
	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

	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

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

		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 practicas de Base de datos solamente utilizando código SQL (No se permite el uso del generador o ayuda de ningún IDE): <ul style="list-style-type: none"> • 2.1 Actividad: Realizar una base de datos para el equipo de futbol del curso. • 2.2 Crear las tabla PERSONA, con los siguientes datos: <ul style="list-style-type: none"> ◦ 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.			
<ul style="list-style-type: none"> ● Crear las tabla PERSONA_TELEFONO, con los siguientes datos: <ul style="list-style-type: none"> ◦ 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 guardaran valores como celular o convencional ◦ Número de teléfono, con un dimensión de 24 caracteres 			

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
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.

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

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

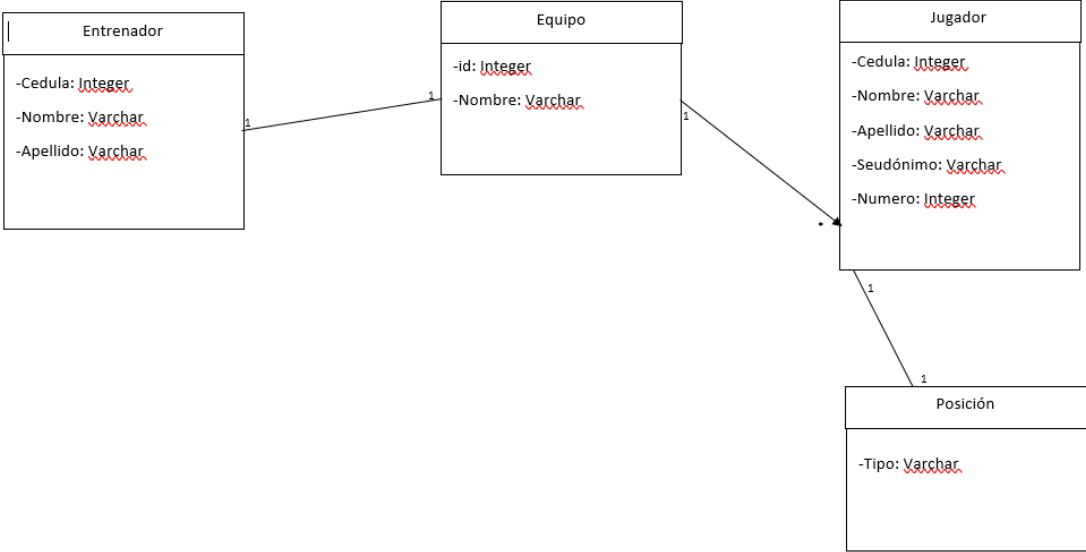
Firma: _____

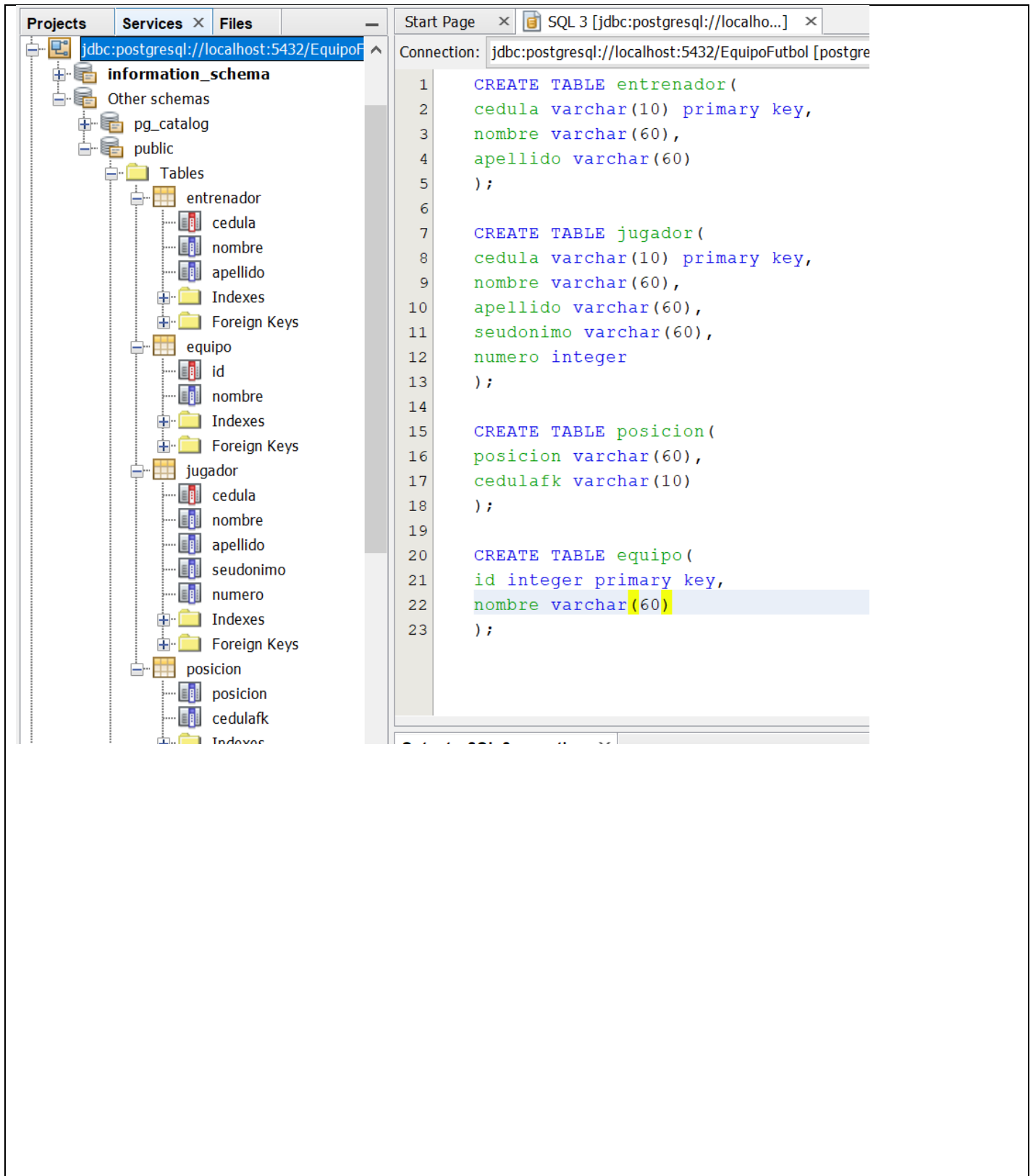


**FORMATO DE INFORME DE PRÁCTICA DE LABORATORIO /
TALLERES / CENTROS DE SIMULACIÓN – PARA
ESTUDIANTES**

Resolución CS N° 076-04-2016-04-20


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

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		
ACTIVIDADES DESARROLLADAS		
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):		
 <pre> erDiagram Entrenador --o{ Equipo : "1" Equipo --o{ Jugador : "1" Jugador --o{ Posición : "1" Entrenador { Integer Cedula Varchar Nombre Varchar Apellido } Equipo { Integer id Varchar Nombre } Jugador { Integer Cedula Varchar Nombre Varchar Apellido Varchar Seudónimo Integer Numero } Posición { Varchar Tipo </pre>		

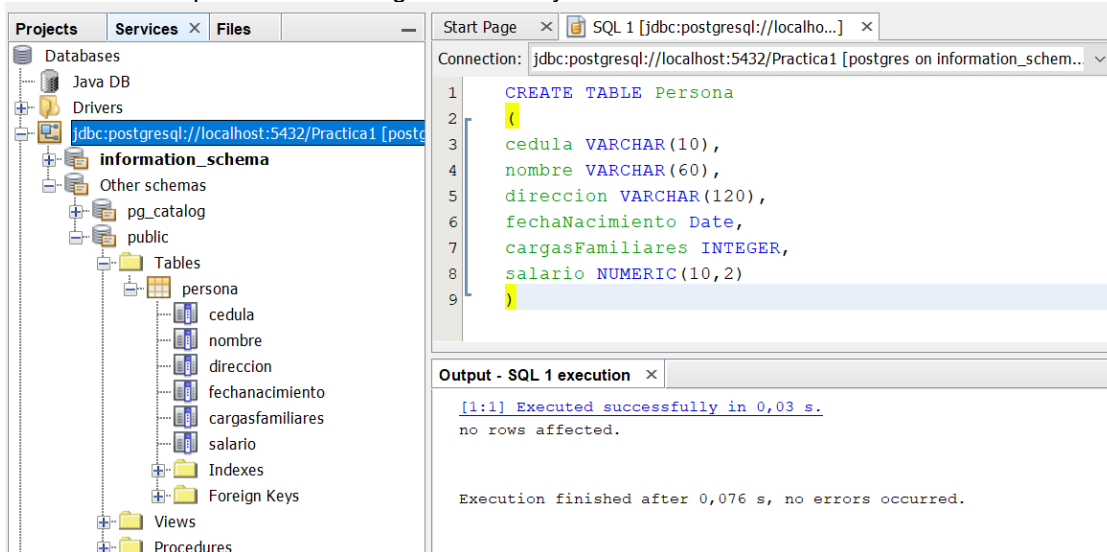


The screenshot displays a PostgreSQL database management tool interface. On the left, a tree view shows the database structure under the connection 'jdbc:postgresql://localhost:5432/EquipoF'. The 'information_schema' is expanded, showing 'Other schemas' (pg_catalog, public) and 'Tables' (entrenador, equipo, jugador, posicion). Each table has associated attributes like 'cedula', 'nombre', 'apellido', 'seudonimo', 'numero', 'id', 'posicion', and 'cedulafk'. On the right, a SQL editor window titled 'SQL 3 [jdbc:postgresql://localho...]' shows the following SQL code:

```
1 CREATE TABLE entrenador(  
2 cedula varchar(10) primary key,  
3 nombre varchar(60),  
4 apellido varchar(60)  
5 );  
6  
7 CREATE TABLE jugador(  
8 cedula varchar(10) primary key,  
9 nombre varchar(60),  
10 apellido varchar(60),  
11 seudonimo varchar(60),  
12 numero integer  
13 );  
14  
15 CREATE TABLE posicion(  
16 posicion varchar(60),  
17 cedulafk varchar(10)  
18 );  
19  
20 CREATE TABLE equipo(  
21 id integer primary key,  
22 nombre varchar(60)  
23 );
```

