

 UNIVERSIDAD POLÍTÉCNICA SALESIANA ECUADOR	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación

Universidad Politécnica Salesiana

Vicerrectorado Docente

Código del Formato:	GUIA-PRL-001
Versión:	VF1.0
Elaborado por:	Directores de Área del Conocimiento Integrantes Consejo Académico
Fecha de elaboración:	2016/04/01
Revisado por:	Consejo Académico
Fecha de revisión:	2016/04/06
Aprobado por:	Lauro Fernando Pesántez Avilés Vicerrector Docente
Fecha de aprobación:	2016/14/06
Nivel de confidencialidad:	Interno

 <p>UNIVERSIDAD POLÍTÉCNICA SALESIANA ECUADOR</p>	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

Descripción General

Propósito

El propósito del presente documento es definir un estándar para elaborar documentación de guías de práctica de laboratorio, talleres o centros de simulación de las Carreras de la Universidad Politécnica Salesiana, con la finalidad de lograr una homogenización en la presentación de la información por parte del personal académico y técnico docente.

Alcance

El presente estándar será aplicado a toda la documentación referente a informes de prácticas de laboratorio, talleres o centros de simulación de las Carreras de la Universidad Politécnica Salesiana.

Formatos

- Formato de Guía de Práctica de Laboratorio / Talleres / Centros de Simulación – para Docentes
- Formato de Informe de Práctica de Laboratorio / Talleres / Centros de Simulación – para Estudiantes



FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES

CARRERA: COMPUTACIÓN

ASIGNATURA: Programación Aplicada

NRO. PRÁCTICA:

1

TÍTULO PRÁCTICA: Base de Datos

OBJETIVO:

Identificar las sentencias SQL

Diseñar e Implementar codigos DDL, DML

Entender cada una de las características del uso de una Base de Datos

INSTRUCCIONES (Detallar las instrucciones que se dará al estudiante):	1. Revisar los conceptos fundamentales de SQL
	2. Establecer como implementar DDL y DML
	3. Implementar y diseñar nuevas estructuras de base de datos y datos.
	4. Realizar el informe respectivo según los datos solicitados.

ACTIVIDADES POR DESARROLLAR

(Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)

1. Revisar la teoría y conceptos de Base de Datos

2. Diseñar e implementar las siguientes prácticas de Base de datos solamente utilizando código SQL (No se permite el uso del generador o ayuda de ningún IDE):

- 2.1 Actividad: Realizar una base de datos para el equipo de futbol del curso.
 - 2.2 Crear la tabla PERSONA, con los siguientes datos:
 - Cedula con una dimensión de 10 caracteres
 - Nombre con una dimensión de 60 caracteres
 - Dirección con una dimensión de 120 caracteres
 - Fecha de nacimiento de tipo fecha
 - Número de cargas familiares
 - Salario de tipo real con 10 dígitos enteros y dos decimales
- 2) Modificar la tabla PERSONA de tal manera que el campo cedula sea clave primaria.
- 3) Modificar la tabla PERSONA de tal manera que los campos nombre, fecha de nacimiento sean obligatorios o requeridos
- 4) Modificar la tabla PERSONA de tal manera la columna nombre cambie a nombres, y se añada una columna para el apellido con un dimensión de 60 caracteres.
- Crear la tabla PERSONA_TELEFONO, con los siguientes datos:
 - Código, de tipo entero
 - Cedula, con una dimensión de 10 caracteres (es la referencia de la tabla persona)
 - Tipo de teléfono, con un dimensión de 12 caracteres, aquí se guardarán valores como celular o convencional
 - Número de teléfono, con un dimensión de 24 caracteres
- 6) Modificar la tabla PERSONA_TELEFONO de tal manera que el campo código sea clave primaria.

7) Modificar la tabla PERSONA_TELEFONO de tal manera que se incluya la referencia a la tabla PERSONA por medio del campo cedula

2.3 SELECT de todos los registros de personas

- Selecciones los nombre y apellidos de las personas cuyo salario es mayor a 500 USD.
- Selecciones los todos los datos de las personas cuyo salario es mayor a 400 USD y menos a 1500.
- Seleccione el número de cedula y nombres de las personas que hayan nacido antes del año 2000.
- Selecciones los datos de las personas cuyo apellido empiece con la letra “a”.
- De su conjunto de personas de su base de datos inserte uno número de teléfono para 3 de ellas , esto en la tabla persona_telefono.
- Cambie el salario a 1000 USD a aquellas personas cuyo salario sea menor a 1000.
- Cambie los nombres y apellidos de todas las personas a mayúsculas
- Borre los registros de las personas cuyo salario sea mayor a 5000 USD.

2.4 FOREIGN KEY

- Seleccione los nombre y apellidos y datos de teléfonos de todas las personas.
- Seleccione los datos de las personas y sus números de teléfonos de aquellas personas que tiene un salario de 1000 USD.
- Seleccione los números de teléfonos que existen en su base de datos.
- Seleccione el nombre y apellido de las personas que tiene como teléfono un numero convencional.

3. Probar y modificar datos utilizando el IDE.

4. Fecha de Entrega: 18 Enero del 2021 23:55

RESULTADO(S) OBTENIDO(S):

Realizar procesos de Bases de Datos

Entender las aplicaciones de codificación de las nuevas características de base de datos

CONCLUSIONES:

Aprenden a trabajar en grupo dentro de plazos de tiempo establecidos, manejando el lenguaje de programación de Java y SQL.

RECOMENDACIONES:

Realizar el trabajo dentro del tiempo establecido.

Docente / Técnico Docente: _____

Firma: _____

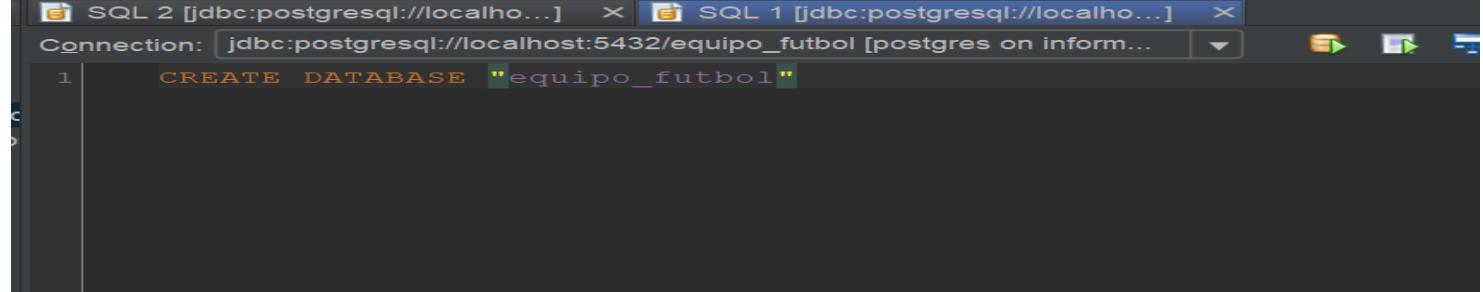
CARRERA: Computacion	ASIGNATURA: Programacion Aplicada
NRO. PRÁCTICA:	TÍTULO PRÁCTICA: Base de Datos

OBJETIVO ALCANZADO:

Identificar las sentencias SQL
 Diseñar e Implementar codigos DDL, DML
 Entender cada una de las características del uso de una Base de Datos

ACTIVIDADES DESARROLLADAS

1. Revisar la teoría y conceptos de Base de Datos
2. Diseñar e implementar las siguientes practicas de Base de datos solamente utilizando código SQL (No se permite el uso del generador o ayuda de ningún IDE):
 - 2.1 Actividad: Realizar una base de datos para el equipo de futbol del curso.

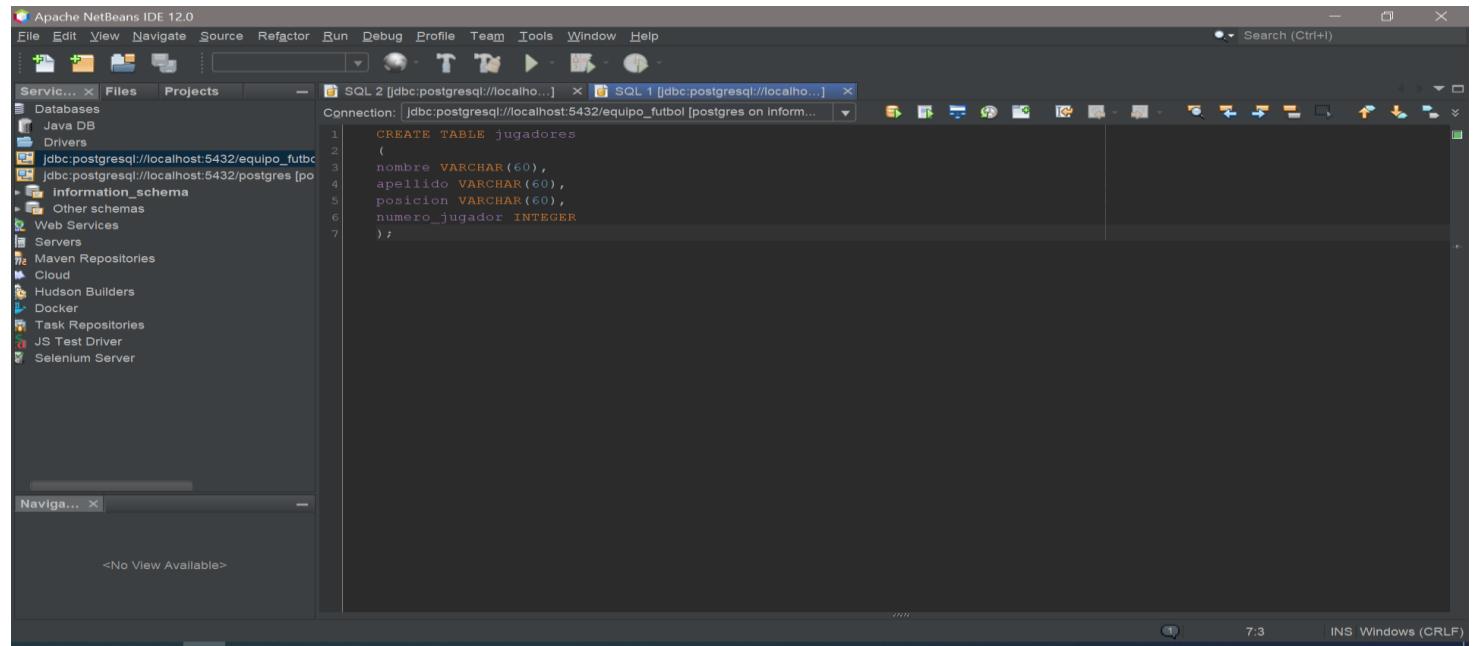


```

SQL 2 [jdbc:postgresql://localhost:5432/equipo_futbol] × SQL 1 [jdbc:postgresql://localhost:5432/postgres] ×
Connection: jdbc:postgresql://localhost:5432/equipo_futbol [postgres on inform...]
1 CREATE DATABASE "equipo_futbol"

SQL 2 [jdbc:postgresql://localhost:5432/equipo_futbol] × SQL 1 [jdbc:postgresql://localhost:5432/postgres] ×
Connection: jdbc:postgresql://localhost:5432/equipo_futbol [postgres on inform...
1 CREATE TABLE jugadores
2 (
3     nombre VARCHAR(60),
4     apellido VARCHAR(60),
5     posicion VARCHAR(60),
6     numero_jugador INTEGER
7 )

```



Apache NetBeans IDE 12.0

Servic... Files Projects Databases Java DB Drivers jdbc:postgresql://localhost:5432/equipo_futbol jdbc:postgresql://localhost:5432/postgres [po... information_schema Other schemas Web Services Servers Maven Repositories Cloud Hudson Builders Docker Task Repositories JS Test Driver Selenium Server

CREATE DATABASE "equipo_futbol"

CREATE TABLE jugadores

nombre VARCHAR(60), apellido VARCHAR(60), posicion VARCHAR(60), numero_jugador INTEGER

Apache NetBeans IDE 12.0

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+)

Servic... X Files Projects

Databases

- Java DB
- Drivers
- jdbc:postgresql://localhost:5432/equip
- information_schema
- Other schemas
- pg_catalog
- public
- Tables
- jugadores**

nombre apellido posicion numero_jugador

- Indexes
- Foreign Keys
- Views
- Procedures

jdbc:postgresql://localhost:5432/postg

Web Services

Servers

Maven Repositories

Cloud

Naviga... X

<No View Available>

SQL 2 [jdbc:postgresql://localhost:5432/equip...] X SQL 3 [jdbc:postgresql://localhost:5432/equip...] X SQL 4 [jdbc:postgresql://localhost:5432/postg...]

