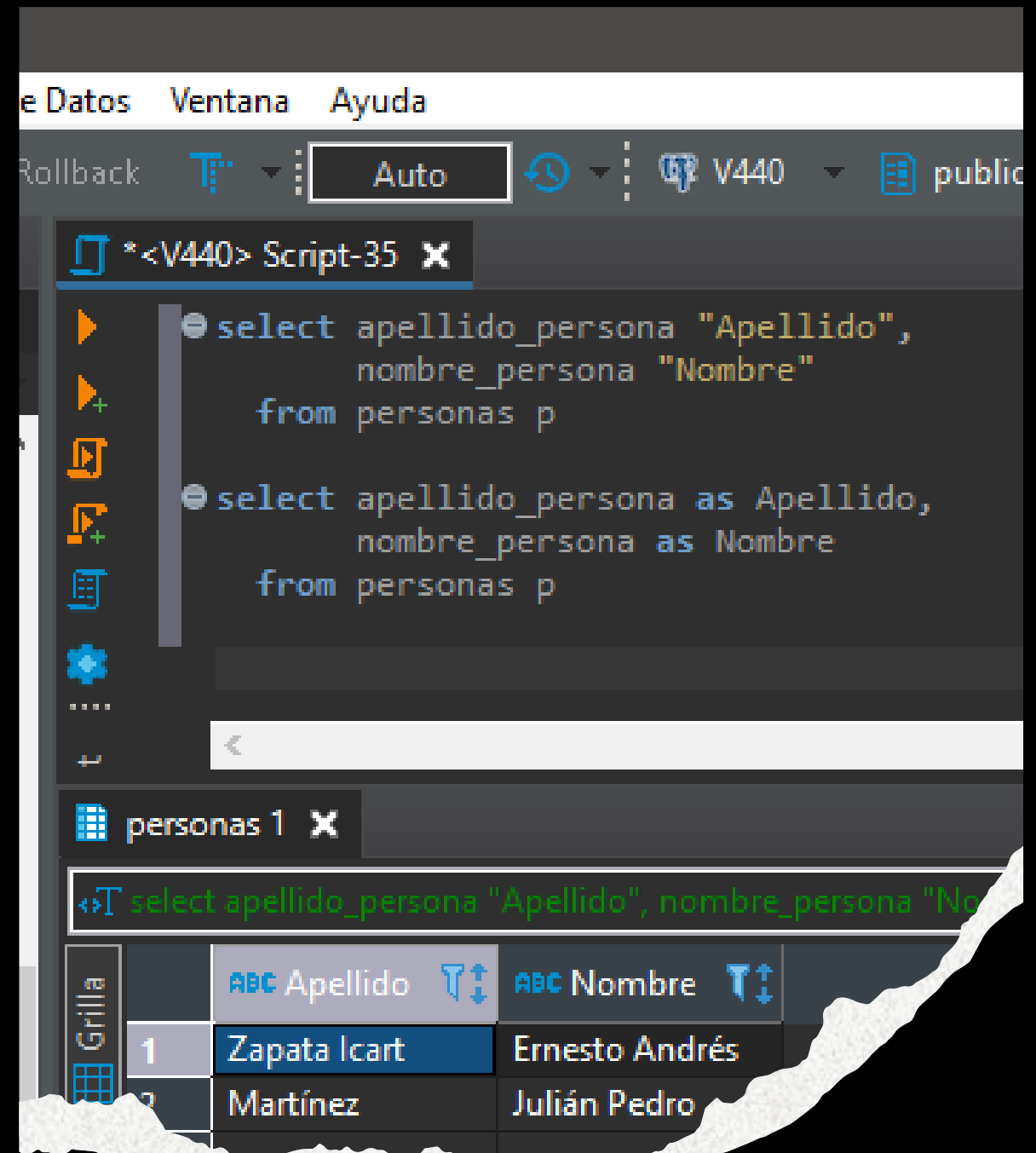


# SQL. Alias de columnas

- Los nombres de las columnas se usan como cabeceras de presentación
- Si son demasiados largos, se pueden cambiar utilizando alias de columnas.
- El alias se pone entre comillas simples o dobles a la derecha de la columna deseada.

