

```
-- Inciso a)
```

```
-- Muestra solo a las personas que su nombre tiene más de 5 letras.
```

```
select * from usuarios where char_length(nombre) > 5;
```




Result Grid									
Filter Rows:									
Export:									
Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	1	Alejandro	Martínez	Vega	alejandro.martinez@example.com	5551239876	1991-08-13	2025-06-18	PassAlejandr
	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18	2025-06-18	PassValeria1
	3	Santiago	Ramos	Ortega	santiago.ramos@example.com	5554567890	1988-11-23	2025-06-18	PassSantiagc
	5	Fernando	Vázquez	Pérez	fernando.vazquez@example.com	5559871234	1994-02-17	2025-06-18	PassFernand
	6	Emilia	Castro	Rivera	emilia.castro@example.com	5551122334	1992-07-25	2025-06-18	PassEmilia12
	7	Jorge Luis	Méndez	Flores	jorge.mendez@example.com	5552233445	1989-03-12	2025-06-18	PassJorge12
	8	Camila	Gutiérrez	Sánchez	camila.gutierrez@example.com	5555544332	1993-09-21	2025-06-18	PassCamila1
	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28	2025-06-18	PassDaniel12
	10	Ana Sofía	Hernández	Gómez	ana.hernandez@example.com	5558899000	1985-12-15	2025-06-18	PassAnaSofi
	11	Roberto	López	Romero	roberto.lopez@example.com	5551122777	1986-01-05	2025-06-18	PassRoberto
	13	Juan Manuel	García	Torres	juan.garcia@example.com	5557788990	1990-08-22	2025-06-18	PassJuanM1

```
-- Muestra a las personas que su nombre tiene entre 5 y 7 caracteres.
```



```
select * from usuarios where char_length(nombre) between 5 and 7;
```

Result Grid									
Filter Rows:									
Export:									
Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18	2025-06-18	PassValeria123
	4	Lucía	Morales	Jiménez	lucia.morales@example.com	5551234567	1995-05-06	2025-06-18	PassLucia123
	6	Emilia	Castro	Rivera	emilia.castro@example.com	5551122334	1992-07-25	2025-06-18	PassEmilia123
	8	Camila	Gutiérrez	Sánchez	camila.gutierrez@example.com	5555544332	1993-09-21	2025-06-18	PassCamila123
	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28	2025-06-18	PassDaniel123
	11	Roberto	López	Romero	roberto.lopez@example.com	5551122777	1986-01-05	2025-06-18	PassRoberto1
	12	Elena	Díaz	Moreno	elena.diaz@example.com	5553344556	1996-06-07	2025-06-18	PassElena123
	14	Marcela	Rojas	Pacheco	marcela.rojas@example.com	5559988776	1995-02-09	2025-06-18	PassMarcela12
	16	Andrea	Mendoza	Aguirre	andrea.mendoza@example.com	5556655447	1998-05-30	2025-06-18	PassAndrea12
	18	Paula	Fuentes	Guzmán	paula.fuentes@example.com	5553344221	1991-12-14	2025-06-18	PassPaula123
	19	Carlos	Martínez	Ríos	carlos.martinez@example.com	5554455667	1988-03-19	2025-06-18	PassCarlos123



```
-- Muestra a las personas que su nombre tiene entre mas de 7
-- caracteres y alguno de sus apellidos tenga entre 5 y 7 caracteres.
select * from usuarios where char_length(nombre) > 7 and (char_length(primer_apellido)
between 5 and 7 or char_length(primer_apellido) between 5 and 7);
```

Result Grid   Filter Rows: <input type="text"/> Export:  Wrap Cell Content: <input checked="" type="checkbox"/>									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	3	Santiago	Ramos	Ortega	santiago.ramos@example.com	5554567890	1988-11-23	2025-06-18	PassSantiago123
	5	Fernando	Vázquez	Pérez	fernando.vazquez@example.com	5559871234	1994-02-17	2025-06-18	PassFernando123
	7	Jorge Luis	Méndez	Flores	jorge.mendez@example.com	5552233445	1989-03-12	2025-06-18	PassJorge123
	13	Juan Manuel	García	Torres	juan.garcia@example.com	5557788990	1990-08-22	2025-06-18	PassJuanM123
	15	Sebastián	Ortiz	Martínez	sebastian.ortiz@example.com	5554433221	1993-11-01	2025-06-18	PassSebastian123
	17	José Luis	Ramírez	Morales	jose.ramirez@example.com	5557766554	1989-04-11	2025-06-18	PassJoseLuis123
	21	Francisco	Navarro	García	francisco.navarro@example.com	5557766553	1993-01-10	2025-06-18	PassFrancisco123
	32	Cristina	Salazar	Méndez	cristina.salazar@example.com	5554455664	1992-12-07	2025-06-18	PassCristina123
	43	Alejandra	Romero	Paredes	alejandra.romero@example.com	5552233442	1993-05-22	2025-06-18	PassAlejandra123
	46	Salvador	Benítez	Flores	salvador.benitez@example.com	5559988779	1985-11-13	2025-06-18	PassSalvador123



```
-- Muestra los primeros tres caracteres del nombre.
select left(nombre, 3) from usuarios;
```