Connection: jdbc:postgresql://localhost:5432/equipo_futbol [postgres on inform...]

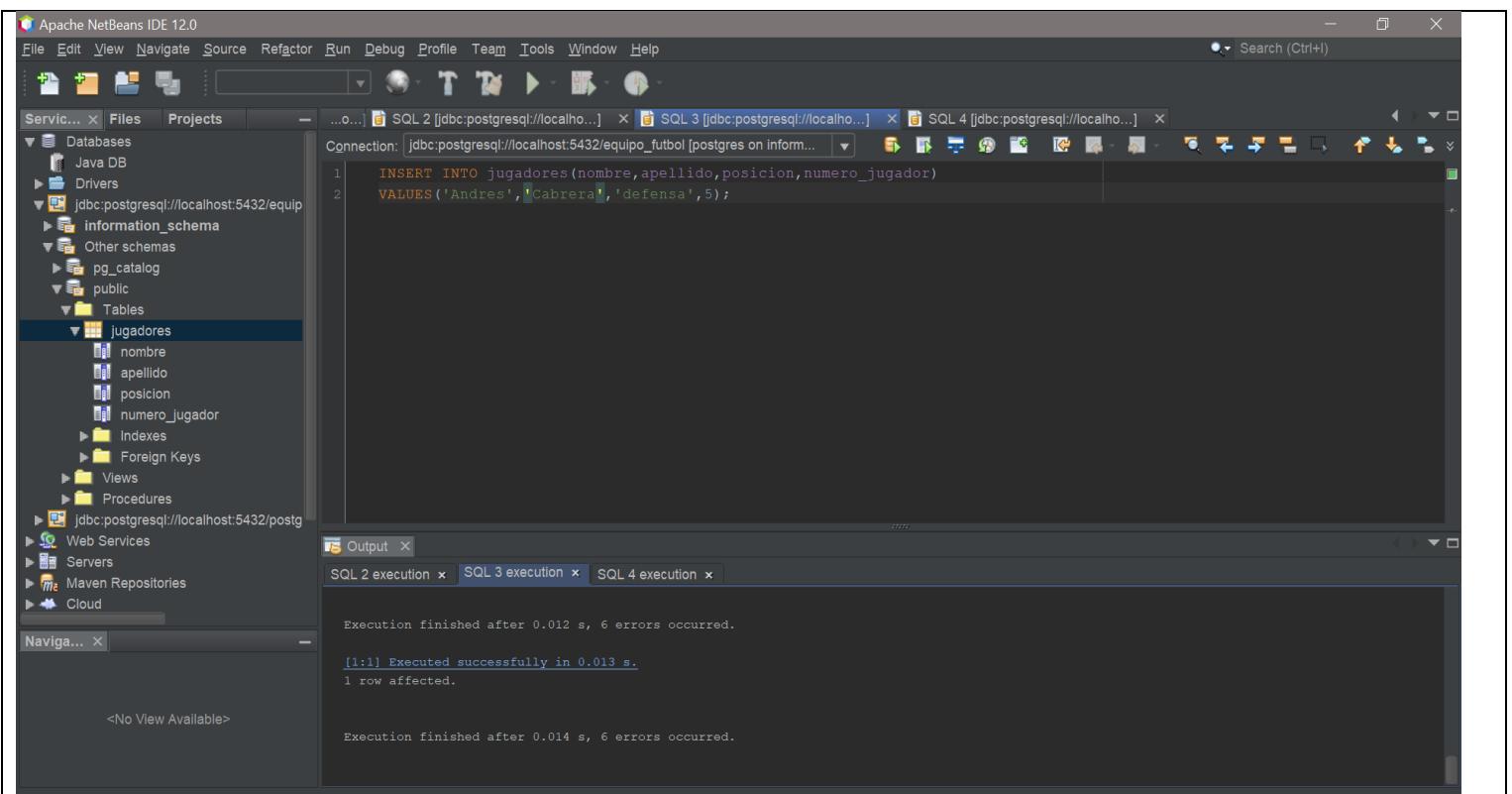
```
1 INSERT INTO jugadores(nombre,apellido,posicion,numero_jugador)
2 VALUES('Andres','Cabrera','defensa',5);
```

Output X

SQL 2 execution x SQL 3 execution x SQL 4 execution x

```
Execution finished after 0.012 s, 6 errors occurred.  
[1:1] Executed successfully in 0.013 s.  
1 row affected.
```

Execution finished after 0.014 s, 6 errors occurred.



Apache NetBeans IDE 12.0

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+)

Servic... X Files Projects

Databases

- Java DB
- Drivers
- jdbc:postgresql://localhost:5432/equip
- information_schema
- Other schemas
- pg_catalog
- public
- Tables
- jugadores**

nombre apellido posicion numero_jugador

- Indexes
- Foreign Keys
- Views
- Procedures

jdbc:postgresql://localhost:5432/postg

Web Services

Servers

Maven Repositories

Cloud

Naviga... X

<No View Available>

SQL 3 [jdbc:postgresql://localhost:5432/equip...] X SQL 4 [jdbc:postgresql://localhost:5432/postg...] X SQL 5 [jdbc:postgresql://localhost:5432/postg...]

Connection: jdbc:postgresql://localhost:5432/equipo_futbol [postgres on inform...]

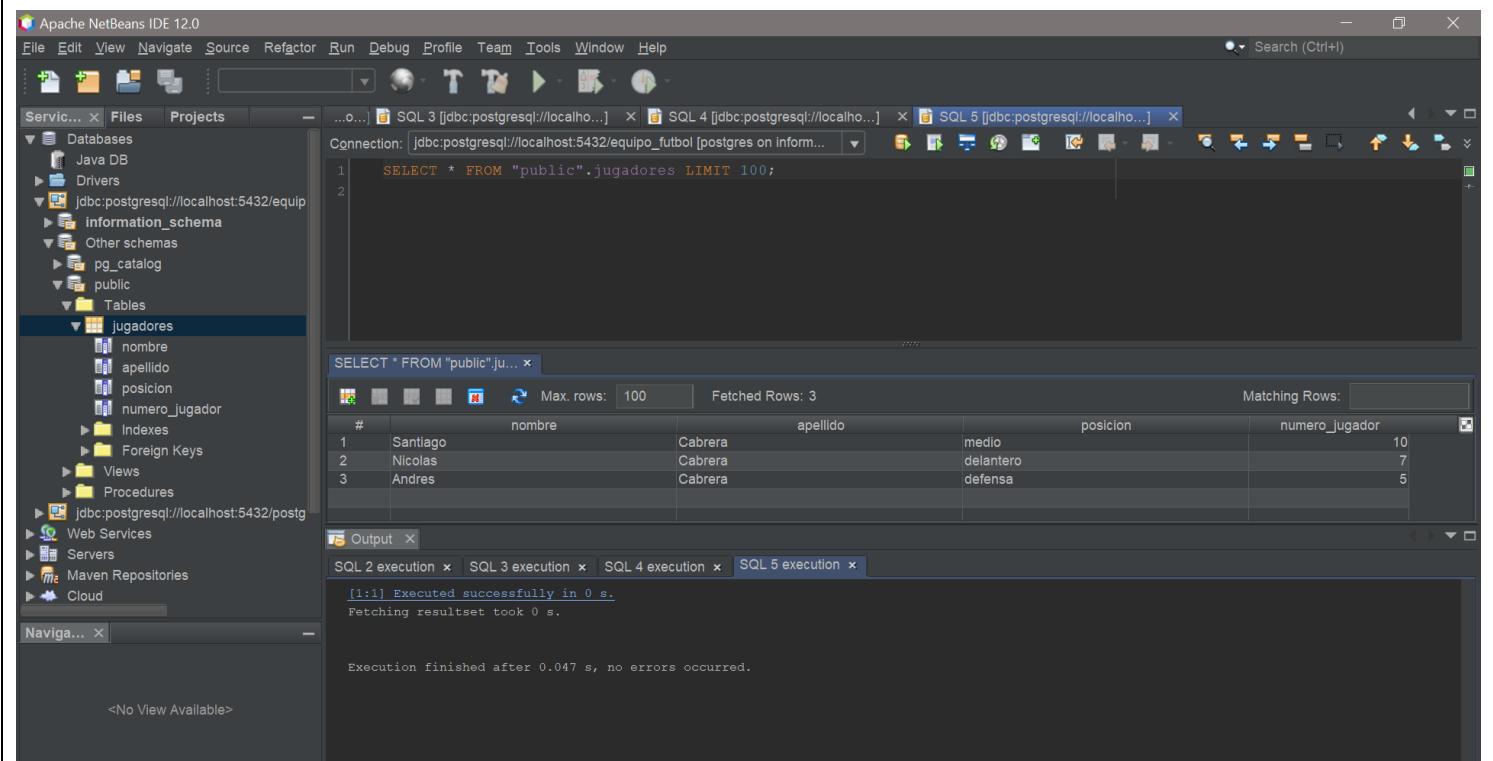
```
1 SELECT * FROM "public".jugadores LIMIT 100;
```

Output X

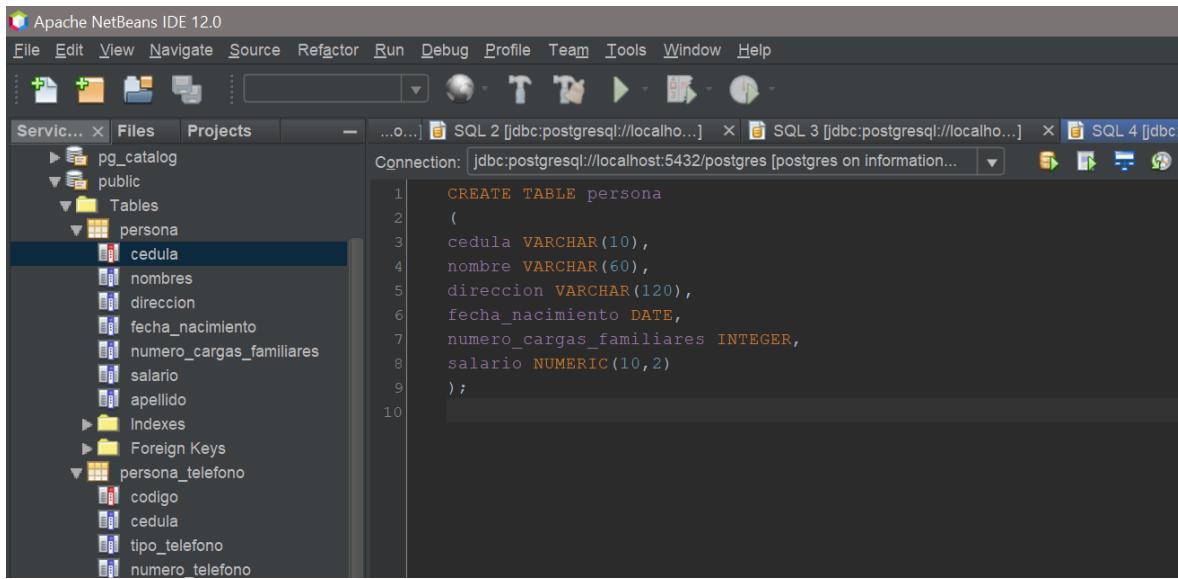
SQL 2 execution x SQL 3 execution x SQL 4 execution x SQL 5 execution x

```
[1:1] Executed successfully in 0 s.  
Fetching resultset took 0 s.
```

Execution finished after 0.047 s, no errors occurred.