# SQL. Alias de columnas

- `select apellido_persona "Apellido", nombre_persona "Nombre" from personas p`
- `select apellido_persona as Apellido, nombre_persona as Nombre from personas p`



# SQL. Columnas calculadas

- Una consulta puede incluir columnas cuyos valores se obtienen a partir de los valores almacenados en columnas de la tabla.

```
select a.codigobarra,
```

```
(a.precio_costo * a.cantidad) "Inventario  
Valorizado" from articulos a
```

The screenshot shows a SQL IDE interface. At the top, there's a menu bar with 'Ventana' and 'Ayuda'. Below it, a toolbar contains icons for text, a dropdown menu with 'Auto', a refresh icon, a server icon labeled 'V440', and a user icon labeled 'public@vis'. A tab labeled '\*<V440> Script-35' is active. The main editor area contains the following SQL query:

```
select a.codigobarra,  
       (a.precio_costo * a.cantidad) "Inventario Val  
from articulos a
```

Below the editor, there's a horizontal scrollbar. A tab labeled 'articulos 1' is visible. The results pane at the bottom shows the query's output as a table:

	ABC codigobarra	123 Inventario Valorizado
1	7790040929906	42.720
2	7790580982102	85.200

# SQL. Cláusula WHERE

- La cláusula WHERE se utiliza para obtener aquellos datos que cumplan la condición expresada.
- Su formato es el siguiente:
- SELECT columna1, columna2, ...
- FROM *nombredetabla*
- WHERE *condición*;



# SQL. Cláusula WHERE

- `select p.apellido_persona,  
p.nombre_persona`  
  
`from personas p`  
  
`where`  
`p.limite_facturasvencidas = 1`

The screenshot shows a SQL IDE interface. At the top, there are menu items: 'e Datos', 'Ventana', and 'Ayuda'. Below the menu is a toolbar with buttons for 'Rollback', a SQL statement icon, a dropdown menu showing 'Auto', a refresh icon, a dropdown menu showing 'V440', and a user icon showing 'public@visu'. The main editor area has a tab labeled '\*<V440> Script-35'. It contains the following SQL query:

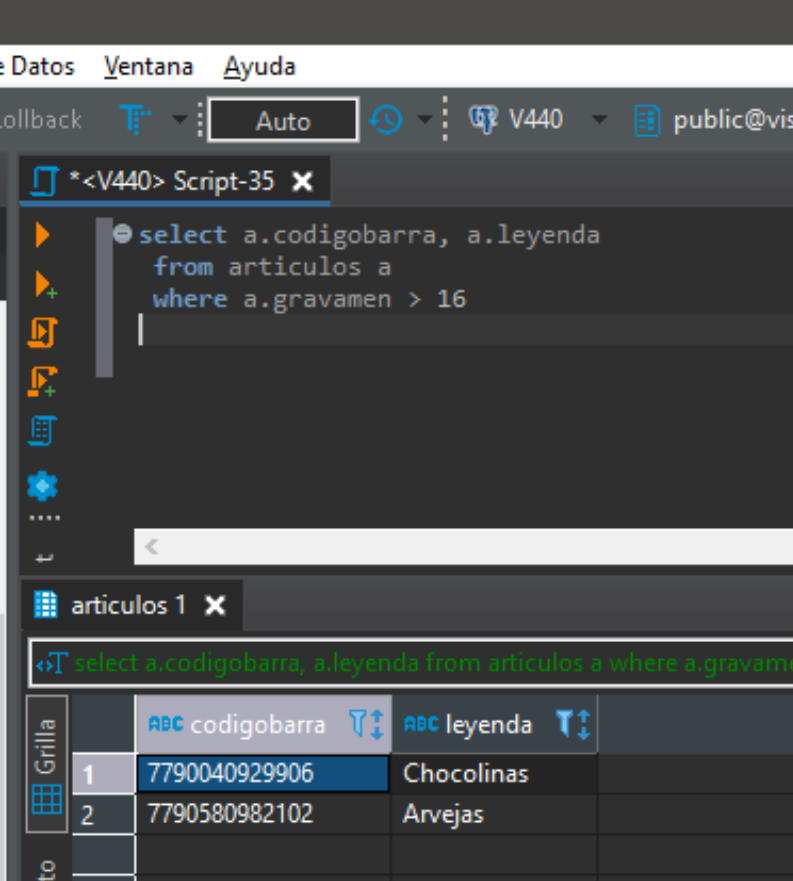
```
select p.apellido_persona,  
       p.nombre_persona  
from personas p  
where p.limite_facturasvencidas = 1
```

Below the editor, there is a tab labeled 'personas 1'. It shows the results of the query in a table format. The table has two columns: 'apellido\_persona' and 'nombre\_persona'. The results are as follows:

	apellido_persona	nombre_persona
1	Zapata Icart	Ernesto Andrés
2	Martínez	Julián Pedro

