



Instituto Politécnico Nacional

ESCOM



LIC. CIENCIA DE DATOS

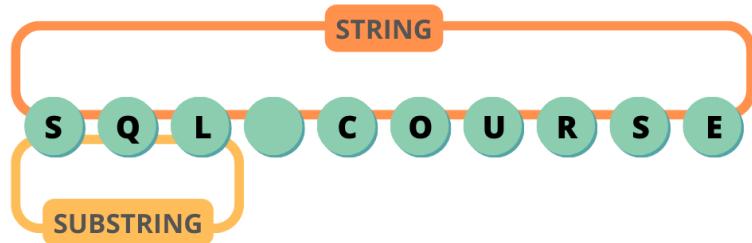
## BASES DE DATOS

Prof. Iván Blanco Almazán

Fernanda Madariaga Villanueva

Grupo 3AV1

### STRING & SUBSTRING



Nombre de actividad: **CADENAS**

Fecha: 18-06-2025

```

1  -- SCRIPT DE CONSULTAS SQL - FUNCIONES DE CADENAS
2  -- 1. FUNCIONES DE CADENAS
3
4  -- Muestra solo a las personas que su nombre tiene mas de 5 letras
5  SELECT * FROM usuarios
6  WHERE LENGTH(nombre) > 5;

```

Data Output Messages Notifications

	<b>id_usuario</b> [PK] integer	<b>nombre</b> character varying (50)	<b>primer_apellido</b> character varying (50)	<b>segundo_apellido</b> character varying (50)	<b>email</b> character varying (100)	<b>telefono</b> character varying (20)
1	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345
2	6	Emilia	Castro	Rivera	emilia.castro@example.com	5551122334

```

8  -- Muestra a las personas que su nombre tiene entre 5 y 7 caracteres
9  SELECT * FROM usuarios
10 WHERE LENGTH(nombre) BETWEEN 5 AND 7;

```

Data Output Messages Notifications

	<b>id_usuario</b> [PK] integer	<b>nombre</b> character varying (50)	<b>primer_apellido</b> character varying (50)	<b>segundo_apellido</b> character varying (50)	<b>email</b> character varying (100)	<b>telefono</b> character varying (20)	<b>fecha_nacimiento</b> date
1	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18
2	4	Lucia	Morales	Jiménez	lucia.morales@example.com	5551234567	1995-05-06
3	6	Emilia	Castro	Rivera	emilia.castro@example.com	5551122334	1992-07-25
4	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28

```

12  -- Muestra a las personas que su nombre tiene mas de 7 caracteres y alguno de sus apellidos tenga entre 5 y 7 caracteres
13  SELECT * FROM usuarios
14  WHERE LENGTH(nombre) > 7 |
15  AND (LENGTH(primer_apellido) BETWEEN 5 AND 7 OR LENGTH(segundo_apellido) BETWEEN 5 AND 7);

```

Data Output Messages Notifications

	<b>id_usuario</b> [PK] integer	<b>nombre</b> character varying (50)	<b>primer_apellido</b> character varying (50)	<b>segundo_apellido</b> character varying (50)	<b>email</b> character varying (100)	<b>telefono</b> character varying (20)	<b>fecha_nacimiento</b> date	<b>fecha</b> date
1	21	Francisco	Navarro	García	francisco.navarro@example.com	5557766553	1993-01-10	2025
2	29	José Carlos	Vega	Padilla	jose.vega@example.com	5557788991	1993-06-15	2025
3	32	Cristina	Salazar	Méndez	cristina.salazar@example.com	5554455664	1992-12-07	2025

```

17  -- Muestra los primeros tres caracteres del nombre
18  SELECT nombre, LEFT(nombre, 3) AS primeros_tres_caracteres
19  FROM usuarios;
20

```

Data Output Messages Notifications

	<b>nombre</b> character varying (50)	<b>primeros_tres_caracteres</b> text
1	Valeria	Val
2	Lucía	Luc
3	Emilia	Emi

```

21  -- Muestra los ultimos 3 caracteres del nombre
22  SELECT nombre, RIGHT(nombre, 3) AS ultimos_tres_caracteres
23  FROM usuarios;
24