- 2.2 Crear las tabla PERSONA, con los siguientes datos:
 - Cedula con una dimensión de 10 caracteres
 - Nombre con una dimensión de 60 caracteres
 - Dirección con una dimensión de 120 caracteres
 - Fecha de nacimiento de tipo fecha
 - Número de cargas familiares
 - Salario de tipo real con 10 dígitos enteros y dos decimales
 -



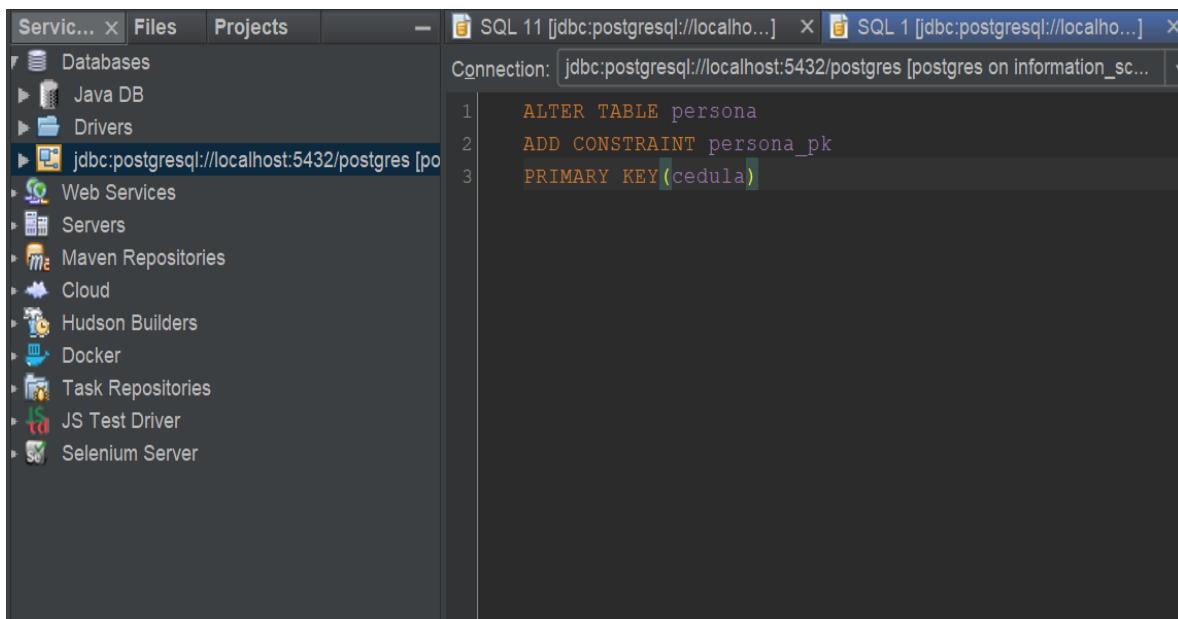
The screenshot shows the Apache NetBeans IDE interface. On the left, the 'Projects' tab is selected, displaying a file tree for a PostgreSQL database named 'postgres'. Under the 'Tables' section, there is a folder named 'persona' containing several columns: 'cedula', 'nombre', 'direccion', 'fecha_nacimiento', 'numero_cargas_familiares', 'salario', and 'apellido'. On the right, the code editor displays the SQL command to create the 'persona' table:

```

1 CREATE TABLE persona
2 (
3     cedula VARCHAR(10),
4     nombre VARCHAR(60),
5     direccion VARCHAR(120),
6     fecha_nacimiento DATE,
7     numero_cargas_familiares INTEGER,
8     salario NUMERIC(10,2)
9 );
10

```

2) Modificar la tabla PERSONA de tal manera que el campo cedula sea clave primaria.

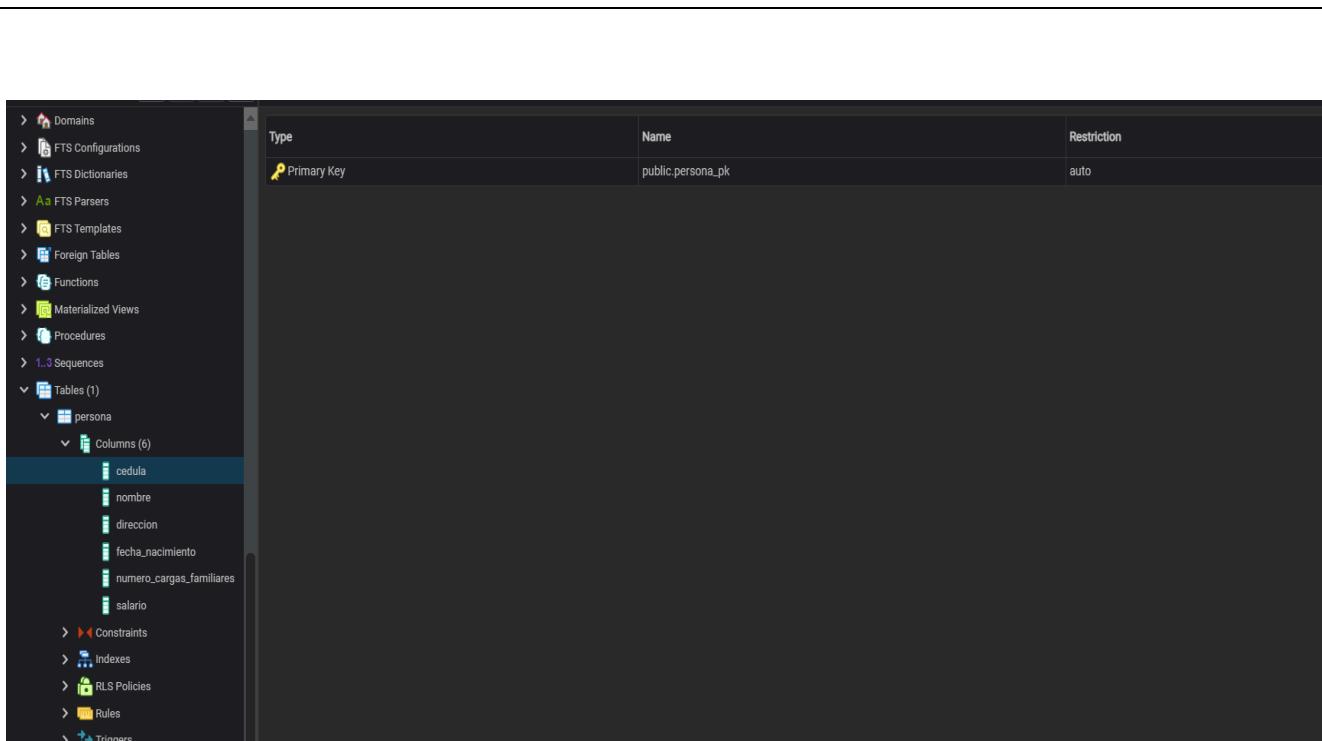


The screenshot shows the Apache NetBeans IDE interface. On the left, the 'Projects' tab is selected, displaying a file tree for a PostgreSQL database named 'postgres'. On the right, the code editor displays the SQL command to alter the 'persona' table and add a primary key constraint:

```

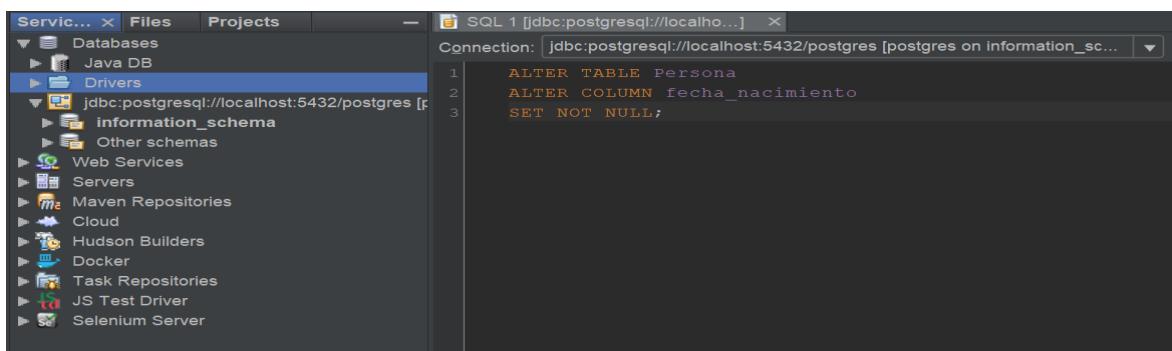
1 ALTER TABLE persona
2 ADD CONSTRAINT persona_pk
3 PRIMARY KEY(cedula)

```



Type	Name	Restriction
Primary Key	public.persona_pk	auto

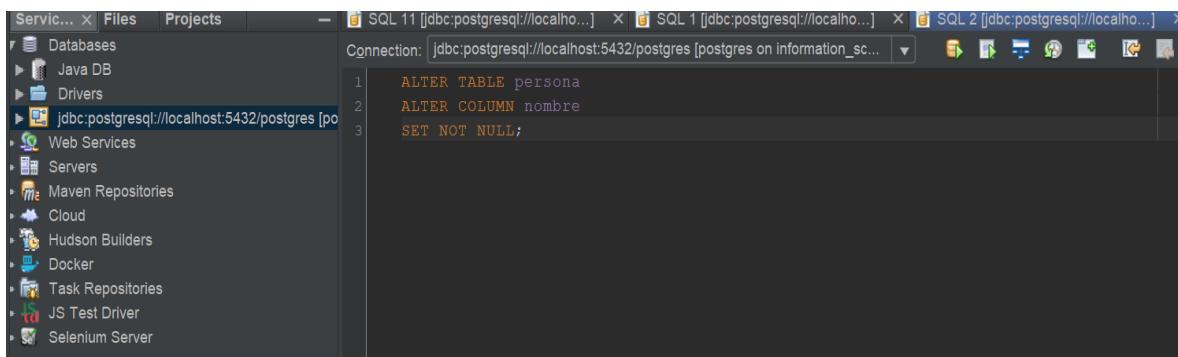
3) Modificar la tabla PERSONA de tal manera que los campos nombre, fecha de nacimiento sean obligatorios o requeridos



```

ALTER TABLE Persona
ALTER COLUMN nombre
SET NOT NULL;

```



```

ALTER TABLE persona
ALTER COLUMN nombre
SET NOT NULL;

```

Screenshot of the pgAdmin interface showing the 'Columns' tab for the 'persona' table. The table has seven columns: cedula, nombres, direccion, fecha_nacimiento, numero_cargas_familiares, salario, and apellido.

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
1	cedula	character varying	10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	nombres	character varying	60		<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	direccion	character varying	120		<input type="checkbox"/>	<input type="checkbox"/>
4	fecha_nacimiento	date			<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	numero_cargas_familiares	integer			<input type="checkbox"/>	<input type="checkbox"/>
6	salario	numeric	10	2	<input type="checkbox"/>	<input type="checkbox"/>
7	apellido	character varying	12		<input type="checkbox"/>	<input type="checkbox"/>

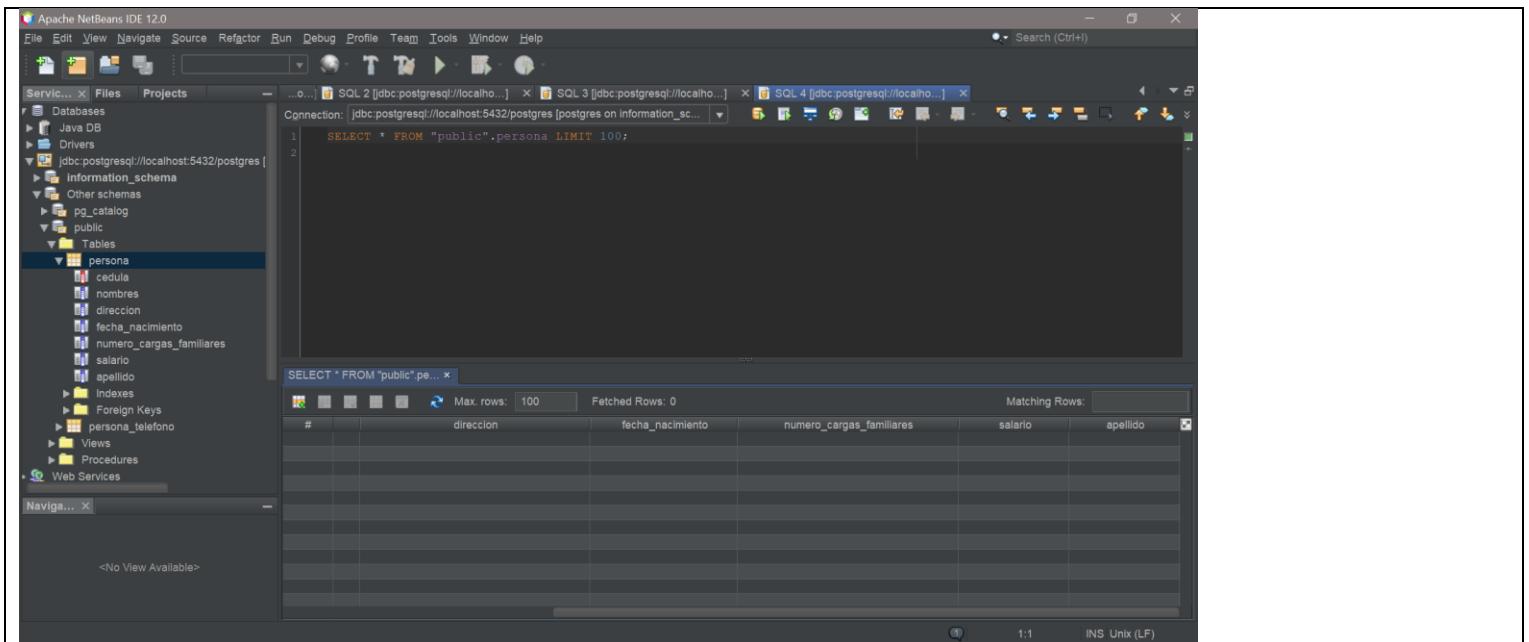
- 4) Modificar la tabla PERSONA de tal manera la columna nombre cambie a nombres, y se añada una columna para el apellido con un dimensión de 60 caracteres.