# SQL. Cláusula WHERE

```
select a.codigobarra, a.leyenda  
from articulos a  
where a.gravamen > 16
```

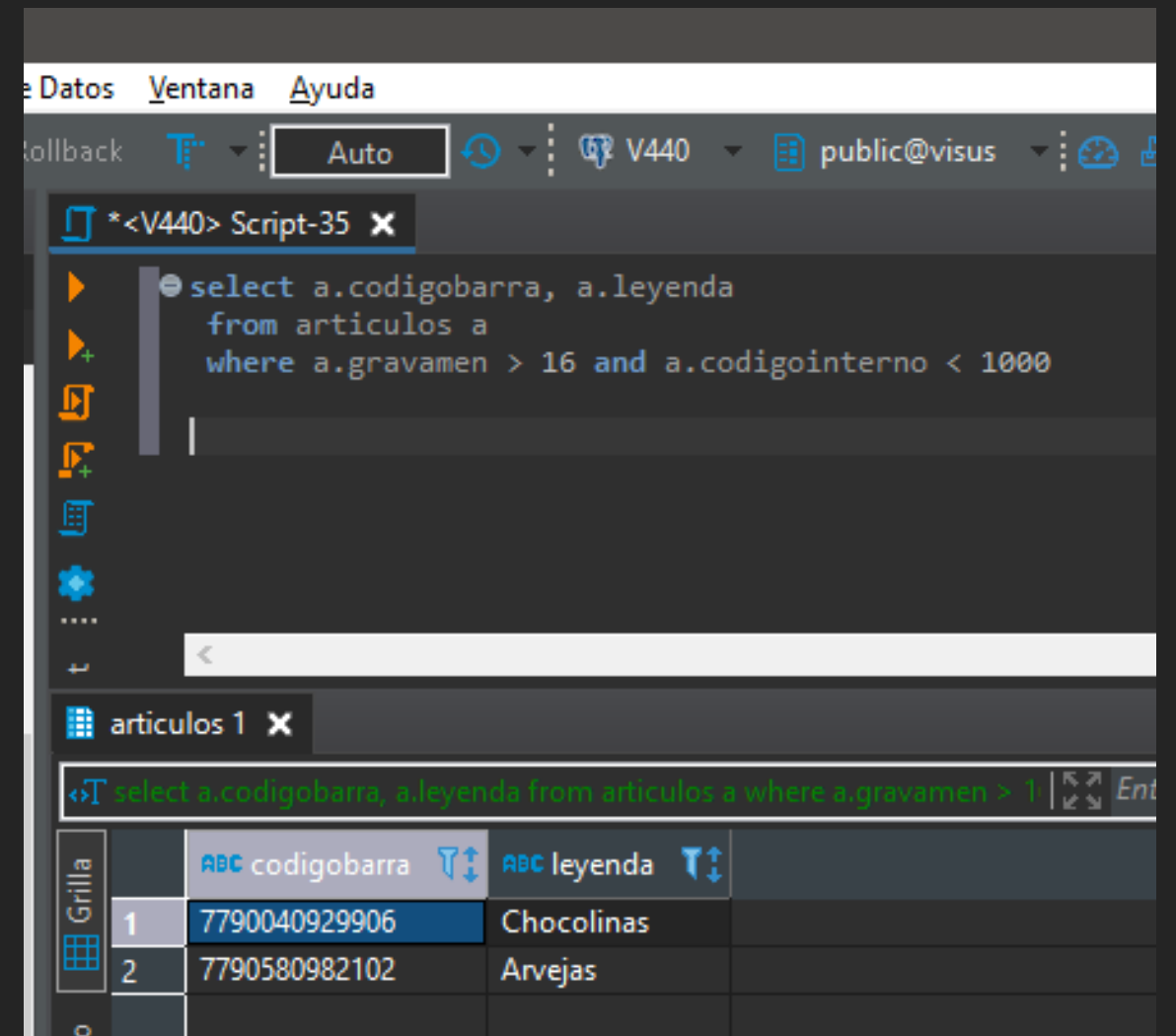


The screenshot shows a SQL IDE interface. At the top, there's a menu bar with 'Datos', 'Ventana', and 'Ayuda'. Below it, a toolbar contains icons for 'Rollback', 'Auto', a refresh icon, and a dropdown menu showing 'V440' and 'public@vis'. The main editor area displays a SQL query: `select a.codigobarra, a.leyenda from articulos a where a.gravamen > 16`. Below the editor, a tab labeled 'articulos 1' is active, showing a table with the results of the query. The table has two columns: 'codigobarra' and 'leyenda'. The first two rows are highlighted in blue.

	ABC codigobarra	ABC leyenda
1	7790040929906	Chocolinas
2	7790580982102	Arvejas

# SQL. Cláusula WHERE

```
select a.codigobarra,  
a.leyenda  
  
from articulos a  
  
where a.gravamen > 16  
and a.codigointerno < 1000
```



# SQL. Cláusula WHERE

```
select a.codigobarra, a.leyenda  
from articulos a  
where a.gravamen > 16 and  
(a.existencia >= 50 or a.existencia <= 100)
```

The screenshot shows a SQL IDE interface. The top menu bar includes 'Datos', 'Ventana', and 'Ayuda'. Below the menu is a toolbar with icons for 'Auto', 'V440', and 'public@visus'. The main editor window displays the following SQL query:

```
select a.codigobarra, a.leyenda  
from articulos a  
where a.gravamen > 16 and  
      (a.existencia >= 50 or a.existencia <= 100)
```

