

Facultad:

Ingeniería

Carrera:

Ingeniería de sistemas

Tema del trabajo:

Consultas trabajo 5 admisión universidad libre

Estudiante:

Josue David Gelvez Jurado

Código:

02210131011

Docente:

Jonathan Rolando Rey Castillo

Materia:

Sistemas gestión base de datos

Cúcuta

2023

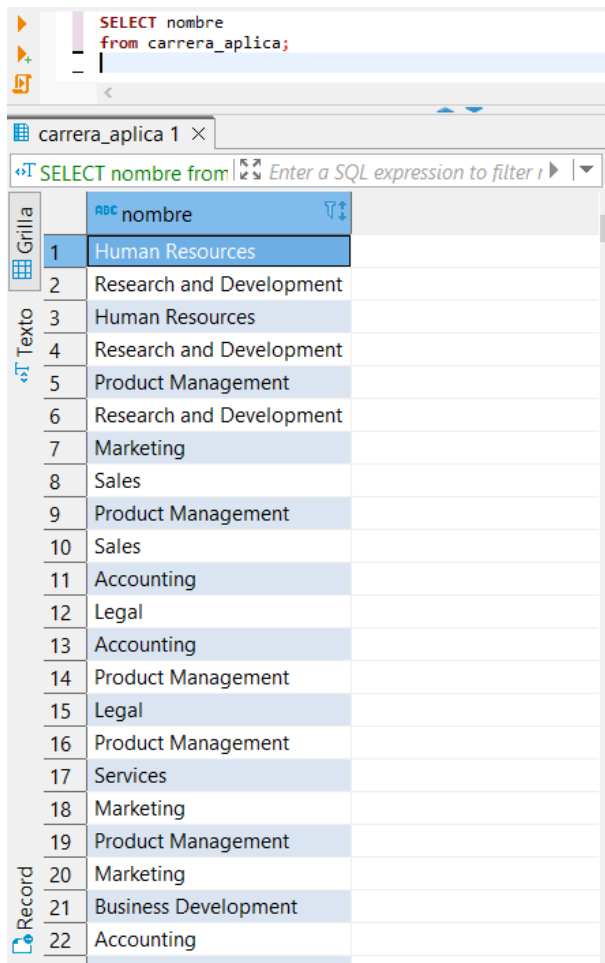
Consultas

1. Select columns

*Se necesita saber el nombre de todas las carreras.

SELECT nombre

from carrera_aplica;



The screenshot shows a database management interface. At the top, a SQL query is entered in a text area: `SELECT nombre
from carrera_aplica;`. Below the query, a tab labeled 'carrera_aplica 1' is active. Underneath the tab, a filter bar shows the query 'SELECT nombre from' followed by a dropdown menu. The main area displays a table with 22 rows and 1 column. The column is labeled 'nombre' and is sorted alphabetically. The rows contain the following values: Human Resources, Research and Development, Human Resources, Research and Development, Product Management, Research and Development, Marketing, Sales, Product Management, Sales, Accounting, Legal, Accounting, Product Management, Legal, Product Management, Services, Marketing, Product Management, Marketing, Business Development, and Accounting.

	nombre
1	Human Resources
2	Research and Development
3	Human Resources
4	Research and Development
5	Product Management
6	Research and Development
7	Marketing
8	Sales
9	Product Management
10	Sales
11	Accounting
12	Legal
13	Accounting
14	Product Management
15	Legal
16	Product Management
17	Services
18	Marketing
19	Product Management
20	Marketing
21	Business Development
22	Accounting

* Se desea conocer todos los nombres de las facultades de la tabla facultad.

SELECT nombre

from facultad

;

SELECT nombre
from facultad

facultad 1 x

SELECT nombre from

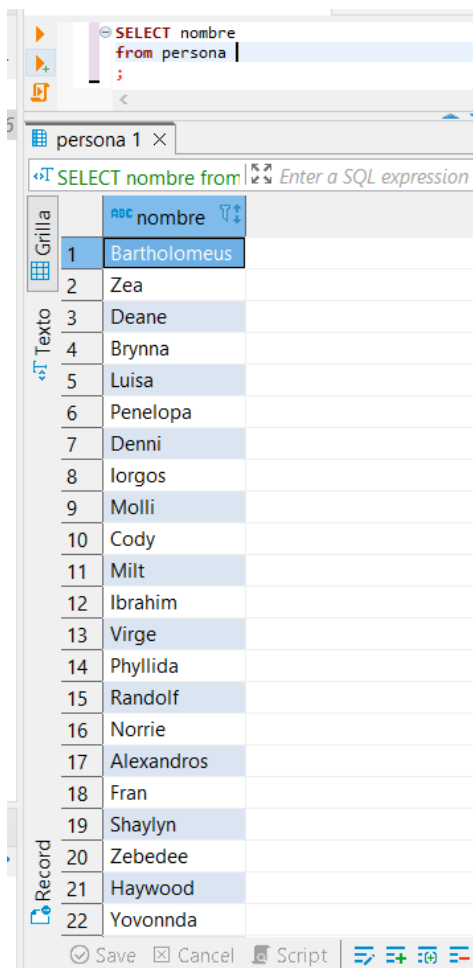
Grilla	nombre
1	Research and Development
2	Business Development
3	Support
4	Marketing
5	Sales
6	Accounting
7	Support
8	Product Management
9	Business Development
10	Marketing
11	Business Development
12	Human Resources
13	Engineering
14	Services
15	Marketing
16	Business Development
17	Marketing
18	Engineering
19	Engineering
20	Services
21	Engineering
22	Accounting

*se desea conocer los nombres de todas las personas de la tabla persona

SELECT nombre

from persona

;



* se desea conocer los apellidos de todas las personas de la tabla persona.

SELECT apellido

from persona

;

SELECT apellido
from persona
;

persona 1 x

SELECT apellido from Enter a SQL expression

Grilla	apellido
1	Raoult
2	Keable
3	Kyte
4	Galbreth
5	De Roos
6	Dougal
7	Keely
8	Brookhouse
9	Cluett
10	Stanyon
11	Condy
12	Wyley
13	Alyokhin
14	Cochet
15	Jessen
16	Cobden
17	Kleisle
18	Boyes
19	Winsiowiecki
20	Archley
21	Clapperton
22	Pygott

Record

Save Cancel Script

* se desea conocer todas las fechas de los registros de la tabla registro.

SELECT fecha_rregis
from registro
;

registro 1 x

SELECT fecha_rregis | Enter a SQL express

	fecha_rregis
1	2022-09-15
2	2022-04-22
3	2022-04-05
4	2023-02-27
5	2022-05-18
6	2022-03-12
7	2022-09-04
8	2023-02-23
9	2022-12-18
10	2022-11-24
11	2023-01-13
12	2022-12-20
13	2022-03-22
14	2022-11-07
15	2022-11-19
16	2022-03-22
17	2022-03-05
18	2023-02-12
19	2023-02-20
20	2022-09-29
21	2022-10-06
22	2022-10-02

Save Cancel Script

2.Select distinct 1 columna

*se necesita saber si en los nombres de las carreras hay valores duplicados

SELECT DISTINCT nombre

from carrera_aplica;

<pre>SELECT DISTINCT nombre from carrera_aplica;</pre>																											
carrera_aplica 1 ×																											
<pre>SELECT DISTINCT no</pre> <i>Enter a SQL express</i>																											
Grilla	<table> <tr> <th>ABC</th><th>nombre</th></tr> <tr><td>1</td><td>Human Resources</td></tr> <tr><td>2</td><td>Research and Development</td></tr> <tr><td>3</td><td>Product Management</td></tr> <tr><td>4</td><td>Marketing</td></tr> <tr><td>5</td><td>Sales</td></tr> <tr><td>6</td><td>Accounting</td></tr> <tr><td>7</td><td>Legal</td></tr> <tr><td>8</td><td>Services</td></tr> <tr><td>9</td><td>Business Development</td></tr> <tr><td>10</td><td>Training</td></tr> <tr><td>11</td><td>Support</td></tr> <tr><td>12</td><td>Engineering</td></tr> </table>	ABC	nombre	1	Human Resources	2	Research and Development	3	Product Management	4	Marketing	5	Sales	6	Accounting	7	Legal	8	Services	9	Business Development	10	Training	11	Support	12	Engineering
ABC	nombre																										
1	Human Resources																										
2	Research and Development																										
3	Product Management																										
4	Marketing																										
5	Sales																										
6	Accounting																										
7	Legal																										
8	Services																										
9	Business Development																										
10	Training																										
11	Support																										
12	Engineering																										
Texto																											

*se desea conocer los nombres no duplicados de la tabla persona.

```
SELECT DISTINCT nombre
```

```
from persona
```

```
;
```