Screenshot of the SQL interface in pgAdmin showing the first step of the modification. The query is:

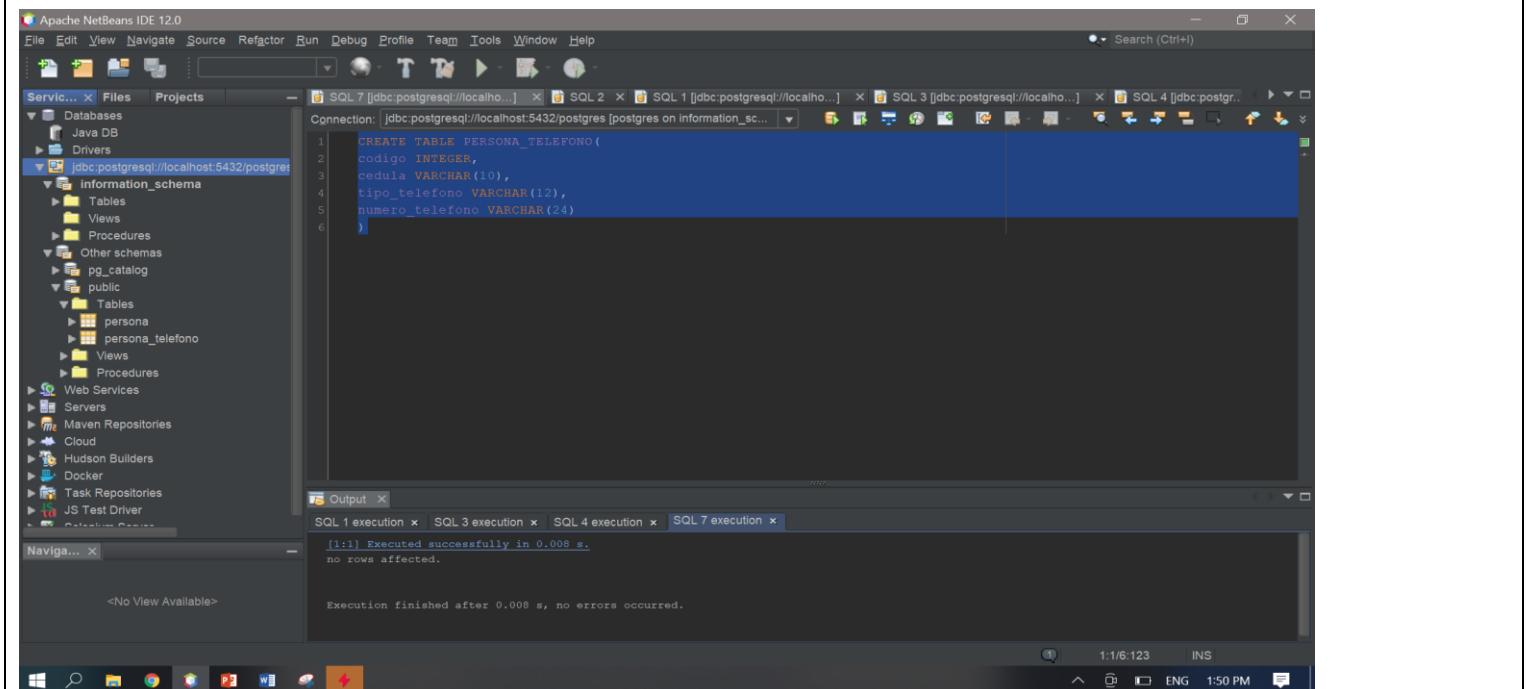
```
ALTER TABLE persona
RENAME COLUMN nombre
TO nombres;
```

Screenshot of the SQL interface in pgAdmin showing the second step of the modification. The query is:

```
ALTER TABLE persona
ADD apellido VARCHAR(60);
```



- Crear las tabla PERSONA_TELEFONO, con los siguientes datos:
 - Código, de tipo entero
 - Cedula, con una dimensión de 10 caracteres (es la referencia de la tabla persona)
 - Tipo de teléfono, con un dimensión de 12 caracteres, aquí se guardarán valores como celular o convencional
 - Número de teléfono, con un dimensión de 24 caracteres



Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación

6) Modificar la tabla PERSONA_TELEFONO de tal manera que el campo código sea clave primaria.

The screenshot shows the Apache NetBeans IDE 12.0 interface. The left sidebar displays the project structure, including databases (Java DB, Drivers), a PostgreSQL connection (localhost:5432/postgres), and various schema and table nodes. The main workspace contains three SQL editor tabs, all connected to the same PostgreSQL database. The central tab shows the following SQL code:

```
1 ALTER TABLE persona_telefono
2 ADD CONSTRAINT persona_telefono_pk
3 PRIMARY KEY (codigo);
```

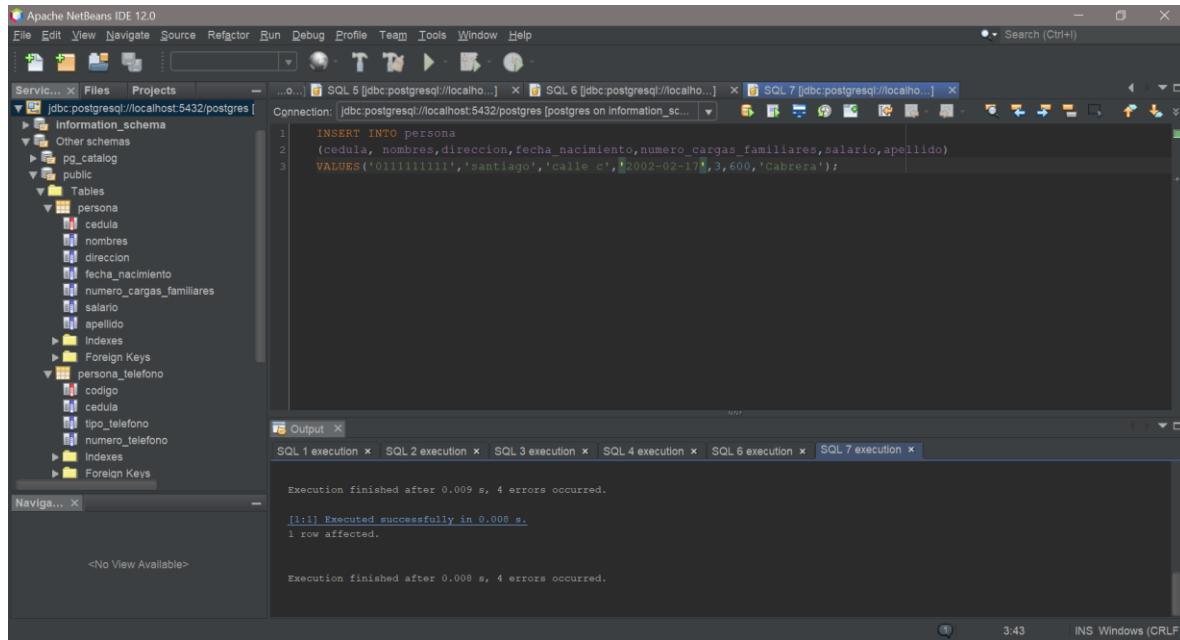
The bottom output panel shows the results of the execution:

```
[1:1] Executed successfully in 0.019 s.
no rows affected.
```

Execution finished after 0.022 s, no errors occurred.

7) Modificar la tabla PERSONA_TELEFONO de tal manera que se incluya la referencia a la tabla PERSONA por medio del campo cedula

INSERT



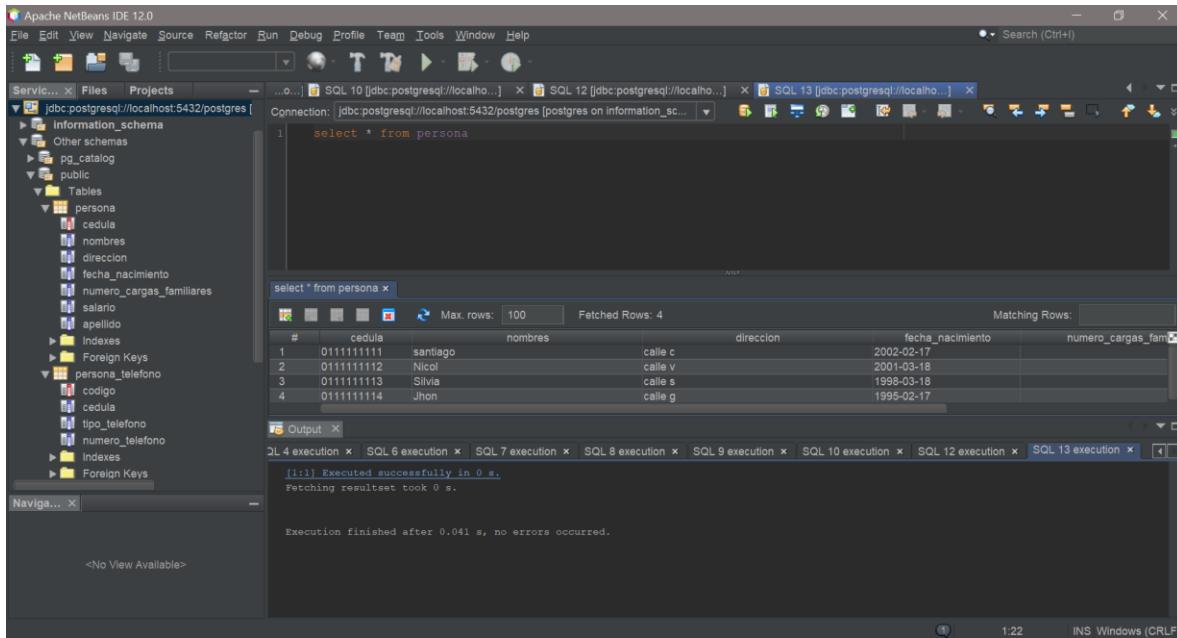
```
Apache NetBeans IDE 12.0
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Service... Files Projects ...o... SQL 5 [jdbc:postgresql://localhost:5432/postgres] X SQL 6 [jdbc:postgresql://localhost:5432/postgres [postgres on information_sc... X SQL 7 [jdbc:postgresql://localhost:5432/postgres [postres on information_sc...
Connection: jdbc:postgresql://localhost:5432/postgres [postgres on information_sc...
1: INSERT INTO persona
2: (cedula, nombres,direccion,fecha_nacimiento,numero_cargas_familiares,salario,apellido)
3: VALUES('0111111111','santiago','calle c',2002-02-17,3,600,'Cabrera');

Output
SQL 1 execution x SQL 2 execution x SQL 3 execution x SQL 4 execution x SQL 6 execution x SQL 7 execution x
Execution finished after 0.009 s, 4 errors occurred.
[1:1] Executed successfully in 0.008 s.
1 row affected.

Execution finished after 0.008 s, 4 errors occurred.

3:43 INS Windows (CRLF)
```