Below the query editor, a tab labeled 'articulos 1' is active. The results are displayed in a table with the following columns: 'Grilla', 'ABC codigobarra', 'ABC leyenda', and an empty column. The data rows are as follows:

Grilla	ABC codigobarra	ABC leyenda	
1	7790040929906	Chocolinas	
2	7790580982102	Arvejas	



# SQL. Cláusula WHERE

```
select a.codigobarra, a.leyenda  
from articulos a  
where a.tipo_producto = 'Estandar' and  
(a.codigoempresa = 580)
```

The screenshot shows a SQL IDE interface. At the top, there's a menu bar with 'de Datos', 'Ventana', and 'Ayuda'. Below it is a toolbar with icons for Rollback, a text editor icon, a dropdown menu set to 'Auto', a refresh icon, a server icon labeled 'V440', and a user icon labeled 'public@visus'. The main editor area has a tab labeled '\*<V440> Script-35' and contains the following SQL query:

```
select a.codigobarra, a.leyenda  
from articulos a  
where a.tipo_producto = 'Estandar' and  
(a.codigoempresa = 580)
```

Below the editor, there's a tab labeled 'articulos 1'. The results are displayed in a table with the following columns: 'Grilla' (with a sub-column '1'), 'ABC codigobarra', 'ABC leyenda', and an empty column. The first row of data shows the value '7790580982102' for 'ABC codigobarra' and 'Arvejas' for 'ABC leyenda'.

Grilla	1	ABC codigobarra	ABC leyenda	
	1	7790580982102	Arvejas	

# SQL. Cláusula ORDER BY

---

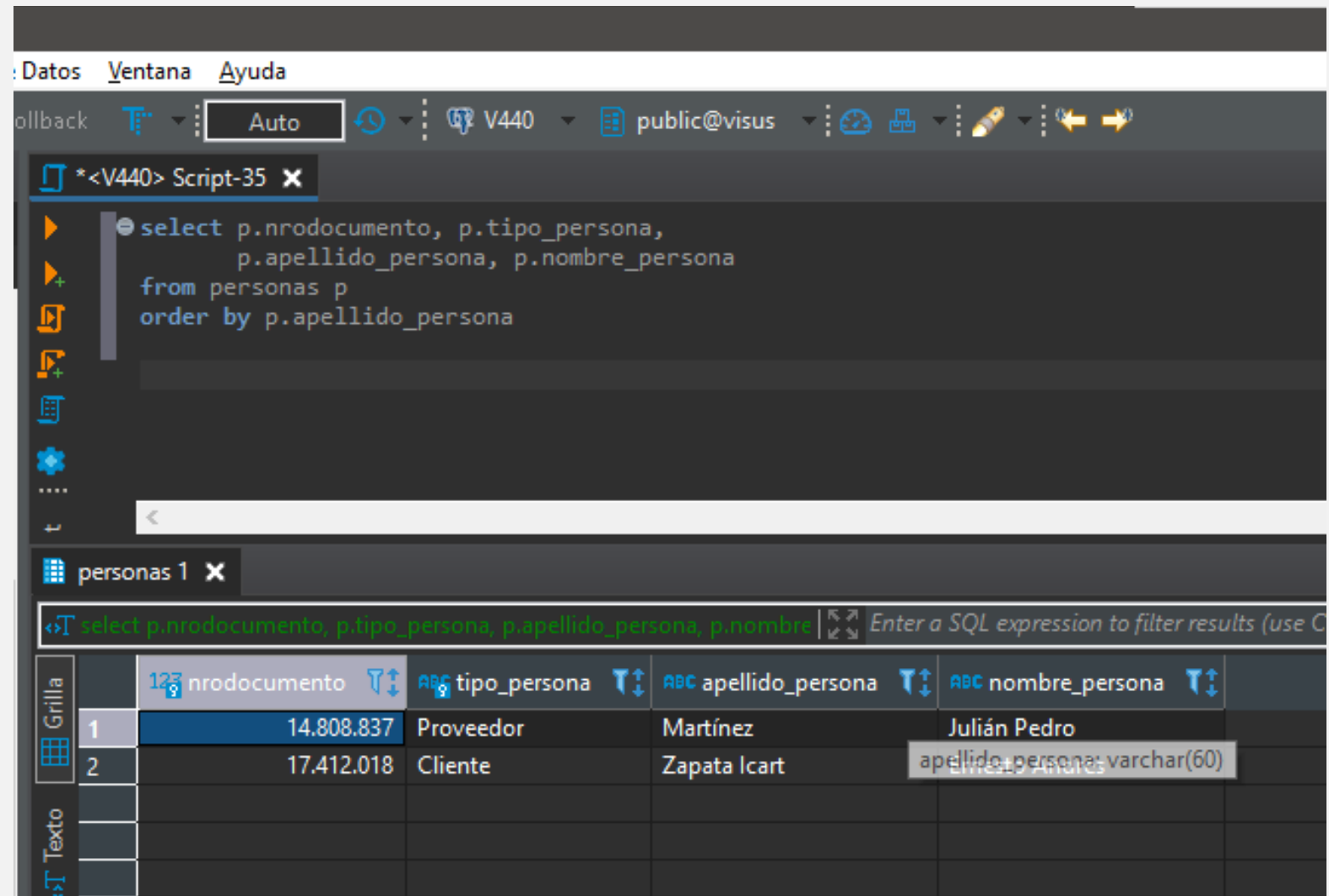
- La cláusula ORDER BY permite ordenar los resultados de una consulta.
  - SELECT columna1, columna2, ...
  - FROM nombredetabla
  - WHERE condición
  - ORDER BY campo1 [ASC|DESC], campo2 [ASC|DESC], ... ;