SELECT DISTINCT nombre
from persona
;

persona 1 x

SELECT DISTINCT noi Enter a SQL expression to filter i

Grilla	nombre
912	Barnard
913	Harbert
914	William
915	Kurt
916	Lezlie
917	Aubry
918	Von
919	Leonardo
920	Rosemaria
921	Leisha
922	Kassey
923	Harvey
924	Derrick
925	Nathanael
926	Frederic
927	Bo
928	Gaelan
929	Ernest
930	Eleanor
931	Raul
932	Donica

Record

*Se desea conocer todos os nombre diferentes de la tabla usuario

SELECT DISTINCT nombre

from usuario

;

SELECT DISTINCT nombre
from usuario
;

usuario 1 x

SELECT DISTINCT no

Grilla

Texto

Record

	nombre
1	Silva
2	Willy
3	Ivory
4	Chelsey
5	Valentina
6	Luke
7	Abbey
8	Orsa
9	Culley
10	Jobina
11	Jaymee
12	Zane
13	Berkley
14	Emmanuel
15	Lutero
16	Byrom
17	Rodolphe
18	Britta
19	Garrott
20	Aland
21	Salvador
22	Linnell

Save Cancel Script

* se desea conocer las fechas no duplicadas de la tabla admision

SELECT DISTINCT fecha_admi

from admision

;

SELECT DISTINCT fecha_admi
from admision
;

admission 1 x

SELECT DISTINCT fec Enter a SQL expression to filter

	fecha_admi
1	2022-12-23
2	2022-03-09
3	2022-05-27
4	2022-06-14
5	2022-03-27
6	2022-10-27
7	2022-09-04
8	2022-11-27
9	2022-09-29
10	2022-09-25
11	2022-07-20
12	2022-12-28
13	2022-09-01
14	2023-01-06
15	2022-05-03
16	2022-07-28
17	2022-08-28
18	2022-05-06
19	2022-06-06
20	2022-10-28
21	2022-03-26
22	2022-12-01

Save Cancel Script

*se desea conocer los precios no duplicados de la tabla matricula.

SELECT DISTINCT precio

from matricula

;

*<02210131011 2> Script-2 x 02210131011

```
SELECT DISTINCT precio
from matricula
;
```

matricula 1 x

SELECT DISTINCT pre Enter a SQL expression to filter

	123 precio
1	2.077.358
2	3.238.584
3	2.177.152
4	2.921.978
5	3.220.475
6	2.201.726
7	2.669.704
8	3.679.298
9	3.510.696
10	2.133.975
11	3.107.115
12	3.423.062
13	2.475.059
14	3.134.396
15	3.819.654
16	2.103.046
17	3.858.775
18	3.768.819
19	2.698.327
20	3.841.904
21	3.964.184
22	2.853.995

Save Cancel Script

200+ Rows: 1

200 row(s) fetched - 128ms, on 2023-03-05 at 19:27:50

3. Select distinct * columnas.

*se necesita saber si dentro de la tabla carrera_aplica hay valores duplicados en sus columnas.

SELECT DISTINCT *

from carrera_aplica

SELECT DISTINCT *
from carrera_aplica

carrera_aplica 1 x

SELECT DISTINCT * fr Enter a SQL expression to filter

	id	nombre	duracion	credito_tot	precio
1	1	Human Resources	5	105	4.215.577
2	2	Research and Dev	4	128	4.015.987,21
3	3	Human Resources	5	114	4.720.733,16
4	4	Research and Dev	5	128	4.797.209,01
5	5	Product Managen	4	100	3.032.847
6	6	Research and Dev	4	157	3.470.706
7	7	Marketing	4	104	4.501.809
8	8	Sales	6	159	4.301.638,12
9	9	Product Managen	6	138	4.096.155,26
10	10	Sales	4	162	3.526.839,31
11	11	Accounting	5	127	3.556.069,58
12	12	Legal	4	105	4.115.955,76
13	13	Accounting	4	112	4.782.468
14	14	Product Managen	5	104	4.334.619,44
15	15	Legal	5	142	3.679.908,79
16	16	Product Managen	5	136	4.436.010
17	17	Services	6	115	3.947.311
18	18	Marketing	4	114	4.263.192,28
19	19	Product Managen	4	128	4.206.770,48
20	20	Marketing	5	117	3.094.533,03
21	21	Business Develop	6	110	4.373.031
22	22	Accounting	5	151	3.578.756,9

Save Cancel Script

* se necesita saber si dentro de la tabla matricula hay datos duplicados.

SELECT DISTINCT *

from matricula

;

SELECT DISTINCT *
from matricula
;

matricula 1 x

SELECT DISTINCT * fr Enter a SQL expression to filter

	era_aplica	precio	id_estudiante	creditos_semestre
1	1	2.077.358	1	11
2	2	3.238.584	2	7
3	3	2.177.152	3	2
4	4	2.921.978	4	17
5	5	3.220.475	5	6
6	6	2.201.726	6	9
7	7	2.669.704	7	13
8	8	3.679.298	8	2
9	9	3.510.696	9	7
10	10	2.133.975	10	19
11	11	3.107.115	11	19
12	12	3.423.062	12	2
13	13	2.475.059	13	10
14	14	3.134.396	14	16
15	15	3.819.654	15	15
16	16	2.103.046	16	18
17	17	3.858.775	17	11
18	18	3.768.819	18	16
19	19	2.698.327	19	13
20	20	3.841.904	20	20
21	21	3.964.184	21	19

Save Cancel Script

*se requiere saber si en la tabla persona hay valores duplicados.

SELECT DISTINCT *

from persona

;

SELECT DISTINCT *
from persona

persona 1 x

SELECT DISTINCT * fr Enter a SQL expression to filter

	id	nombre	apellido	direccion	telefono
1	1	Bartholomeus	Raoult	7274 Roxbury Hill	+46 678
2	2	Zea	Keable	98210 Darwin Center	+420 59
3	3	Deane	Kyte	000 Waxwing Trail	+7 299 9
4	4	Brynna	Galbreth	0280 Farragut Plaza	+30 268
5	5	Luisa	De Roos	28 Lillian Lane	+63 451
6	6	Penelopa	Dougal	9699 Center Circle	+7 319 9
7	7	Denni	Keely	685 Glacier Hill Plaza	+7 636 8
8	8	Iorgos	Brookhouse	8941 Emmet Circle	+963 19
9	9	Molli	Cluett	94 Warbler Circle	+672 77
10	10	Cody	Stanyon	19660 Merry Court	+1 530 2
11	11	Milt	Condy	91923 Stuart Alley	+63 783
12	12	Ibrahim	Wyley	9 Briar Crest Avenue	+86 228
13	13	Virge	Alyokhin	5 Anhalt Parkway	+64 793
14	14	Phyllida	Cochet	45515 Lillian Pass	+62 959
15	15	Randolf	Jessen	74021 Shelley Way	+351 35
16	16	Norrie	Cobden	14754 Dryden Road	+51 993
17	17	Alexandros	Kleisle	5 Nancy Crossing	+86 196
18	18	Fran	Boyes	48 Vahlen Hill	+48 550
19	19	Shaylyn	Winsiowiecki	27 Carberry Junction	+98 657
20	20	Zebedee	Archley	5 Muir Lane	+86 352
21	21	Haywood	Clapperton	073 Michigan Pass	+420 61

*Se requiere saber si en la tabla estudiante hay valores duplicados

SELECT DISTINCT *

from estudiante

;

SELECT DISTINCT *
from estudiante

estudiante 1

SELECT DISTINCT * fr Enter a SQL expression to filter

	id	id_admision	id_usuario	nombre	avatar
1	1	1	1	Sherill	https://rc
2	2	2	2	Isaac	https://rc
3	3	3	3	Dannel	https://rc
4	4	4	4	Archie	https://rc
5	5	5	5	Jo-anne	https://rc
6	6	6	6	Jacquelynn	https://rc
7	7	7	7	Yolande	https://rc
8	8	8	8	Wayne	https://rc
9	9	9	9	Deloria	https://rc
10	10	10	10	Louis	https://rc
11	11	11	11	Monique	https://rc
12	12	12	12	Llewellyn	https://rc
13	13	13	13	Adara	https://rc
14	14	14	14	Valaree	https://rc
15	15	15	15	Rafaellle	https://rc
16	16	16	16	Davon	https://rc
17	17	17	17	Constantin	https://rc
18	18	18	18	Berri	https://rc
19	19	19	19	Ely	https://rc
20	20	20	20	Eolanda	https://rc
21	21	21	21	Murial	https://rc

* se requiere conocer si en la tabla notas hay datos duplicados.