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

- 2.1 Actividad: Realizar una base de datos para el equipo de futbol del curso.
- 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 SQL Developer interface. On the left, the 'Projects' pane displays the database structure: 'Databases' > 'Java DB' > 'Drivers' > 'jdbc:postgresql://localhost:5432/Practica1 [postgres on information_schem...]' > 'information_schema' > 'Other schemas' > 'pg_catalog' > 'public' > 'Tables' > 'persona'. The 'persona' table is expanded, showing columns: 'cedula', 'nombre', 'direccion', 'fechanacimiento', 'cargasfamiliares', and 'salario'. The 'Output - SQL 1 execution' pane shows the successful execution of the following SQL statement:

```

1 CREATE TABLE Persona
2 (
3     cedula VARCHAR(10),
4     nombre VARCHAR(60),
5     direccion VARCHAR(120),
6     fechaNacimiento Date,
7     cargasFamiliares INTEGER,
8     salario NUMERIC(10,2)
9 )

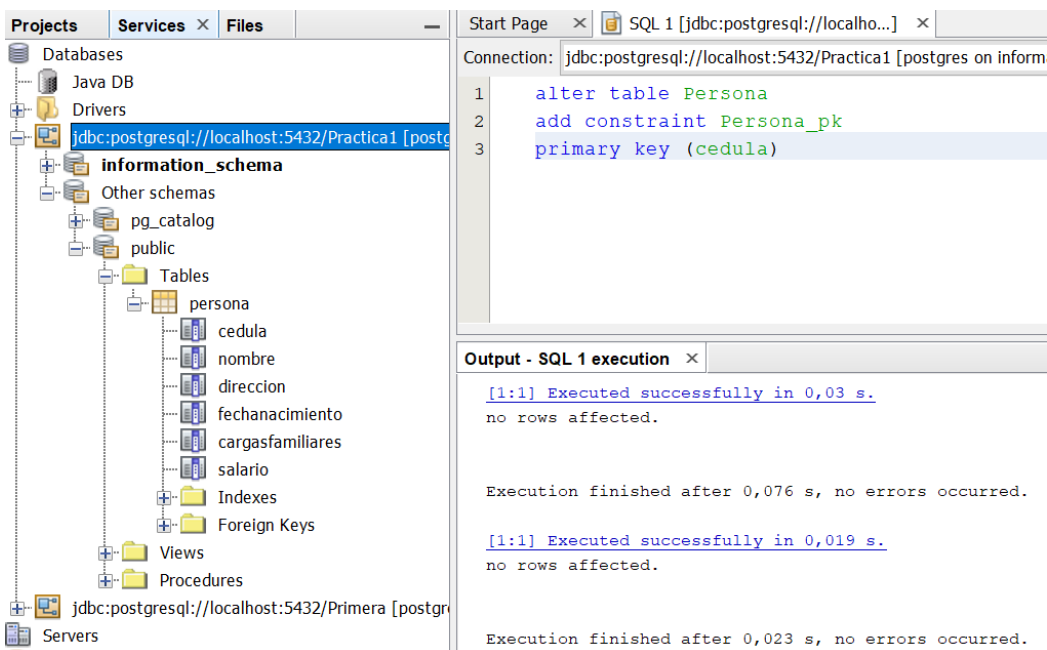
```

Output - SQL 1 execution

[1:1] Executed successfully in 0,03 s.
no rows affected.

Execution finished after 0,076 s, no errors occurred.

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



The screenshot shows the SQL Developer interface. On the left, the 'Projects' pane displays the database structure: 'Databases' > 'Java DB' > 'Drivers' > 'jdbc:postgresql://localhost:5432/Practica1 [postgres on information_schem...]' > 'information_schema' > 'Other schemas' > 'pg_catalog' > 'public' > 'Tables' > 'persona'. The 'persona' table is expanded, showing columns: 'cedula', 'nombre', 'direccion', 'fechanacimiento', 'cargasfamiliares', and 'salario'. The 'Output - SQL 1 execution' pane shows the successful execution of the following SQL statement:

```

1 alter table Persona
2 add constraint Persona_pk
3 primary key (cedula)

```


Output - SQL 1 execution

[1:1] Executed successfully in 0,03 s.
no rows affected.

Execution finished after 0,076 s, no errors occurred.

[1:1] Executed successfully in 0,019 s.
no rows affected.

Execution finished after 0,023 s, no errors occurred.

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

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

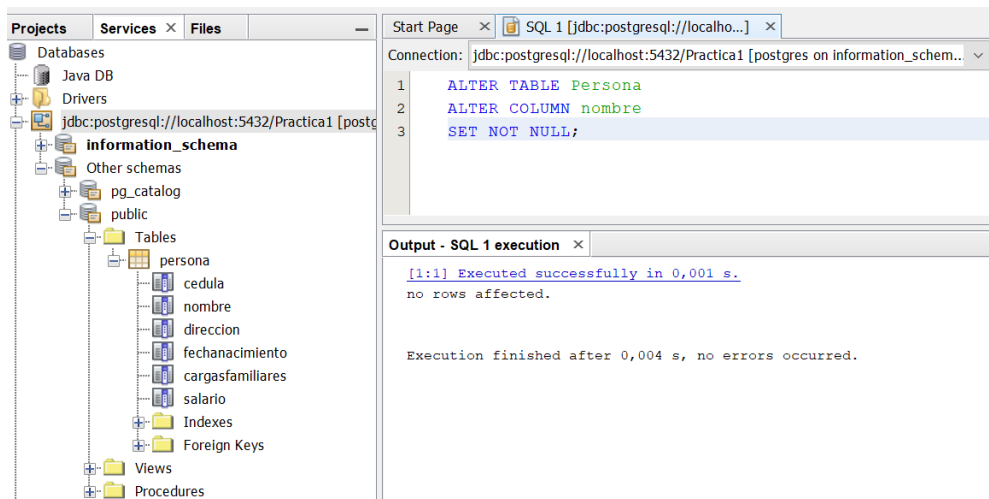
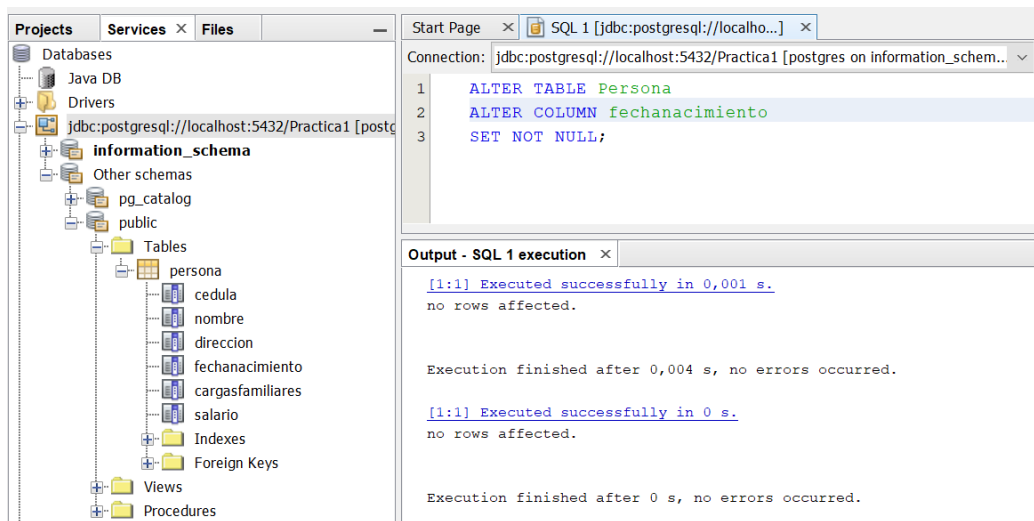
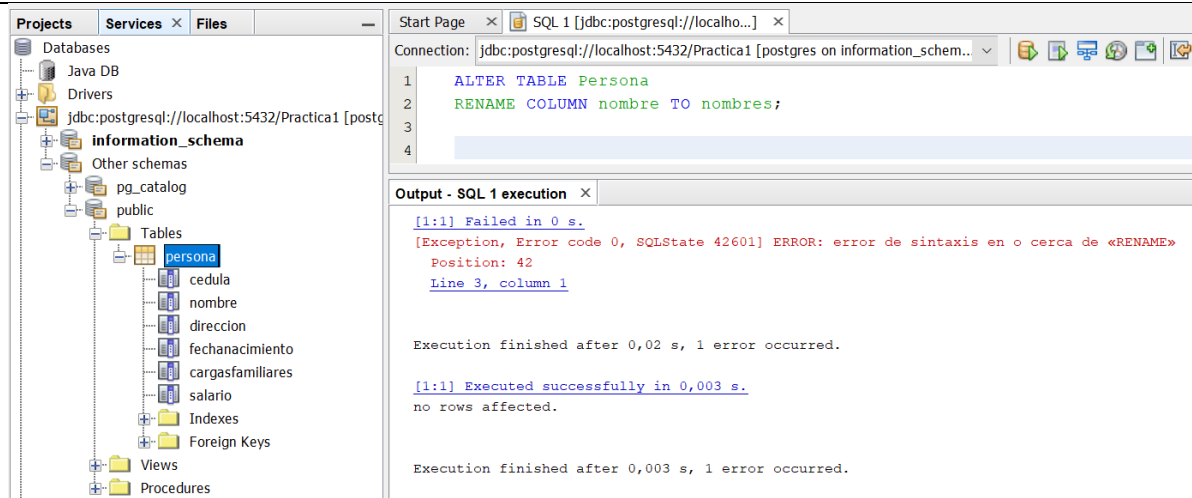


Imagen de la modificación del nombre



Modificación fecha de nacimiento

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



The screenshot shows a SQL IDE interface. On the left, the 'Projects' pane displays a database structure with 'information_schema' expanded, showing 'public' and 'Tables'. The 'persona' table is selected. The main editor shows the following SQL code:

```
1 ALTER TABLE Persona
2 RENAME COLUMN nombre TO nombres;
3
4
```

The 'Output - SQL 1 execution' pane shows the following error message:

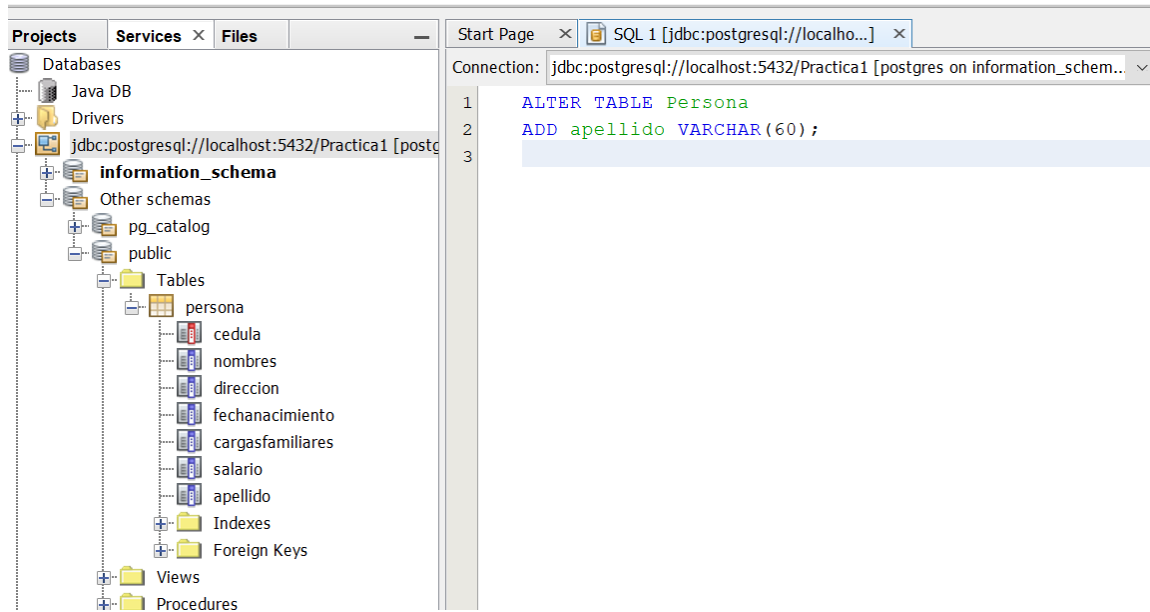
```
[1:1] Failed in 0 s.
[Exception, Error code 0, SQLState 42601] ERROR: error de sintaxis en o cerca de «RENAME»
Position: 42
Line 3, column 1

Execution finished after 0,02 s, 1 error occurred.

[1:1] Executed successfully in 0,003 s.
no rows affected.

Execution finished after 0,003 s, 1 error occurred.
```