```

Data Output Messages Notifications

	<b>nombre</b> character varying (50)	<b>ultimos_tres_caracteres</b> text
1	Valeria	ria
2	Lucía	cía
3	Emilia	lia
4	Daniel	iel
5	Roberto	rto

```

25 -- Muestra del 2do al 5to caracter del nombre
26 ✓ SELECT nombre, SUBSTRING(nombre, 2, 4) AS caracteres_2_al_5
27 FROM usuarios;

```

Data Output Messages Notifications

Showing rows: 1 to 61 Page No: 1 of 1

	nombre character varying (50)	caracteres_2_al_5 text
1	Valeria	aler
2	Lucía	ucía
3	Emilia	mili

```

29 -- Reemplaza las d por s en nombre
30 ✓ SELECT nombre, REPLACE(nombre, 'd', 's') AS nombre_con_s
31 FROM usuarios;

```

Data Output Messages Notifications

Showing rows: 1 to 61 Page No: 1 of 1

	nombre character varying (50)	nombre_con_s text
7	Andrea	Ansrea
8	Carlos	Carlos
9	Francisco	Francisco
10	Lorena	Lorena
11	David	Davis
12	Adriana	Asriana

```

33 -- Obten la longitud del apellido paterno
34 ✓ SELECT primer_apellido, LENGTH(primer_apellido) AS longitud_apellido_paterno
35 FROM usuarios;

```

Data Output Messages Notifications

Showing rows: 1 to 61 Page No: 1 of 1

	primer_apellido character varying (50)	longitud_apellido_paterno integer
1	Reyes	5
2	Morales	7
3	Castro	6
4	Cruz	4
5	López	5

```

37 -- Muestra en mayusculas el nombre
38 ✓ SELECT nombre, UPPER(nombre) AS nombre_mayusculas
39 FROM usuarios;

```

Data Output Messages Notifications

Showing rows: 1 to 61 Page No: 1 of 1

	nombre character varying (50)	nombre_mayusculas text
1	Valeria	VALERIA
2	Lucía	LUCÍA
3	Emilia	EMILIA

```

41 -- Muestra en minusculas el apellido paterno
42 ✓ SELECT primer_apellido, LOWER(primer_apellido) AS apellido_minusculas
43 FROM usuarios;
44

```

Data Output Messages Notifications

Showing rows: 1 to 61 Page No: 1 of 1

	primer_apellido character varying (50)	apellido_minusculas text
1	Reyes	reyes
2	Morales	morales
3	Castro	castro

```

45 -- Muestra el nombre completo empezando por el apellido paterno con mayusculas en una sola columna
46 v SELECT UPPER(primer_apellido) || ' ' ||
47     COALESCE(segundo_apellido || ' ', '') || 
48     nombre AS nombre_completo
49 FROM usuarios;

```

Data Output Messages Notifications

	nombre_completo	text
1	REYES Soto Valeria	
2	MORALES Jiménez Lucia	
3	CASTRO Rivera Emilia	

```

51 -- Muestra el nombre de las personas con las E reemplazadas con el numero 3
52 v SELECT nombre, REPLACE(REPLACE(nombre, 'E', '3'), 'e', '3') AS nombre_con_3
53 FROM usuarios;

```

Data Output Messages Notifications

	nombre	character varying (50)	nombre_con_3	text
1	Valeria		Val3ria	
2	Lucía		Lucia	
3	Emilia		3milia	

```

55 -- Muestra el nombre completo de las personas con las o reemplazados con el numero 0 en una sola columna
56 v SELECT REPLACE(REPLACE(nombre || ' ' || primer_apellido || ' ' || COALESCE(segundo_apellido, ''), 'o', '0'), 'o', '0') AS nombre_completo_con_0
57 FROM usuarios;
58