SELECT DISTINCT *

from notas

;

SELECT DISTINCT *
from notas

notas 1

SELECT DISTINCT * fr Enter a SQL expression to filter

	id	id_parcial	nota_participacion	mota_asistencia
1	1	1	3,78	1,65
2	2	2	2,23	1,22
3	3	3	4,06	2,31
4	4	4	1,28	4,79
5	5	5	1,04	4,32
6	6	6	4,97	1,96
7	7	7	4,2	2,63
8	8	8	1,76	4,67
9	9	9	4,26	4,71
10	10	10	4,43	2,22
11	11	11	1,46	1,33
12	12	12	4,63	1,78
13	13	13	4,06	1,62
14	14	14	3,95	2,83
15	15	15	2,05	2,02
16	16	16	1,5	3,01
17	17	17	1,35	3,08
18	18	18	1,59	3,74
19	19	19	3,97	4,02
20	20	20	2,48	4,49
21	21	21	1,47	3,68

Save Cancel Script

1000

4. Select *

*se necesita revisar todos los datos de la tabla carrera_aplica.

SELECT
from carrera_aplica

carrera_aplica 1 x

SELECT * from carrer Enter a SQL expression to filter

	id	nombre	duracion	credito_tot	precio
1	1	Human Resources	5	105	4.215.577
2	2	Research and Dev	4	128	4.015.987,21
3	3	Human Resources	5	114	4.720.733,16
4	4	Research and Dev	5	128	4.797.209,01
5	5	Product Managen	4	100	3.032.847
6	6	Research and Dev	4	157	3.470.706
7	7	Marketing	4	104	4.501.809
8	8	Sales	6	159	4.301.638,12
9	9	Product Managen	6	138	4.096.155,26
10	10	Sales	4	162	3.526.839,31
11	11	Accounting	5	127	3.556.069,58
12	12	Legal	4	105	4.115.955,76
13	13	Accounting	4	112	4.782.468
14	14	Product Managen	5	104	4.334.619,44
15	15	Legal	5	142	3.679.908,79
16	16	Product Managen	5	136	4.436.010
17	17	Services	6	115	3.947.311
18	18	Marketing	4	114	4.263.192,28
19	19	Product Managen	4	128	4.206.770,48
20	20	Marketing	5	117	3.094.533,03
21	21	Business Develop	6	110	4.373.031
22	22	Accounting	5	151	3.578.756,9

*se requiere conocer todos los datos de la tabla notas.

SELECT *

from notas

;

SELECT *
from notas

notas 1 x

SELECT * from notas Enter a SQL expression to filter

	id	id_parcial	nota_participacion	mota_asistencia
1	1	1	3,78	1,65
2	2	2	2,23	1,22
3	3	3	4,06	2,31
4	4	4	1,28	4,79
5	5	5	1,04	4,32
6	6	6	4,97	1,96
7	7	7	4,2	2,63
8	8	8	1,76	4,67
9	9	9	4,26	4,71
10	10	10	4,43	2,22
11	11	11	1,46	1,33
12	12	12	4,63	1,78
13	13	13	4,06	1,62
14	14	14	3,95	2,83
15	15	15	2,05	2,02
16	16	16	1,5	3,01
17	17	17	1,35	3,08
18	18	18	1,59	3,74
19	19	19	3,97	4,02
20	20	20	2,48	4,49
21	21	21	1,47	3,68

Save Cancel Script 1000

*se requiere conocer todos los datos de la tabla materia.

SELECT *

from materia

;

SELECT *

from materia

;

materia 1 x

SELECT * from mater Enter a SQL expression to filter

	id	id_matricula	id_tipo_materia	nombre
1	1	1	1	Geological Engineer
2	2	2	2	Senior Editor
3	3	3	3	Research Assistant III
4	4	4	4	Speech Pathologist
5	5	5	5	Financial Advisor
6	6	6	6	Professor
7	7	7	7	Senior Quality Engineer
8	8	8	8	Safety Technician I
9	9	9	9	Senior Financial Analyst
10	10	10	10	Research Assistant I
11	11	11	11	Help Desk Operator
12	12	12	12	Occupational Therapist
13	13	13	13	Software Test Engineer
14	14	14	14	Nuclear Power Engineer
15	15	15	15	Professor
16	16	16	16	Data Coordinator
17	17	17	17	Media Manager I
18	18	18	18	VP Quality Control
19	19	19	19	Electrical Engineer
20	20	20	20	Web Developer IV
21	21	21	21	Statistician I

*se requiere conocer todos los datos de la tabla matricula.

SELECT *

from matricula

;

SELECT *
from matricula

matricula 1 x

SELECT * from matric Enter a SQL expression to filter

	id	id_admision	id_carrera_aplica	precio	id
1	1	1	1	2.077.358	
2	2	2	2	3.238.584	
3	3	3	3	2.177.152	
4	4	4	4	2.921.978	
5	5	5	5	3.220.475	
6	6	6	6	2.201.726	
7	7	7	7	2.669.704	
8	8	8	8	3.679.298	
9	9	9	9	3.510.696	
10	10	10	10	2.133.975	
11	11	11	11	3.107.115	
12	12	12	12	3.423.062	
13	13	13	13	2.475.059	
14	14	14	14	3.134.396	
15	15	15	15	3.819.654	
16	16	16	16	2.103.046	
17	17	17	17	3.858.775	
18	18	18	18	3.768.819	
19	19	19	19	2.698.327	
20	20	20	20	3.841.904	
21	21	21	21	3.964.184	

*se requiere conocer todos los datos de la tabla tipo_materia.

SELECT *
from tipo_materia

tipo_materia 1 x

SELECT * from tipo_n Enter a SQL expression to filter

	id	nombre
1	1	Financial Analyst
2	2	Help Desk Operator
3	3	Registered Nurse
4	4	Account Representative IV
5	5	Cost Accountant
6	6	Budget/Accounting Analyst I
7	7	Pharmacist
8	8	Financial Analyst
9	9	Paralegal
10	10	Data Coordinator
11	11	Technical Writer
12	12	Marketing Assistant
13	13	Sales Associate
14	14	Web Developer IV
15	15	Marketing Manager
16	16	Teacher
17	17	General Manager
18	18	Sales Associate
19	19	Electrical Engineer
20	20	Analog Circuit Design manager
21	21	Pharmacist
22	22	Recruiter

Save Cancel Script

5. Order by

*Se necesita saber el valor de los precios de forma ascendente de la tabla carrera_aplica.

SELECT precio

from carrera_aplica

order by precio ASC

SELECT precio
from carrera_aplica
order by precio ASC

carrera_aplica 1 x

SELECT precio from c Enter a SQL expression to filter r

Grilla	123	precio
1	3.000.774,18	
2	3.001.340,53	
3	3.005.165,63	
4	3.007.839	
5	3.008.252,47	
6	3.011.043	
7	3.011.104,74	
8	3.019.741	
9	3.019.973,99	
10	3.024.811,85	
11	3.026.264	
12	3.032.847	
13	3.033.795,85	
14	3.040.345,88	
15	3.041.473,59	
16	3.041.526	
17	3.043.655,1	
18	3.043.791,05	
19	3.044.607,7	
20	3.044.822,81	
21	3.047.691,87	
22	3.049.708,13	

* Se requiere ordenar de manera ascendente el nombre de las materias

SELECT nombre

from materia

order by nombre ASC

SELECT nombre
from materia
order by nombre ASC
;

materia 1 x

SELECT nombre from Enter a SQL expression to filter

	nombre
1	Account Coordinator
2	Account Coordinator
3	Account Executive
4	Account Executive
5	Account Executive
6	Account Executive
7	Account Executive
8	Account Executive
9	Account Executive
10	Account Executive
11	Account Representative I
12	Account Representative I
13	Account Representative II
14	Account Representative II
15	Account Representative II
16	Account Representative III
17	Account Representative III
18	Account Representative IV
19	Accountant I
20	Accountant I
21	Accountant I

Save Cancel Script

* Se quiere ordenar de manera descendente los precios de la tabla matricula

SELECT precio

from matricula m

order by precio DESC

;

SELECT precio
from matricula m
order by precio DESC ;

matricula 1 x

SELECT precio from r Enter a SQL expression to filter i

Grilla	123 precio
1	3.999.080
2	3.996.572
3	3.994.697
4	3.992.081
5	3.990.435
6	3.989.766
7	3.982.741
8	3.981.926
9	3.981.831
10	3.978.706
11	3.978.233
12	3.977.978
13	3.972.745
14	3.970.462
15	3.969.662
16	3.968.262
17	3.964.184
18	3.961.846
19	3.961.281
20	3.959.722
21	3.959.390

Save Cancel Script

1.000+ Rows: 0

* Se quiere ordenar de manera descendente los nombres de los grupos

SELECT nombre

from grupo g

order by nombre DESC

;