Renombrar la columna nombre



The screenshot shows the same SQL IDE interface. The 'Output - SQL 1 execution' pane now shows a successful execution:

```
[1:1] Executed successfully in 0,003 s.
no rows affected.

Execution finished after 0,003 s, 1 error occurred.
```


The main editor shows the following SQL code:

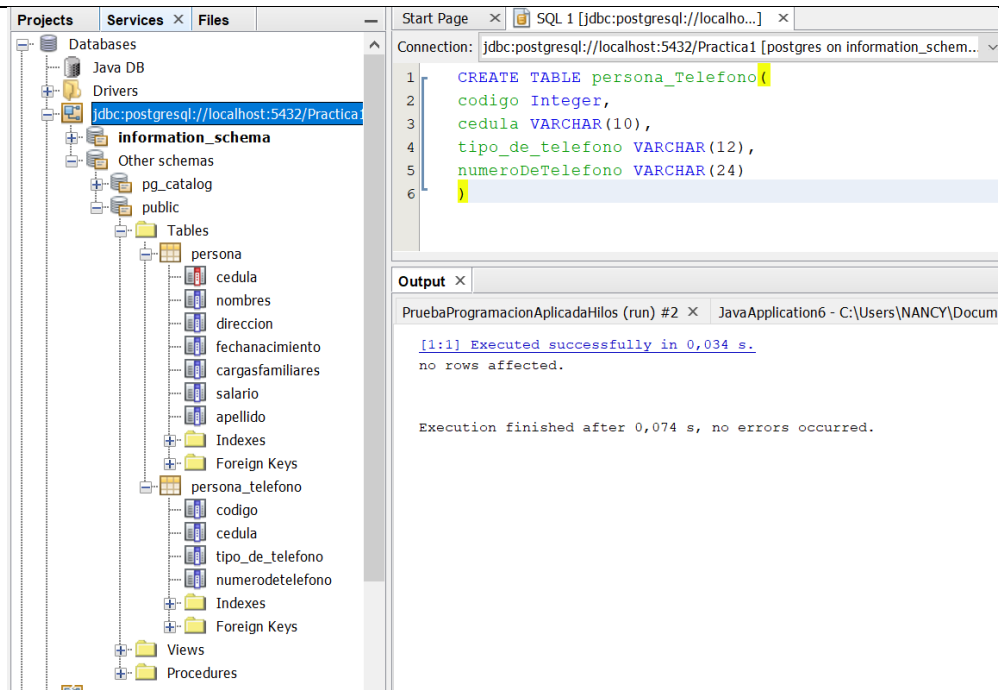
```
1 ALTER TABLE Persona
2 ADD apellido VARCHAR(60);
3
```

The 'Projects' pane on the left shows the 'persona' table with the following columns: cedula, nombres, direccion, fechanacimiento, cargasfamiliares, salario, and apellido.

5) 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 guardaran valores como celular o convencional
- Número de teléfono, con un dimensión de 24 caracteres

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



The screenshot shows the NetBeans IDE interface. On the left, the 'Projects' pane displays a tree view of the database structure, including 'Databases', 'Drivers', 'information_schema', 'pg_catalog', 'public', 'Tables', 'Indexes', 'Foreign Keys', 'Views', and 'Procedures'. The 'personas' table is selected under 'Tables'. The 'SQL 1' editor on the right contains the following SQL code:

```

1 CREATE TABLE persona_telefono(
2     codigo Integer,
3     cedula VARCHAR(10),
4     tipo_de_telefono VARCHAR(12),
5     numeroDeTelefono VARCHAR(24)
6 )

```

The 'Output' pane at the bottom shows the execution results:

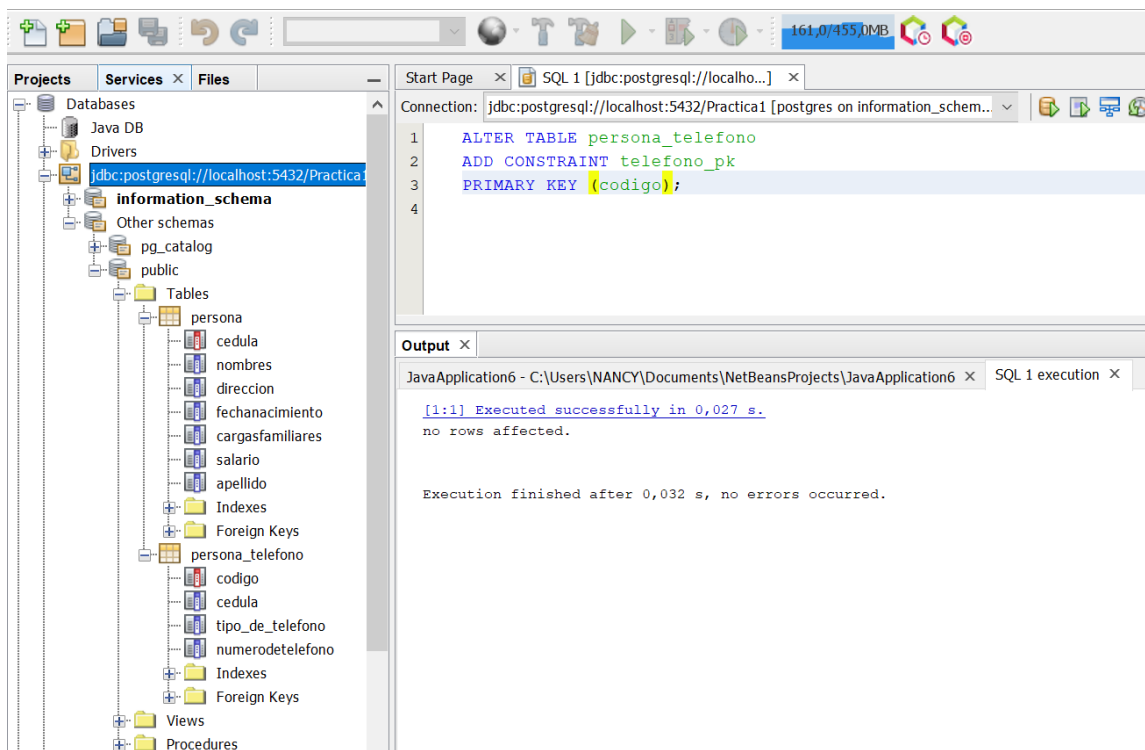
```

PruebaProgramacionAplicadaHilos (run) #2 x JavaApplication6 - C:\Users\NANCY\Docum
[1:1] Executed successfully in 0,034 s.
no rows affected.

Execution finished after 0,074 s, no errors occurred.

```

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



The screenshot shows the NetBeans IDE interface. The 'SQL 1' editor on the right contains the following SQL code:

```

1 ALTER TABLE persona_telefono
2 ADD CONSTRAINT telefono_pk
3 PRIMARY KEY (codigo);
4

```

The 'Output' pane at the bottom shows the execution results:

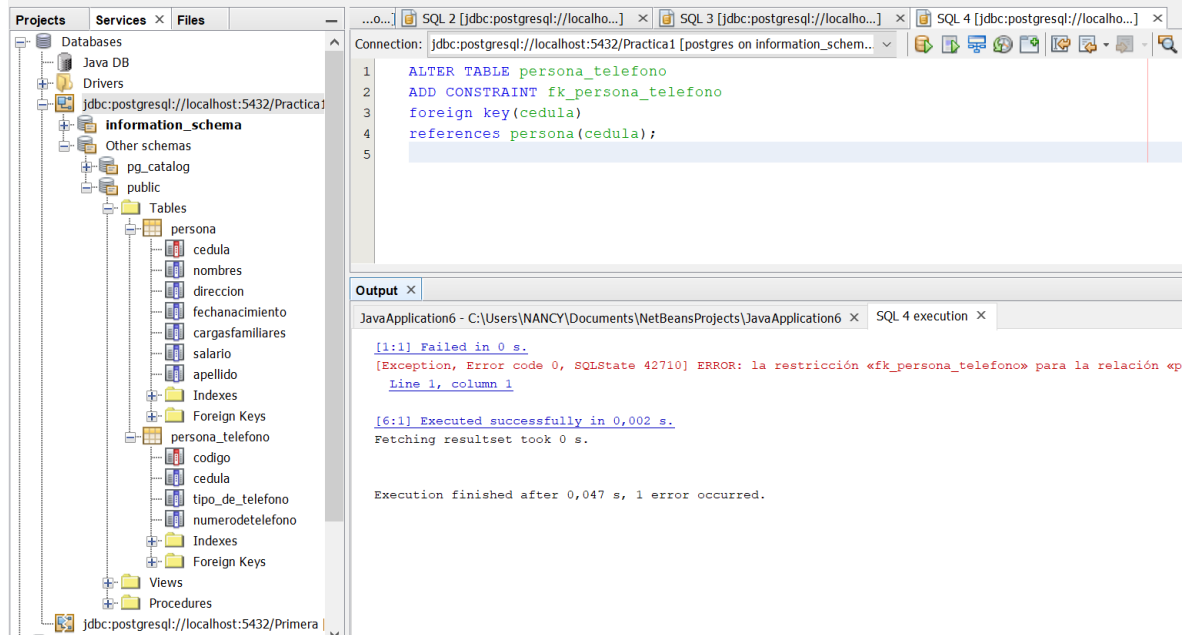
```

JavaApplication6 - C:\Users\NANCY\Documents\NetBeansProjects\JavaApplication6 x SQL 1 execution x
[1:1] Executed successfully in 0,027 s.
no rows affected.

Execution finished after 0,032 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



Projects Services Files

Databases

- Java DB
- Drivers
- jdbc:postgresql://localhost:5432/Practica1
 - information_schema
 - Other schemas
 - pg_catalog
 - public
 - Tables
 - persona
 - cedula
 - nombres
 - direccion
 - fechanacimiento
 - cargasfamiliares
 - salario
 - apellido
 - Indexes
 - Foreign Keys
 - persona_telefono
 - codigo
 - cedula
 - tipo_de_telefono
 - numerodetelefono
 - Indexes
 - Foreign Keys
 - Views
 - Procedures

jdbc:postgresql://localhost:5432/Primera

SQL 2 [jdbc:postgresql://localhost:...] SQL 3 [jdbc:postgresql://localhost:...] SQL 4 [jdbc:postgresql://localhost:...]

Connection: jdbc:postgresql://localhost:5432/Practica1 [postgres on information_schem...

