1

```
/*1. Mostrar los alumnos del instituto, pero que solo aparezcan desde el quinto al noveno.*/
select * from alumno limit 5 offset 4;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows:

id	nombre	apellido1	apellido2	fecha_nacimiento	es_repetidor	telefono
5	Paco	Martinez	Lopez	1995-11-24	no	692735409
6	Irene	Gutierrez	Sanchez	1991-03-28	si	NULL
7	Cristina	Fernandez	Ramirez	1996-09-17	no	628349590
8	Antonio	Carretero	Ortega	1994-05-20	si	612345633
9	Manuel	Dominguez	Hernandez	1999-07-08	no	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

2

```
/*2. Realiza una consulta de todos los alumnos repetidores.*/
select * from alumno where es_repetidor="si";
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Co

id	nombre	apellido1	apellido2	fecha_nacimiento	es_repetidor	telefono
4	Lucia	Sanchez	Ortega	1993-06-13	si	678516294
6	Irene	Gutierrez	Sanchez	1991-03-28	si	NULL
8	Antonio	Carretero	Ortega	1994-05-20	si	612345633
11	Pepe	Sanchez	Acedo	1997-05-23	si	251222555
12	Samuel	Sanchez	Acedo	1998-11-23	si	251333777
13	Juan	Sanchez	Acedo	2000-12-20	si	251333444
14	Zacarias	Sanchez	Acedo	2001-12-25	si	251333555
NULL	NULL	NULL	NULL	NULL	NULL	NULL

3

```
/*3. Realiza una consulta de todos los alumnos repetidores nacidos entre el año 1995 y el
2000.*/
select * from alumno where es_repetidor ="si" and fecha_nacimiento > "1994-12-31" and fecha_nacimiento < "2000-01-01";
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	nombre	apellido1	apellido2	fecha_nacimiento	es_repetidor	telefono
11	Pepe	Sanchez	Acedo	1997-05-23	si	251222555
12	Samuel	Sanchez	Acedo	1998-11-23	si	251333777
NULL	NULL	NULL	NULL	NULL	NULL	NULL

4

```
/*4. Realiza una consulta de los no repetidores nacidos después de 1998*/
select * from alumno where es_repetidor="no" and fecha_nacimiento > "1997-12-31";
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	nombre	apellido1	apellido2	fecha_nacimiento	es_repetidor	telefono
2	Juan	Saez	Vega	1998-04-02	no	618253876
9	Manuel	Dominguez	Hernandez	1999-07-08	no	NULL
10	Daniel	Moreno	Ruiz	1998-02-03	no	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

5

```
/*5. Realiza una consulta de alumnos que nacieron en 1998, y otra de los que no nacieron
ese año.*/
select * from alumno where fecha_nacimiento > "1997-12-31" and fecha_nacimiento < "1999-01-01";
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	nombre	apellido1	apellido2	fecha_nacimiento	es_repetidor	telefono
2	Juan	Saez	Vega	1998-04-02	no	618253876
10	Daniel	Moreno	Ruiz	1998-02-03	no	NULL
12	Samuel	Sanchez	Acedo	1998-11-23	si	251333777
NULL	NULL	NULL	NULL	NULL	NULL	NULL

6

```
/*6. Consulta a los alumnos que tienen un identificador entre 4 y 9.*/
select * from alumno limit 6 offset 3;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows:

id	nombre	apellido1	apellido2	fecha_nacimiento	es_repetidor	telefono
4	Lucia	Sanchez	Ortega	1993-06-13	si	678516294
5	Paco	Martinez	Lopez	1995-11-24	no	692735409
6	Irene	Gutierrez	Sanchez	1991-03-28	si	NULL
7	Cristina	Fernandez	Ramirez	1996-09-17	no	628349590
8	Antonio	Carretero	Ortega	1994-05-20	si	612345633
9	Manuel	Dominguez	Hernandez	1999-07-08	no	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

7

```
/*7. Consulta a los alumnos mayores de edad y menores de 25 que no tienen teléfono.*/
select * from alumno where telefono is null and fecha_nacimiento between '1997-01-01' and '2003-12-31';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	nombre	apellido1	apellido2	fecha_nacimiento	es_repetidor	telefono
9	Manuel	Dominguez	Hernandez	1999-07-08	no	NULL
10	Daniel	Moreno	Ruiz	1998-02-03	no	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

8

```
/*8. Calcular la circunferencia de un círculo de radio 5 cm. (circunferencia = 2 * pi * radio) y
mostrarla en los siguientes formatos:*/
/*a) Sin truncar ni redondear*/
set @y = 5;
select @y;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