# SQL. Cláusula ORDER BY

—

- ASC especifica criterio de ordenación ascendente. Es la opción por defecto.
- DESC indica ordenación descendente.

# SQL. Cláusula ORDER BY



The screenshot shows a SQL IDE interface. The top menu bar includes 'Datos', 'Ventana', and 'Ayuda'. Below the menu is a toolbar with icons for 'Auto', 'V440', 'public@visus', and other functions. The main editor window displays a SQL query:

```
select p.nrodocumento, p.tipo_persona,  
       p.apellido_persona, p.nombre_persona  
from personas p  
order by p.apellido_persona
```

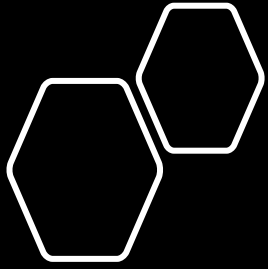
Below the query editor, a results pane titled 'personas 1' shows the output of the query. It includes a filter bar with the text 'Enter a SQL expression to filter results (use C)'. The results are displayed in a table with columns: 'nrodocumento', 'tipo\_persona', 'apellido\_persona', and 'nombre\_persona'. The first two rows are visible:

	nrodocumento	tipo_persona	apellido_persona	nombre_persona
1	14.808.837	Proveedor	Martínez	Julián Pedro
2	17.412.018	Cliente	Zapata lcart	apellido_persona: varchar(60)

- Obtenemos los datos de las personas ordenando el resultado por apellidos.

```
select p.nrodocumento, p.tipo_persona,  
       p.apellido_persona, p.nombre_persona  
from personas p  
order by p.apellido_persona
```





# SQL. Cláusula ORDER BY

```
select p.nrodocumento,  
p.tipo_persona,  
p.apellido_persona,  
p.nombre_persona
```

```
from personas p
```

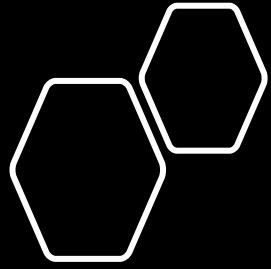
```
order by  
p.tipo_persona desc,
```

```
p.apellido_persona asc
```

- Ordenamos el resultado por tipo de persona y luego, por apellido

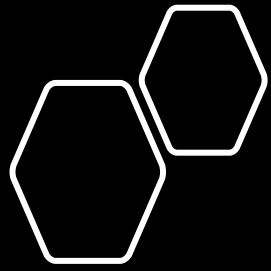
The screenshot shows a SQL IDE window with a menu bar (Datos, Ventana, Ayuda) and a toolbar. The main editor displays a SQL query: `select p.nrodocumento, p.tipo_persona, p.apellido_persona, p.nombre_persona from personas p order by p.tipo_persona desc, p.apellido_persona asc`. Below the editor, a results pane titled 'personas 1' shows a table with 5 columns: 'nrodocumento', 'tipo\_persona', 'apellido\_persona', and 'nombre\_persona'. The table contains two rows of data. The first row has '14.808.837' for nrodocumento, 'Proveedor' for tipo\_persona, 'Martínez' for apellido\_persona, and 'Julián Pedro' for nombre\_persona. The second row has '17.412.018' for nrodocumento, 'Cliente' for tipo\_persona, 'Zapata Icart' for apellido\_persona, and 'Ernesto Andrés' for nombre\_persona. The table is displayed in a grid view with column headers and row numbers.

	nrodocumento	tipo_persona	apellido_persona	nombre_persona
1	14.808.837	Proveedor	Martínez	Julián Pedro
2	17.412.018	Cliente	Zapata Icart	Ernesto Andrés



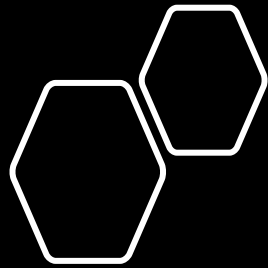
# SQL. Cláusula ORDER BY

- También se puede especificar la ordenación por un número.
- Este número indica la posición de la columna a la derecha de SELECT por la que se quiere ordenar el resultado
- `SELECT * FROM personas ORDER BY 2 ASC, 3 DESC;`