2.3 SELECT de todos los registros de personas



```
Apache NetBeans IDE 12.0
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Service... Files Projects ...o... SQL 10 [jdbc:postgresql://localhost:5432/postgres] X SQL 12 [jdbc:postgresql://localhost:5432/postgres [postgres on information_sc... X SQL 13 [jdbc:postgresql://localhost:5432/postgres [postres on information_sc...
Connection: jdbc:postgresql://localhost:5432/postgres [postgres on information_sc...
1: select * from persona

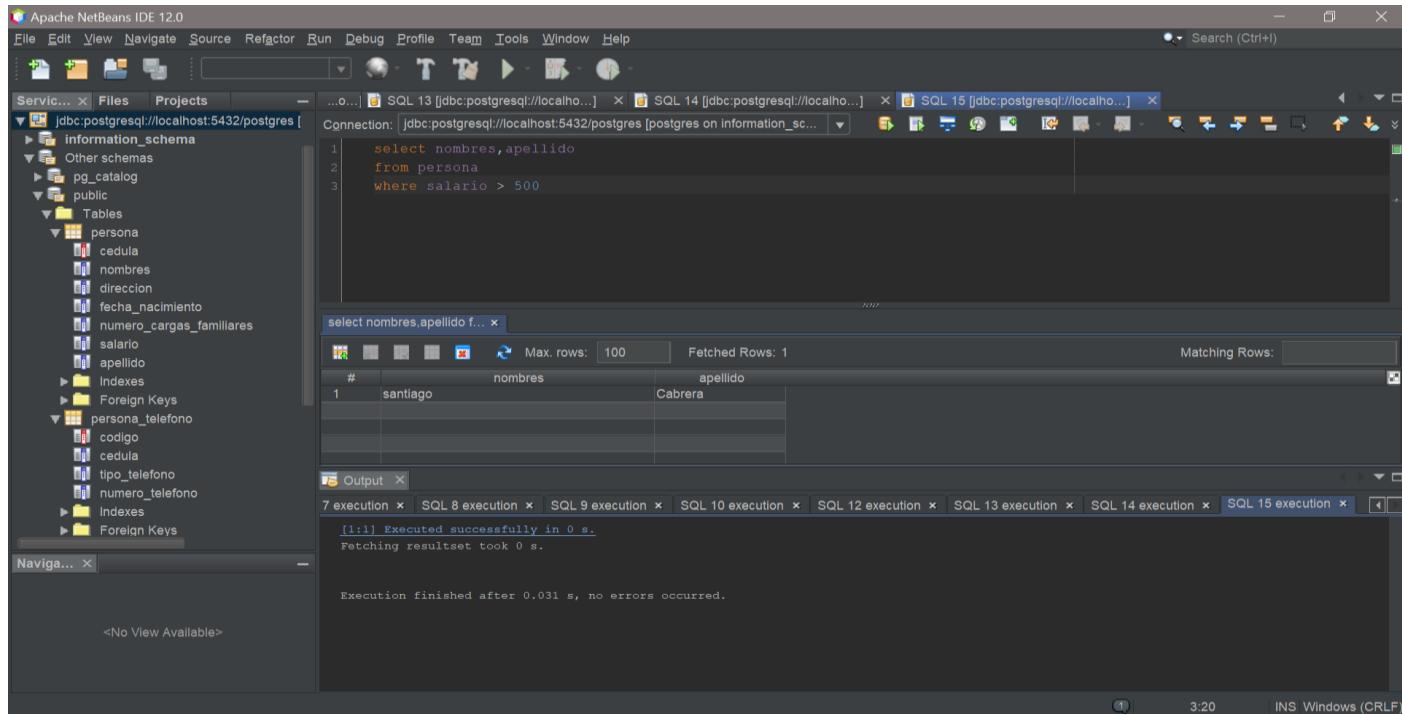
select * from persona x
Max. rows: 100 Fetched Rows: 4 Matching Rows:
# cedula nombres direccion fecha_nacimiento numero_cargas_familiares
1 0111111111 santiago calle c 2002-02-17 3
2 0111111112 Nicol calle v 2001-03-18
3 0111111113 Silvia calle s 1998-03-18
4 0111111114 Jhon calle g 1995-02-17

Output
SQL 4 execution x SQL 6 execution x SQL 7 execution x SQL 8 execution x SQL 9 execution x SQL 10 execution x SQL 12 execution x SQL 13 execution x
[1:1] Executed successfully in 0 s.
Fetching resultset took 0 s.

Execution finished after 0.041 s, no errors occurred.

1:22 INS Windows (CRLF)
```

- Selecciona los nombre y apellidos de las personas cuyo salario es mayor a 500 USD.



The screenshot shows the Apache NetBeans IDE interface. On the left, the Project Explorer displays a database connection named 'jdbc:postgresql://localhost:5432/postgres'. The Database Navigator shows the schema structure, including the 'persona' table which contains columns: cedula, nombres, direccion, fecha_nacimiento, numero_cargas_familiares, salario, apellido. A SQL editor window contains the following query:

```
1 select nombres, apellido
2 from persona
3 where salario > 500
```

The results pane shows a single row of data:

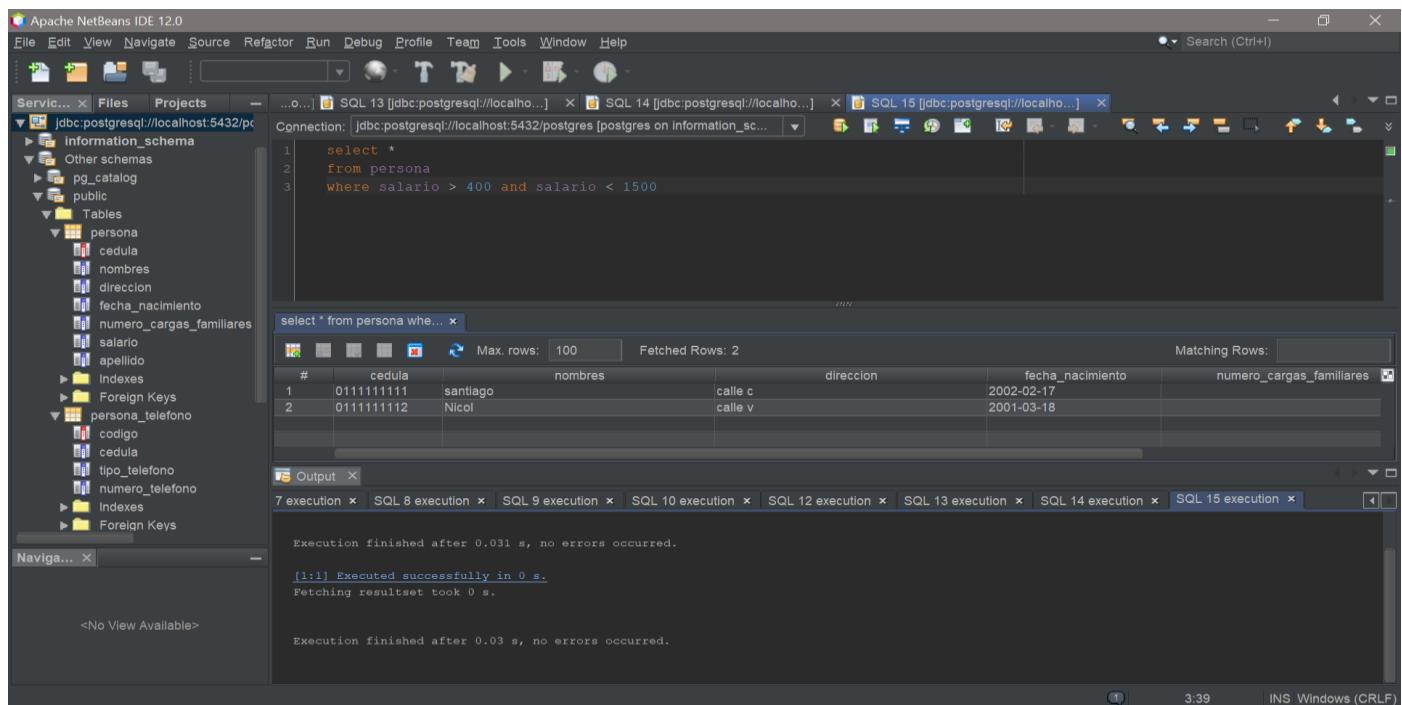
#	nombres	apellido
1	santiago	Cabrera

The Output pane shows the execution results:

```
[1:1] Executed successfully in 0 s.
Fetching resultset took 0 s.

Execution finished after 0.031 s, no errors occurred.
```

- Selecciona todos los datos de las personas cuyo salario es mayor a 400 USD y menor a 1500.



The screenshot shows the Apache NetBeans IDE interface. On the left, the Project Explorer displays a database connection named 'jdbc:postgresql://localhost:5432/postgres'. The Database Navigator shows the schema structure, including the 'persona' table which contains columns: cedula, nombres, direccion, fecha_nacimiento, numero_cargas_familiares, salario, apellido. A SQL editor window contains the following query:

```
1 select *
2 from persona
3 where salario > 400 and salario < 1500
```

The results pane shows two rows of data:

#	cedula	nombres	direccion	fecha_nacimiento	numero_cargas_familiares
1	0111111111	santiago	calle c	2002-02-17	
2	0111111112	Nicol	calle v	2001-03-18	

The Output pane shows the execution results:

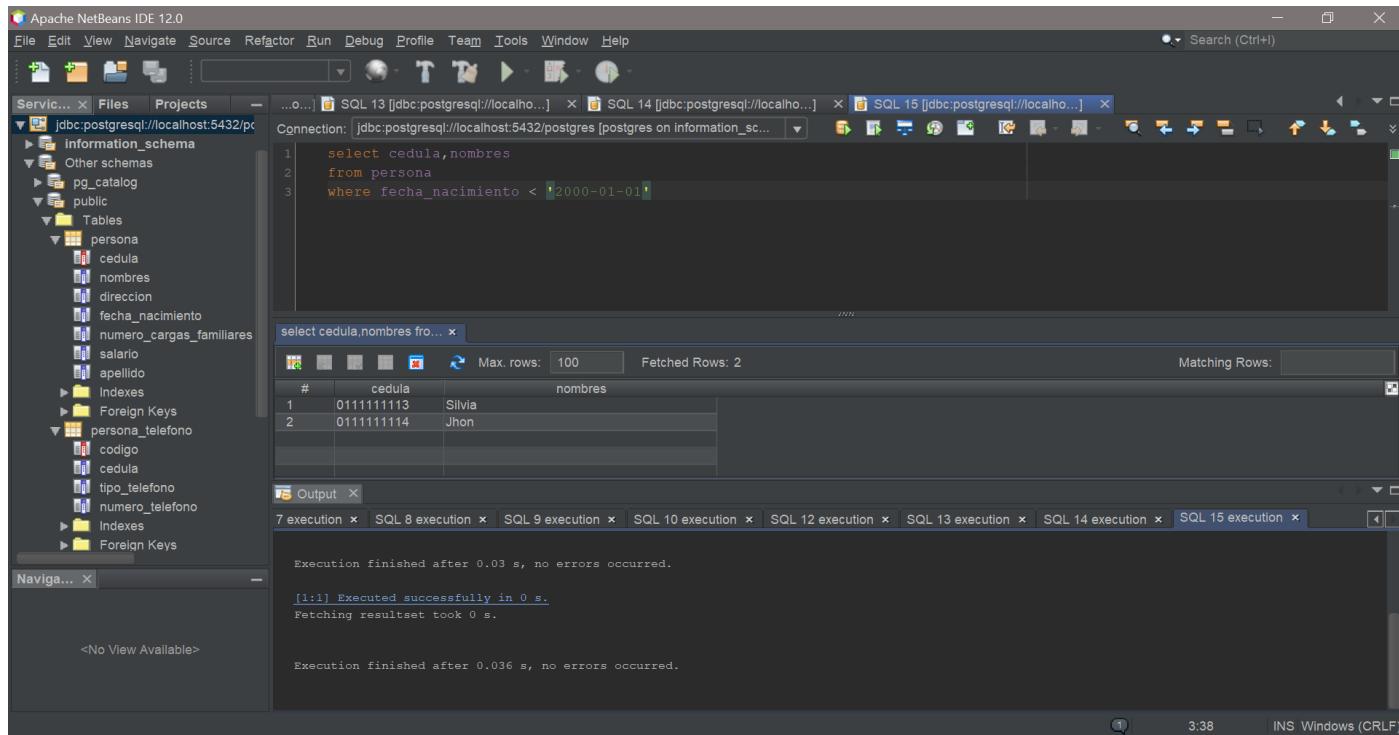
```
[1:1] Executed successfully in 0 s.
Fetching resultset took 0 s.

Execution finished after 0.031 s, no errors occurred.

[1:1] Executed successfully in 0 s.
Fetching resultset took 0 s.

Execution finished after 0.03 s, no errors occurred.
```

- Seleccione el número de cedula y nombres de las personas que hayan nacido antes del año 2000.