Result Grid  	
	left(nombre, 3)
▶	Ale
	Val
	San
	Luc
	Fer
	Emi
	Jor
	Cam
	Dan
	Ana
	Rob
	Flo



```
-- Muestra los últimos 3 caracteres del nombre.  
select right(nombre, 3) from usuarios;
```

Result Grid  	
	right(nombre, 3)
▶	dro
	ria
	ago
	cía
	ndo
	lia
	uis
	ila
	iel
	fía
	rto
	ena

```
-- Muestra del 2do al 5to carácter del nombre.  
select substring(nombre, 2, 4) from usuarios;
```

Result Grid			 Filter Rows
	substring(nombre, 2, 4)		
▶	leja		
	aler		
	anti		
	ucía		
	erna		
	mili		
	orge		
	amil		
	anie		
	na S		
	ober		
	lena		

```
-- Reemplaza las d por s en nombre.  
select replace(nombre, 'd', 's') from usuarios;
```

Result Grid			 Filter Rows
	replace(nombre, 'd', 's')		
▶	Alejansro		
	Valeria		
	Santiago		
	Lucía		
	Fernanso		
	Emilia		
	Jorge Luis		
	Camila		
	Daniel		
	Ana Sofía		
	Roberto		
	Elena		


```
-- Obtén la longitud del apellido paterno.  
select primer_apellido, char_length(primer_apellido) from usuarios;
```

Result Grid			Filter Rows:
	primer_apellido	char_length(primer_apellido)	
▶	Martínez	8	
	Reyes	5	
	Ramos	5	
	Morales	7	
	Vázquez	7	
	Castro	6	
	Méndez	6	
	Gutiérrez	9	
	Cruz	4	
	Hernández	9	
	López	5	
	Díaz	4	

```
-- Muestra en mayúsculas el nombre.  
select upper(nombre) from usuarios;
```

Result Grid		Filter Rows:
	upper(nombre)	
▶	ALEJANDRO	
	VALERIA	
	SANTIAGO	
	LUCÍA	
	FERNANDO	
	EMILIA	
	JORGE LUIS	
	CAMILA	
	DANIEL	
	ANA SOFÍA	
	ROBERTO	
	ELENA	

```
-- Muestra en minúsculas el apellido paterno.  
select lower(primer_apellido) from usuarios;
```

Result Grid			 Filter Rows
	lower(primer_apellido)		
▶	martínez		
	reyes		
	ramos		
	morales		
	vázquez		
	castro		
	méndez		
	gutierrez		
	cruz		
	hernández		
	lópez		
	díaz		

```
-- Muestra el nombre completo empezando por el apellido paterno con
-- mayúsculas en una sola columna.
select concat(upper(primer_apellido), ' ', segundo_apellido, ' ', nombre) from usuarios;
```

Result Grid		Filter Rows:	Export:
	concat(upper(primer_apellido), ' ', segundo_apellido, ' ', nombre)		
▶	MARTÍNEZ Vega Alejandro		
	REYES Soto Valeria		
	RAMOS Ortega Santiago		
	MORALES Jiménez Lucía		
	VÁZQUEZ Pérez Fernando		
	CASTRO Rivera Emilia		
	MÉNDEZ Flores Jorge Luis		
	GUTIÉRREZ Sánchez Camila		
	CRUZ Martínez Daniel		
	HERNÁNDEZ Gómez Ana Sofía		
	LÓPEZ Romero Roberto		
	NÚÑEZ Moreno Elena		

```
-- Muestra el nombre de las personas con las E reemplazadas con el número 3.
select replace(replace(nombre, 'E', 3), 'e', 3) from usuarios;
```

Result Grid		Filter Rows:	Export:
	replace(replace(nombre, 'E', 3), 'e', 3)		
▶	Al3jandro		
	Val3ria		
	Santiago		
	Lucía		
	F3rmando		
	3milia		
	Jorg3 Luis		
	Camila		
	Dani3l		
	Ana Sofía		
	Rob3rto		
	El3na		

```
-- Muestra el nombre completo de las personas con las o reemplazados con el número 0 en una sola columna.
select replace(concat(nombre, ' ', primer_apellido, ' ', segundo_apellido), 'o', 0) from usuarios;
```