SELECT nombre
from grupo g
order by nombre DESC
;

matricula 1 x

SELECT precio from i Enter a SQL expression to filter i

	123 precio
1	3.999.080
2	3.996.572
3	3.994.697
4	3.992.081
5	3.990.435
6	3.989.766
7	3.982.741
8	3.981.926
9	3.981.831
10	3.978.706
11	3.978.233
12	3.977.978
13	3.972.745
14	3.970.462
15	3.969.662
16	3.968.262
17	3.964.184
18	3.961.846
19	3.961.281
20	3.959.722
21	3.959.390

Save Cancel Script

* Se quiere ordenar de manera ascendente las notas finales de la tabla notas

SELECT nota_final

from notas n

order by nota_final ASC

;

The screenshot shows a database query tool interface. At the top, a SQL query is entered in a blue box:

```
SELECT nota_final
from notas n
order by nota_final ASC
;
```

Below the query, the results are displayed in a grid view titled "notas 1". The grid has two columns: an index from 1 to 21 and a column labeled "nota_final". The data is sorted in ascending order of the final grade.

	nota_final
1	1
2	1,01
3	1,02
4	1,02
5	1,02
6	1,03
7	1,03
8	1,04
9	1,04
10	1,04
11	1,04
12	1,05
13	1,07
14	1,07
15	1,07
16	1,07
17	1,08
18	1,08
19	1,09
20	1,09
21	1,09

At the bottom of the interface, there is a status bar showing "1.000+ Rows: 1" and "1000 row(s) fetched - 370ms (1ms fetch), on 2023-03-05 at 19:53:1".

6. Order by *

*Se desea saber los datos conocer algunos datos de la tabla carrera_aplica de forma ascendente.

```
SELECT nombre, duracion, credito_total, precio
```

```
from carrera_aplica
```

```
order by nombre ASC , duracion ASC, credito_total ASC , precio ASC ;
```

SELECT nombre, duracion, credito_total, precio
from carrera_aplica
order by nombre ASC , duracion ASC, credito_total ASC , precio ASC ;

carrera_aplica 1 x

SELECT nombre, duracion, credito_total, precio

	nombre	duracion	credito_total	precio
1	Accounting	4	105	3.300.543,72
2	Accounting	4	112	3.859.485,74
3	Accounting	4	112	4.782.468
4	Accounting	4	112	4.836.553
5	Accounting	4	114	3.532.319,49
6	Accounting	4	116	3.391.876,4
7	Accounting	4	126	4.916.409,18
8	Accounting	4	132	3.092.108,07
9	Accounting	4	133	3.484.692,32
10	Accounting	4	134	3.540.342,08
11	Accounting	4	134	4.291.301
12	Accounting	4	136	4.999.295
13	Accounting	4	137	3.190.359,57
14	Accounting	4	137	4.484.025
15	Accounting	4	137	4.942.205
16	Accounting	4	142	4.600.970
17	Accounting	4	143	3.425.830
18	Accounting	4	149	4.091.298,62
19	Accounting	4	164	3.989.117
20	Accounting	4	164	4.711.748
21	Accounting	5	102	3.484.121,57
22	Accounting	5	102	3.554.281,94

*Se desea ordenar los datos nombre y creditos de la tabla materia de forma ascendente.

SELECT nombre , creditos

from materia m

order by nombre ASC, creditos ASC

;

SELECT nombre , creditos
from materia m
order by nombre ASC, creditos ASC

materia 1 x

SELECT nombre , cre Enter a SQL expression to filter.

	ABC nombre	123 creditos
1	Account Coordinator	1
2	Account Coordinator	1
3	Account Executive	1
4	Account Executive	1
5	Account Executive	2
6	Account Executive	2
7	Account Executive	2
8	Account Executive	2
9	Account Executive	2
10	Account Executive	3
11	Account Representative I	1
12	Account Representative I	1
13	Account Representative II	2
14	Account Representative II	3
15	Account Representative II	3
16	Account Representative III	1
17	Account Representative III	3
18	Account Representative IV	3
19	Accountant I	2
20	Accountant I	2
21	Accountant I	3

Save Cancel Script

*Se desea ordenar los datos nota_participacion y mota_asistencia de la tabla notas de forma descendente.

SELECT nota_participacion , mota_asistencia

from notas n

order by nota_participacion DESC , mota_asistencia desc

;

SELECT nota_participacion , mota_asistencia
from notas n
order by nota_participacion DESC , mota_asistencia desc
;

notas 1 x

SELECT nota_participacion | Enter a SQL expression to filter |

	123 nota_participacion	123 mota_asistencia
1	5	3,9
2	4,99	4,4
3	4,99	3,2
4	4,99	3,18
5	4,99	2,64
6	4,99	1,87
7	4,98	2,41
8	4,98	1,83
9	4,97	4,95
10	4,97	3,2
11	4,97	2,46
12	4,97	1,96
13	4,96	4,78
14	4,96	1,78
15	4,95	2,09
16	4,95	2,05
17	4,94	3,23
18	4,93	3,03
19	4,93	3,01
20	4,93	2,79
21	4,93	1,68

Save Cancel Script

* Se desea ordenar los datos nombres de la tabla estudiante de forma ascendente y saber de qué país son.

SELECT nombre , pais

from estudiante e

order by nombre DESC , pais

;

SELECT nombre , pais
from estudiante e
order by nombre DESC , pais
;

estudiante 1 x

SELECT nombre , pais Enter a SQL expression to filter

	asc nombre	asc pais
1	Zorina	Russia
2	Zondra	China
3	Zita	Gambia
4	Zebulon	Slovenia
5	Zea	Guatemala
6	Zarla	Brazil
7	Zandra	Sweden
8	Yuri	Indonesia
9	Yule	United States
10	Yolande	Sweden
11	Ynez	Angola
12	Ynes	Indonesia
13	Yardley	China
14	Yard	China
15	Yancey	Cameroon
16	Yance	Mexico
17	Wynny	China
18	Wylie	Malaysia
19	Wye	Angola
20	Wright	China
21	Wolf	Indonesia

Save Cancel Script

*Se desea ordenar los datos estado y fecha_admi de la taba admision de forma descendente.

SELECT estado , fecha_admi

from admision a

order by estado DESC

;

SQL Query Editor:

```
SELECT estado , fecha_admi
from admision a
order by estado DESC
;
```

admision 1 x

SQL Query:

```
SELECT estado , fecha_admi
```

Enter a SQL expression to filter

	estado	fecha_admi
1	true	2022-10-24
2	true	2022-10-02
3	true	2022-09-17
4	true	2022-06-20
5	true	2022-10-27
6	true	2022-08-09
7	true	2022-07-04
8	true	2022-09-23
9	true	2023-02-05
10	true	2023-01-13
11	true	2022-07-01
12	true	2022-12-12
13	true	2023-02-02
14	true	2022-09-01
15	true	2022-04-26
16	true	2023-02-17
17	true	2023-02-09
18	true	2022-12-30
19	true	2022-10-26
20	true	2022-11-19
21	true	2022-05-07

Save Cancel Script

7. Where predicados

*Se desea conocer los precios en carrera_aplica que sean mayores a 4500000

SELECT precio

from carrera_aplica

where precio > 4500000

SELECT precio
from carrera_aplica
where precio > 4500000

carrera_aplica 1 x

SELECT precio from c Enter a SQL expression to filter

Grilla	123 precio
1	4.720.733,16
2	4.797.209,01
3	4.501.809
4	4.782.468
5	4.599.724
6	4.683.380,65
7	4.595.877,25
8	4.876.661
9	4.804.728
10	4.837.268
11	4.702.884
12	4.587.379,01
13	4.916.409,18
14	4.679.679
15	4.928.448,28
16	4.632.143
17	4.897.464
18	4.758.096
19	4.745.538
20	4.684.125
21	4.672.895
22	4.550.889,05

Record

*Se desea conocer solo los precios que tengan un valor diferente a 4000000

SELECT precio

FROM matricula m

where precio <> 4000000

SELECT precio
FROM matricula m
where precio <> 4000000

matricula 1 x

SELECT precio FROM *Enter a SQL expression to filter*

	123 precio
1	2.077.358
2	3.238.584
3	2.177.152
4	2.921.978
5	3.220.475
6	2.201.726
7	2.669.704
8	3.679.298
9	3.510.696
10	2.133.975
11	3.107.115
12	3.423.062
13	2.475.059
14	3.134.396
15	3.819.654
16	2.103.046
17	3.858.775
18	3.768.819
19	2.698.327
20	3.841.904
21	3.964.184