# SQL. Cláusulas DISTINCT Y ALL

- DISTINCT recupera las filas que son distintas.
- ALL recupera todas las filas aunque algunas estén repetidas. Es la opción por defecto.
- Ambas palabras vienen detrás de SELECT



# DISTINCT Y ALL

Base de Datos Ventana Ayuda

Rollback Auto V440 public@

\*<V440> Script-35 x

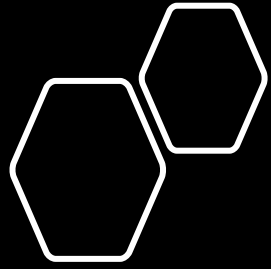
```
select distinct l.codigopostal
from localidades l
order by l.codigopostal
```

localidades 1 x

```
select distinct l.codigopostal from localidades l order by l.cod
```

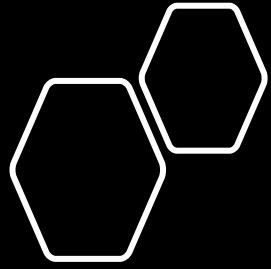
Grilla	ABC codigopostal
1	
2	1647
3	1649
4	2100
5	2805
6	2820
7	2821
8	2823
9	2824
10	2826
11	2828





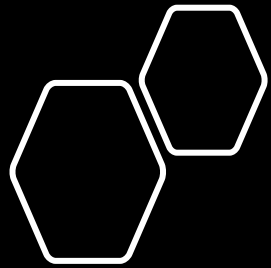
# SQL. Test de comparación

- Condición de búsqueda más utilizada en la cláusula WHERE
- Compara el valor de una expresión con otra.
- expresión operador expresión
- Operadores: <, >, >=, <=, !=, <>



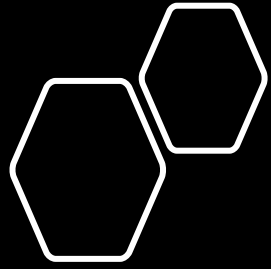
# SQL. Operador LIKE

- El operador LIKE permite realizar comparaciones con cadenas de caracteres.
- WHERE campo LIKE 'caracteres';
- \*: cualquier cadena de 0 o más caracteres
- \_: representa un carácter cualquiera



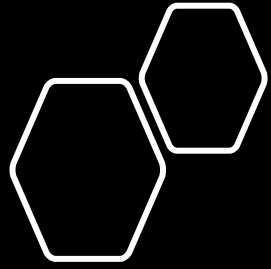
# SQL. Operador LIKE

- Empleados cuyo apellido comience por la letra A.
- `SELECT * FROM emple WHERE apellido LIKE 'A%';`
- Empleados cuyo apellido termine por la letra S.
- `SELECT * FROM emple WHERE apellido LIKE '%S';`



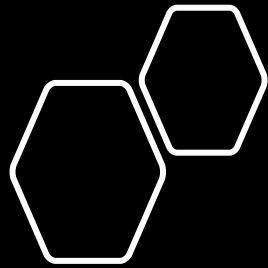
# SQL. Operador LIKE

- Empleados cuyo apellido tenga una M en cualquier posición.
- `SELECT * FROM emple WHERE apellido LIKE '%M%';`
- Empleados cuyo apellido tenga una M en la tercera posición.
- `SELECT * FROM emple WHERE apellido LIKE '__M%';`



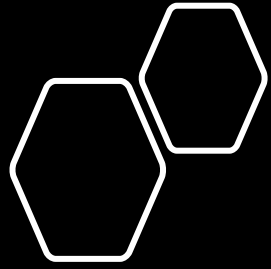
# SQL. Operador LIKE

- Empleados cuyo apellido tenga 3 caracteres y termine en M.
- `SELECT * FROM emple WHERE apellido LIKE '__M';`
- Empleados cuyo apellido tenga 2 caracteres y empiece por la letra M.
- `SELECT * FROM emple WHERE apellido LIKE 'M_';`



# SQL. IS NULL

- El campo de una fila es NULL si no contiene ningún valor.
- Para comprobar si un campo tiene un valor nulo utilizamos la expresión IS NULL.
- `columna IS NULL`



# SQL. IS NULL

- Localidades que no tienen Código postal

*select \* from  
localidades l*

*where  
l.codigopostal is  
null*

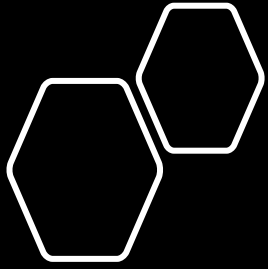
The screenshot shows a SQL client window with a menu bar (Datos, Ventana, Ayuda) and a toolbar. The main editor displays the following SQL query:

```
select * from localidades l  
where l.codigopostal is null
```