```

Data Output Messages Notifications

	nombre_completo_con_0	text
1	Valeria Reyes S0t0	
2	Lucia M0rales Jiménez	
3	Emilia Castr0 Rivera	

```

59 -- Libros con edicion entre 1 y 5 y cuyo titulo comience con letra mayuscula
60 v SELECT * FROM libros
61 WHERE edicion BETWEEN 1 AND 5
62 AND titulo ~ '^([A-Z])';
63

```

Data Output Messages Notifications

	id_libro	[PK] integer	isbn	character varying (20)	titulo	character varying (255)	anio_publicacion	integer	editorial	character varying (100)	edicion	integer	cantidad_total	integer
1			1	978-3-16-148410-0	Cien Años de Soledad			1967	Sudamericana			1		
2			2	978-84-376-0494-7	Don Quijote de la Mancha			1605	Francisco de Robles			2		
3			4	978-0-7432-7356-5	Fahrenheit 451			1953	Ballantine Books			1		
4			5	978-0-19-953556-9	Orgullo y Prejuicio			1813	T. Egerton			3		
5			6	978-0-06-112008-4	Matar a un Ruisenor			1960	J.B. Lippincott & Co.			1		
6			7	978-1-85715-091-6	Crimen y Castigo			1866	The Russian Messenger			2		

```

64 -- Usuarios cuyo segundo apellido sea NULL y cuyo primer apellido tenga mas de 6 caracteres
65 v SELECT * FROM usuarios
66 WHERE segundo_apellido IS NULL
67 AND LENGTH(primer_apellido) > 6;
68

```

Data Output Messages Notifications

	id_usuario	[PK] integer	nombre	character varying (50)	primer_apellido	character varying (50)	segundo_apellido	character varying (50)	email	character varying (100)	telefono	character varying (20)	fecha_nacimiento	date
1			8	Camila	Gutiérrez		[null]		camila.gutierrez@example.com	5555544332		1993-09-21		
2			18	Paula	Fuentes		[null]		paula.fuentes@example.com	5553344221		1991-12-14		
3			23	Raúl	Dominguez		[null]		raul.dominguez@example.com	5553344223		1987-06-27		

```

69 -- Muestra el nombre de las personas reemplazando los siguientes caracteres A-->@,E-->3,I-->!, O-->0
70 v SELECT nombre,
71      REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(nombre, 'A', '@'), 'a', '@'), 'E', '3'), 'e', '3'), 'I', '!'),
72 FROM usuarios;
73

```

Data Output Messages Notifications

Showing rows: 1 to 61 | | Page No: 1 of 1 |

	nombre character varying (50)	nombre_codificado text
1	Valeria	V@l3rl@
2	Lucía	Lucí@
3	Emilia	3mll!@
4	Daniel	D@n3l

```

74 -- Convierte los primeros tres caracteres del nombre en mayuscula
75 v SELECT nombre,
76      UPPER(LEFT(nombre, 3)) || SUBSTRING(nombre, 4) AS nombre_modificado
77 FROM usuarios
78 WHERE LENGTH(nombre) >= 3;

```

Data Output Messages Notifications

Showing rows: 1 to 61 | | Page No: 1 of 1 |

	nombre character varying (50)	nombre_modificado text
1	Valeria	VALeria
2	Lucía	LUCcía
3	Emilia	EMilia
4	Daniel	DANiel

```

80 -- Convierte el ultimo caracter del nombre en mayusculas
81 v SELECT nombre,
82      LEFT(nombre, LENGTH(nombre) - 1) || UPPER(RIGHT(nombre, 1)) AS nombre_modificado
83 FROM usuarios
84 WHERE LENGTH(nombre) >= 1;

```

Data Output Messages Notifications

Showing rows: 1 to 61 | | Page No: 1 of 1 |

	nombre character varying (50)	nombre_modificado text
1	Valeria	ValerIA
2	Lucía	LucíA
3	Emilia	EmiliA

```

86 -- Convierte el 3er caracter del nombre en Mayuscula
87 v SELECT nombre,
88      LEFT(nombre, 2) || UPPER(SUBSTRING(nombre, 3, 1)) || SUBSTRING(nombre, 4) AS nombre_modificado
89 FROM usuarios
90 WHERE LENGTH(nombre) >= 3;