Save Cancel Script

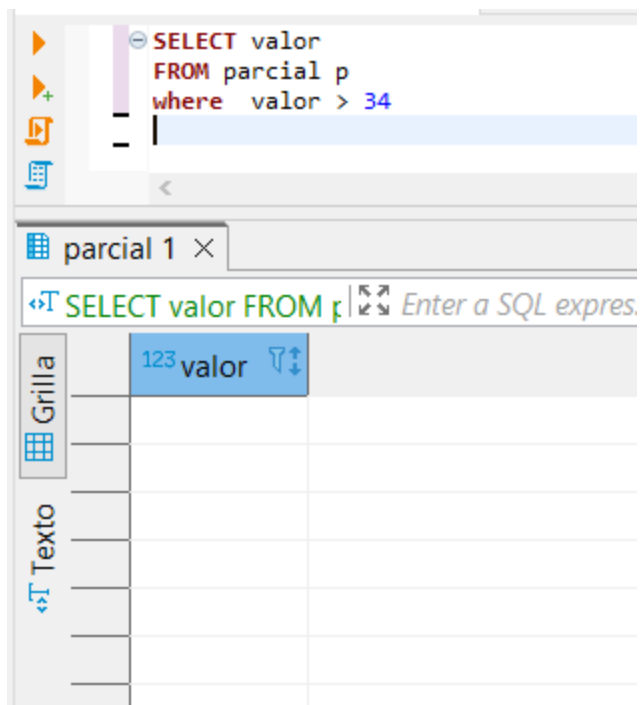
1.000+ Rows: 1

*Se desea conocer solos los valores mayores a 34 en la tabla parcial

SELECT valor

FROM parcial p

where valor > 34



*Se desea conocer solo las notas finales que sean mayores o iguales a 3 en la tabla notas

```
SELECT nota_final
```

```
FROM notas n
```

```
where nota_final >= 3
```

SELECT nota_final
FROM notas n
where nota_final >= 3

notas 1 x

SELECT nota_final FR Enter a SQL expression to filter

Grilla	123 nota_final
1	3,61
2	3,9
3	3,58
4	3,29
5	3,38
6	4,26
7	3,84
8	3,63
9	4,78
10	3,35
11	4,53
12	3,42
13	4,67
14	3,13
15	3
16	3,36
17	4,94
18	4,87
19	3,97
20	4,16
21	3,03

Record

Save Cancel Script

* Se desea conocer solo las notas finales que sean menores o iguales a 2 en la tabla notas

SELECT nota_final

FROM notas n

where nota_final <= 2

SELECT nota_final
FROM notas n
where nota_final <= 2

notas 1 x

SELECT nota_final FR Enter a SQL expression to

Grilla	123 nota_final
1	1,26
2	1,32
3	1,23
4	1,52
5	1,71
6	1,54
7	1,78
8	1,98
9	1,85
10	1,36
11	1,78
12	1,3
13	1,44
14	2
15	1,49
16	1,27
17	1,59
18	1,49
19	1,19
20	1,29
21	1,92

Record

Save Cancel Script

8. Where and or

* Se desea conocer los creditos de sean mayores a 120 y los precios que sean mayores a 3700000 de la tabla carrera_aplica

SELECT credito_total, precio

from carrera_aplica

where credito_total > 120 and precio > 4500000;

SELECT credito_total, precio
from carrera_aplica
where credito_total > 120 and precio > 4500000;

carrera_aplica 1 x

SELECT credito_total, Enter a SQL expression to filter i

Grilla	123 credito_total	123 precio
1	128	4.797.209,01
2	168	4.599.724
3	138	4.683.380,65
4	147	4.595.877,25
5	152	4.876.661
6	162	4.837.268
7	146	4.702.884
8	158	4.587.379,01
9	126	4.916.409,18
10	156	4.679.679
11	168	4.928.448,28
12	169	4.632.143
13	128	4.897.464
14	121	4.745.538
15	134	4.550.889,05
16	125	4.779.535,72
17	132	4.710.615
18	142	4.520.931,68
19	146	4.828.527
20	124	4.899.489,32
21	167	4.623.862,64
22	133	4.832.538

Record

*Se desea conocer las notas de participacion que sean mayores o iguales a 3 o las notas de asistencia que sean menores o iguales a 4 en la tabla notas.

SELECT nota_participacion, mota_asistencia

FROM notas n

where nota_participacion >=3 or mota_asistencia <= 4

SELECT nota_participacion, mota_asistencia
FROM notas n
where nota_participacion >=3 or mota_asistencia <= 4

notas 1 x

SELECT nota_participacion | Enter a SQL expression to filter |

	123 nota_participacion	123 mota_asistencia
1	3,78	1,65
2	2,23	1,22
3	4,06	2,31
4	4,97	1,96
5	4,2	2,63
6	4,26	4,71
7	4,43	2,22
8	1,46	1,33
9	4,63	1,78
10	4,06	1,62
11	3,95	2,83
12	2,05	2,02
13	1,5	3,01
14	1,35	3,08
15	1,59	3,74
16	3,97	4,02
17	1,47	3,68
18	4,44	2,31
19	2,68	3,28
20	3,19	3,65
21	2,56	3,81

Save Cancel Script

*Se desea conocer los pines validos entre 2022-08-05 y 2023-02-23

SELECT valido_inicio , valido_fin, id , estado

FROM pin p

where valido_inicio >'2022-08-05' and valido_fin <'2023-02-23'

SELECT valido_inicio , valido_fin, id , estado
FROM pin p
where valido_inicio >'2022-08-05' and valido_fin <'2023-02-23'

pin 1 x

SELECT valido_inicio Enter a SQL expression to filter

	valido_inicio	valido_fin	id	estado
1	2023-02-22	2022-06-22	2	true
2	2022-10-20	2022-11-22	3	true
3	2023-02-22	2023-01-11	4	false
4	2023-01-02	2022-11-25	6	true
5	2023-02-27	2022-08-18	8	true
6	2023-01-08	2022-11-26	9	true
7	2023-02-18	2023-02-02	10	false
8	2022-08-14	2022-09-03	12	false
9	2022-12-28	2022-06-27	14	true
10	2022-11-15	2022-06-17	16	false
11	2023-02-17	2022-07-07	17	true
12	2022-11-05	2022-12-28	20	true
13	2023-03-02	2022-12-14	22	true
14	2022-08-06	2022-04-02	24	true
15	2022-12-25	2022-05-28	26	true
16	2022-10-10	2023-01-27	27	false
17	2022-10-04	2022-06-14	29	false
18	2022-12-11	2022-07-22	30	false
19	2022-11-28	2022-06-13	31	false
20	2022-11-06	2022-09-23	33	true
21	2022-12-08	2023-01-01	35	true

Save Cancel Script

*Se desea conocer el id, estado de las admisiones en el periodo de tiempo 2022-08-05 y 2023-02-23

SELECT id , estado , fecha_admi

FROM admision a

where fecha_admi >'2022-08-05' and fecha_admi <'2023-02-23'

SQL Query:

```
SELECT id , estado , fecha_admi
FROM admision a
where fecha_admi >'2022-08-05' and fecha_admi <'2023-02-23'
```

admision 1

SQL Query: `SELECT id , estado , fecha_admi`

	id	estado	fecha_admi
1	1	true	2022-12-23
2	6	true	2022-10-27
3	7	false	2022-09-04
4	9	true	2022-11-27
5	10	true	2022-09-29
6	11	true	2022-09-25
7	13	true	2022-12-28
8	14	true	2022-09-01
9	15	true	2023-01-06
10	18	true	2022-08-28
11	21	false	2022-10-28
12	23	false	2022-12-01
13	25	false	2022-11-20
14	26	true	2022-09-13
15	29	true	2022-10-08
16	30	false	2022-12-04
17	31	false	2022-09-21
18	32	true	2023-02-13
19	34	false	2022-10-07
20	35	false	2023-01-19
21	38	true	2023-02-06

Save Cancel Script

*se desea conocer los id de las matrículas con un precio entre 3000000 y 3500000 en orden

SELECT id , precio

FROM matricula m

where precio >'3000000' and precio <'3500000'

ORDER by precio

SQL Query:

```
SELECT id , precio
FROM matricula m
where precio >'3000000' and precio <'3500000'
ORDER by precio
```

matricula 1 ×

SQL Query: `SELECT id , precio FR` Enter a SQL expression to filter

	id	precio
1	473	3.000.822
2	939	3.001.694
3	27	3.002.024
4	287	3.002.223
5	81	3.005.223
6	323	3.005.325
7	751	3.008.604
8	316	3.008.623
9	159	3.014.605
10	716	3.017.385
11	398	3.022.070
12	104	3.022.403
13	113	3.024.025
14	577	3.026.248
15	846	3.027.477
16	878	3.028.033
17	250	3.029.129
18	951	3.029.439
19	683	3.030.137
20	962	3.030.843
21	687	3.031.138

Save Cancel Script

9. Where between

* se desea conocer solo las carreras con los creditos entre 120 y 140 en la tabla carrera_aplica.