Result Grid		Filter Rows:	Export:
	replace(concat(nombre, ' ', primer_apellido, ' ', segundo_apellido), 'o', 0)		
▶	Alejandro Martínez Vega		
	Valeria Reyes Soto		
	Santiago Ramos Ortega		
	Lucía Morales Jiménez		
	Fernando Vázquez Pérez		
	Emilia Castro Rivera		
	Jorge Luis Méndez Flores		
	Camila Gutiérrez Sánchez		
	Daniel Cruz Martínez		
	Ana Sofía Hernández Gómez		
	Roberto López Romero		
	Elena Díaz Moreno		

```
-- Libros con edición entre 1 y 5 y cuyo título comience con letra mayúscula
select * from libros where edicion between 1 and 5 and titulo REGEXP '^[A-Z]';
```

Result Grid





Filter Rows:

Export:

Wrap Cell Content:

	id_libro	ISBN	titulo	anio_publicacion	editorial	edicion	cantidad_total	cantidad_disponible	clas
▶	1	978-3-16-148410-0	Cien Años de Soledad	1967	Sudamericana	1	10	7	5
	2	978-84-376-0494-7	Don Quijote de la Mancha	1605	Francisco de Robles	2	8	5	5
	4	978-0-7432-7356-5	Fahrenheit 451	1953	Ballantine Books	1	6	4	4
	5	978-0-19-953556-9	Orgullo y Prejuicio	1813	T. Egerton	3	15	12	5
	6	978-0-06-112008-4	Matar a un Ruiseñor	1960	J.B. Lippincott & Co.	1	10	9	5
	7	978-1-85715-091-6	Crimen y Castigo	1866	The Russian Messenger	2	8	6	5
	8	978-0-15-601219-5	El Principito	1943	Reynal & Hitchcock	1	20	16	5
	9	978-0-14-143955-6	Cumbres Borrascosas	1847	Thomas Cautley Newby	1	5	3	4
	10	978-0-375-72472-6	El Gran Gatsby	1925	Charles Scribner's Sons	1	7	6	5
	11	978-0-679-72841-5	Rayuela	1963	Sudamericana	1	10	8	5
	12	978-0-7432-7358-9	El Guardián Entre el Centeno	1951	Little, Brown and Com...	1	9	7	5



Result Grid			Filter Rows: <input type="text"/>	Export: 	Wrap Cell Content: 					
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol

Result Grid		Filter Rows:		Export:		Wrap Cell
	replace(replace(replace(replace(replace(replace(replace('A', '@'), 'a', '@'), 'E', '3'), 'e', '3'), 'I', '!'), 'i', '!'), 'O', '0'), 'o', '0')					
▶	@l3j@ndr0					
	V@l3r!@					
	S@nt!@g0					
	Luci@					
	F3rn@nd0					
	3m!!@					
	J0rg3 Lu!s					
	C@m!!@					
	D@n!3l					
	@n@ S0fi@					
	R0b3rt0					

```
-- Convierte los primeros tres caracteres del nombre en mayúscula (Edgar-->EDgar).
select concat(upper(left(nombre, 3)), lower(substring(nombre, 4))) from usuarios;
```

Result Grid		Filter Rows:	Export:	Wrap Cell
	concat(upper(left(nombre, 3)), lower(substring(nombre, 4)))			
▶	ALEjandro			
	VALeria			
	SANTIago			
	LUCía			
	FERNando			
	EMÍlia			
	JORge luis			
	CAMila			
	DANiel			
	ANA sofía			
	ROBerto			
	El Ena			

```
-- Convierte el ultimo carácter del nombre en mayúsculas (Edgar-->EdgaR).
select concat(substring(nombre, 1, char_length(nombre) - 1),
upper(substring(nombre, -1))) from usuarios;
```

Result Grid		Filter Rows:
	concat(substring(nombre, 1, char_length(nombre) - 1), upper(substring(nombre, -1)))	
▶	Alejandro	
	ValeriA	
	SantiagoO	
	LucíA	
	FernandoO	
	Emilia	
	Jorge Luis	
	Camila	
	Daniel	
	Ana Sofía	
	Roberto	

```
-- Convierte el 3er carácter del nombre en Mayúscula (Edgar--> EdGar).
select concat(substring(nombre, 1, 2), upper(substring(nombre, 3, 1)),
substring(nombre, 4)) from usuarios;
```