The screenshot shows the Apache NetBeans IDE interface with three tabs in the top navigation bar: SQL 13 [jdbc:postgresql://localhost:5432/postgres], SQL 14 [jdbc:postgresql://localhost:5432/postgres], and SQL 15 [jdbc:postgresql://localhost:5432/postgres]. The SQL 15 tab contains the following query:

```
select cedula,nombres
from persona
where fecha_nacimiento < '2000-01-01'
```

The results pane displays the output of the query:

#	cedula	nombres
1	0111111113	Silvia
2	0111111114	Jhon

The Output pane shows the execution details:

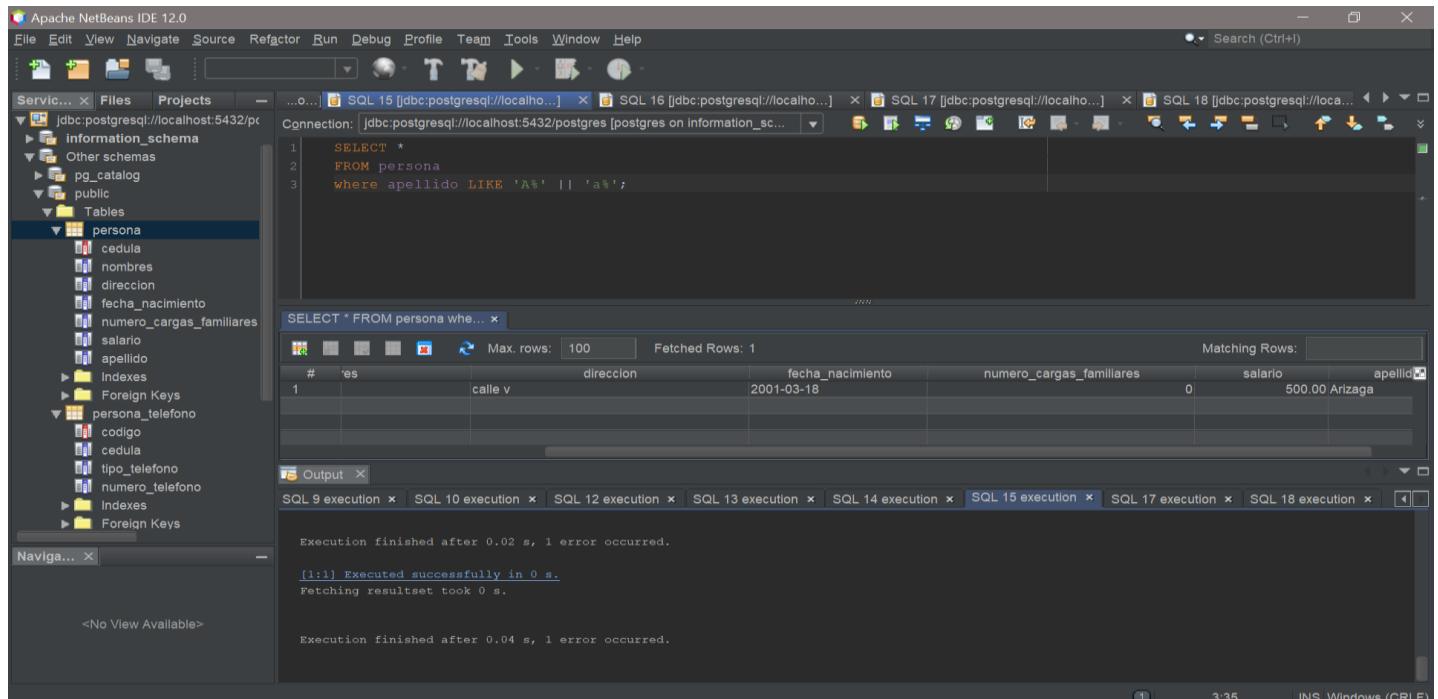
```
Execution finished after 0.03 s, no errors occurred.  

[1:1] Executed successfully in 0 s.  

Fetching resultset took 0 s.  

Execution finished after 0.036 s, no errors occurred.
```

- Seleccione los datos de las personas cuyo apellido empiece con la letra “a”.



The screenshot shows the Apache NetBeans IDE interface with four tabs in the top navigation bar: SQL 15 [jdbc:postgresql://localhost:5432/postgres], SQL 16 [jdbc:postgresql://localhost:5432/postgres], SQL 17 [jdbc:postgresql://localhost:5432/postgres], and SQL 18 [jdbc:postgresql://localhost:5432/postgres]. The SQL 16 tab contains the following query:

```
SELECT *
FROM persona
where apellido LIKE 'A%' || 'a%';
```

The results pane displays the output of the query:

#	res	direccion	fecha_nacimiento	numero_cargas_familiares	salario	apellido
1	calle v	2001-03-18	0	500.00	Arizaga	

The Output pane shows the execution details:

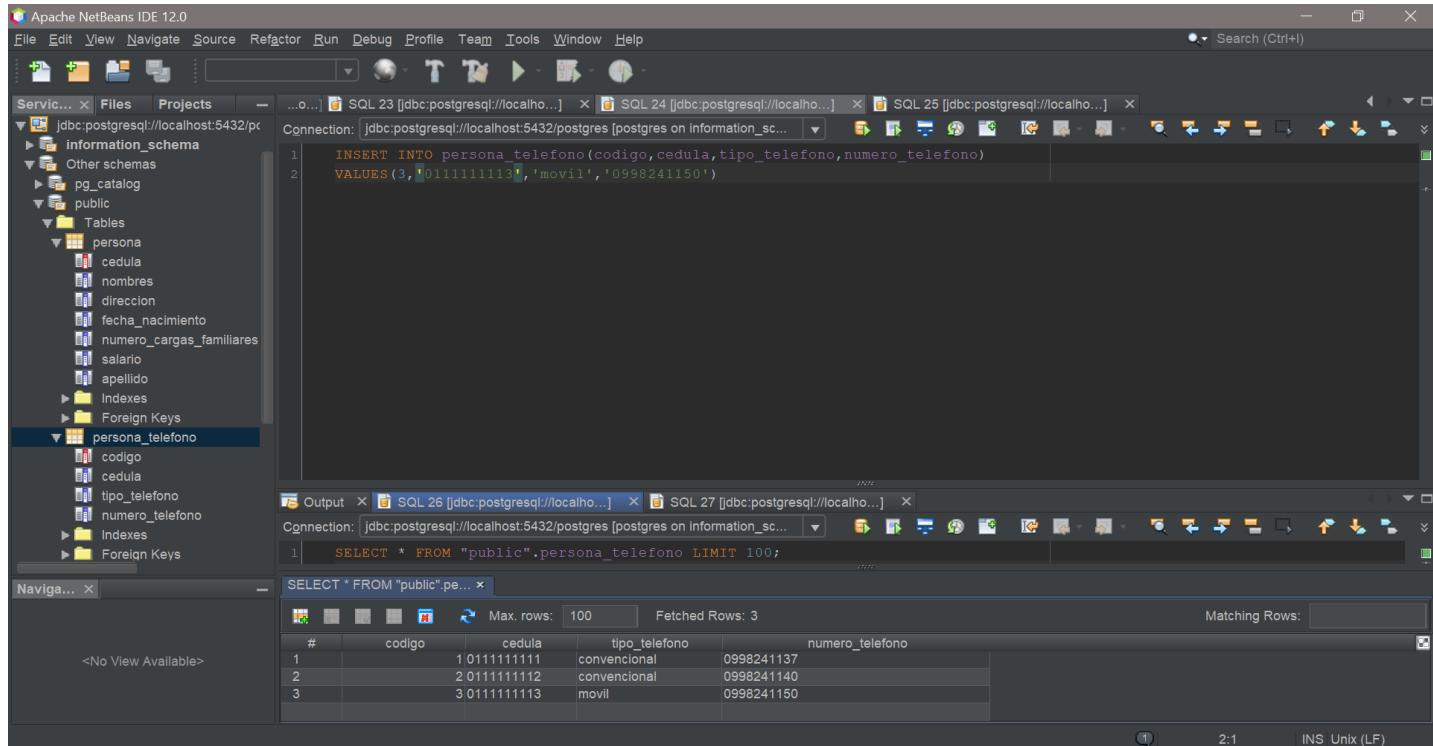
```
Execution finished after 0.02 s, 1 error occurred.  

[1:1] Executed successfully in 0 s.  

Fetching resultset took 0 s.  

Execution finished after 0.04 s, 1 error occurred.
```

- De su conjunto de personas de su base de datos inserte uno número de teléfono para 3 de ellas , esto en la tabla persona_telefono.



The screenshot shows the Apache NetBeans IDE interface. On the left, the Database Navigator displays the schema structure, including the 'persona' table and its columns: cedula, nombres, direccion, fecha_nacimiento, numero_cargas_familiares, salario, apellido, and persona_telefono. The 'persona_telefono' table is selected. In the center, the SQL Editor contains two queries:

```

1 INSERT INTO persona_telefono(codigo,cedula,tipo_telefono,numero_telefono)
VALUES(3,'0111111113','movil','0998241150')

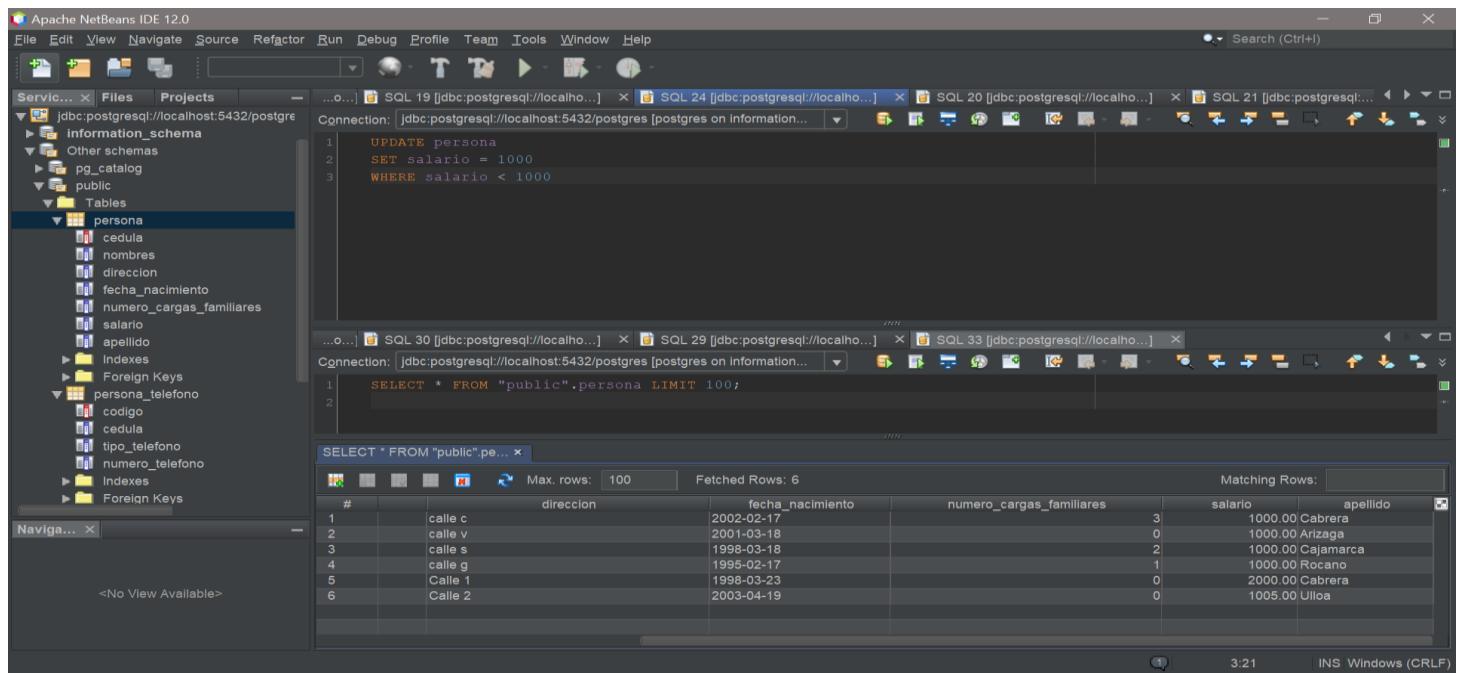
1 SELECT * FROM "public".persona_telefono LIMIT 100;

```

On the right, the Results Editor displays the output of the second query, showing three rows of data:

#	codigo	cedula	tipo_telefono	numero_telefono
1		1 0111111111	convencional	0998241137
2		2 0111111112	convencional	0998241140
3		3 0111111113	movil	0998241150

- Cambie el salario a 1000 USD a aquellas personas cuyo salario sea menor a 1000.