SELECT nombre, credito_total

from carrera_aplica

Where credito_total BETWEEN 120 and 140

SELECT nombre, credito_total
from carrera_aplica
Where credito_total BETWEEN 120 and 140

carrera_aplica 1

SELECT nombre, credito_total

	nombre	credito_total
1	Research and Development	128
2	Research and Development	128
3	Product Management	138
4	Accounting	127
5	Product Management	136
6	Product Management	128
7	Research and Development	138
8	Legal	127
9	Legal	129
10	Services	128
11	Research and Development	126
12	Accounting	137
13	Research and Development	128
14	Sales	120
15	Business Development	128
16	Accounting	126
17	Business Development	128
18	Accounting	130
19	Marketing	127
20	Marketing	126
21	Training	137
22	Services	128

Save Cancel Script

*Se desea conocer el nombre de las materias que tengan de 1 a 2 créditos

SELECT nombre, creditos

FROM materia m

where creditos BETWEEN 1 and 2

SELECT nombre, credits
FROM materia m
where credits BETWEEN 1 and 2

materia 1 x

SELECT nombre, cred | Enter a SQL expression to filter |

	nombre	credits
1	Senior Editor	2
2	Research Assistant III	2
3	Speech Pathologist	2
4	Professor	1
5	Senior Quality Engineer	1
6	Safety Technician I	2
7	Senior Financial Analyst	1
8	Software Test Engineer III	1
9	Nuclear Power Engineer	2
10	Professor	1
11	Data Coordinator	2
12	Media Manager I	2
13	VP Quality Control	1
14	Statistician I	2
15	Marketing Assistant	1
16	Operator	1
17	Account Executive	2
18	VP Marketing	2
19	Senior Developer	2
20	Senior Quality Engineer	1
21	Geological Engineer	1

Save Cancel Script | 657 Rows: 1

*Se desea conocer los pines que tienen un valor entre 55000 y 65000, también conocer su estado.

SELECT id , precio, estado

FROM pin p

where precio BETWEEN 55000 and 65000

SELECT id , precio, estado
FROM pin p
where precio BETWEEN 55000 and 65000

pin 1 x

SELECT id , precio, es Enter a SQL expression to filter

	id	precio	estado
1	7	58.770	false
2	13	60.359	false
3	15	56.878	false
4	21	56.807	true
5	24	59.481	true
6	28	63.544	false
7	37	58.080	true
8	40	58.494	true
9	41	60.637	false
10	47	58.368	true
11	51	59.983	false
12	52	63.479	false
13	54	60.588	false
14	62	63.758	true
15	67	59.055	false
16	72	58.192	true
17	75	55.909	false
18	77	57.589	true
19	78	61.316	true
20	84	62.821	false
21	85	62.320	false

Save Cancel Script

226 Rows: 1

* Se desea conocer el estado de los usuario en un periodo de tiempo entre 2022-07-29 y 2023-02-24

SELECT id_usuario, estado , fecha_admi

FROM admision a

where fecha_admi BETWEEN '2022-07-29' and '2023-02-24'

SELECT id_usuario, estado, fecha_admi
FROM admision a
where fecha_admi BETWEEN '2022-07-29' and '2023-02-24'

admision 1 x

SELECT id_usuario, estado, fecha_admi

	id_usuario	estado	fecha_admi
1	1	true	2022-12-23
2	6	true	2022-10-27
3	7	false	2022-09-04
4	9	true	2022-11-27
5	10	true	2022-09-29
6	11	true	2022-09-25
7	13	true	2022-12-28
8	14	true	2022-09-01
9	15	true	2023-01-06
10	18	true	2022-08-28
11	21	false	2022-10-28
12	23	false	2022-12-01
13	25	false	2022-11-20
14	26	true	2022-09-13
15	29	true	2022-10-08
16	30	false	2022-12-04
17	31	false	2022-09-21
18	32	true	2023-02-13
19	34	false	2022-10-07
20	35	false	2023-01-19
21	38	true	2023-02-06

*Se desea conocer las carreras que tienen un precio entre 3250000 y 3750000 y que este ordenado el precio de manera ascendente.

SELECT nombre, precio

FROM carrera_aplica ca

where precio BETWEEN 3250000 and 3750000

ORDER by precio ASC

SELECT nombre, precio
FROM carrera_aplica ca
where precio BETWEEN 3250000 and 3750000
ORDER by precio ASC

carrera_aplica 1 x

SELECT nombre, precio Enter a SQL expression to filter

	nombre	precio
1	Research and Development	3.254.064,03
2	Legal	3.262.961,45
3	Marketing	3.263.896,95
4	Business Development	3.264.247
5	Marketing	3.264.430,04
6	Services	3.265.850,42
7	Training	3.265.961,39
8	Engineering	3.267.023,56
9	Support	3.273.126,68
10	Business Development	3.276.480,37
11	Business Development	3.278.015,14
12	Accounting	3.292.820,74
13	Product Management	3.293.761,2
14	Training	3.297.598,92
15	Accounting	3.300.543,72
16	Support	3.301.430,43
17	Legal	3.302.011
18	Marketing	3.302.907
19	Research and Development	3.303.074
20	Marketing	3.305.118
21	Sales	3.308.142,51

Save Cancel Script

273 Rows: 1

273 row(s) fetched - 527ms, on 2023-03-05 at 21:20:14

10. Where in

* se desea conocer solo los precios de las siguientes carreras Legal, Sales, Research and Development de la tabla carrera_aplica.

```
SELECT nombre, precio
```

```
from carrera_aplica
```

```
Where nombre in('Legal', 'Sales', 'Research and Development' );
```

SELECT nombre, precio
from carrera_aplica
where nombre in('Legal', 'Sales', 'Research and Development');

carrera_aplica 1 x

SELECT nombre, precio Enter a SQL expression to filter

	nombre	precio
1	Research and Development	4.015.987,21
2	Research and Development	4.797.209,01
3	Research and Development	3.470.706
4	Sales	4.301.638,12
5	Sales	3.526.839,31
6	Legal	4.115.955,76
7	Legal	3.679.908,79
8	Research and Development	4.599.724
9	Research and Development	4.683.380,65
10	Legal	3.727.880,02
11	Sales	4.876.661
12	Sales	4.446.153
13	Legal	3.376.466,02
14	Sales	3.916.932
15	Legal	3.794.707,27
16	Research and Development	3.137.606,99
17	Sales	4.216.240,87
18	Legal	4.392.611
19	Research and Development	3.443.905,35
20	Research and Development	3.647.647,86
21	Research and Development	3.620.628,12
22	Sales	3.102.560

*Se desea conocer el teléfono de las siguientes personas Penelopa, Alexandros , Pierson

SELECT nombre, telefono

FROM persona p

where nombre('Penelopa','Alexandros','Pierson');

SELECT nombre, telefono
FROM persona p
where nombre in ('Penelopa', 'Alexandros', 'Pierson');

persona 1 x

SELECT nombre, telefono

	nombre	telefono
1	Penelopa	+7 319 511 2695
2	Alexandros	+86 196 692 8664
3	Pierson	+966 404 639 1855

*Se desea conocer el pais de origen de los siguientes estudiantes Wayne, Davon, Llewellyn

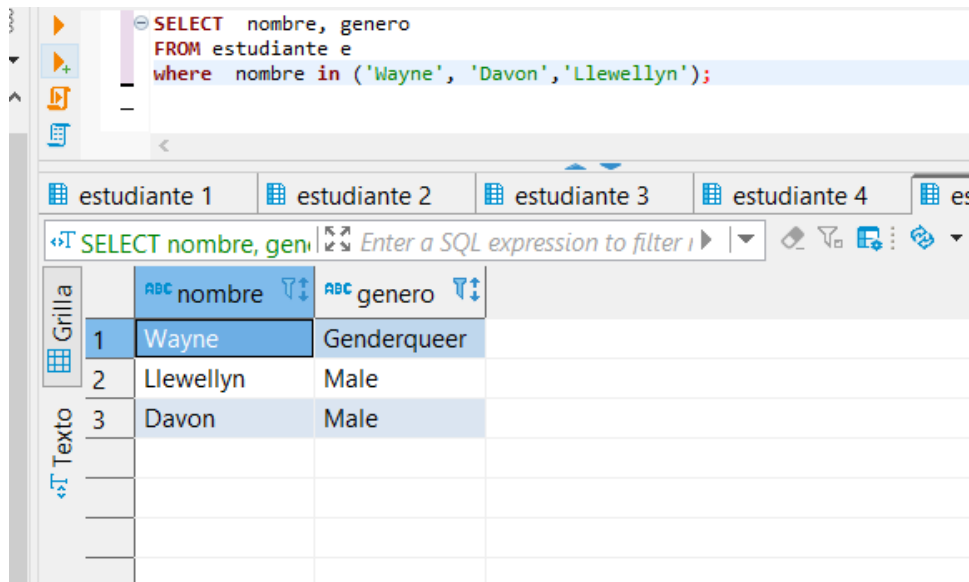
SELECT nombre, pais
FROM estudiante e
where nombre in ('Wayne', 'Davon', 'Llewellyn');