```

Data Output Messages Notifications

Showing rows: 1 to 61 | | Page No: 1 of 1 |

	nombre character varying (50)	nombre_modificado text
1	Valeria	VaLeria
2	Lucía	LuCía
3	Emilia	EmlliA

```

92 -- Convierte el 2do y 4to caracter del nombre a Mayuscula
93 v SELECT nombre,
94     LEFT(nombre, 1) ||
95     UPPER(SUBSTRING(nombre, 2, 1)) ||
96     SUBSTRING(nombre, 3, 1) ||
97     UPPER(SUBSTRING(nombre, 4, 1)) ||
98     SUBSTRING(nombre, 5) AS nombre_modificado
99 FROM usuarios
100 WHERE LENGTH(nombre) >= 4;
101

```

Data Output Messages Notifications

	nombre	nombre_modificado
1	Valeria	VAleRia
2	Lucía	LUCÍa
3	Emilia	EMILia
4	Daniel	DAneL

```

102 -- Convierte a mayusculas el segundo y ultimo caracter (el primero es por default mayúscula)
103 v SELECT nombre,
104     LEFT(nombre, 1) ||
105     UPPER(SUBSTRING(nombre, 2, 1)) ||
106     SUBSTRING(nombre, 3, LENGTH(nombre) - 3) ||
107     UPPER(RIGHT(nombre, 1)) AS nombre_modificado
108 FROM usuarios
109 WHERE LENGTH(nombre) >= 2;
110

```

Data Output Messages Notifications

	nombre	nombre_modificado
1	Valeria	VALeriA
2	Lucía	LUciA
3	Emilia	EMiliA

```

111 -- Convierte a mayuscula el segundo, cuarto y penultimo caracter del nombre
112 v SELECT nombre,
113     LEFT(nombre, 1) ||
114     UPPER(SUBSTRING(nombre, 2, 1)) ||
115     SUBSTRING(nombre, 3, 1) ||
116     UPPER(SUBSTRING(nombre, 4, 1)) ||
117     SUBSTRING(nombre, 5, GREATEST(LENGTH(nombre)-5, 0)) ||
118     UPPER(SUBSTRING(nombre, LENGTH(nombre)-2, 1)) ||
119     RIGHT(nombre, 1) AS nombre_modificado
120 FROM usuarios
121 WHERE LENGTH(nombre) >= 5;
122

```

Data Output Messages Notifications

	nombre	nombre_modificado
1	Valeria	VALEriRa
2	Lucía	LUCÍCa
3	Emilia	EMILiLa

```

125 -- Selecciona los usuarios con correo electronico registrado, pero cuya direccion de email no incluya mas de 10 caracteres al
126 v SELECT * FROM usuarios
127 WHERE email IS NOT NULL
128 AND LENGTH(SPLIT_PART(email, '@', 1)) <= 10;
129

```

Data Output Messages Notifications

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro
1	29	José Carlos	Vega	Padilla	jose.vega@example.com	5557788991	1993-06-15	2025-06-12
2	39	Ivan	León	Serrano	ivan.leon@example.com	5551122771	1988-06-03	2025-06-12
3	51	Juan	Pérez	López	juan.perez@biblioteca.com	5551234567	1980-03-15	2025-06-12
4	54	Eduardo	Lopez	Alcantara	edu.lopez@example.com	5566843987	1999-08-13	2025-06-12
5	12	Elena	Díaz	Moreno	elena.diaz@example.com	5553344556	[null]	2025-06-12

```

136 -- Usuarios cuyo nombre completo tenga exactamente 30 caracteres, y cuya contraseña tenga al menos 8 caracteres
137 ✓ INSERT INTO usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, id_rol, contraseña)
138   VALUES
139   ('Fernanda', 'Villanueva', 'Montenegro', 'fernanda.montenegro@example.com', '5633998273', '2005-06-19', 3, 'PassFernanda123')
140 ✓ SELECT * FROM usuarios
141 WHERE LENGTH(nombre || ' ' || primer_apellido || ' ' || COALESCE(segundo_apellido, '')) = 30
142 AND LENGTH(contraseña) >= 8;
143