The screenshot shows the Apache NetBeans IDE interface. On the left, the Database Navigator displays the schema structure, including the 'persona' table and its columns: cedula, nombres, direccion, fecha_nacimiento, numero_cargas_familiares, salario, apellido, and persona_telefono. The 'persona' table is selected. In the center, the SQL Editor contains the following update query:

```

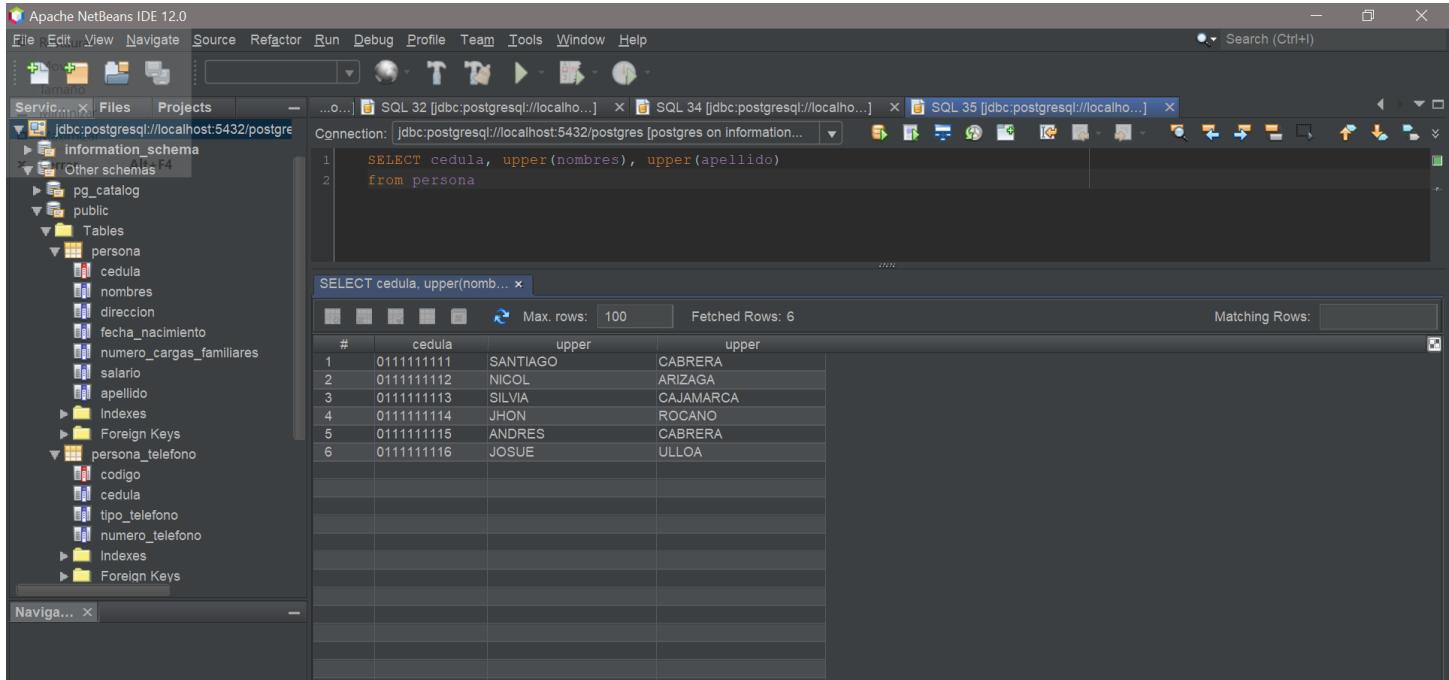
1 UPDATE persona
2 SET salario = 1000
3 WHERE salario < 1000

```

On the right, the Results Editor displays the output of the update query, followed by the output of the second query from the previous screenshot, showing six rows of data:

#	direccion	fecha_nacimiento	numero_cargas_familiares	salario	apellido
1	calle c	2002-02-17	3	1000.00	Cabrera
2	calle v	2001-03-18	0	1000.00	Arizaga
3	calle s	1998-03-18	2	1000.00	Cajamarca
4	calle g	1995-02-17	1	1000.00	Rocano
5	Calle 1	1998-03-23	0	2000.00	Cabrera
6	Calle 2	2003-04-19	0	1005.00	Ulloa

- Cambie los nombres y apellidos de todas las personas a mayúsculas

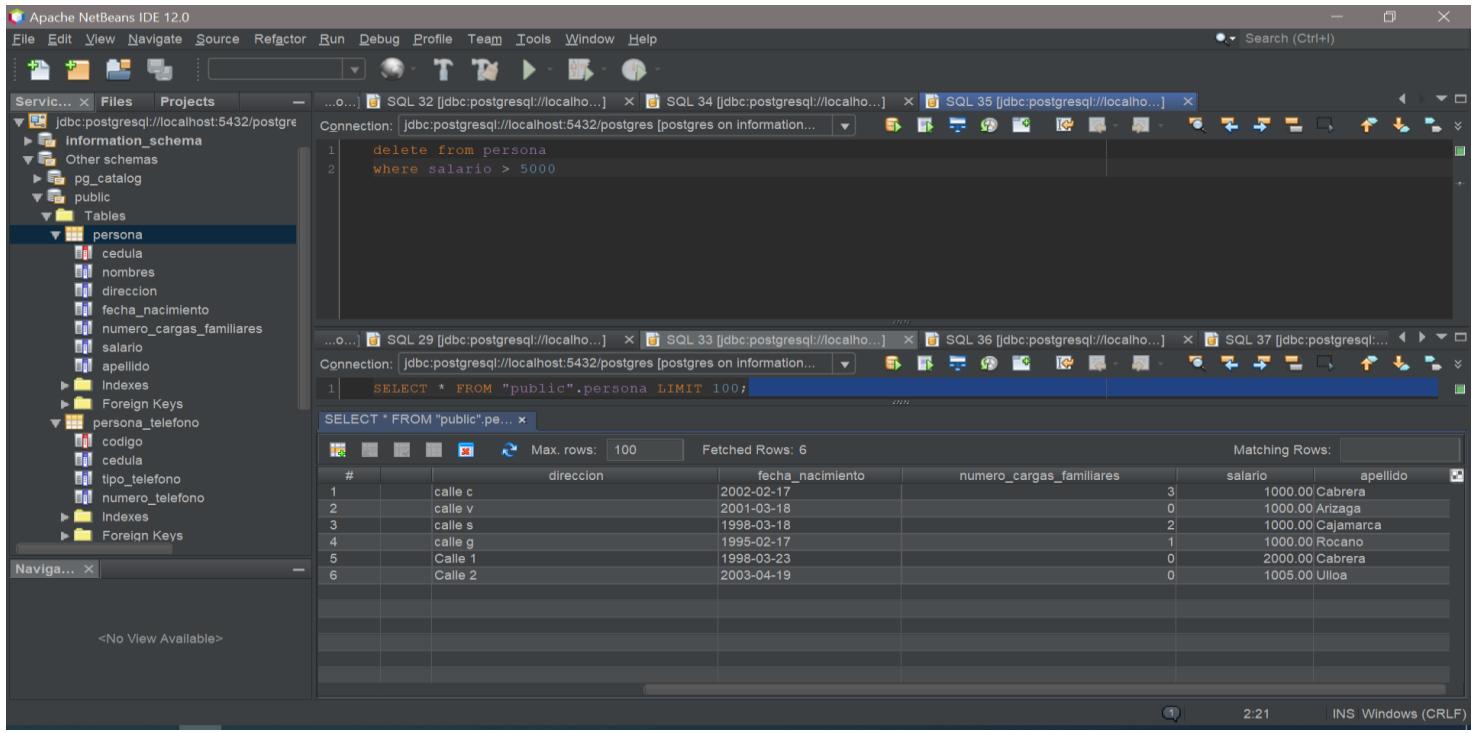


Apache NetBeans IDE 12.0

```
SELECT cedula, upper(nombres), upper(apellido)
from persona
```

#	cedula	upper	upper
1	0111111111	SANTIAGO	CABRERA
2	0111111112	NICOL	ARIZAGA
3	0111111113	SILVIA	CAJAMARCA
4	0111111114	JHON	ROCANO
5	0111111115	ANDRES	CABRERA
6	0111111116	JOSUE	ULLOA

- Borre los registros de las personas cuyo salario sea mayor a 5000 USD.



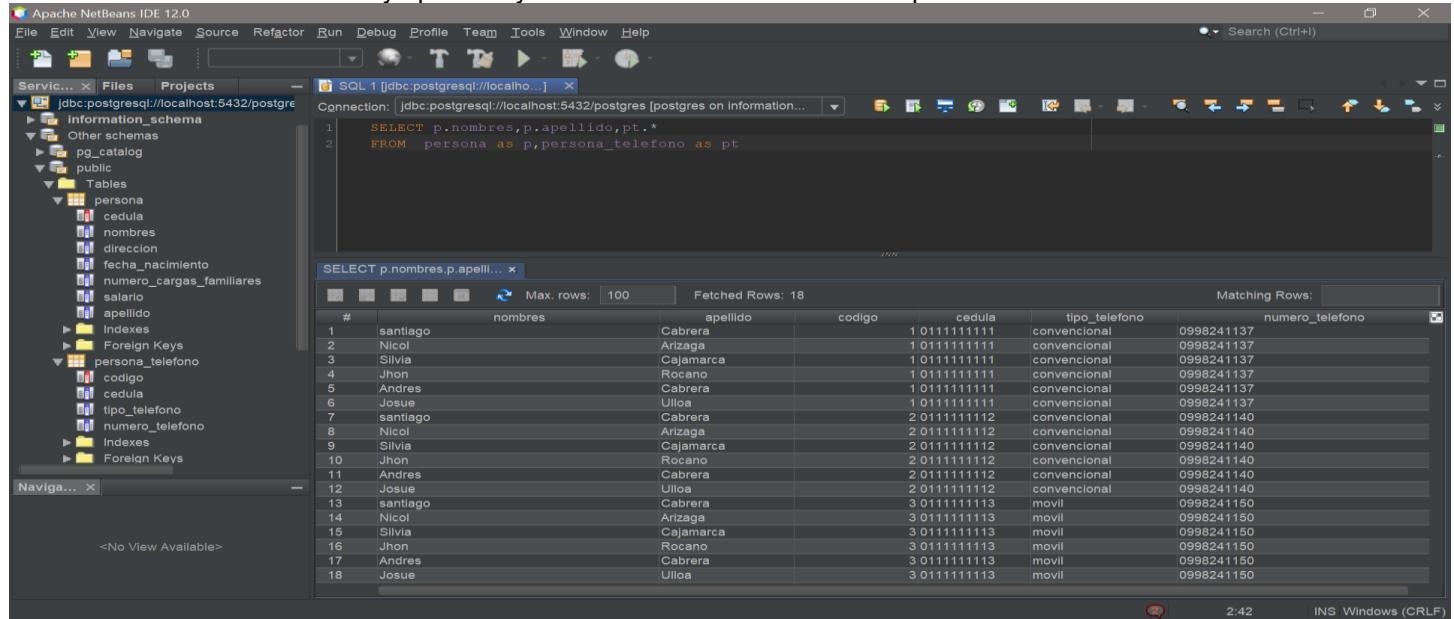
Apache NetBeans IDE 12.0

```
delete from persona
where salario > 5000
```

#	direccion	fecha_nacimiento	numero_cargas_familiares	salario	apellido
1	calle c	2002-02-17	3	1000.00	Cabrera
2	calle v	2001-03-18	0	1000.00	Arizaga
3	calle s	1998-03-18	2	1000.00	Cajamarca
4	calle g	1995-02-17	1	1000.00	Rocano
5	Calle 1	1998-03-23	0	2000.00	Cabrera
6	Calle 2	2003-04-19	0	1005.00	Ulloa

2.4 FOREIGN KEY

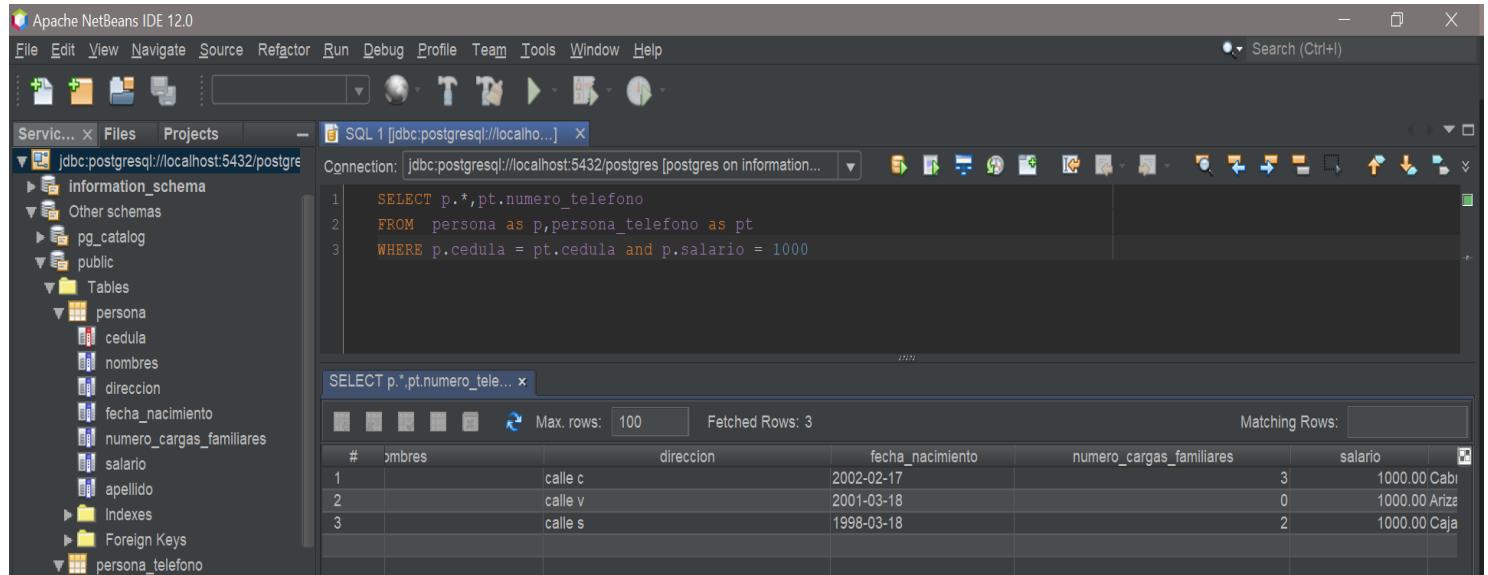
- Seleccione los nombre y apellidos y datos de teléfonos de todas las personas.



```
Apache NetBeans IDE 12.0
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Servic... Files Projects SQL 1 [jdbc:postgresql://localhost:5432/postgre]
Connection: jdbc:postgresql://localhost:5432/postgres [postgres on information...]
1 SELECT p.nombres,p.apellido,pt.*
2 FROM persona as p,persona_telefono as pt
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
2:42 INS Windows (CRLF)
```