```

1 ALTER TABLE persona_telefono
2 ADD CONSTRAINT fk_persona_telefono
3 foreign key (cedula)
4 references persona (cedula);
5

```

Output

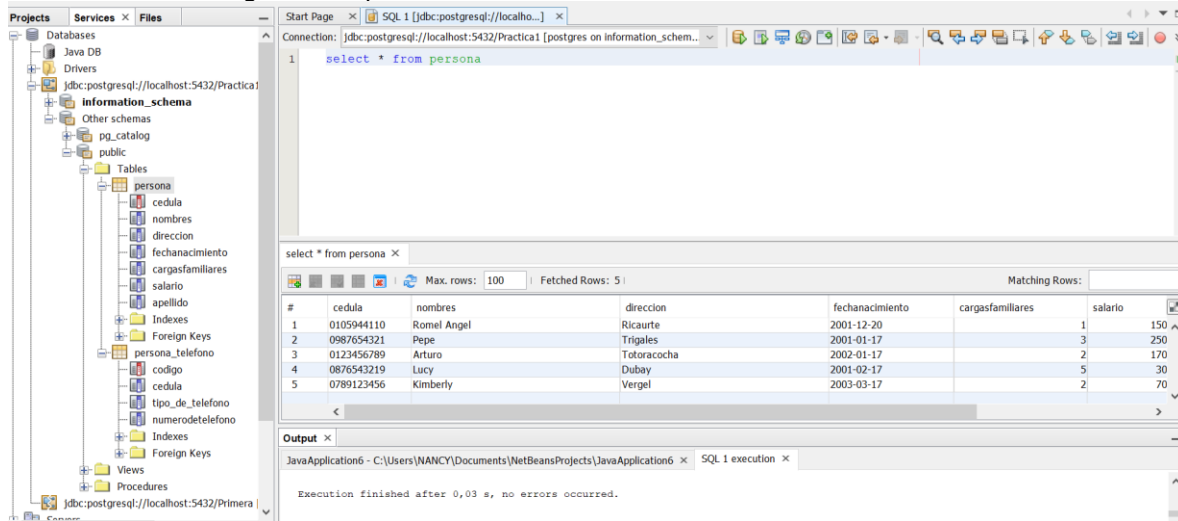
JavaApplication6 - C:\Users\NANCY\Documents\NetBeansProjects\JavaApplication6 x SQL 4 execution x

[1:1] Failed in 0 s.
[Exception, Error code 0, SQLState 42710] ERROR: la restricción «fk_persona_telefono» para la relación «p
Line 1, column 1

[6:1] Executed successfully in 0,002 s.
Fetching resultset took 0 s.

Execution finished after 0,047 s, 1 error occurred.

2.3 SELECT de todos los registros de personas



Projects Services Files

Databases

- Java DB
- Drivers
- jdbc:postgresql://localhost:5432/Practica1
 - information_schema
 - Other schemas
 - pg_catalog
 - public
 - Tables
 - persona
 - cedula
 - nombres
 - direccion
 - fechanacimiento
 - cargasfamiliares
 - salario
 - apellido
 - Indexes
 - Foreign Keys
 - persona_telefono
 - codigo
 - cedula
 - tipo_de_telefono
 - numerodetelefono
 - Indexes
 - Foreign Keys
 - Views
 - Procedures

jdbc:postgresql://localhost:5432/Primera

Start Page SQL 1 [jdbc:postgresql://localhost:...] x

Connection: jdbc:postgresql://localhost:5432/Practica1 [postgres on information_schem...

```

1 select * from persona

```

select * from persona x

Max. rows: 100 | Fetched Rows: 5 | Matching Rows:


#	cedula	nombres	direccion	fechanacimiento	cargasfamiliares	salario
1	0105944110	Romel Angel	Ricaurte	2001-12-20	1	150
2	0987654321	Pepe	Trigales	2001-01-17	3	250
3	0123456789	Arturo	Totoracocha	2002-01-17	2	170
4	0876543219	Lucy	Dubay	2001-02-17	5	30
5	0789123456	Kimberly	Vergel	2003-03-17	2	70

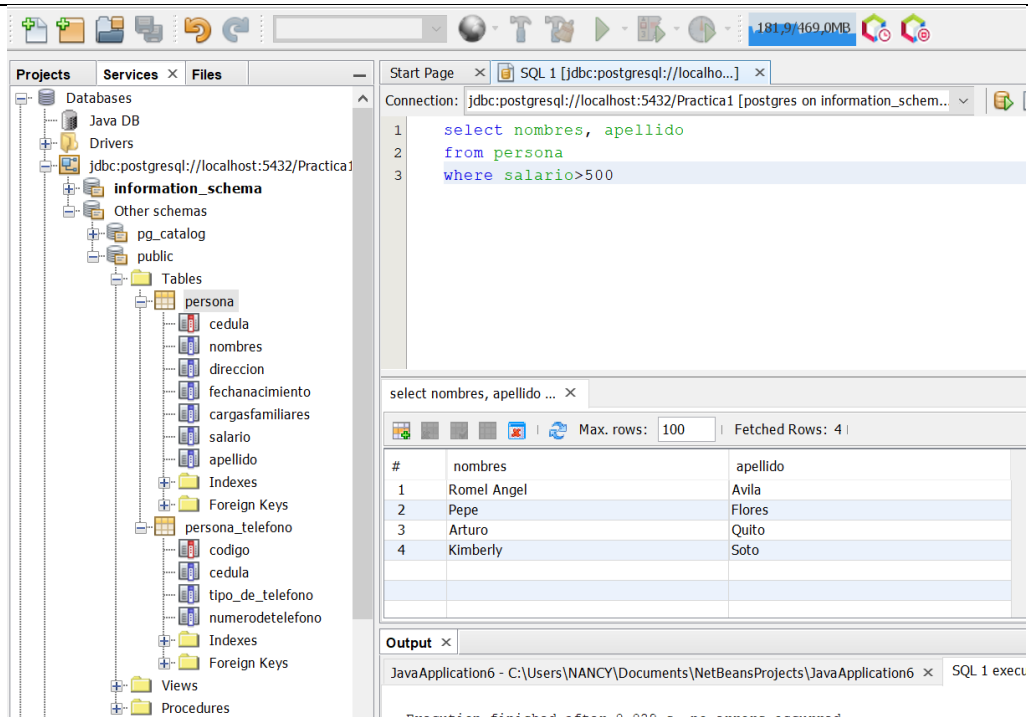
Output

JavaApplication6 - C:\Users\NANCY\Documents\NetBeansProjects\JavaApplication6 x SQL 1 execution x

Execution finished after 0,03 s, no errors occurred.

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

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



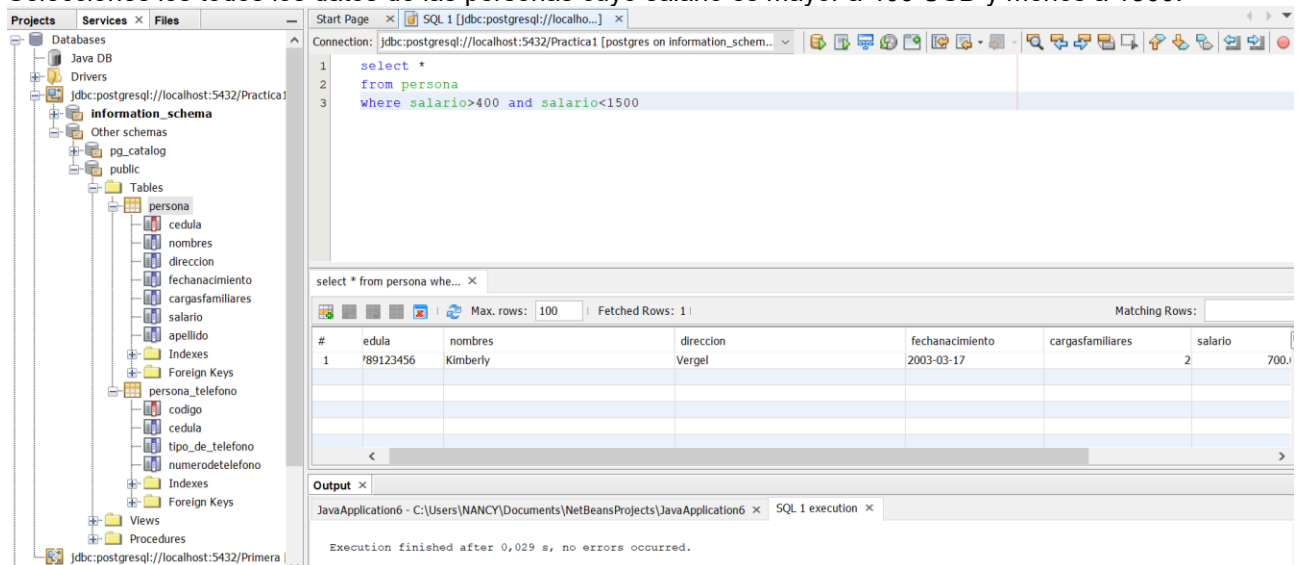
The screenshot shows the NetBeans IDE interface. On the left, the 'Projects' pane displays a database connection 'jdbc:postgresql://localhost:5432/Practica1' with an 'information_schema' database. The 'Tables' list shows a 'persona' table. The main editor displays a SQL query:

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

The 'Output' pane shows the results of the query:

#	nombres	apellido
1	Romel Angel	Avila
2	Pepe	Flores
3	Arturo	Quito
4	Kimberly	Soto

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



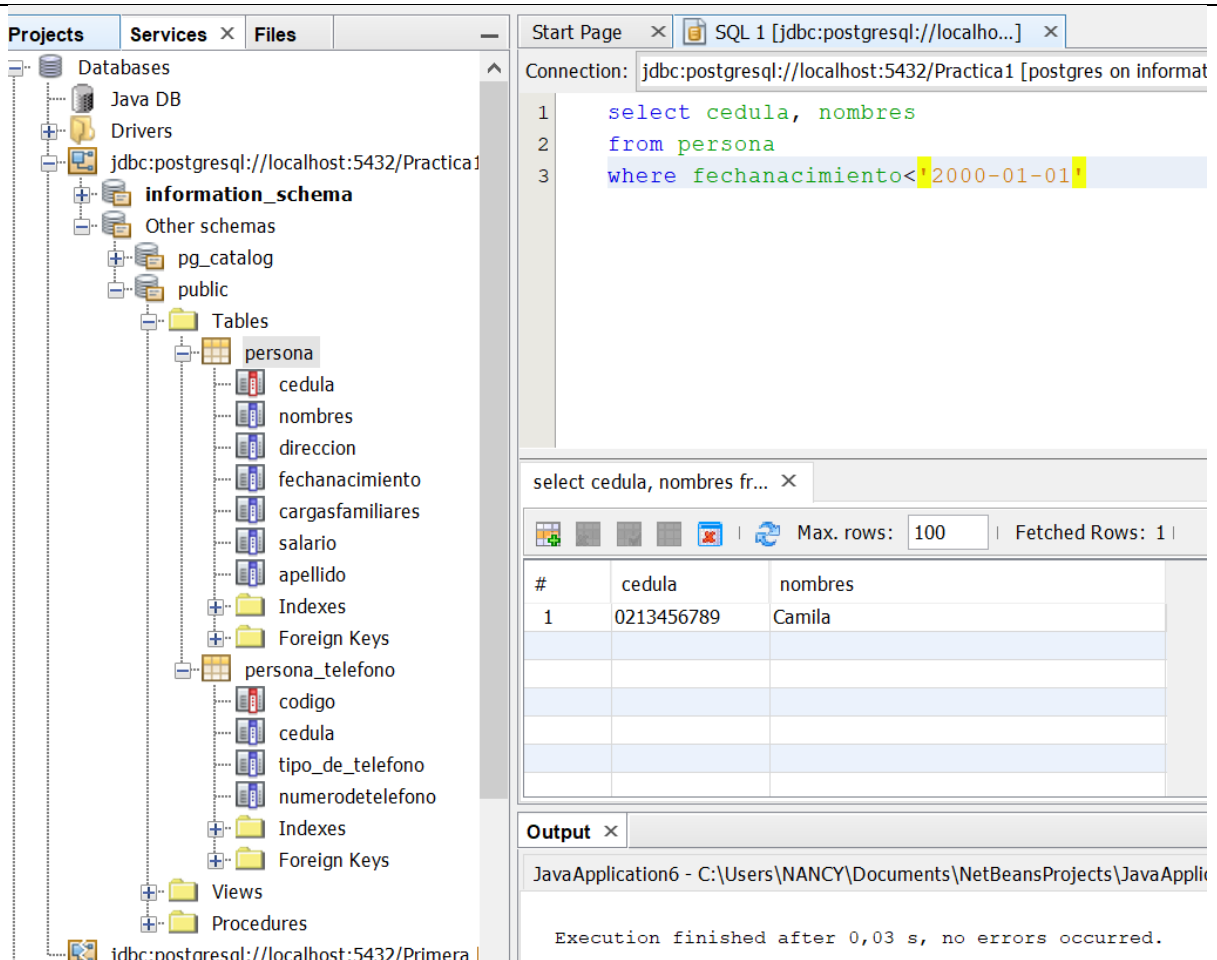
The screenshot shows the NetBeans IDE interface. On the left, the 'Projects' pane displays a database connection 'jdbc:postgresql://localhost:5432/Practica1' with an 'information_schema' database. The 'Tables' list shows a 'persona' table. The main editor displays a SQL query:

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

The 'Output' pane shows the results of the query:

#	edula	nombres	direccion	fechanacimiento	cargasfamiliares	salario
1	789123456	Kimberly	Vergel	2003-03-17	2	700.0

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



The screenshot shows the NetBeans IDE interface. On the left, the 'Projects' pane displays a tree structure of a PostgreSQL database named 'Practica1'. The tree includes 'Databases', 'Java DB', 'Drivers', 'jdbc:postgresql://localhost:5432/Practica1', 'information_schema', 'Other schemas', 'pg_catalog', 'public', 'Tables', 'persona' (with sub-entries for 'cedula', 'nombres', 'direccion', 'fechanacimiento', 'cargasfamiliares', 'salario', 'apellido', 'Indexes', 'Foreign Keys'), 'persona_telefono' (with sub-entries for 'codigo', 'cedula', 'tipo_de_telefono', 'numerodetelefono', 'Indexes', 'Foreign Keys'), 'Views', and 'Procedures'.

The 'SQL 1' editor shows the following query:

```

1  select cedula, nombres
2  from persona
3  where fechanacimiento < 2000-01-01

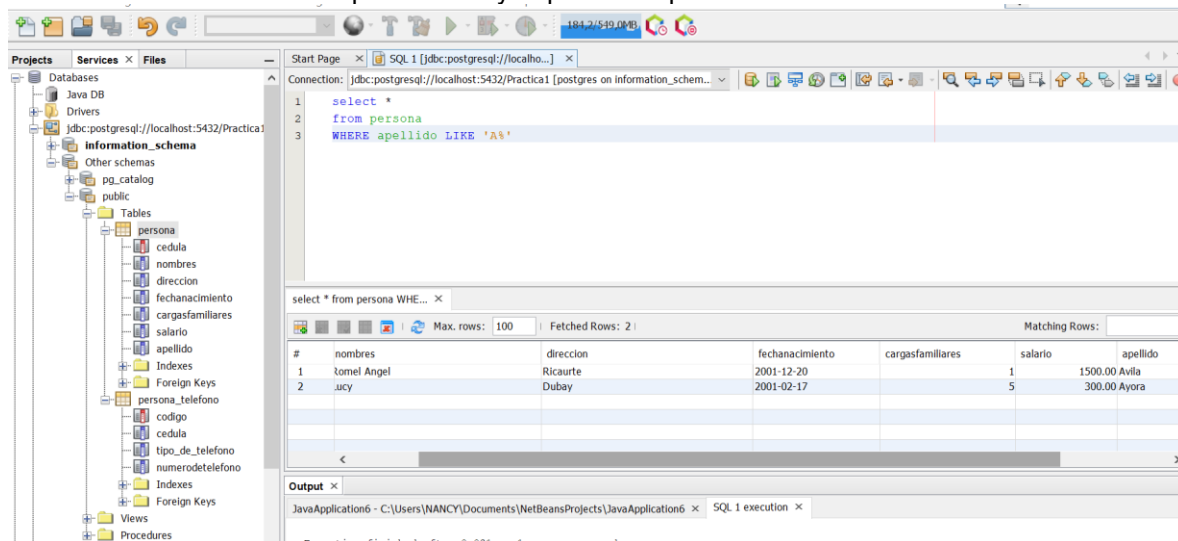
```

The 'Output' pane shows the execution results:

#	cedula	nombres
1	0213456789	Camila

The 'Output' pane also displays the message: 'Execution finished after 0,03 s, no errors occurred.'

- Selecciones los datos de las personas cuyo apellido empiece con la letra "a".



The screenshot shows the NetBeans IDE interface. On the left, the 'Projects' pane displays a tree structure of a PostgreSQL database named 'Practica1'. The tree includes 'Databases', 'Java DB', 'Drivers', 'jdbc:postgresql://localhost:5432/Practica1', 'information_schema', 'Other schemas', 'pg_catalog', 'public', 'Tables', 'persona' (with sub-entries for 'cedula', 'nombres', 'direccion', 'fechanacimiento', 'cargasfamiliares', 'salario', 'apellido', 'Indexes', 'Foreign Keys'), 'persona_telefono' (with sub-entries for 'codigo', 'cedula', 'tipo_de_telefono', 'numerodetelefono', 'Indexes', 'Foreign Keys'), 'Views', and 'Procedures'.

The 'SQL 1' editor shows the following query:

```

1  select *
2  from persona
3  WHERE apellido LIKE 'A%'

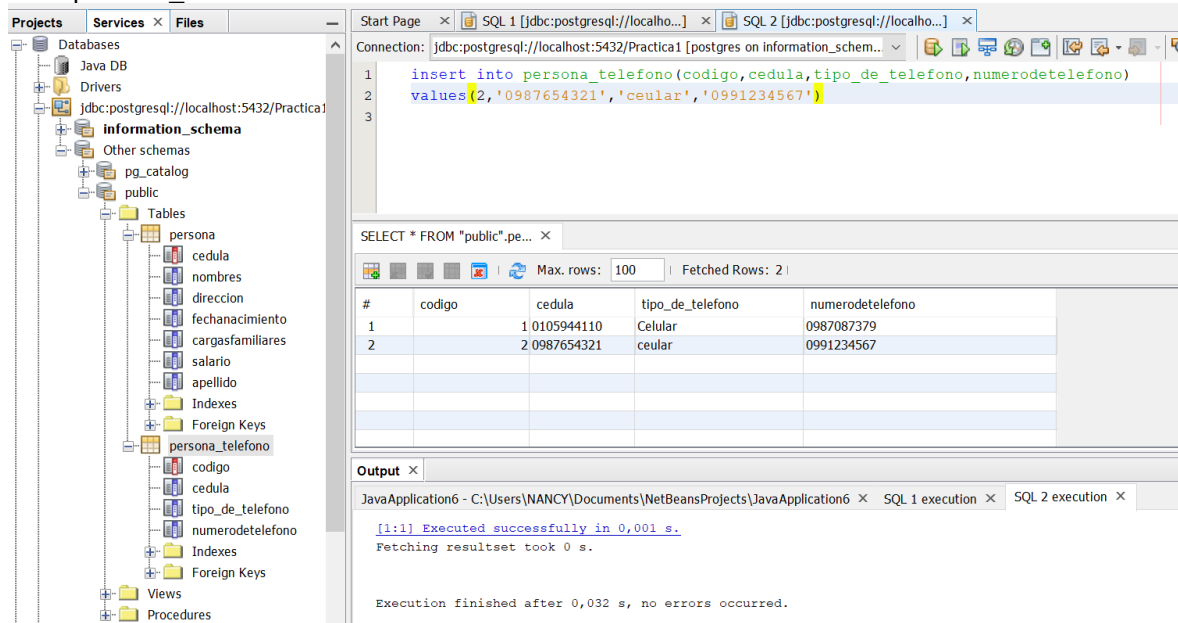
```

The 'Output' pane shows the execution results:

#	nombres	direccion	fechanacimiento	cargasfamiliares	salario	apellido
1	Ricard Angel	Dubay	2001-12-20	1	1500.00	Avila
2	Lucy	Dubay	2001-02-17	5	300.00	Ayora

The 'Output' pane also displays the message: 'Execution finished after 0,03 s, no errors 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 NetBeans IDE with a PostgreSQL connection. The left sidebar displays the database structure, including the 'persona' table and the 'persona_telefono' table. The main editor shows an SQL query to insert a new record into 'persona_telefono'.