```

Data Output Messages Notifications

Showing rows: 1 to 1 | | Page No: 1 of 1 |

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date	<code>fecha_registro</code> date
1	65	Fernanda	Villanueva	Montenegro	fernanda.montenegro@example.com	5633998273	2005-06-19	202

```

146 -- Las personas que se llamen Eduardo sin importar que tengan 2 nombres
147 ✓ SELECT * FROM usuarios
148 WHERE nombre LIKE '%Eduardo%';
149

```

Data Output Messages Notifications

Showing rows: 1 to 1 | | Page No: 1 of 1 |

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date	<code>fecha_registro</code> date
1	54	Eduardo	Lopez	Alcantara	edu.lopez@example.com	5566843987	1999-08-13	2025-06-12

```

150 -- Las personas que su segundo caracter sea una "d"
151 ✓ SELECT * FROM usuarios
152 WHERE nombre LIKE '_d%';
153

```

Data Output Messages Notifications

Showing rows: 1 to 2 | | Page No: 1 of 1 |

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date	<code>fecha_registro</code> date
1	26	Adriana	Moreno	Castillo	adriana.moreno@example.com	5551239876	1991-09-13	2025-06-12
2	54	Eduardo	Lopez	Alcantara	edu.lopez@example.com	5566843987	1999-08-13	2025-06-12

```

154 -- Los que no empiecen su nombre con una vocal
155 ✓ SELECT * FROM usuarios
156 WHERE nombre NOT SIMILAR TO '[AEIOUaeiou]%';
157

```

Data Output Messages Notifications

Showing rows: 1 to 50 | | Page No: 1 of 1 |

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date	<code>fecha_registro</code> date
1	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18	2025-06-12
2	4	Lucia	Morales	Jiménez	lucia.morales@example.com	5551234567	1995-05-06	2025-06-12
3	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28	2025-06-12
4	11	Roberto	López	Romero	roberto.lopez@example.com	5551122777	1986-01-05	2025-06-12

```

158 -- Los que empiecen su nombre con una vocal y terminen con s
159 -- aqui no habia entonces inserte
160 ✓ INSERT INTO usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, id_rol, contraseña)
161   VALUES
162   ('Alexis', 'Bettencourt', 'Montenegro', 'alexis.bettencourt@example.com', '5633927273', '2005-06-29', 3, 'PassAlexis123');
163 ✓ SELECT * FROM usuarios
164 WHERE nombre SIMILAR TO '[AEIOUaeiou][sS]';
165

```

Data Output Messages Notifications

Showing rows: 1 to 1 | | Page No: 1 of 1 |

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date	<code>fecha_registro</code> date
1	66	Alexis	Bettencourt	Montenegro	alexis.bettencourt@example.com	5633927273	2005-06-29	2025-06-12

```

1 -- Los que su tercer caracter del nombre sea una G
2 -- no habia entonces insertar
3 v INSERT INTO usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, id_rol, contrasena)
4 VALUES
5 ('Logan', 'Mercuri', 'Vega', 'logan.mercuri@example.com', '5598269876', '1991-08-13', 3, 'PassLogan123');
6 v SELECT * FROM usuarios
7 WHERE nombre SIMILAR TO '_-[Gg]%';
```