Below the editor, a tab labeled 'localidades 1' is active, showing a table with the following data:

	idlocalidad	nombre	codigopostal	provincia
1	722	8° DISTRITO	[NULL]	AR-E (Entre Ríos)





# SQL. IS NULL

- Medicamentos que tienen alguna observación

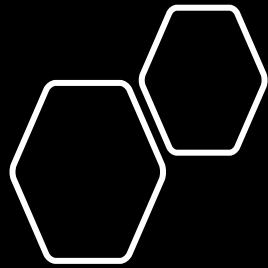
```
select * from  
medicamentos m  
  
where  
m.observaciones  
is not null
```

The screenshot shows a SQL client window with a menu bar (Datos, Ventana, Ayuda) and a toolbar. The main area displays a query in a script editor:

```
select * from medicamentos m  
where m.observaciones is not null
```

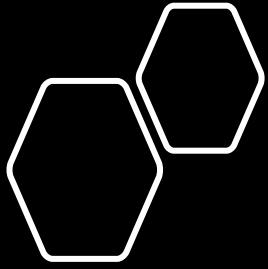
Below the script editor, a results grid titled 'medicamentos 1' is shown. It contains one row of data:

	idmedicamento	codigo interno	nombre comercial	droga	observaciones
1	2	T14295	TETAVAX	anatoxina tetánica	Temperatura= -10



# SQL. Operador BETWEEN

- Comprueba si un valor está comprendido en un rango de valores.
- <expresión> BETWEEN valor\_inicial AND valor\_final
- Empleados que tengan un salario entre 1000 y 2000 euros.
- `SELECT * FROM emple WHERE salario BETWEEN 1000 AND 2000;`



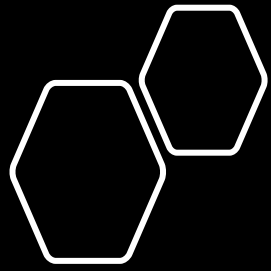
# SQL. Operador BETWEEN

`select * from  
reactivos r`

`where  
r.cantdeterminacione  
s between 2 and 8`

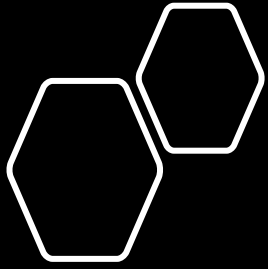
The screenshot shows a SQL IDE interface. The top menu bar includes 'File', 'Datos', 'Ventana', and 'Ayuda'. The toolbar contains icons for 'Rollback', 'Auto', a refresh icon, a connection icon labeled 'V440', a user icon labeled 'public@databalthus', and other standard database tools. The main editor window displays a SQL query: `select * from reactivos r where r.cantdeterminaciones between 2 and 8`. Below the editor, a results pane titled 'reactivos 1' shows the query results in a table format. The table has columns: 'idreactivo', 'identfab', 'nombre', 'fechavencimiento', 'nrolote', 'cantdeterminaciones', and 'contenido'. The first two rows are visible, corresponding to the filtered results.

	idreactivo	identfab	nombre	fechavencimiento	nrolote	cantdeterminaciones	contenido
1	3	3	ABIA MAX	2020-01-19	7444	3	Ninguno
2	2	421	Reactivo1	2020-09-13	2121	8	



# SQL. Operador IN

- Permite comprobar si una expresión pertenece a un conjunto de valores.
- <expresión> IN (lista de valores separados por comas)
- Obtener los empleados de los departamentos 10 ó 20.
- `SELECT * FROM emple WHERE dept_no IN (10,20);`



# SQL. Operador IN

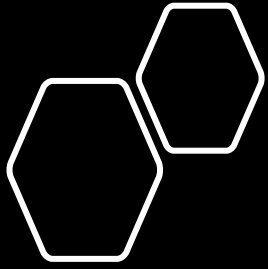
```
select * from reactivos r  
where  
r.cantdeterminaciones in  
(3,8)
```

The screenshot shows a SQL IDE interface. The top menu bar includes 'Archivo', 'Ventana', and 'Ayuda'. The toolbar contains icons for 'Back', 'Auto', 'Refresh', 'V440', 'public@databalthus', 'Execute', 'Print', 'Find', and 'Run'. The main editor window displays the following SQL query:

```
select * from reactivos r  
where r.cantdeterminaciones in (3,8)
```

Below the editor, a tab labeled 'reactivos 1' is active. The results pane shows the following table:

	idreactivo	identfab	nombre	fechavencimiento	nrolote	cantdeterminaciones
1	3	3	ABIA MAX	2020-01-19	7444	3
2	2	421	Reactivo1	2020-09-13	2121	8

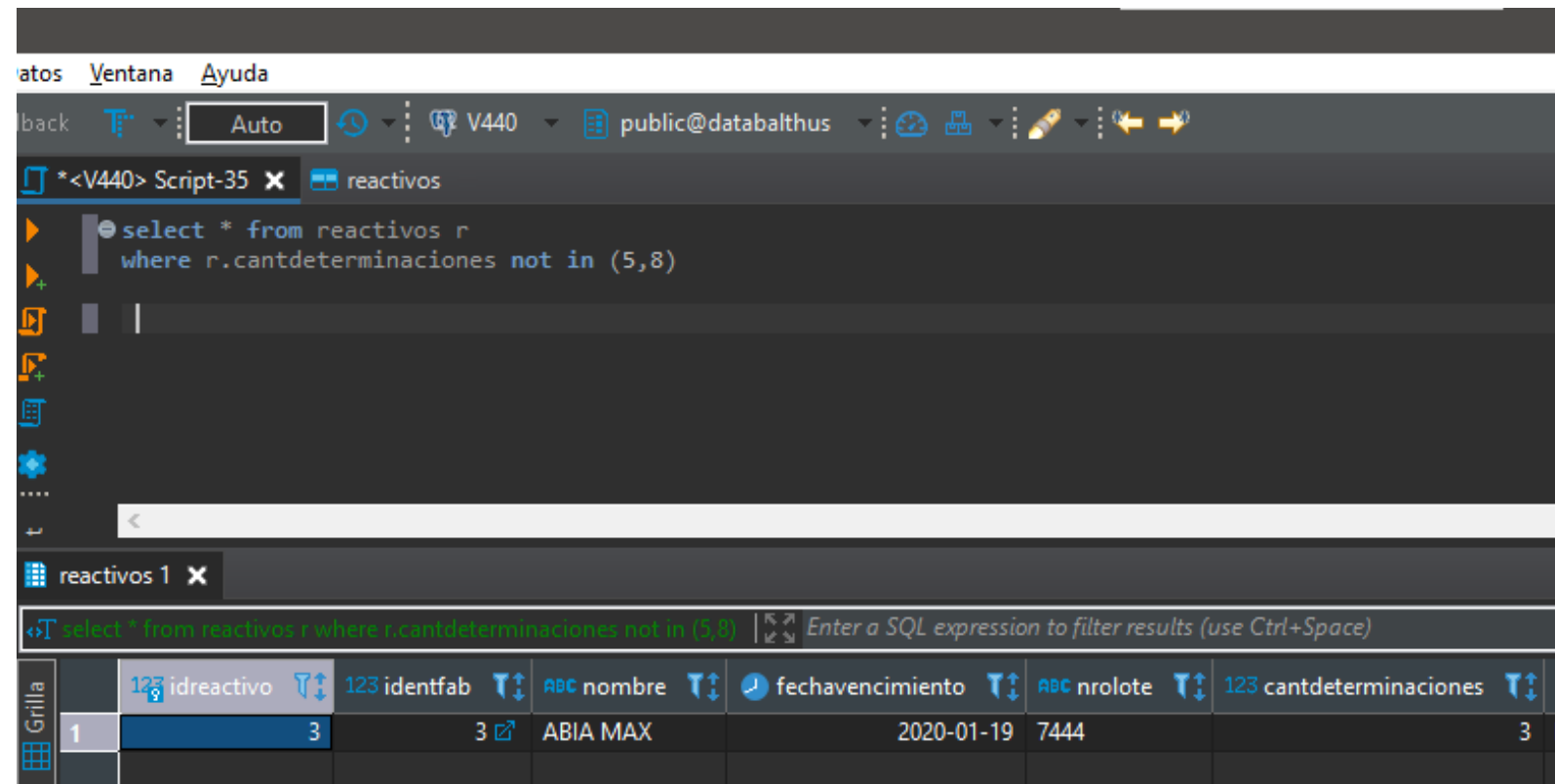


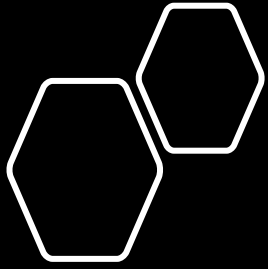
SQL.

Operador IN

`select * from  
reactivos r`

`where  
r.cantdeterminacione  
s not in (5,8)`





SQL.

Operador IN

`select * from staff s  
where s.titulo in  
('Técnico','Empleado')`

Enter a SQL expression to filter results (use Ctrl+Space)					
123 idcategoria	ABC apellido	ABC nombre	ABC sexo	fechanacimiento	ABC titulo
1	Zapata Icart	Ernesto Andrés	M	1964-10-01	Técnico
1	Galetto	Susana Beatriz	F	1962-02-24	Analista

DatosVentanaAyuda

IlbackAutoV440public@databalthus

\*<V440> Script-35reactivos