```

1 insert into persona_telefono(codigo,cedula,tipo_de_telefono,numerodetelefono)
2 values(2,'0987654321','ceular','0991234567')
3

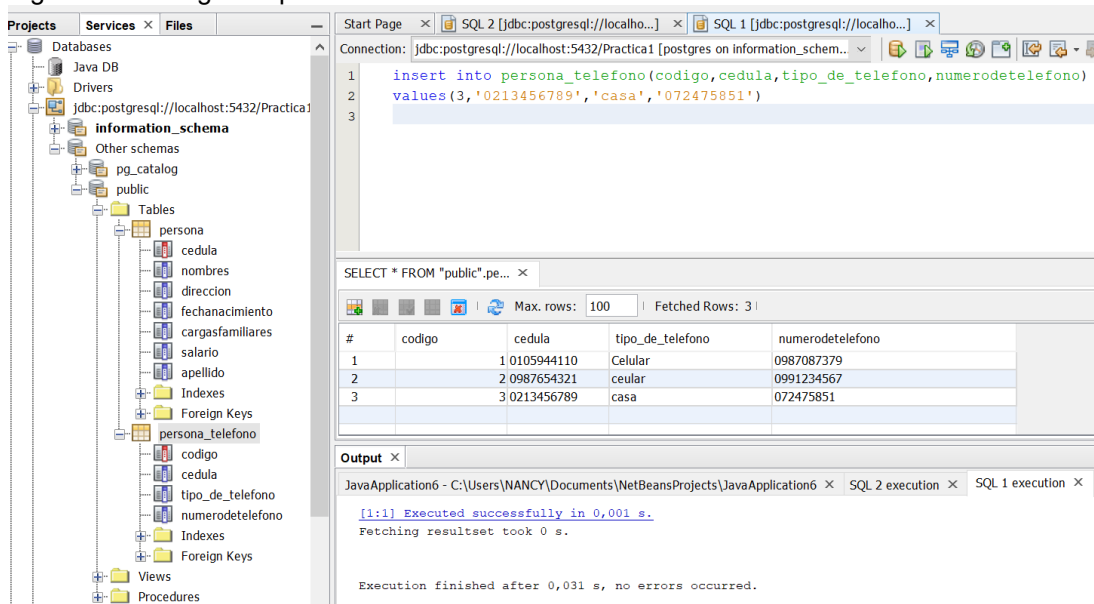
```

The output window shows the execution results:

#	codigo	cedula	tipo_de_telefono	numerodetelefono
1		1 0105944110	Celular	0987087379
2		2 0987654321	ceular	0991234567

Output: [1:1] Executed successfully in 0,001 s. Fetching resultset took 0 s. Execution finished after 0,032 s, no errors occurred.

Ingreso de la segunda persona



The screenshot shows the NetBeans IDE with a PostgreSQL connection. The left sidebar displays the database structure, including the 'persona' table and the 'persona_telefono' table. The main editor shows an SQL query to insert a new record into 'persona_telefono'.

```

1 insert into persona_telefono(codigo,cedula,tipo_de_telefono,numerodetelefono)
2 values(3,'0213456789','casa','072475851')
3

```

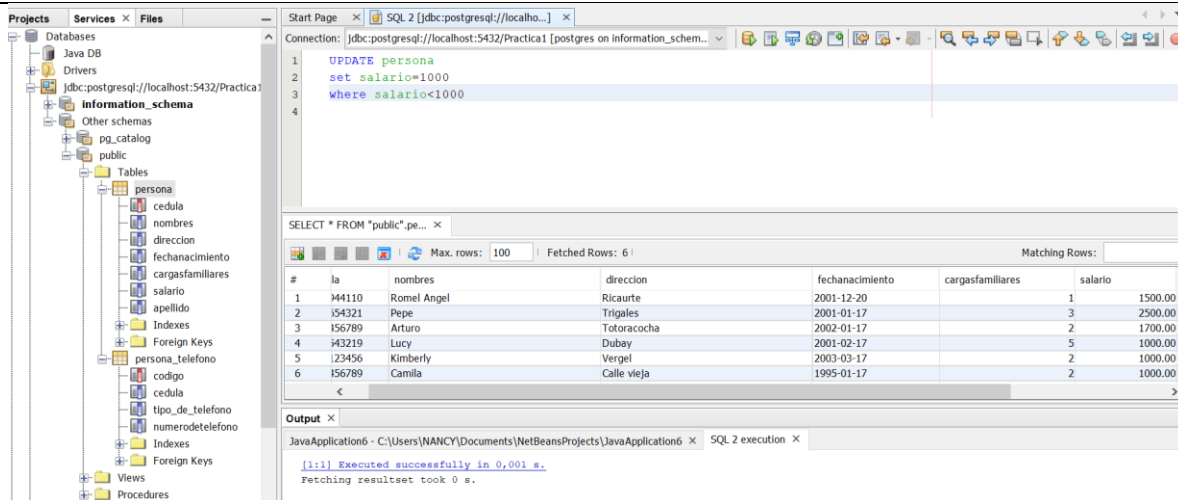
The output window shows the execution results:

#	codigo	cedula	tipo_de_telefono	numerodetelefono
1		1 0105944110	Celular	0987087379
2		2 0987654321	ceular	0991234567
3		3 0213456789	casa	072475851

Output: [1:1] Executed successfully in 0,001 s. Fetching resultset took 0 s. Execution finished after 0,031 s, no errors occurred.

Ingreso de la tercera persona

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



The screenshot shows the NetBeans IDE with a PostgreSQL database connection. The left sidebar displays the database structure, including the 'personas' table. The main window shows an SQL query being executed:

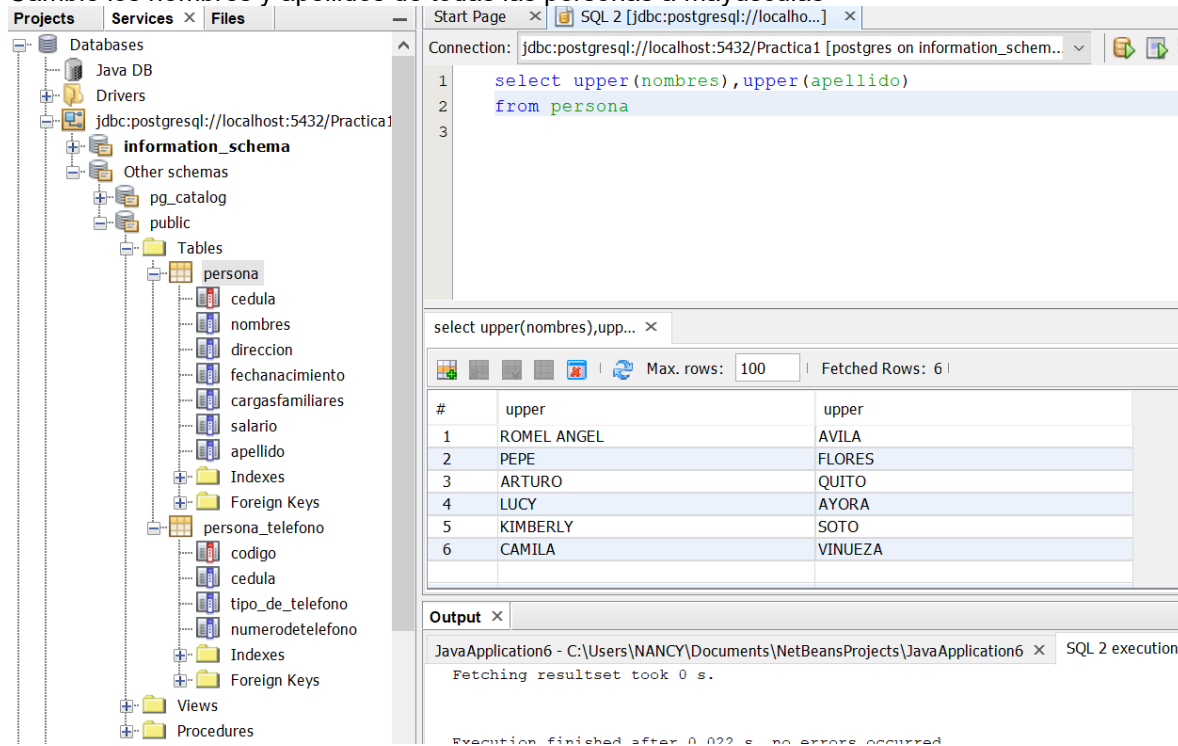
```
UPDATE personas
set salario=1000
where salario<1000
```

The results pane shows the following data:

#	id	nombres	direccion	fechanacimiento	cargasfamiliares	salario
1	44110	Romel Angel	Ricaurte	2001-12-20	1	1500.00
2	54321	Pepe	Trigales	2001-01-17	3	2500.00
3	56789	Arturo	Totoracocha	2002-01-17	2	1700.00
4	43219	Lucy	Dubay	2001-02-17	5	1000.00
5	23456	Kimberly	Vergel	2003-03-17	2	1000.00
6	56789	Camila	Calle vieja	1995-01-17	2	1000.00

The output pane shows the execution status: "Executed successfully in 0.001 s." and "Fetching resultset took 0 s."

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



The screenshot shows the NetBeans IDE with a PostgreSQL database connection. The left sidebar displays the database structure, including the 'personas' table. The main window shows an SQL query being executed:

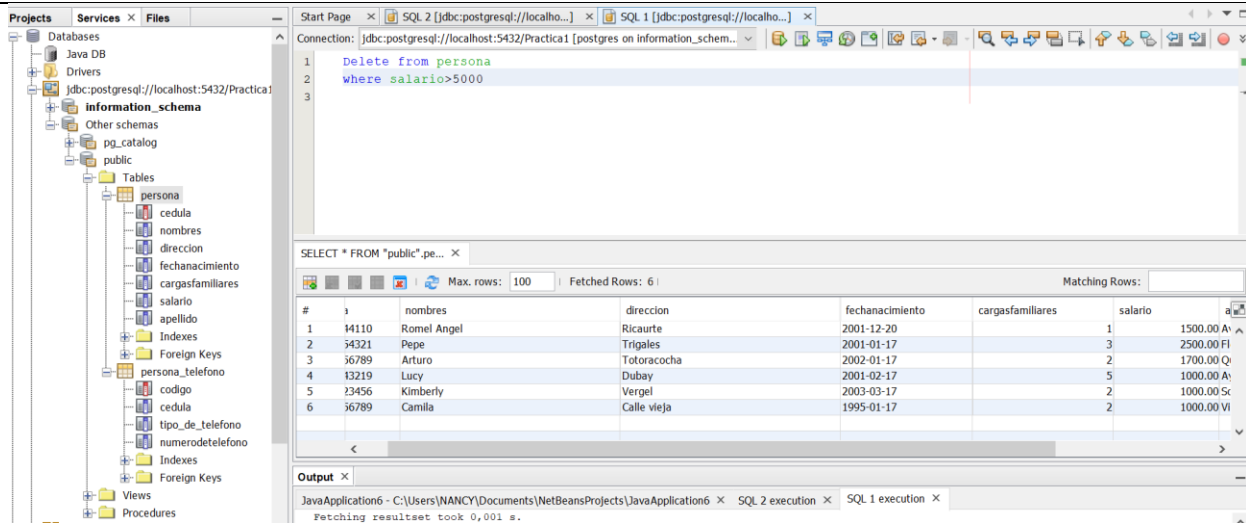
```
select upper(nombres), upper(apellido)
from personas
```

The results pane shows the following data:

#	upper	upper
1	ROMEL ANGEL	AVILA
2	PEPE	FLORES
3	ARTURO	QUITO
4	LUCY	AYORA
5	KIMBERLY	SOTO
6	CAMILA	VINUEZA

The output pane shows the execution status: "Execution finished after 0.022 s. no errors occurred"

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

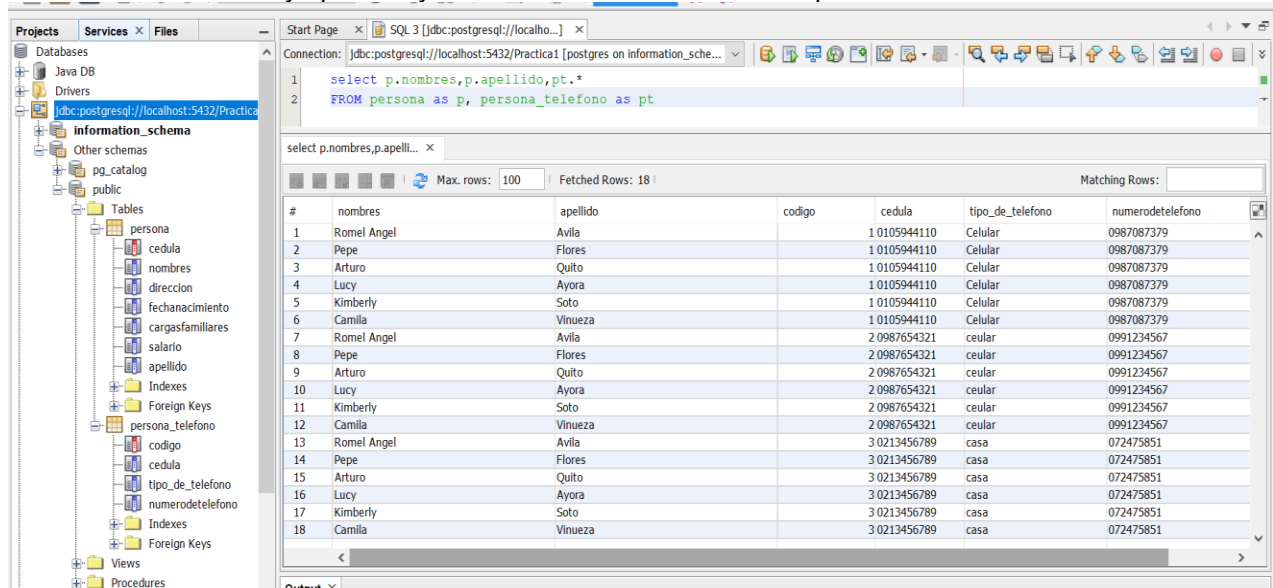


The screenshot shows the NetBeans IDE with a PostgreSQL database connection. The left sidebar displays the database structure, including the 'information_schema' and 'public' schemas. The main editor shows a SQL query: `Delete from persona where salario>5000`. The output window displays the results of a query: `SELECT * FROM "public".pe...`. The results table shows 6 rows of data.

#	id	nombres	direccion	fechanacimiento	cargasfamiliares	salario
1	14110	Romel Angel	Ricaurte	2001-12-20	1	1500.00
2	14321	Pepe	Trigales	2001-01-17	3	2500.00
3	16789	Arturo	Totoracocha	2002-01-17	2	1700.00
4	13219	Lucy	Dubay	2001-02-17	5	1000.00
5	13456	Kimberly	Vergel	2003-03-17	2	1000.00
6	16789	Camila	Calle vieja	1995-01-17	2	1000.00


2.4 FOREIGN KEY

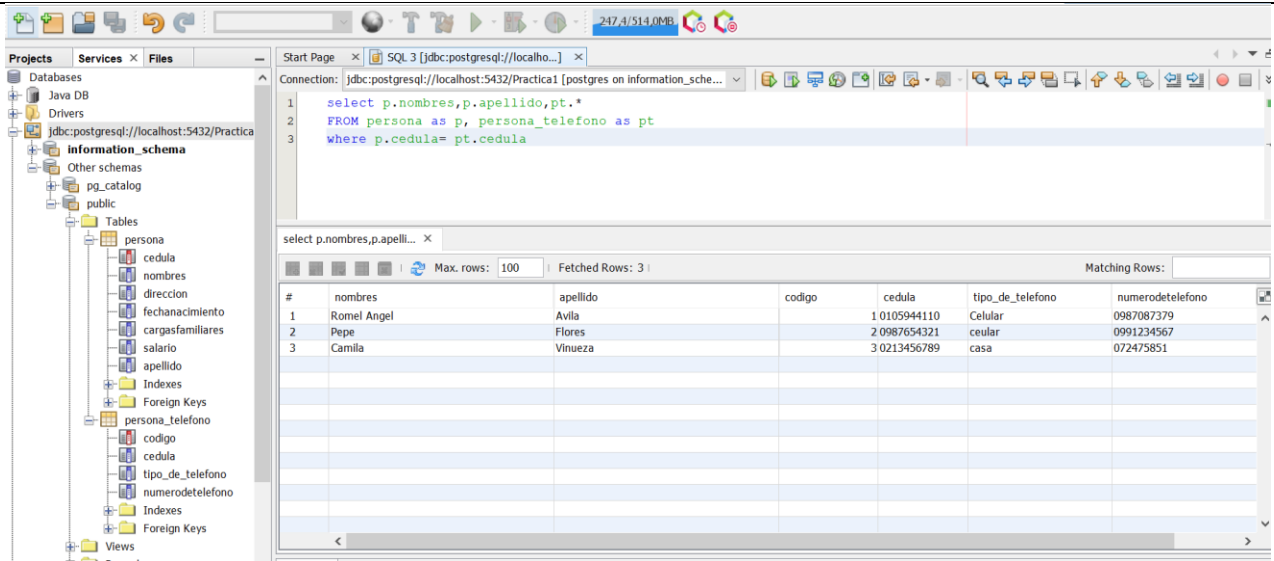
- Seleccione los nombre y apellidos y datos de teléfonos de todas las personas.



The screenshot shows the NetBeans IDE with a PostgreSQL database connection. The left sidebar displays the database structure, including the 'information_schema' and 'public' schemas. The main editor shows a SQL query: `select p.nombres,p.apellido,pt.* FROM persona as p, persona_telefono as pt`. The output window displays the results of the query: `select p.nombres,p.apelli...`. The results table shows 18 rows of data.

#	nombres	apellido	codigo	cedula	tipo_de_telefono	numerodetelefono
1	Romel Angel	Avila		1 0105944110	Celular	0987087379
2	Pepe	Flores		1 0105944110	Celular	0987087379
3	Arturo	Quito		1 0105944110	Celular	0987087379
4	Lucy	Ayora		1 0105944110	Celular	0987087379
5	Kimberly	Soto		1 0105944110	Celular	0987087379
6	Camila	Vinueza		1 0105944110	Celular	0987087379
7	Romel Angel	Avila		2 0987654321	ceular	0991234567
8	Pepe	Flores		2 0987654321	ceular	0991234567
9	Arturo	Quito		2 0987654321	ceular	0991234567
10	Lucy	Ayora		2 0987654321	ceular	0991234567
11	Kimberly	Soto		2 0987654321	ceular	0991234567
12	Camila	Vinueza		2 0987654321	ceular	0991234567
13	Romel Angel	Avila		3 0213456789	casa	072475851
14	Pepe	Flores		3 0213456789	casa	072475851
15	Arturo	Quito		3 0213456789	casa	072475851
16	Lucy	Ayora		3 0213456789	casa	072475851
17	Kimberly	Soto		3 0213456789	casa	072475851
18	Camila	Vinueza		3 0213456789	casa	072475851

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



The screenshot shows the SQL Studio interface. The left pane displays the database schema with tables 'persona' and 'persona_telefono'. The main editor contains the following SQL query:

```

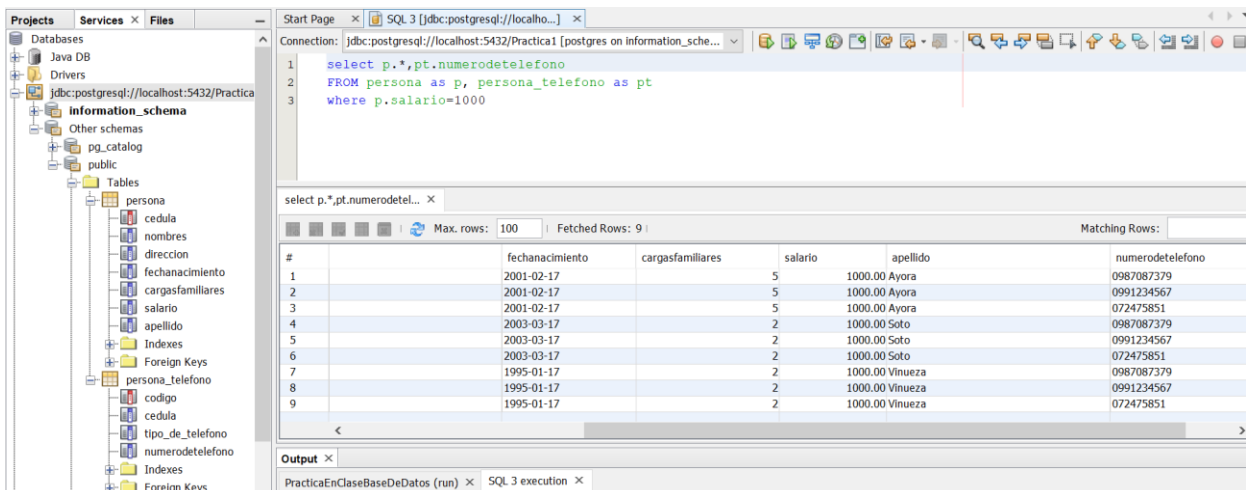
1 select p.nombres,p.apellido,pt.*
2 FROM persona as p, persona_telefono as pt
3 where p.cedula= pt.cedula

```

The results pane shows 3 matching rows:

#	nombres	apellido	codigo	cedula	tipo_de_telefono	numerodetelefono
1	Romel Angel	Avila		1 0105944110	Celular	0987087379
2	Pepe	Flores		2 0987654321	ceular	0991234567
3	Camila	Vinueza		3 0213456789	casa	072475851

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



The screenshot shows the SQL Studio interface. The left pane displays the database schema. The main editor contains the following SQL query:

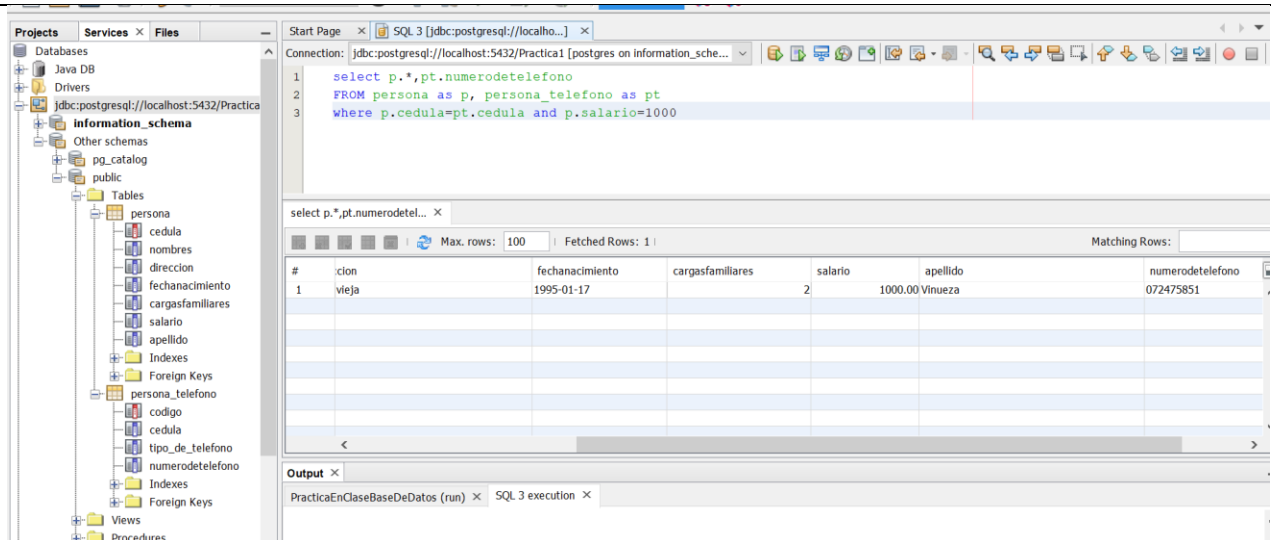
```