Data Output Messages Notifications

Showing rows: 1 to 4 |  Page No: 1 of 1 |

	<b>id_usuario</b> [PK] integer	<b>nombre</b> character varying (50)	<b>primer_apellido</b> character varying (50)	<b>segundo_apellido</b> character varying (50)	<b>email</b> character varying (100)	<b>telefono</b> character varying (20)	<b>fecha_nacimiento</b> date	<b>fecha_registro</b> date
1	37	Ángel	Valencia	Campos	angel.valencia@example.com	5554433228	1994-09-05	2025-06-1
2	60	Miguel	Ruiz	Torres	miguel.ruiz@example.com	5553456789	1991-07-18	2025-06-1
3	67	Begona	Molina	Antana	begona.molina@example.com	8293927273	2004-06-29	2025-06-1
4	69	Logan	Mercuri	Vega	logan.mercuri@example.com	5598269876	1991-08-13	2025-06-1

```

9 -- Los que su primer caracter en el apellido paterno sea 'E' y el 4 sea 'A'
10 --NO HABIA entonces insertar
11 v INSERT INTO usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, id_rol, contrasena)
12 VALUES
13 ('Morgan', 'Estaño', 'Hull', 'morgan.estano@example.com', '5998269876', '1999-09-17', 3, 'PassMorgan123');
14 v SELECT * FROM usuarios
15 WHERE primer_apellido SIMILAR TO 'E-[Aa]%';
```

Data Output Messages Notifications

Showing rows: 1 to 1 |  Page No: 1 of 1 |

	<b>id_usuario</b> [PK] integer	<b>nombre</b> character varying (50)	<b>primer_apellido</b> character varying (50)	<b>segundo_apellido</b> character varying (50)	<b>email</b> character varying (100)	<b>telefono</b> character varying (20)	<b>fecha_nacimiento</b> date	<b>fecha_registro</b> date
1	70	Morgan	Estaño	Hull	morgan.estano@example.com	5998269876	1999-09-17	2025-06-1

```

17 -- Los que tengan por lo menos una 'E' en su nombre
18 v SELECT * FROM usuarios
19 WHERE nombre LIKE '%E%' OR nombre LIKE '%e%';
```

Data Output Messages Notifications

Showing rows: 1 to 33 |  Page No: 1 of 1 |

	<b>id_usuario</b> [PK] integer	<b>nombre</b> character varying (50)	<b>primer_apellido</b> character varying (50)	<b>segundo_apellido</b> character varying (50)	<b>email</b> character varying (100)	<b>telefono</b> character varying (20)	<b>fecha_nacimiento</b> date	<b>fecha_registro</b> date
1	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18	20
2	6	Emilia	Castro	Rivera	emilia.castro@example.com	5551122334	1992-07-25	20
3	9	Daniel	Cruz	Martinez	daniel.cruz@example.com	5556677889	1997-10-28	20
4	11	Roberto	López	Romero	roberto.lopez@example.com	5551122777	1986-01-05	20

```

21 -- Los que se llaman Eduardo y Cualquiera de sus apellidos empieza con 'C'
22 --NO HABIA entonces insertar
23 v INSERT INTO usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, id_rol, contrasena)
24 VALUES
25 ('Eduardo', 'Cobalto', 'Bautista', 'eduardo.cobalto@example.com', '5998288876', '2000-09-10', 3, 'PassEduardo123');
26 v SELECT * FROM usuarios
27 WHERE nombre LIKE '%Eduardo%'
28 AND (primer_apellido LIKE 'C%' OR segundo_apellido LIKE 'C%');
```

Data Output Messages Notifications

Showing rows: 1 to 1 |  Page No: 1 of 1 |

	<b>id_usuario</b> [PK] integer	<b>nombre</b> character varying (50)	<b>primer_apellido</b> character varying (50)	<b>segundo_apellido</b> character varying (50)	<b>email</b> character varying (100)	<b>telefono</b> character varying (20)	<b>fecha_nacimiento</b> date	<b>fecha_registro</b> date
1	71	Eduardo	Cobalto	Bautista	eduardo.cobalto@example.com	5998288876	2000-09-10	2025-06-1

```

30 -- Las personas que su apellido materno empieza con la primera mitad del alfabeto [A-M] pero que no empiecen ni con A ni con I
31 v SELECT * FROM usuarios
32 WHERE segundo_apellido SIMILAR TO '[B-DF-Mb-df-m]%';
```

Data Output Messages Notifications

Showing rows: 1 to 32 |  Page No: 1 of 1 |

	<b>id_usuario</b> [PK] integer	<b>nombre</b> character varying (50)	<b>primer_apellido</b> character varying (50)	<b>segundo_apellido</b> character varying (50)	<b>email</b> character varying (100)	<b>telefono</b> character varying (20)	<b>fecha_nacimiento</b> date	<b>fecha_registro</b> date
1	4	Lucía	Morales	Jiménez	lucia.morales@example.com	5551234567	1995-05-06	20
2	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28	20
3	21	Francisco	Navarro	García	francisco.navarro@example.com	5557766553	1993-01-10	20
4	25	David	Estrada	Mejía	david.estraza@example.com	5554455669	1990-02-26	20

```

34 -- Las personas que su apellido paterno empiece con la segunda mitad del alfabeto [N-Z]
35 v SELECT * FROM usuarios
36 WHERE primer_apellido SIMILAR TO '[N-Zn-z]%';
37

```

Data Output Messages Notifications

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date	<code>fech</code> date
1	2	Valeria	Reyes	Soto	valeria.reyes@example.com	5556782345	1990-09-18	20
2	14	Marcela	Rojas	Pacheco	marcela.rojas@example.com	5559988776	1995-02-09	20
3	21	Francisco	Navarro	Garcia	francisco.navarro@example.com	5557766553	1993-01-10	20
4	24	Lorena	Rivas	Serrano	lorena.rivas@example.com	5556677884	1995-08-09	20
5	28	Daniela	Palacios	Cabrera	daniela.palacios@example.com	5556677887	1994-03-04	20

```

42 -- Listar los usuarios cuyo apellido paterno termine con "ez"
43 v SELECT * FROM usuarios
44 WHERE primer_apellido LIKE '%ez';
45

```

Data Output Messages Notifications

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date	<code>fech</code> date
1	11	Roberto	López	Romero	roberto.lopez@example.com	5551122777	1986-01-05	202
2	19	Carlos	Martínez	Ríos	carlos.martinez@example.com	5554455667	1988-03-19	202
3	33	Javier	Gómez	Ortiz	javier.gomez@example.com	5552233441	1995-08-22	202
4	34	Laura	Gutiérrez	Rodríguez	laura.gutierrez@example.com	5556655443	1993-04-08	202
5	35	Oscar	Martinez	Santos	oscar.martinez@example.com	5551122333	1990-10-17	202

```

46 -- Seleccionar todos los usuarios cuyo correo electronico sea de Gmail
47 --No habia entonces inserto
48 v INSERT INTO usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, id_rol, contrasena)
49 VALUES
50 ('Fernando', 'Farias', 'Villanueva', 'fernando.farias@gmail.com', '5982288876', '2006-12-26', 3, 'PassFernando123');
51 v SELECT * FROM usuarios
52 WHERE email LIKE '%gmail.com';
53

```

Data Output Messages Notifications

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date	<code>fecha_registr</code> date
1	72	Fernando	Farias	Villanueva	fernando.farias@gmail.com	5982288876	2006-12-26	2025-06-18

```

54 -- Consultar los libros cuya editorial contenga la palabra "Editores"
55 --No habia entonces inserto
56 v INSERT INTO libros (isbn, titulo, anio_publicacion, editorial, edicion, cantidad_total, cantidad_disponible, clasificacion)
57 VALUES
58 ('978-3-16-118410-0', 'Camino a la luz', 2005, 'Editores Unidos', 2, 20, 13, 5);
59 v SELECT * FROM libros
60 WHERE editorial LIKE '%Editores%';
61

```

Data Output Messages Notifications

	<code>id_libro</code> [PK] integer	<code>isbn</code> character varying (20)	<code>titulo</code> character varying (255)	<code>anio_publicacion</code> integer	<code>editorial</code> character varying (100)	<code>edicion</code> integer	<code>cantidad_total</code> integer	<code>cantidad_disponible</code> integer	<code>clasificaci</code> integer
1	49	978-3-16-118410-0	Camino a la luz	2005	Editores Unidos	2	20	13	

```

62 -- Listar los usuarios cuyos nombres contengan una vocal seguida de una "n"
63 v SELECT * FROM usuarios |
64 WHERE nombre SIMILAR TO '%[AEIOUaeiou]n%';
65

```

Data Output Messages Notifications

	<code>id_usuario</code> [PK] integer	<code>nombre</code> character varying (50)	<code>primer_apellido</code> character varying (50)	<code>segundo_apellido</code> character varying (50)	<code>email</code> character varying (100)	<code>telefono</code> character varying (20)	<code>fecha_nacimiento</code> date
1	9	Daniel	Cruz	Martínez	daniel.cruz@example.com	5556677889	1997-10-28
2	16	Andrea	Mendoza	Aguirre	andrea.mendoza@example.com	5556655447	1998-05-30
3	21	Francisco	Navarro	Garcia	francisco.navarro@example.com	5557766553	1993-01-10
4	24	Lorena	Rivas	Serrano	lorena.rivas@example.com	5556677884	1995-08-09

```

66 -- Obtener todos los usuarios cuyos numeros de telefono terminen en "00"
67 --No hay enones inserto
68 ✓ INSERT INTO usuarios (nombre, primer_apellido, segundo_apellido, email, telefono, fecha_nacimiento, id_rol, contrasena)
69 VALUES
70 ('Rosa', 'Oliva', 'Oliva', 'rosa.oliva@gmail.com', '5982280000', '1993-12-13', 3, 'PassRosa123');
71 ✓ SELECT * FROM usuarios
72 WHERE telefono LIKE '%00';
73

```

Data Output Messages Notifications

	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro	contrasena
	character varying (50)	character varying (50)	character varying (50)	character varying (100)	character varying (20)	date	date	character v
1	Rosa	Oliva	Oliva	rosa.oliva@gmail.com	5982280000	1993-12-13	2025-06-18	PassRosa123

74 -- Consultar los autores cuyo nombre contenga la letra "J" en cualquier parte

```

75 ✓ SELECT * FROM autores |
76 WHERE nombre_autor LIKE '%J%' OR nombre_autor LIKE '%j%';
77

```

Data Output Messages Notifications

	id_autor	nombre_autor	apellido_paterno	apellido_materno
	[PK] integer	character varying (50)	character varying (50)	character varying (50)
1	5	Jane	Austen	[null]
2	11	Julio	Cortázar	[null]
3	12	J.D.	Salinger	[null]
4	13	J.R.R.	Tolkien	[null]

78 -- Obtener los usuarios cuyo nombre contenga exactamente cinco caracteres

```

79 ✓ SELECT * FROM usuarios
80 WHERE nombre SIMILAR TO '_____';
81

```

Data Output Messages Notifications

	id_usuario	nombre	primer_apellido	segundo_apellido	email	telefono	fecha_nacimiento	fecha_registro
	[PK] integer	character varying (50)	character varying (50)	character varying (50)	character varying (100)	character varying (20)	date	date
1	4	Lucía	Morales	Jiménez	lucia.morales@example.com	5551234567	1995-05-06	2025-06-12
2	25	David	Estrada	Mejía	david.estrada@example.com	5554455669	1990-02-26	2025-06-12
3	34	Laura	Gutiérrez	Rodríguez	laura.gutierrez@example.com	5556655443	1993-04-08	2025-06-12

82 -- Obtener los titulos de libros que contengan numeros en cualquier parte

```

83 ✓ SELECT * FROM libros
84 WHERE titulo SIMILAR TO '%[0-9]%' ;
85

```

Data Output Messages Notifications

	id_libro	isbn	titulo	anio_publicacion	editorial	edicion	cantidad_total	cantidad_disponible	clasificacion
	[PK] integer	character varying (20)	character varying (255)	integer	character varying (100)	integer	integer	integer	integer
1	3	978-0-452-28423-4	1984	1949	Secker & Warburg	1	12	10	
2	4	978-0-7432-7356-5	Fahrenheit 451	1953	Ballantine Books	1	6	4	