#	nombres	apellido	codigo	cedula	tipo_telefono	numero_telefono
1	santiago	Cabrera	1 0111111111		convenional	0998241137
2	Nicol	Arizaga	1 0111111111		convenional	0998241137
3	Silvia	Cajamarca	1 0111111111		convenional	0998241137
4	Jhon	Rocano	1 0111111111		convenional	0998241137
5	Andres	Cabrera	1 0111111111		convenional	0998241137
6	Josue	Ulloa	1 0111111111		convenional	0998241137
7	santiago	Cabrera	2 0111111112		convenional	0998241140
8	Nicol	Arizaga	2 0111111112		convenional	0998241140
9	Silvia	Cajamarca	2 0111111112		convenional	0998241140
10	Jhon	Rocano	2 0111111112		convenional	0998241140
11	Andres	Cabrera	2 0111111112		convenional	0998241140
12	Josue	Ulloa	2 0111111112		convenional	0998241140
13	santiago	Cabrera	3 0111111113		movil	0998241150
14	Nicol	Arizaga	3 0111111113		movil	0998241150
15	Silvia	Cajamarca	3 0111111113		movil	0998241150
16	Jhon	Rocano	3 0111111113		movil	0998241150
17	Andres	Cabrera	3 0111111113		movil	0998241150
18	Josue	Ulloa	3 0111111113		movil	0998241150

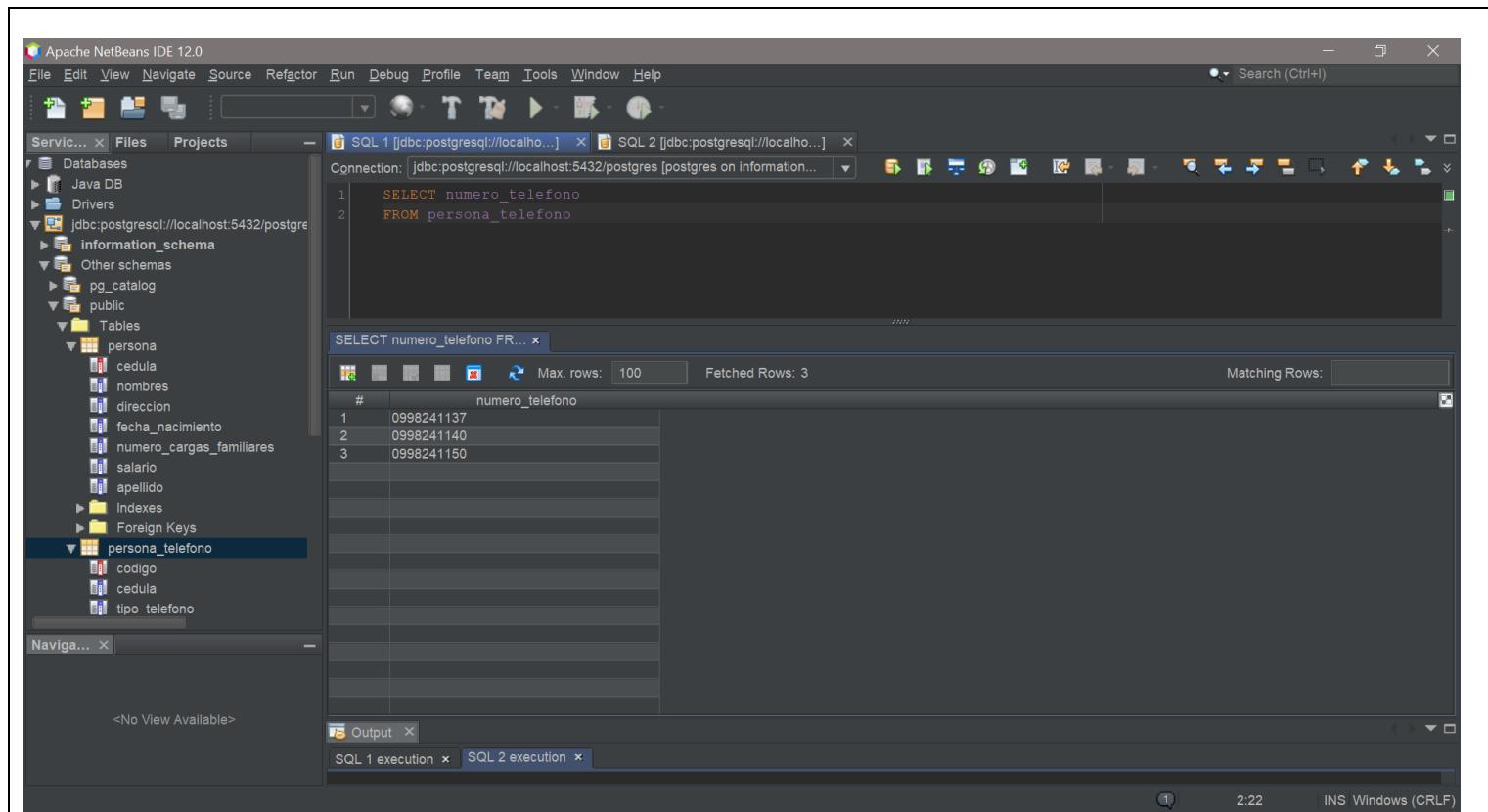
- Seleccione los datos de las personas y sus números de teléfonos de aquellas personas que tiene un salario de 1000 USD.



```
Apache NetBeans IDE 12.0
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Servic... Files Projects SQL 1 [jdbc:postgresql://localhost:5432/postgre]
Connection: jdbc:postgresql://localhost:5432/postgres [postgres on information...]
1 SELECT p.*,pt.numero_telefono
2 FROM persona as p,persona_telefono as pt
3 WHERE p.cedula = pt.cedula and p.salario = 1000
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
2:42 INS Windows (CRLF)
```

#	nombres	direccion	fecha_nacimiento	numero_cargas_familiares	salario
1	calle c	2002-02-17		3	1000.00 Cab
2	calle v	2001-03-18		0	1000.00 Arize
3	calle s	1998-03-18		2	1000.00 Caja

- Seleccione los números de teléfonos que existen en su base de datos.



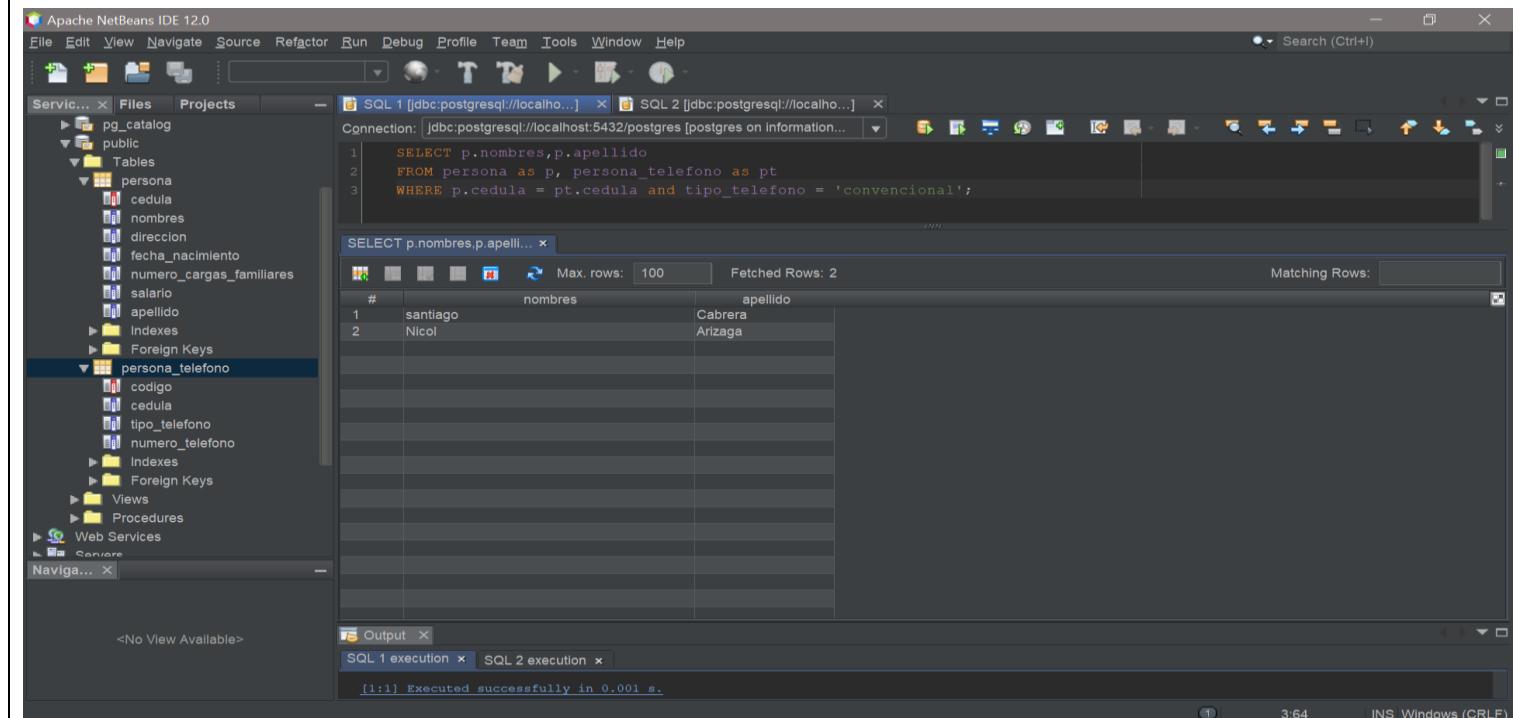
The screenshot shows the Apache NetBeans IDE 12.0 interface. On the left, the Navigator pane displays the database schema, including the 'person' table and its sub-table 'persona_telefono'. In the center, the SQL Editor pane contains the following query:

```
SELECT numero_telefono
FROM persona_telefono
```

The Results pane below the editor shows the fetched rows:

#	numero_telefono
1	0998241137
2	0998241140
3	0998241150

- Seleccione el nombre y apellido de las personas que tiene como teléfono un numero convencional.



The screenshot shows the Apache NetBeans IDE 12.0 interface. The Navigator pane displays the database schema, including the 'person' table and its sub-table 'persona_telefono'. In the center, the SQL Editor pane contains the following query:

```
SELECT p.nombres, p.apellido
FROM persona AS p, persona_telefono AS pt
WHERE p.cedula = pt.cedula AND tipo_telefono = 'convencional';
```

The Results pane below the editor shows the fetched rows:

#	nombres	apellido
1	santiago	Cabrera
2	Nicol	Arizaga

At the bottom of the Results pane, the message '[1:1] Executed successfully in 0.001 s.' is displayed.

RESULTADO(S) OBTENIDO(S):

Realizar procesos de Bases de Datos

Entender las aplicaciones de codificación de las nuevas características de base de datos

CONCLUSIONES:

Como conclusión puedo decir que debemos tener muy claro este tema debido a que es una parte muy importante. Debemos tener conocimiento de cómo crear tablas, actualizar, relacionarlas, insertar, seleccionar. Al trabajar con la herramienta postgres nos damos cuenta de que es muy completa y dinámica.

RECOMENDACIONES:

Recomiendo haber asistido a las clases ya que nos dio una introducción de esta misma y nos aportó con material muy útil el cual tiene toda la información para realizar los debidos procesos.

Nombre de estudiante: Jorge Santiago Cabrera Arias

Firma de estudiante:



CS Escaneado con CamScanner