@y
5

```
/*8. Calcular la circunferencia de un círculo de radio 5 cm. (circunferencia = 2 * pi * radio) y
mostrarla en los siguientes formatos:*/
/*a) Sin truncar ni redondear*/
set @y = 5;
select @y;
set @circunferencia = 2 * PI() + @y;
select @circunferencia;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

@circunferencia
11.283185307179586

```
/*b) Redondeando sin decimales.*/
select round (@circunferencia);
/*c) Redondear con dos decimales.*/
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

round (@circunferencia)
11

```
/*c) Redondear con dos decimales.*/
select round (@circunferencia,2);
```

Result Grid | Filter Rows: | Export: | Wr

round (@circunferencia,2)
11.28

9

```
/*9. ¿Cuántos bytes tiene un KB? ¿Y un MB?*/
select pow(2, 10) as 'kb(2^10)', pow(10,6) as 'mb(1*10^6)';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

kb(2^10)	mb(1*10^6)
1024	1000000

10

```
/*10. Mostrar la hora actual mostrando 4 columnas:
a) Hora completa: hh:mm:ss
b) Solo la hora.
c) Solo los minutos.
d) Solo segundos.*/
select curtime(), date_format(now(), "%H"), date_format(now(), "%i"), date_format(now(), "%s");
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

curtime()	date_format(now(), "%H")	date_format(now(), "%i")	date_format(now(), "%s")
19:46:38	19	46	38

11

```
/*11. De la tabla alumnos queremos obtener un listado de las onomásticas ordenado. Repite
la búsqueda sobre la tabla contactos. Mostrar de 15 en 15.*/
select nombre, apellido1, apellido2 from alumno order by apellido1, apellido2, nombre limit 15 offset 0;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

nombre	apellido1	apellido2
Antonio	Carretero	Ortega
Manuel	Dominguez	Hernandez
Cristina	Fernandez	Ramirez
Irene	Gutierrez	Sanchez
Paco	Martinez	Lopez
Daniel	Moreno	Ruiz
Pepe	Ramirez	Gea
Juan	Saez	Vega
Maria	Sanchez	NULL
Juan	Sanchez	Acedo
Pepe	Sanchez	Acedo
Samuel	Sanchez	Acedo
Zacarias	Sanchez	Acedo
Lucia	Sanchez	Ortega

12

```
/*12. Mostrar un listado de los alumnos en el que aparezcan tres columnas: id, nombre
completo (apellido1, apellido2, nombre) y es_repetidor. Estará ordenada por: el campo
es_repetidor y por nombre completo (apellido1, apellido2, nombre). Ambos criterios
ascendentes. Observa cuidadosamente los resultados y explica la ordenación.*/
select id, concat_ws(' ', apellido1, apellido2, nombre) as 'nombre completo', es_repetidor from alumno
order by es_repetidor, apellido1, apellido2, nombre;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id	nombre completo	es_repetidor
8	Carretero Ortega Antonio	si
6	Gutierrez Sanchez Irene	si
13	Sanchez Acedo Juan	si
11	Sanchez Acedo Pepe	si
12	Sanchez Acedo Samuel	si
14	Sanchez Acedo Zacarias	si
4	Sanchez Ortega Lucia	si
9	Dominguez Hernandez Manuel	no
7	Fernandez Ramirez Cristina	no
5	Martinez Lopez Paco	no
10	Moreno Ruiz Daniel	no
3	Ramirez Gea Pepe	no
2	Saez Vega Juan	no
1	Sanchez Maria	no

13

```
/*13.Listado de todos los alumnos con su edad actual. Para el cálculo considera que todos los
años tienen 365.242 días (365 días, 5 horas y 48 minutos).*/
select round(datediff (now(), fecha_nacimiento)/365) from alumno;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

round(datediff (now(), fecha_nacimiento)/365)
33
25
35
30
28
32
27
29
24
25
26
25
22
21

14

```
/*14. Listado de todos los alumnos cuyo segundo apellido termine en “ez”.*/
select * from alumno where apellido2 like "%ez";
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	nombre	apellido1	apellido2	fecha_nacimiento	es_repetidor	telefono
5	Paco	Martinez	Lopez	1995-11-24	no	692735409
6	Irene	Gutierrez	Sanchez	1991-03-28	si	NULL
7	Cristina	Fernandez	Ramirez	1996-09-17	no	628349590
9	Manuel	Dominguez	Hernandez	1999-07-08	no	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

15

```
/*15.De la base de datos institutodb:
a) Eliminar el segundo apellido del primer alumno. (..set apellido2= NULL..).
b) Mostrar un listado en el que aparezcan dos columnas: id y “apellido1 apellido2, nombre”
En cabecera ha de aparecer: Código y Nombre de alumno.
*/
update alumno set apellido2 = null where id = 1;
select id as codigo, concat_ws(" ", apellido1, apellido2, nombre) as
"Nombre del alumno" from alumno;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

codigo	Nombre del alumno
1	Sanchez Maria
2	Saez Vega Juan
3	Ramirez Gea Pepe
4	Sanchez Ortega Lucia
5	Martinez Lopez Paco
6	Gutierrez Sanchez Irene
7	Fernandez Ramirez Cristina
8	Carretero Ortega Antonio
9	Dominguez Hernandez Manuel
10	Moreno Ruiz Daniel
11	Sanchez Acedo Pepe
12	Sanchez Acedo Samuel
13	Sanchez Acedo Juan
14	Sanchez Acedo Zacarias

16

```
/*16. .Queremos asignar una cuenta de correo a los alumnos que estará compuesta por: letra
inicial, tres primeras letras del primer apellido, tres primeras letras del segundo apellido y
año de nacimiento (unidad y decena). El dominio será: politecnico.com*/
select id, nombre, lower(concat_ws('', left(trim(nombre),1), left(trim(apellido1),3), left(trim(apellido2),3), '@politecnico.com'))
as correo from alumno;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id	nombre	correo
1	Maria	msan@politecnico.com
2	Juan	jsaeveg@politecnico.com
3	Pepe	pramgea@politecnico.com
4	Lucia	lsanort@politecnico.com
5	Paco	pmarlop@politecnico.com
6	Irene	igutsan@politecnico.com
7	Cristina	cferram@politecnico.com
8	Antonio	acarort@politecnico.com
9	Manuel	mdomher@politecnico.com
10	Daniel	dmorrui@politecnico.com
11	Pepe	psanace@politecnico.com
12	Samuel	ssanace@politecnico.com
13	Juan	jsanace@politecnico.com
14	Zacarias	zsanace@politecnico.com

17

```
/*17. Se ha de mostrar tres columnas: id, nombre completo (apellido1,apellido2, “, “ , nombre)
y correo. En las cabeceras ha de aparecer: Código, Nombre, Correo electrónico.*/
select id as 'codigo', concat_ws('', apellido1,',', apellido2, " ", nombre) as 'Nombre', lower(concat_ws('', left
(trim(nombre),1), left(trim(apellido1),3), left(trim(apellido2),3), '@politecnico.com')) as correo from alumno;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

codigo	Nombre	correo
1	Sanchez, Maria	msan@politecnico.com
2	Saez,Vega Juan	jsaeveg@politecnico.com
3	Ramirez,Gea Pepe	pramgea@politecnico.com
4	Sanchez,Ortega Lucia	lsanort@politecnico.com
5	Martinez,Lopez Paco	pmarlop@politecnico.com
6	Gutierrez,Sanchez Irene	igutsan@politecnico.com
7	Fernandez,Ramirez Cristina	cferram@politecnico.com
8	Carretero,Ortega Antonio	acarort@politecnico.com
9	Dominguez,Hernandez Manuel	mdomher@politecnico.com
10	Moreno,Ruiz Daniel	dmorrui@politecnico.com
11	Sanchez,Acedo Pepe	psanace@politecnico.com
12	Sanchez,Acedo Samuel	ssanace@politecnico.com
13	Sanchez,Acedo Juan	jsanace@politecnico.com
14	Sanchez,Acedo Zacarias	zsanace@politecnico.com

18

```
/*18. En la tabla amo (veterinariodb) aparece el atributo: “apellidos”. Sobre esa tabla haz una
consulta en la que aparezcan tres columnas: primer apellido, segundo apellido y los dos
apellidos juntos (comprueba los resultados). Puede haber dueños de mascotas que solo
tengan un apellido o un número indeterminado de espacios en blanco al principio, final o en
medio.*/
select left(apellidos, (locate(" ", apellidos, 1))), right(apellidos, (length(apellidos)-(locate(" ", apellidos, 1)))), apellidos from amo;
select * from amo;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

left(apellidos, (locate(" ", apellidos, 1)))	right(apellidos, (length(apellidos)-(locate(" ", apellidos, 1))))	apellidos
Pérez	Gúzman	Pérez Gúzman
Santos	Gómez	Santos Gómez
	Subirez	Subirez
Diaz	Baeza	Diaz Baeza
Caldito	Porrón	Caldito Porrón
	Saiz-Vicuña	Saiz-Vicuña
Pérez	Gúzman	Pérez Gúzman