estudiante 1 estudiante 2 estudiante 3 estudiante 4

SELECT nombre, pais

	nombre	pais
1	Wayne	China
2	Llewellyn	Brazil
3	Davon	Indonesia

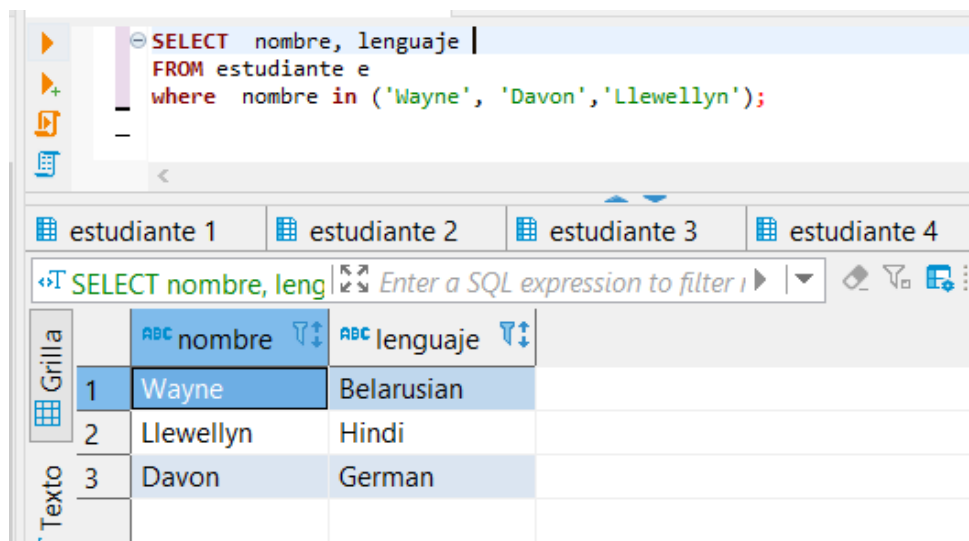
*Se desea conocer el genero de los siguientes estudiantes Wayne, Davon, Llewellyn



```
SELECT nombre, genero
FROM estudiante e
where nombre in ('Wayne', 'Davon', 'Llewellyn');
```

	nombre	genero
1	Wayne	Genderqueer
2	Llewellyn	Male
3	Davon	Male

*Se desea conocer el lenguaje de los siguientes estudiantes Wayne, Davon, Llewellyn



```
SELECT nombre, lenguaje
FROM estudiante e
where nombre in ('Wayne', 'Davon', 'Llewellyn');
```

	nombre	lenguaje
1	Wayne	Belarusian
2	Llewellyn	Hindi
3	Davon	German

11. Where like

*se desea conocer las carreras que empiecen con la letra R en la tabla carrera_aplica

SELECT nombre

from carrera_aplica

Where nombre like 'R%';

The screenshot shows a database query tool interface. At the top, a SQL query is entered in a text area:

```
SELECT nombre
from carrera_aplica
where nombre like 'R%';
```

Below the query, a tab labeled "carrera_aplica 1" is selected. The results are displayed in a table with the following structure:

	nombre
1	Research and D
2	Research and D
3	Research and D
4	Research and D
5	Research and D
6	Research and D
7	Research and D
8	Research and D
9	Research and D
10	Research and D
11	Research and D
12	Research and D
13	Research and D
14	Research and D
15	Research and D
16	Research and D
17	Research and D
18	Research and D
19	Research and D
20	Research and D
21	Research and D
22	Research and D

On the left side of the table, there are three view options: "Grilla" (selected), "Texto", and "Record".

* Se desea conocer el nombre de los estudiantes que su segunda letra sea a

```
SELECT nombre
```

```
FROM estudiante e
```

```
where nombre LIKE '_a%';
```

SQL Query Editor:

```
SELECT nombre
FROM estudiante e
where nombre LIKE '_a%';
```

Database: estudiante 1 | estudiante 2 | estudiante 3 | estuc

SQL Query: SELECT nombre FROM

Grilla

	nombre
1	Dannel
2	Jacquelynn
3	Wayne
4	Valaree
5	Rafaellle
6	Davon
7	Galvin
8	Lavinia
9	Yance
10	Cassey
11	Mamie
12	Farrel
13	Hadley
14	Bale
15	Karl
16	Vasilis
17	Hanan
18	Hale
19	Barbaraanne
20	Van
21	Barron

Record

Save Cancel Script

* Se desea conocer el apellido de las personas que su segunda letra sea s

Select apellido

From persona p

Where apellido like '_s%';

SQL Query Editor:

```
SELECT apellido
FROM persona p
where apellido LIKE 's%';
```

Database: estudiante 1 | estudiante 2 | estudiante 3

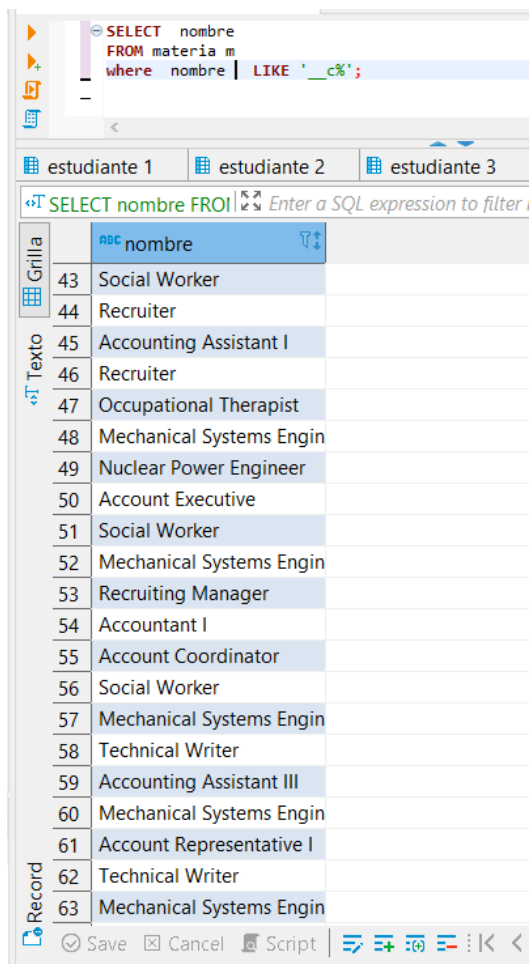
SQL Query: SELECT apellido FROM persona p where apellido LIKE 's%';

Grilla	Apellido
1	Isakowicz
2	Usmar
3	Escoffrey
4	Askem
5	Ashbe
6	Aslott
7	Escala
8	Asken
9	Ashburner
10	Island
11	Asaaf

* se desea conocer los nombres de las materias que empiezan se tercera letra sea s

<pre>SELECT nombre FROM materia m where nombre LIKE 'c%';</pre>		
<div> <div>estudiante 1</div> <div>estudiante 2</div> <div>estudiante 3</div> <div>est</div> </div>		
<div> <div>SELECT nombre FROM</div> <div>Enter a SQL expression to filter</div> </div>		
<div> <div>Grilla</div> <div>Texto</div> </div>	43	Business Systems Develop
	44	Associate Professor
	45	Research Assistant III
	46	Design Engineer
	47	Associate Professor
	48	GIS Technical Architect
	49	Research Associate
	50	Research Nurse
	51	Research Nurse
	52	Assistant Media Planner
	53	Assistant Manager
	54	Design Engineer
	55	Associate Professor
	56	Desktop Support Technicia
	57	Design Engineer
	58	Cost Accountant
	59	Desktop Support Technicia
	60	Associate Professor

*se desea conocer los nombres de las materias que empiezan se tercera letra sea c



12. Where not like

* se desea conocer las carreras que no empiezan con la letra R de la tabla carrera aplica.

```
SELECT nombre
```

```
from carrera_aplica
```

```
Where nombre not like 'R%';
```

SELECT nombre
from carrera_aplica
Where nombre not like 'R%';

carrera_aplica 1

SELECT nombre from

	nombre
1	Human Resources
2	Human Resources
3	Product Management
4	Marketing
5	Sales
6	Product Management
7	Sales
8	Accounting
9	Legal
10	Accounting
11	Product Management
12	Legal
13	Product Management
14	Services
15	Marketing
16	Product Management
17	Marketing
18	Business Development
19	Accounting
20	Product Management
21	Services
22	Training

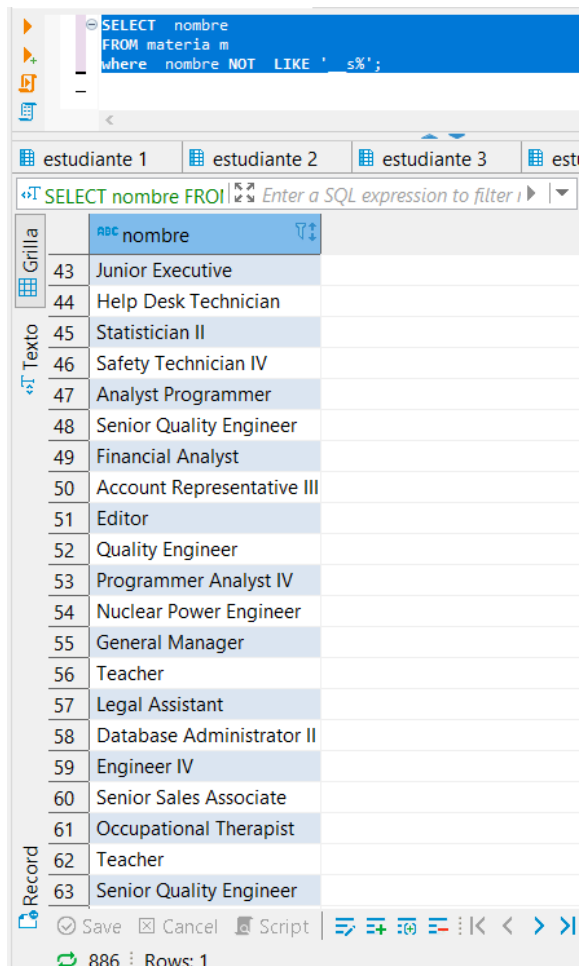
Save Cancel Script

* Se desea conocer los nombres de las materias que su tercera letra no sea s

SELECT nombre

FROM materia m

where nombre NOT LIKE '___s%';

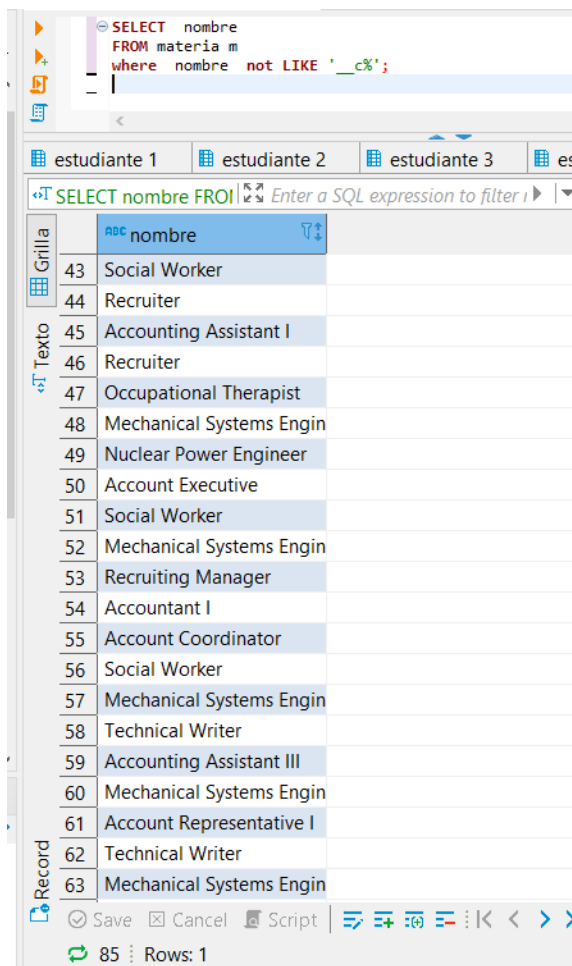


* Se desea conocer los nombres de las materias que su tercera letra no sea c

SELECT nombre

FROM materia m

where nombre not LIKE '___c%';



* Se desea conocer el apellido de las personas que su segunda letra no sea s

Select apellido

From persona p

Where apellido not like '_s%';

Select apellido
From persona p
Where apellido not like '_s%';

estudiante 1 estudiante 2 estudian

Select apellido From Enter a SQL expression

Grilla	ABC apellido
1	Raoult
2	Keable
3	Kyte
4	Galbreth
5	De Roos
6	Dougal
7	Keely
8	Brookhouse
9	Cluett
10	Stanyon
11	Condy
12	Wyley
13	Alyokhin
14	Cochet
15	Jessen
16	Cobden
17	Kleisle
18	Boyes
19	Winsiowiecki
20	Archley

cord

*Se desea conocer el nombre de los estudiantes que su segunda letra no sea a

SELECT nombre

FROM estudiante e

where nombre not LIKE '_a%';

*<02210131011 2> Script-2 x 02210131011

```
SELECT nombre
FROM estudiante e
where nombre not LIKE '_a%';
```

estudiante 1 estudiante 2 estudiante 3

SELECT nombre FROM Enter a SQL expression to filter

Grilla	nombre
1	Sherill
2	Isaac
3	Archie
4	Jo-anne
5	Yolande
6	Deloria
7	Louis
8	Monique
9	Llewellyn
10	Adara
11	Constantin
12	Berri
13	Ely
14	Eolanda
15	Murial
16	Elwira
17	Pearl

13. Select as

* se desea revisar el nombre de en la tabla carrera_aplica pero ahora con el nombre carrera.

```
SELECT nombre as carrera
```

```
from carrera_aplica
```

```
;
```

SELECT nombre as carrera
from carrera_aplica
;

carrera_aplica 1 X

SELECT nombre as ca Enter a SQL expression to j

Grilla	ABC	carrera
1		Human Resources
2		Research and Development
3		Human Resources
4		Research and Development
5		Product Management
6		Research and Development
7		Marketing
8		Sales
9		Product Management
10		Sales
11		Accounting
12		Legal
13		Accounting
14		Product Management
15		Legal
16		Product Management
17		Services
18		Marketing
19		Product Management
20		Marketing
21		Business Development
22		Accounting

Save Cancel Script

* SELECT nombre as estudiante

FROM estudiante e

SQL Query Editor:

```
SELECT nombre as estudiante
FROM estudiante e
```

Query Results:

	estudiante 1	estudiante 2	estudiante 3
Grilla	ABC estudiante		
1	Sherill		
2	Isaac		
3	Dannel		
4	Archie		
5	Jo-anne		
6	Jacquelynn		
7	Yolande		
8	Wayne		
9	Deloria		
10	Louis		
11	Monique		
12	Llewellyn		
13	Adara		
14	Valaree		
15	Rafaellle		
16	Davon		

*SELECT nombre as estudiante

FROM usuario u

<div> <div> SELECT nombre as usuario FROM usuario u </div> </div>	
<div> <div>estudiante 1</div> <div>estudiante 2</div> <div>estudiant</div> </div>	
<div> <div> SELECT nombre as es </div> <div>Enter a SQL expression to</div> </div>	
Grilla	<div> <div>ABC</div> <div>estudiante</div> </div>
1	Silva
2	Willy
3	Ivory
4	Chelsey
5	Valentina
6	Luke
7	Abbey
8	Orsa
9	Culley
10	Jobina
11	Jaymee
12	Zane
13	Berkley
14	Emmanuel
15	Lutero
16	Byrom
17	Rodolphe
18	Britta
19	Garrott
20	Aland
21	Salvador

*SELECT nombre as asignatura

FROM materia m

<div> <div> SELECT nombre as asignatura FROM materia m </div> </div>	
<div> <div> <div>estudiante 1</div> <div>estudiante 2</div> <div>estudiante 3</div> </div> <div> SELECT nombre as as Enter a SQL expression to fi </div> </div>	
Grilla	<div> <div>asignatura</div> </div>
1	Geological Engineer
2	Senior Editor
3	Research Assistant III
4	Speech Pathologist
5	Financial Advisor
6	Professor
7	Senior Quality Engineer
8	Safety Technician I
9	Senior Financial Analyst
10	Research Assistant I
11	Help Desk Operator
12	Occupational Therapist
13	Software Test Engineer III
14	Nuclear Power Engineer
15	Professor
16	Data Coordinator
17	Media Manager I
18	VP Quality Control
19	Electrical Engineer
20	Web Developer IV
21	Statistician I

*SELECT nombre as conjunto

FROM grupo g

SELECT nombre as conjunto
FROM grupo g

estudiante 1 estudiante 2 estudiante 3 estudi

SELECT nombre as as Enter a SQL expression to filter i

Grilla

	asignatura
1	Geological Engineer
2	Senior Editor
3	Research Assistant III
4	Speech Pathologist
5	Financial Advisor
6	Professor
7	Senior Quality Engineer
8	Safety Technician I
9	Senior Financial Analyst
10	Research Assistant I
11	Help Desk Operator
12	Occupational Therapist
13	Software Test Engineer III
14	Nuclear Power Engineer
15	Professor
16	Data Coordinator
17	Media Manager I
18	VP Quality Control
19	Electrical Engineer
20	Web Developer IV
21	Statistician I

Record

Save Cancel Script