Result Grid		Filter Rows:	Export:
	concat(substring(nombre, 1, 2), upper(substring(nombre, 3, 1)), substring(nombre, 4))		
▶	AlEjandro		
	VaLeria		
	SaNtiago		
	LuCía		
	FeRnando		
	EmÍlia		
	JoRge Luis		
	CaMila		
	DaNiel		
	AnA Sofía		
	RoBerto		

```
-- Convierte el 2do y 4to carácter del nombre a Mayúscula(Edgar-->EDgAr).
select concat(substring(nombre, 1, 1), upper(substring(nombre, 2, 1)),
substring(nombre, 3, 1), upper(substring(nombre, 4, 1)), substring(nombre, 5)) from usuarios;
```

Result Grid		Filter Rows:	Export:
	concat(substring(nombre, 1, 1), upper(substring(nombre, 2, 1)), substring(nombre, 3, 1), upper(substring(nombre, 4, 1)), substring(nombre, 5))		
▶	AlEjandro		
	VALeria		
	SANTiago		
	LUCía		
	FERnando		
	EMÍlia		
	JORge Luis		
	CAMILa		
	DANIel		

```
-- Convierte a mayúsculas el segundo y último carácter (Armando-->ArmandO)
select concat(substring(nombre, 1, 1), upper(substring(nombre, 2, 1)),
substring(nombre, 3, char_length(nombre)-1), upper(substring(nombre, -1))) from usuarios;
```

Result Grid		Filter Rows:	Export:
	concat(substring(nombre, 1, 1), upper(substring(nombre, 2, 1)), substring(nombre, 3, char_length(nombre)-1), upper(substring(nombre, -1)))		
	LUcíaA		
	FErnandoO		
	EMiliaA		
	JOrgé LuisS		
	CAmílaA		
	DAnielL		
	ANa SofíaA		
	RObertoO		
	ELenaA		

```
-- Convierte a mayúscula el segundo, cuarto y penúltimo carácter del nombre (Armando--> ARmAnDo)
select concat(substring(nombre, 1, 1), upper(substring(nombre, 2, 1)),
substring(nombre, 3, 1), upper(substring(nombre, 4, 1)), substring(nombre, 5, char_length(nombre)-3),
upper(substring(nombre, -2, 1)), substring(nombre, -1, 1)) from usuarios;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	concat(substring(nombre, 1, 1), upper(substring(nombre, 2, 1)), substring(nombre, 3, 1), upper(substring(nombre, 4, 1)), substring(nombre, 5, char_length(nombre)-3), upper(substring(nombre, -2, 1)), substring(nombre, -1, 1))			
▶	JUaN ManuelE			
	MArCelaLa			
	SEbAstiánÁn			
	ANdReaEa			
	JOsÉ LuisIs			
	PAuLaLa			
	CArLosOs			
	MÓNICAca			
	FRaNciscoCo			
	MArIanaNa			
	RAÚLÚl			
	LOrEnaNa			
	DAvIdId			

```
-- Selecciona los usuarios con correo electrónico registrado, pero
-- cuya dirección de email no incluya más de 10 caracteres antes del símbolo @
select * from usuarios where char_length(substring(email, 1, position('@' in email)-1)) <= 10;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rc
▶	12	Elena	Díaz	Moreno	elena.diaz@example.com	5553344556	1996-06-07	2025-06-18	PassElena123	3
	29	José Carlos	Vega	Padilla	jose.vega@example.com	5557788991	1993-06-15	2025-06-18	PassJoseC123	3
	39	Iván	León	Serrano	ivan.leon@example.com	5551122771	1988-06-03	2025-06-18	PassIvan123	3
	51	Juan	Pérez	López	juan.perez@biblioteca.com	5551234567	1980-03-15	2025-06-18	adminJuan123	1
	57	Sofia	Díaz	Moreno	sofia.diaz@biblioteca.com	5556543210	1988-02-28	2025-06-18	biblioSofia123	2

```
-- Usuarios cuyo nombre tenga exactamente dos palabras y cuyo rol esté entre 'administrador' y 'bibliotecario'
select * from usuarios where (LENGTH(TRIM(nombre)) -
LENGTH(REPLACE(TRIM(nombre), ' ', ''))) = 1 and id_rol between 1 and 2;
insert into usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, id_rol, contrasena)
values
('Pedro Eduardo', 'Zaragoza', 'Mendez', 'pedro.zaragoza@example.com', '5523456789', '1994-01-15', 2, 'PassPedro123');
select * from usuarios where (LENGTH(TRIM(nombre)) -
LENGTH(REPLACE(TRIM(nombre), ' ', ''))) = 1 and id_rol between 1 and 2;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol
------------	--------	-----------------	------------------	-------	----------	------------------	----------------	------------	--------

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

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	
▶	61	Pedro Eduardo	Zaragoza	Mendez	pedro.zaragoza@example.com	5523456789	1994-01-15	2025-06-18	PassPedro123	2

```
-- Usuarios cuyo nombre completo (nombre, primer apellido, segundo
-- apellido) tenga exactamente 30 caracteres, y cuya contraseña tenga al menos 8 caracteres
select * from usuarios
where char_length(concat(nombre, ' ', primer_apellido, ' ', segundo_apellido)) = 30 and char_length(contrasena) >= 8;
insert into usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, contrasena, id_rol)
values ('Jonathan', 'Fernández', 'Montenegros', 'jonathan.fernández@example.com', '5551234567', '1990-05-10', 'PassJonathan123', 2);
select * from usuarios
where char_length(concat(nombre, ' ', primer_apellido, ' ', segundo_apellido)) = 30 and char_length(contrasena) >= 8;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
	62	Jonathan	Fernández	Montenegros	jonathan.fernández@example.com	5551234567	1990-05-10	2025-06-18	PassJonathan1



```
-- Inciso b)
-- Las personas que se llamen Eduardo sin importar que tengan 2 nombres.
select * from usuarios where nombre like 'Eduardo%' or nombre like '% Eduardo%' or nombre = 'Eduardo';
insert into usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, contrasena, id_rol
)
values ('Luis Eduardo', 'Fernández', 'Montenegros', 'luis.fernandez@example.com', '5551234567', '1990-05-10', 'PassLuis123
select * from usuarios where nombre like 'Eduardo%' or nombre like '% Eduardo%' or nombre = 'Eduardo';
```

id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
60	Eduardo	Zaragoza	NULL	eduardo.zaragoza@example.com	5512345678	1995-09-15	2025-06-18	PassEduardo123
61	Pedro Eduardo	Zaragoza	Mendez	pedro.zaragoza@example.com	5523456789	1994-01-15	2025-06-18	PassPedro123

id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
60	Eduardo	Zaragoza	NULL	eduardo.zaragoza@example.com	5512345678	1995-09-15	2025-06-18	PassEduardo123
61	Pedro Eduardo	Zaragoza	Mendez	pedro.zaragoza@example.com	5523456789	1994-01-15	2025-06-18	PassPedro123
63	Luis Eduardo	Fernández	Montenegros	luis.fernandez@example.com	5551234567	1990-05-10	2025-06-18	PassLuis123

```
-- Las personas que su segundo carácter sea una "d".
```

```
select * from usuarios where nombre like '_d%';
```

id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
26	Adriana	Moreno	Castillo	adriana.moreno@example.com	5551239876	1991-09-13	2025-06-18	PassAdriana123
60	Eduardo	Zaragoza	NULL	eduardo.zaragoza@example.com	5512345678	1995-09-15	2025-06-18	PassEduardo123

```
-- Los que no empiecen su nombre con una vocal.
```

```
select * from usuarios where nombre not regexp '^[AEIOUaeiou]';
```

id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18	2025-06-18	PassValeria12
3	Santiago	Ramos	Ortega	santiago.ramos@example.com	5554567890	1988-11-23	2025-06-18	PassSantiago
4	Lucía	Morales	Jiménez	lucia.morales@example.com	5551234567	1995-05-06	2025-06-18	PassLucia123
5	Fernando	Vázquez	Pérez	fernando.vazquez@example.com	5559871234	1994-02-17	2025-06-18	PassFernando
7	Jorge Luis	Méndez	Flores	jorge.mendez@example.com	5552233445	1989-03-12	2025-06-18	PassJorge12
8	Camila	Gutiérrez	Sánchez	camila.gutierrez@example.com	5555544332	1993-09-21	2025-06-18	PassCamila12
9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28	2025-06-18	PassDaniel12
11	Roberto	López	Romero	roberto.lopez@example.com	5551122777	1986-01-05	2025-06-18	PassRoberto
13	Juan Manuel	García	Torres	juan.garcia@example.com	5557788990	1990-08-22	2025-06-18	PassJuanM12

```
-- Los que empiecen su nombre con una vocal y terminen con s.
select * from usuarios where nombre regexp '^[AEIOUaeiou].*s$';
) insert into usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, contrasena, id_rol
values ('Alexis', 'Hernández', 'Mendéz', 'alexis.hernandez@example.com', '5534567891', '1991-02-10', 'PassAlexis123', 3);
select * from usuarios where nombre regexp '^[AEIOUaeiou].*s$';
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol
64	Alexis	Hernández	Mendéz	alexis.hernandez@example.com	5534567891	1991-02-10	2025-06-18	PassAlexis123	

```
-- Los que su tercer carácter del nombre sea una G.
select * from usuarios where nombre like '__G%';
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol
37	Ángel	Valencia	Campos	angel.valencia@example.com	5554433228	1994-09-05	2025-06-18	PassAngel123	3

```
-- Los que su primer carácter en el apellido paterno sea 'E' y el 4 sea 'A'
select * from usuarios where primer_apellido like 'E__A%';
) insert into usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, contrasena, id_rol
values ('Luis', 'Edwards', null, 'luis.edwards@example.com', '5545678912', '1992-08-13', 'PassLuis123', 1);
select * from usuarios where primer_apellido like 'E__A%';
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol
65	Luis	Edwards	NULL	luis.edwards@example.com	5545678912	1992-08-13	2025-06-18	PassLuis123	1



```
-- Los que tengan por lo menos una 'E' en su nombre.
select * from usuarios where nombre like '%E%';
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	1	Alejandro	Martínez	Vega	alejandro.martinez@example.com	5551239876	1991-08-13	2025-06-18	PassAlejand
	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18	2025-06-18	PassValeria1
	5	Fernando	Vázquez	Pérez	fernando.vazquez@example.com	5559871234	1994-02-17	2025-06-18	PassFernanc
	6	Emilia	Castro	Rivera	emilia.castro@example.com	5551122334	1992-07-25	2025-06-18	PassEmilia12
	7	Jorge Luis	Méndez	Flores	jorge.mendez@example.com	5552233445	1989-03-12	2025-06-18	PassJorge12
	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28	2025-06-18	PassDaniel1
	11	Roberto	López	Romero	roberto.lopez@example.com	5551122777	1986-01-05	2025-06-18	PassRoberto
	12	Elena	Díaz	Moreno	elena.diaz@example.com	5553344556	1996-06-07	2025-06-18	PassElena12
	13	Juan Manuel	García	Torres	juan.garcia@example.com	5557788990	1990-08-22	2025-06-18	PassJuanM1

```
-- Los que se llaman Eduardo y Cualquiera de sus apellidos empiece con 'C'
select * from usuarios where nombre like 'Eduardo%'
and (primer_apellido like 'C%' or segundo_apellido like 'C%');
insert into usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, contrasena, id_rol
)
values ('Eduardo', 'Castro', null, 'eduardo.castro@example.com', '5556789123', '1995-10-24', 'PassEduardo123', 1);
select * from usuarios where nombre like 'Eduardo%'
and (primer_apellido like 'C%' or segundo_apellido like 'C%');
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	66	Eduardo	Castro	NULL	eduardo.castro@example.com	5556789123	1995-10-24	2025-06-18	PassEduardo123

```
-- Las personas que su apellido materno empiece con la primera
-- mitad del alfabeto [A-M] pero que no empiecen ni con A ni con E
select * from usuarios where segundo_apellido regexp '^[B-DF-Mb-df-m]';
```

Result Grid									
Filter Rows:									
Export:									
Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	4	Lucía	Morales	Jiménez	lucia.morales@example.com	5551234567	1995-05-06	2025-06-18	PassLucia123
	7	Jorge Luis	Méndez	Flores	jorge.mendez@example.com	5552233445	1989-03-12	2025-06-18	PassJorge123
	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28	2025-06-18	PassDaniel12
	10	Ana Sofía	Hernández	Gómez	ana.hernandez@example.com	5558899000	1985-12-15	2025-06-18	PassAnaSofia
	12	Elena	Díaz	Moreno	elena.diaz@example.com	5553344556	1996-06-07	2025-06-18	PassElena123
	15	Sebastián	Ortiz	Martínez	sebastian.ortiz@example.com	5554433221	1993-11-01	2025-06-18	PassSebastian
	17	José Luis	Ramírez	Morales	jose.ramirez@example.com	5557766554	1989-04-11	2025-06-18	PassJoseLuis
	18	Paula	Fuentes	Guzmán	paula.fuentes@example.com	5553344221	1991-12-14	2025-06-18	PassPaula123
	21	Francisco	Navarro	García	francisco.navarro@example.com	5557766553	1993-01-10	2025-06-18	PassFrancisco

```
-- Las personas que su apellido paterno empiece con la segunda mitad del alfabeto [N-Z]
select * from usuarios where primer_apellido regexp '^[N-Zn-z]';
```

Result Grid									
Filter Rows:									
Export:									
Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18	2025-06-18	PassValeria1
	3	Santiago	Ramos	Ortega	santiago.ramos@example.com	5554567890	1988-11-23	2025-06-18	PassSantiago
	5	Fernando	Vázquez	Pérez	fernando.vazquez@example.com	5559871234	1994-02-17	2025-06-18	PassFernando
	14	Marcela	Rojas	Pacheco	marcela.rojas@example.com	5559988776	1995-02-09	2025-06-18	PassMarcela
	15	Sebastián	Ortiz	Martínez	sebastian.ortiz@example.com	5554433221	1993-11-01	2025-06-18	PassSebastian
	17	José Luis	Ramírez	Morales	jose.ramirez@example.com	5557766554	1989-04-11	2025-06-18	PassJoseLuis
	20	Mónica	Román	Pérez	monica.roman@example.com	5556677889	1994-07-23	2025-06-18	PassMonica1
	21	Francisco	Navarro	García	francisco.navarro@example.com	5557766553	1993-01-10	2025-06-18	PassFrancisco
	24	Lorena	Rivas	Serrano	lorena.rivas@example.com	5556677884	1995-08-09	2025-06-18	PassLorena1

```
-- Obtener los libros cuyo título contenga la palabra "Historia".
select * from libros where titulo like '%Historia%';
insert into Libros (isbn, titulo, anio_publicacion, editorial, edicion, cantidad_total, cantidad_disponible, clasificacion)
values
('978-9-68-134266-1', 'La otra historia de México', 2006, 'Editorial Ariel', 1, 10, 7, 4);
select * from libros where titulo like '%Historia%';
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

id_libro	ISBN	titulo	anio_publicacion	editorial	edicion	cantidad_total	cantidad_disponible	clasificacion
----------	------	--------	------------------	-----------	---------	----------------	---------------------	---------------

Result Grid									
Filter Rows:									
Export:									
Wrap Cell Content:									
	id_libro	ISBN	titulo	anio_publicacion	editorial	edicion	cantidad_total	cantidad_disponible	clasificacion
▶	48	978-9-68-134266-1	La otra historia de México	2006	Editorial Ariel	1	10	7	4

```
-- Listar los usuarios cuyo apellido paterno termine con "ez".
select * from usuarios where primer_apellido like '%ez';
```

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	1	Alejandro	Martínez	Vega	alejandro.martinez@example.com	5551239876	1991-08-13	2025-06-18	PassAlejand
	5	Fernando	Vázquez	Pérez	fernando.vazquez@example.com	5559871234	1994-02-17	2025-06-18	PassFernanc
	7	Jorge Luis	Méndez	Flores	jorge.mendez@example.com	5552233445	1989-03-12	2025-06-18	PassJorge12
	8	Camila	Gutiérrez	Sánchez	camila.gutierrez@example.com	5555544332	1993-09-21	2025-06-18	PassCamila1
	10	Ana Sofía	Hernández	Gómez	ana.hernandez@example.com	5558899000	1985-12-15	2025-06-18	PassAnaSofi
	11	Roberto	López	Romero	roberto.lopez@example.com	5551122777	1986-01-05	2025-06-18	PassRoberto
	17	José Luis	Ramírez	Morales	jose.ramirez@example.com	5557766554	1989-04-11	2025-06-18	PassJoseLuis
	19	Carlos	Martínez	Ríos	carlos.martinez@example.com	5554455667	1988-03-19	2025-06-18	PassCarlos1
	23	Raúl	Domínguez	Vargas	raul.dominguez@example.com	5553344223	1987-06-27	2025-06-18	PassRaul123

```
-- Seleccionar todos los usuarios cuyo correo electrónico sea de Gmail.
select * from usuarios where email like '%@gmail.com';
insert into usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, contrasena, id_rol)
values ('Raul', 'Rodríguez', null, 'raul.rodriquez@gmail.com', '5551234567', '1998-12-24', 'PassRaul123', 2);
select * from usuarios where email like '%@gmail.com';
```

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol
--	------------	--------	-----------------	------------------	-------	----------	------------------	----------------	------------	--------

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena	id_rol
▶	67	Raul	Rodríguez	NULL	raul.rodriquez@gmail.com	5551234567	1998-12-24	2025-06-18	PassRaul123	2

```
-- Consultar los libros cuya editorial contenga la palabra "Editores".
select * from libros where editorial like '%Editores%';
insert into libros (ISBN, titulo, anio_publicacion, editorial, edicion, cantidad_total, cantidad_disponible, clasificacion)
value ('978-607-00-1234-5', 'Introducción a la Programación', 2021, 'Alfa Editores Técnicos', '3', 10, 7, '005.13');
select * from libros where editorial like '%Editores%';
```

	id_libro	ISBN	titulo	anio_publicacion	editorial	edicion	cantidad_total	cantidad_disponible	clasificacion
--	----------	------	--------	------------------	-----------	---------	----------------	---------------------	---------------

	id_libro	ISBN	titulo	anio_publicacion	editorial	edicion	cantidad_total	cantidad_disponible	clasificacion
▶	49	978-607-00-1234-5	Introducción a la Programación	2021	Alfa Editores Técnicos	3	10	7	5



```
-- Listar los usuarios cuyos nombres contengan una vocal seguida de una "n".
select * from usuarios where nombre regexp '[AEIOUaeiou]n';
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	1	Alejandro	Martínez	Vega	alejandro.martinez@example.com	5551239876	1991-08-13	2025-06-18	PassAlejand
	3	Santiago	Ramos	Ortega	santiago.ramos@example.com	5554567890	1988-11-23	2025-06-18	PassSantiag
	5	Fernando	Vázquez	Pérez	fernando.vazquez@example.com	5559871234	1994-02-17	2025-06-18	PassFernanc
	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28	2025-06-18	PassDaniel1
	10	Ana Sofía	Hernández	Gómez	ana.hernandez@example.com	5558899000	1985-12-15	2025-06-18	PassAnaSofi
	12	Elena	Díaz	Moreno	elena.diaz@example.com	5553344556	1996-06-07	2025-06-18	PassElena12
	13	Juan Manuel	García	Torres	juan.garcia@example.com	5557788990	1990-08-22	2025-06-18	PassJuanM1
	16	Andrea	Mendoza	Aguirre	andrea.mendoza@example.com	5556655447	1998-05-30	2025-06-18	PassAndrea
	21	Francisco	Navarro	García	francisco.navarro@example.com	5557766553	1993-01-10	2025-06-18	PassFrancisc

```
-- Obtener todos los usuarios cuyos números de teléfono terminen en "00".
select * from usuarios where telefono like '%00';
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	10	Ana Sofía	Hernández	Gómez	ana.hernandez@example.com	5558899000	1985-12-15	2025-06-18	PassAnaSofia123

```
-- Consultar los autores cuyo nombre contenga la letra "j" en cualquier parte (ya sea minúscula o mayúsculas).
select * from autores where nombre_autor like '%J%' or nombre_autor like '%j%';
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content:				
	id_autor	nombre_autor	apellido_paterno	apellido_materno
▶	5	Jane	Austen	NULL
	11	Julio	Cortázar	NULL
	12	J.D.	Salinger	NULL
	13	J.R.R.	Tolkien	NULL
	14	J.R.R.	Tolkien	NULL
	15	J.K.	Rowling	NULL
	29	James	Joyce	NULL
	31	Jonathan	Swift	NULL
	40	Alejandro	Dumas	NULL
	41	John	Milton	NULL

```
-- Obtener los usuarios cuyo nombre contenga exactamente cinco caracteres.
select * from usuarios where CHAR_LENGTH(nombre) = 5;
```

Result Grid									
Filter Rows:									
Export:									
Wrap Cell Content:									
	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
▶	4	Lucía	Morales	Jiménez	lucia.morales@example.com	5551234567	1995-05-06	2025-06-18	PassLucia123
	12	Elena	Díaz	Moreno	elena.diaz@example.com	5553344556	1996-06-07	2025-06-18	PassElena123
	18	Paula	Fuentes	Guzmán	paula.fuentes@example.com	5553344221	1991-12-14	2025-06-18	PassPaula123
	25	David	Estrada	Mejía	david.estrada@example.com	5554455669	1990-02-26	2025-06-18	PassDavid123
	34	Laura	Gutiérrez	Rodríguez	laura.gutierrez@example.com	5556655443	1993-04-08	2025-06-18	PassLaura123
	35	Oscar	Martínez	Santos	oscar.martinez@example.com	5551122333	1990-10-17	2025-06-18	PassOscar123
	37	Ángel	Valencia	Campos	angel.valencia@example.com	5554433228	1994-09-05	2025-06-18	PassAngel123
	47	Karla	Castillo	Arias	karla.castillo@example.com	5551122773	1995-02-07	2025-06-18	PassKarla123
	50	Diego	Pérez	López	diego.perez@example.com	5557788992	1990-03-18	2025-06-18	PassDiego123

```
-- Obtener los títulos de libros que contengan números en cualquier parte.
select * from libros where titulo regexp '[0-9]';
```

Result Grid									
Filter Rows:									
Export:									
Wrap Cell Content:									
	id_libro	ISBN	titulo	anio_publicacion	editorial	edicion	cantidad_total	cantidad_disponible	clasificacion
▶	3	978-0-452-28423-4	1984	1949	Secker & Warburg	1	12	10	5
	4	978-0-7432-7356-5	Fahrenheit 451	1953	Ballantine Books	1	6	4	4