19

```
/*19.  Listado en dos columnas ordenado alfabéticamente por apellidos y nombre. En la que
aparezca: a) Nombre completo: Nombre (Primera letra mayúscula y resto minúscula), una
“,” seguida de los apellidos en mayúsculas. b) Número de caracteres que tiene el nombre
completo.*/
select concat_ws('', concat(upper(left(nombre, 1)), lower(substr(nombre, 2, length(nombre)))), ', ', upper(apellido1), ',', upper(apellido2)) as nombres,
length(concat_ws('', nombre, apellido1, apellido2)) as numCar from alumno;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombres	numCar
Maria, SANCHEZ,	12
Juan, SAEZ,VEGA	12
Pepe, RAMIREZ,GEA	14
Lucia, SANCHEZ,ORTEGA	18
Paco, MARTINEZ,LOPEZ	17
Irene, GUTIERREZ,SANCHEZ	21
Cristina, FERNANDEZ,RAMIREZ	24
Antonio, CARRETERO,ORTEGA	22
Manuel, DOMINGUEZ,HERNANDEZ	24
Daniel, MORENO,RUIZ	16
Pepe, SANCHEZ,ACEDO	16
Samuel, SANCHEZ,ACEDO	18
Juan, SANCHEZ,ACEDO	16
Zacarias, SANCHEZ,ACEDO	20

20

```
/*20. Expresar la fecha de hoy en el formato: **lunes, 31 de enero de 2021** */
select date_format("2022-01-31",'**%W, %d de %M de %Y**');
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

date_format("2022-01-31",'**%W, %d de %M de %Y**')
Monday, 31 de January de 2022

21

```
/*21. De la base de datos de contactos, obtener un listado de las personas en el que
aparezcan las siguientes columnas:
a) Nombre.
b) Apellidos.
c) Nombre del mes de nacimiento.
d) Día de nacimiento.
e) Año de nacimiento. El objetivo es tener una lista de los próximos cumpleaños, por lo que
se ha de ordenar por: mes, día y año. Para verlo cómodamente se han de mostrar de 15 en
15.
*/
select nombre, apellido1, date_format(fecha_nacimiento, '%M') as Mes_nacimiento, date_format(fecha_nacimiento, '%e')
as dia_nacimiento, date_format(fecha_nacimiento, '%Y') as año_nacimiento from alumno order by date_format(fecha_nacimiento, '%m'),
dia_nacimiento, año_nacimiento;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre	apellido1	Mes_nacimiento	dia_nacimiento	año_nacimiento
Pepe	Ramirez	January	3	1988
Daniel	Moreno	February	3	1998
Irene	Gutierrez	March	28	1991
Juan	Saez	April	2	1998
Antonio	Carretero	May	20	1994
Pepe	Sanchez	May	23	1997
Lucia	Sanchez	June	13	1993
Manuel	Dominguez	July	8	1999
Cristina	Fernandez	September	17	1996
Samuel	Sanchez	November	23	1998
Paco	Martinez	November	24	1995
Maria	Sanchez	December	1	1990
Juan	Sanchez	December	20	2000
Zacarias	Sanchez	December	25	2001

22

```
/*22. Establecer la expresión regular que sea verdadera con los siguientes formatos:
a) Código postal: cinco números.
b) Número de DNI: siete u ocho números.
c) Teléfono: 9 números.
d) Nif: siete u ocho números seguidos o no de un guion y un letra
e) Matrícula de un coche, formato nuevo: 4 números, un guión y tres letras.
f) Matrícula de un coche, formato antiguo: una o dos letras (según provincia), guion, cuatro
números, guión y una o dos letras. Guiones opcionales.
g) Matrícula de un coche, formato antiquísimo: una o dos letras (según provincia), un guion
y entre un número y seis.*/

select '32478' regexp '^[0-9]{5}$';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

'32478' regexp '^[0-9]{5}$'
1

```
127 •   select '32478601' regexp '^[0-9]{8}$';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	'32478601' regexp '^[0-9]{8}$'
▶	1

```
128 •   select '952451278' regexp '^[0-9]{9}$';
129
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	'952451278' regexp '^[0-9]{9}$'
▶	1

```
129 •   select '95245127-b' regexp '[0-9]{7,8}.[a-z]';
130
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	'95245127-b' regexp '[0-9]{7,8}.[a-z]'
▶	1

```
130 •   select '9524-bcv' regexp '[0-9]{4}.[a-z]{3}';
131
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	'9524-bcv' regexp '[0-9]{4}.[a-z]{3}'
▶	1

```
131 •   select 'ab-567' regexp '^[a-z]{2}-[0-9]{1,6}';
132
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	'ab-567' regexp '^[a-z]{2}-[0-9]{1,6}'
▶	1