1 select p.*,pt.numerodetelefono
2 FROM persona as p, persona_telefono as pt
3 where p.salario=1000

```

The results pane shows 9 matching rows:

#	fechanacimiento	cargasfamiliares	salario	apellido	numerodetelefono
1	2001-02-17	5	1000.00	Ayora	0987087379
2	2001-02-17	5	1000.00	Ayora	0991234567
3	2001-02-17	5	1000.00	Ayora	072475851
4	2003-03-17	2	1000.00	Soto	0987087379
5	2003-03-17	2	1000.00	Soto	0991234567
6	2003-03-17	2	1000.00	Soto	072475851
7	1995-01-17	2	1000.00	Vinueza	0987087379
8	1995-01-17	2	1000.00	Vinueza	0991234567
9	1995-01-17	2	1000.00	Vinueza	072475851



The screenshot shows the SQL Studio interface. On the left, the 'Databases' tree is expanded to 'information_schema' > 'Tables'. The main editor shows the following SQL query:

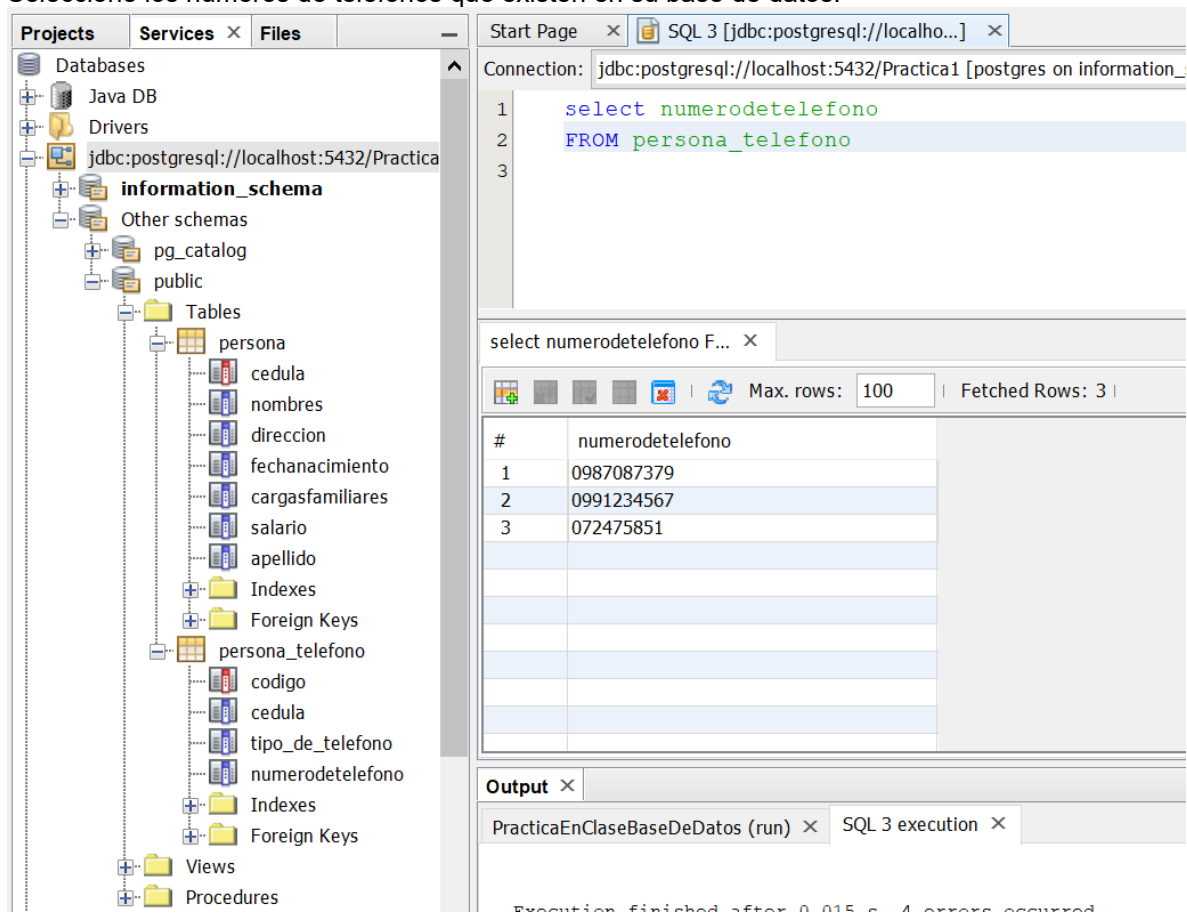
```
1 select p.*,pt.numerodetelfono
2 FROM persona as p, persona_telefono as pt
3 where p.cedula=pt.cedula and p.salario=1000
```

The results pane shows one row of data:

#	cion	fechanacimiento	cargasfamiliares	salario	apellido	numerodetelfono
1	vieja	1995-01-17	2	1000.00	Vinueza	072475851

The 'Output' pane at the bottom shows the execution of the query.

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



The screenshot shows the SQL Studio interface. On the left, the 'Databases' tree is expanded to 'information_schema' > 'Tables'. The main editor shows the following SQL query:


```
1 select numerodetelfono
2 FROM persona_telefono
3
```

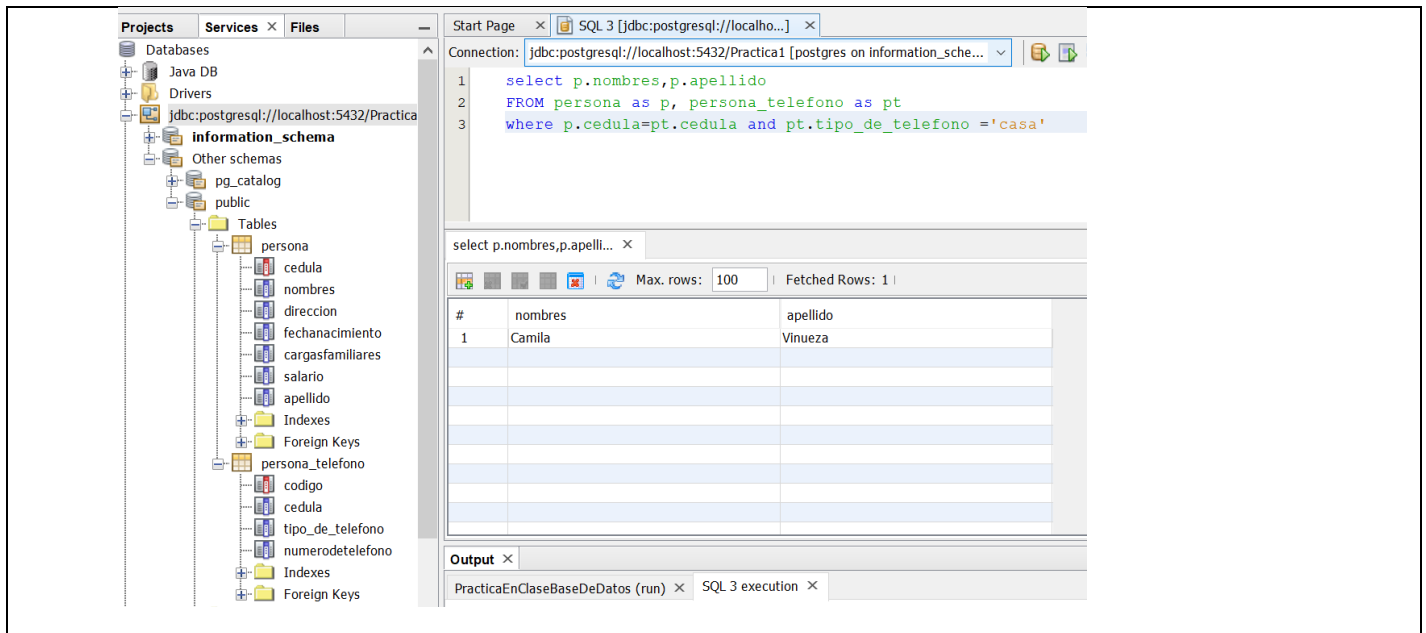
The results pane shows three rows of data:

#	numerodetelfono
1	0987087379
2	0991234567
3	072475851

The 'Output' pane at the bottom shows the execution of the query.

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

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



The screenshot shows a PostgreSQL IDE interface. On the left, a tree view displays the database structure, including tables like 'persona' and 'persona_telefono'. The main window shows a SQL query:

```

1 select p.nombres,p.apellido
2 FROM persona as p, persona_telefono as pt
3 where p.cedula=pt.cedula and pt.tipo_de_telefono ='casa'

```

Below the query, the results are displayed in a table:

#	nombres	apellido
1	Camila	Vinueza

The 'Output' pane at the bottom shows the execution status: 'PracticaEnClaseBaseDeDatos (run)' and 'SQL 3 execution'.

RESULTADO(S) OBTENIDO(S):

Como resultados podemos encontrar el manejo de algunos comandos para poder manejar las bases de datos tanto con los datos guardados como con las columnas.

CONCLUSIONES:

En conclusión, Podemos decir que aprendimos un poco mas sobre el manejo de base de datos y ya tenemos un conocimiento introductorio al mismo gracias a esta práctica.

RECOMENDACIONES:

No hay recommendation

Estudiantes: Romel Ávila

Firma:

