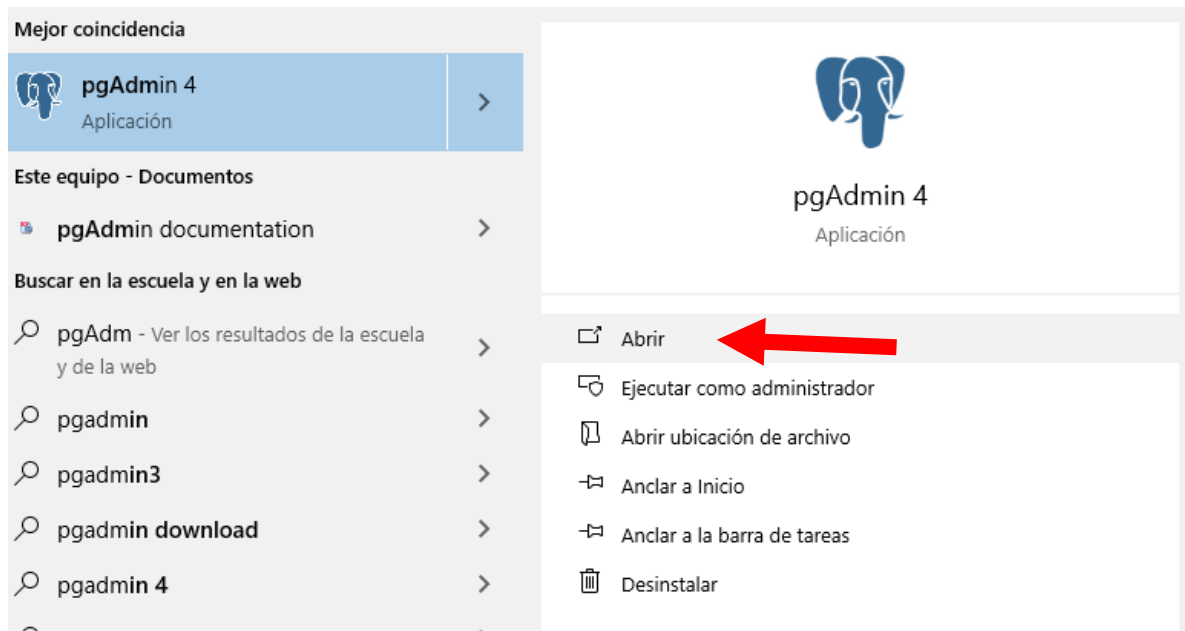
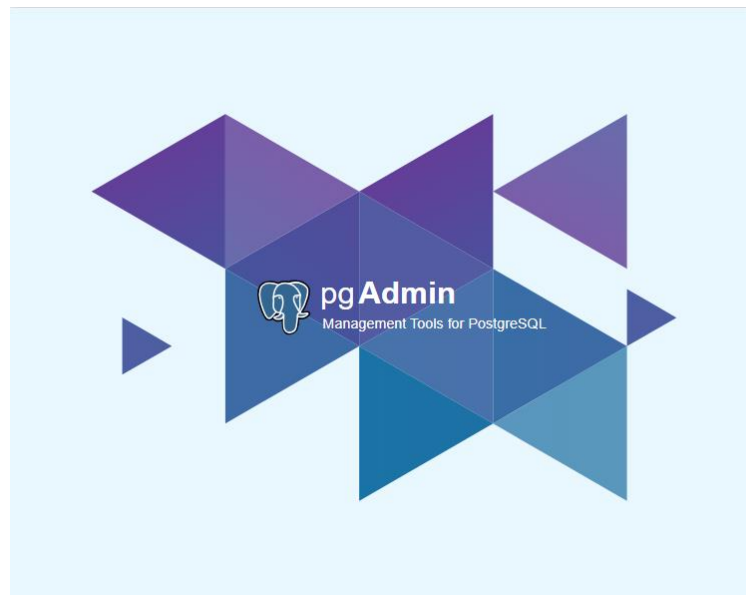


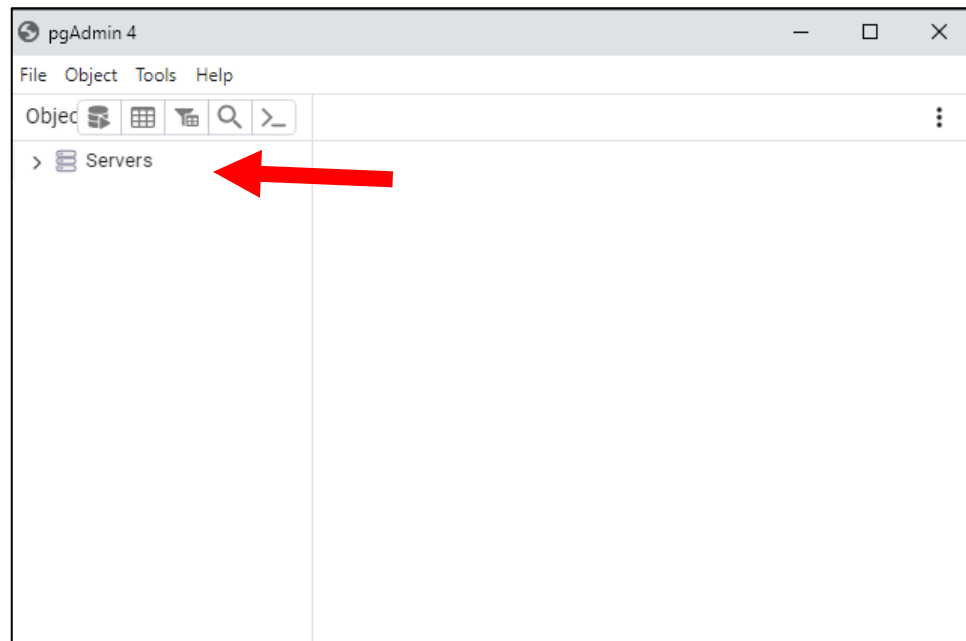
- Buscamos pgAdmin4 en nuestra barra de búsquedas y abrimos pgAdmin4



- Esperamos que se abra la aplicación de pgAdmin4

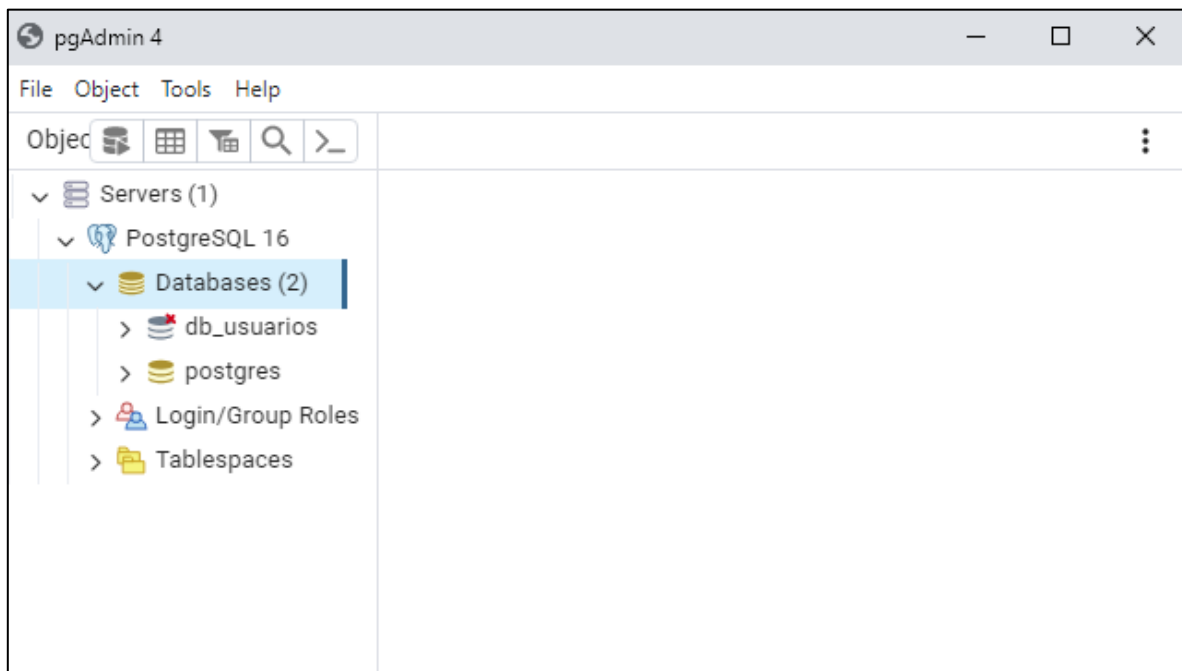


- Ya que abra el pgAdmin4 nos aparecerá lo que se muestra en la siguiente imagen, lo que seguirá será presionar en el apartado donde aparece Servers

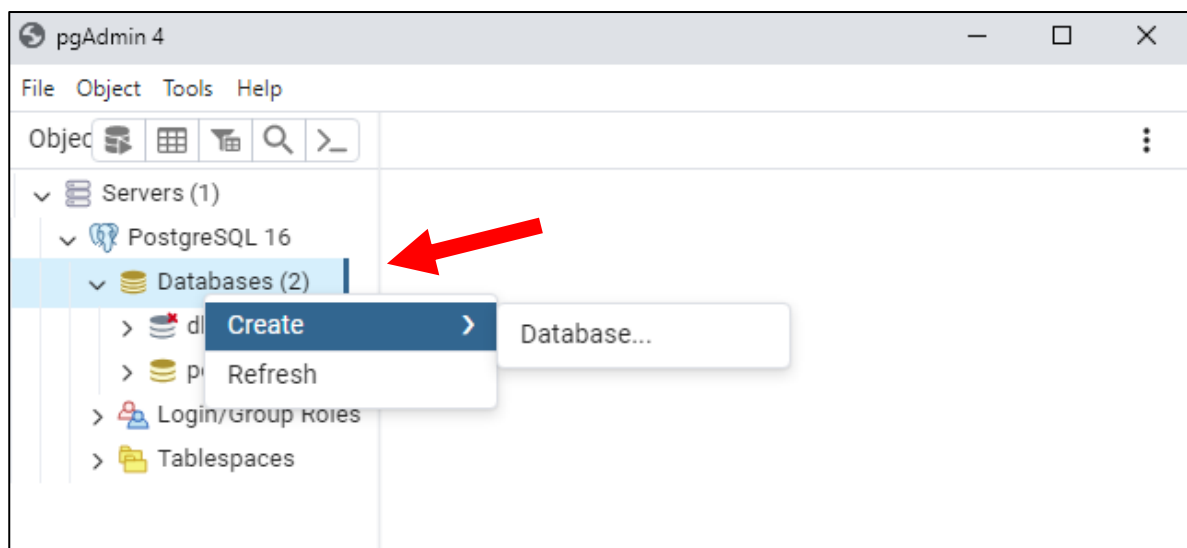


- Iniciamos sesión con el usuario que tenemos por defecto en postgres al instalarlo (en mi caso es ADMIN).

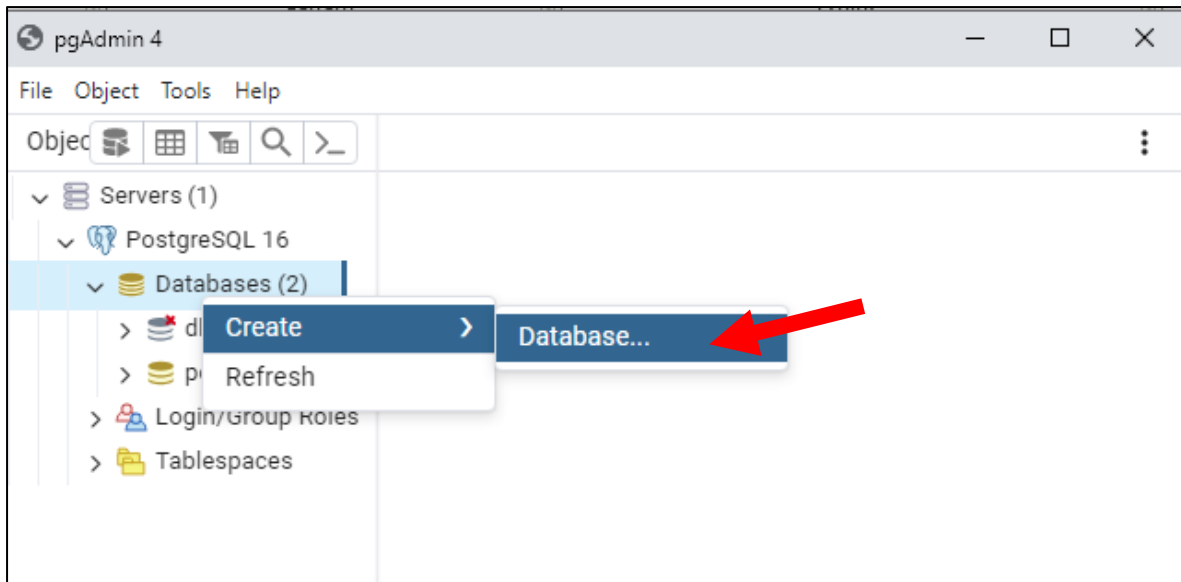
The image shows a 'Connect to Server' dialog box. The title bar says 'Connect to Server' with a close button (X). The main text says 'Please enter the password for the user 'postgres' to connect the server - "PostgreSQL 16"'. Below this is a password input field with five dots. Under the input field is a checkbox labeled 'Save Password'. At the bottom right are two buttons: 'Cancel' with a close icon (X) and 'OK' with a checkmark icon (✓).



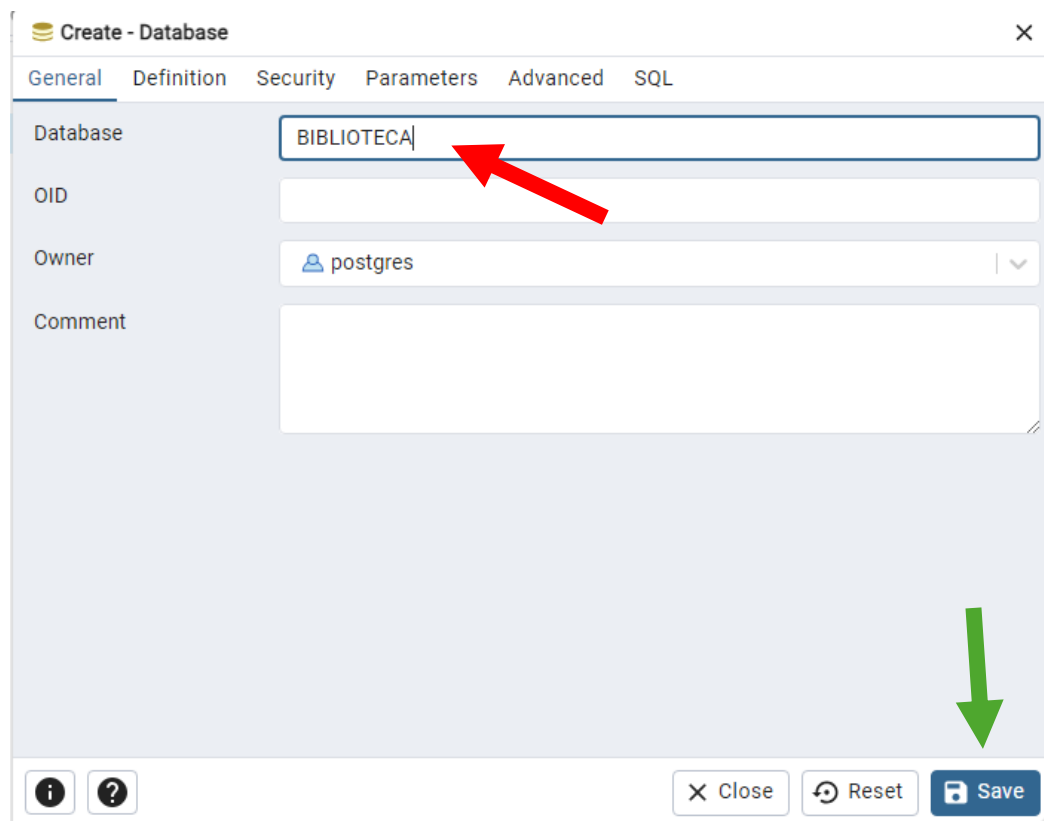
- Para la creación de la tabla de base de datos le damos clic derecho donde dice Databases



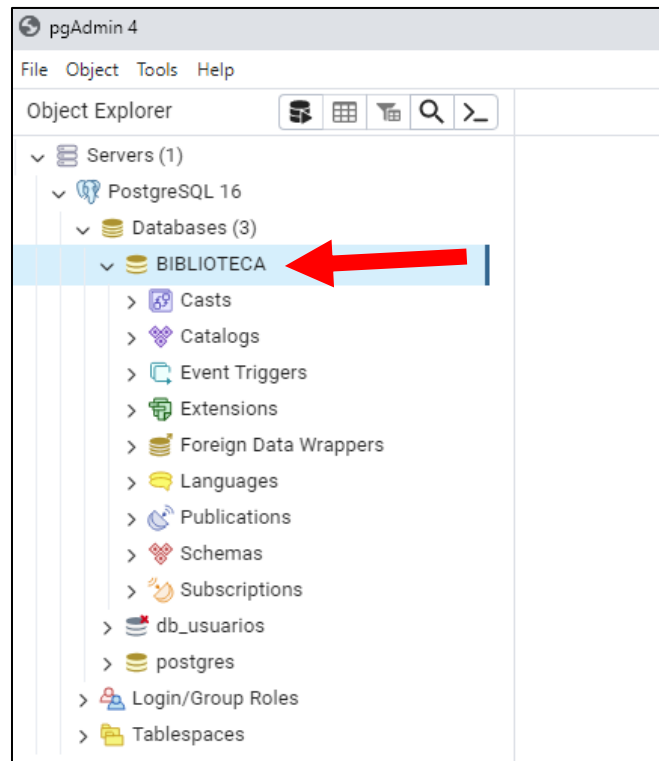
- Ahora presionaremos en Database



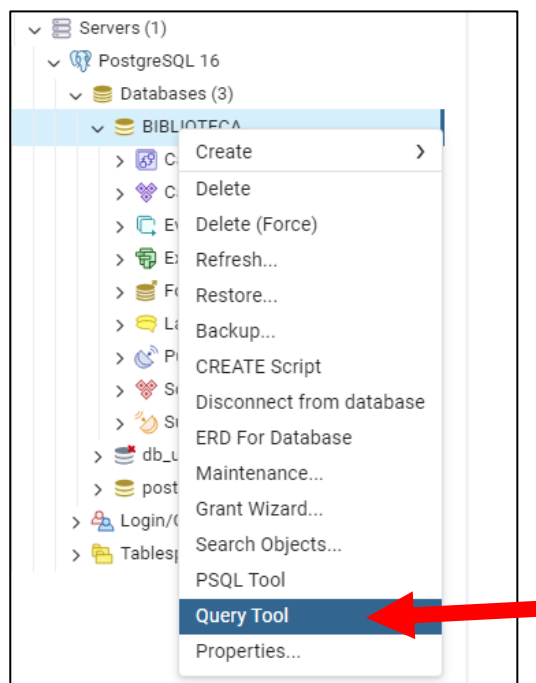
- Nos abrirá una nueva pestaña en esta agregaremos un nombre a nuestra base de datos (en este caso le pondré BIBLIOTECA) y le damos en Save



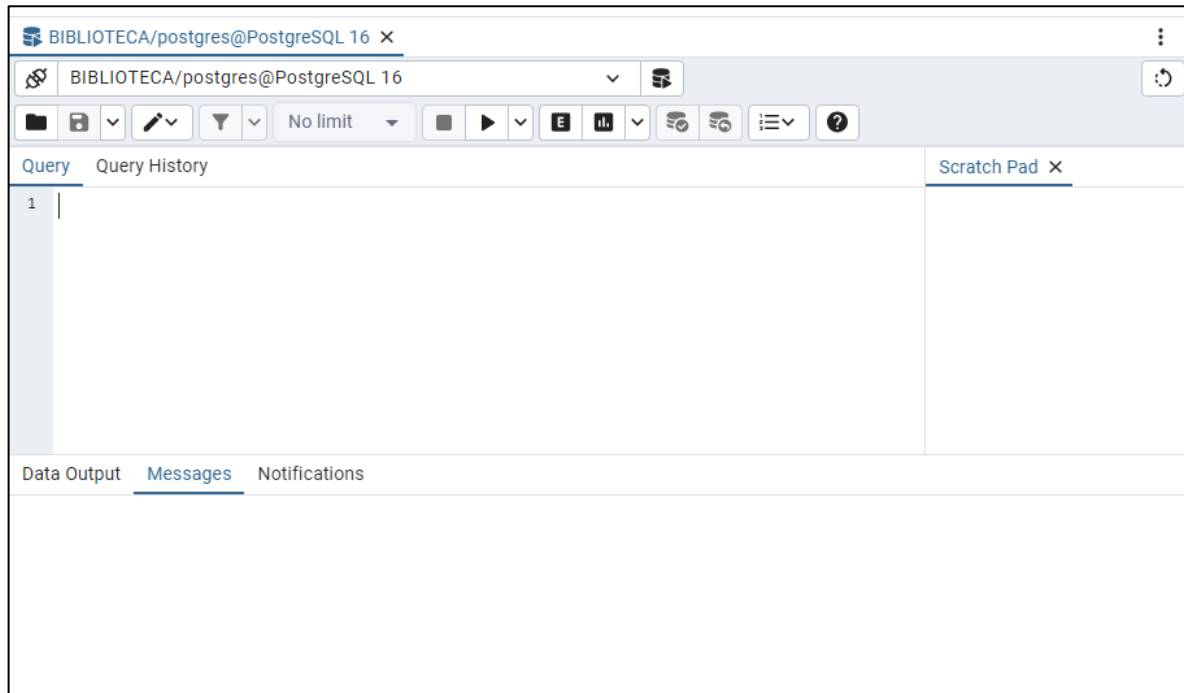
- De lado izquierdo nos aparecerá nuestra base de datos que acabamos de crear



- Para la creación de las tablas dentro de la base de datos en el nombre de la base de datos presionaremos clic derecho y presionaremos la opción de Query Tool



- Del lado derecho nos aparecerá un apartado donde escribiremos las sentencias SQL para la creación de las tablas



- Ahora bien, empezaremos creando la tabla " usuario " mediante la siguiente sentencia SQL:

```
CREATE TABLE usuario (
```

```
    ID_usuario INTEGER GENERATED ALWAYS AS IDENTITY PRIMARY  
    KEY,
```

```
    nombre TEXT NOT NULL,
```

```
    telefono INTEGER NOT NULL,
```

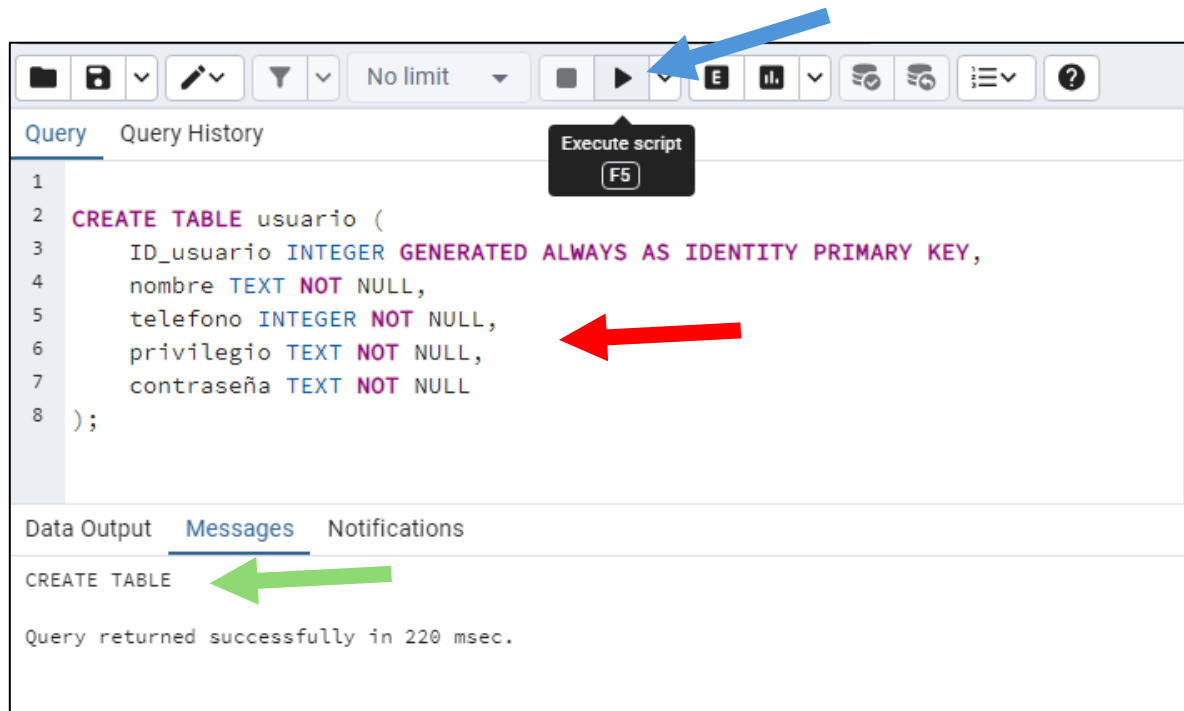
```
    privilegio TEXT NOT NULL,
```

```
    contraseña TEXT NOT NULL
```

```
);
```

tenemos en esta los atributos ID_usuario, nombre, teléfono, privilegio, contraseña, siendo el atributo ID_usuario la llave primaria siendo always as identiti para que se autoincremente de 1 en 1, cada uno tiene su tipo de dato integer o text y todos son not null

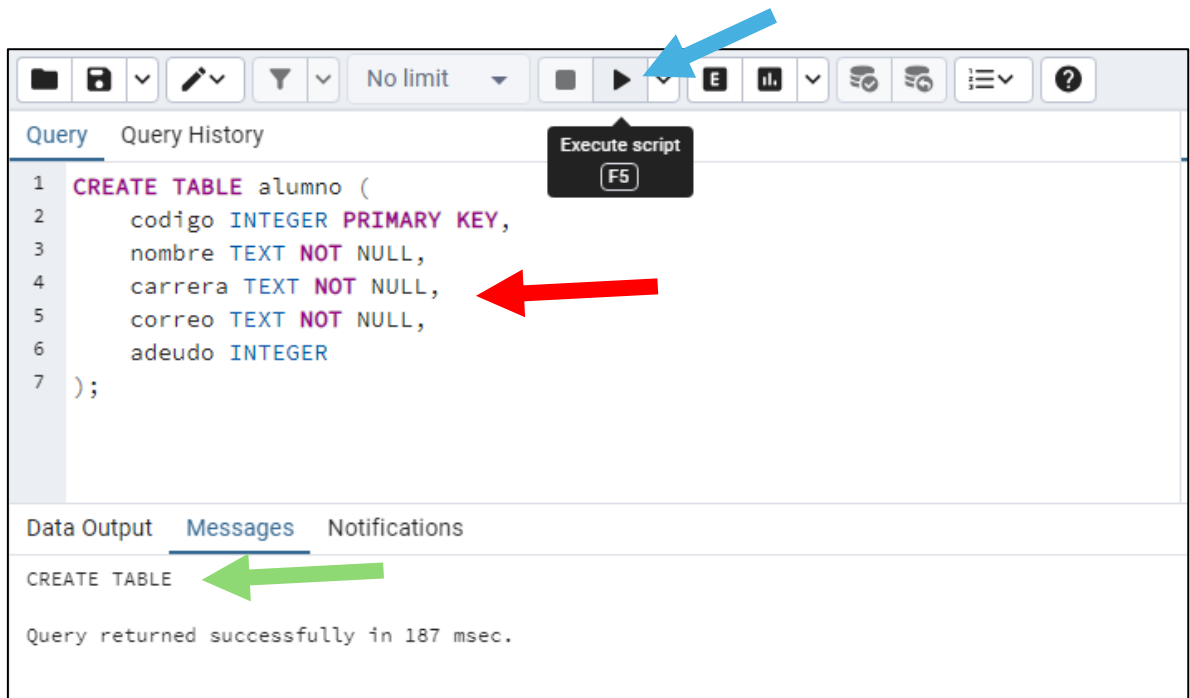
despues de escribir la sentencia del lado derecho, presionaremos la flechita superior para que se ejecute todo el scrip y confirmaremos si se creo correctamente observando lo que aparece abajo en messages.



- Ahora crearemos la tabla de " alumno " mediante la siguiente sentencia SQL:

```
CREATE TABLE alumno (  
    codigo INTEGER PRIMARY KEY,  
    nombre TEXT NOT NULL,  
    carrera TEXT NOT NULL,  
    correo TEXT NOT NULL,  
    adeudo INTEGER  
);
```

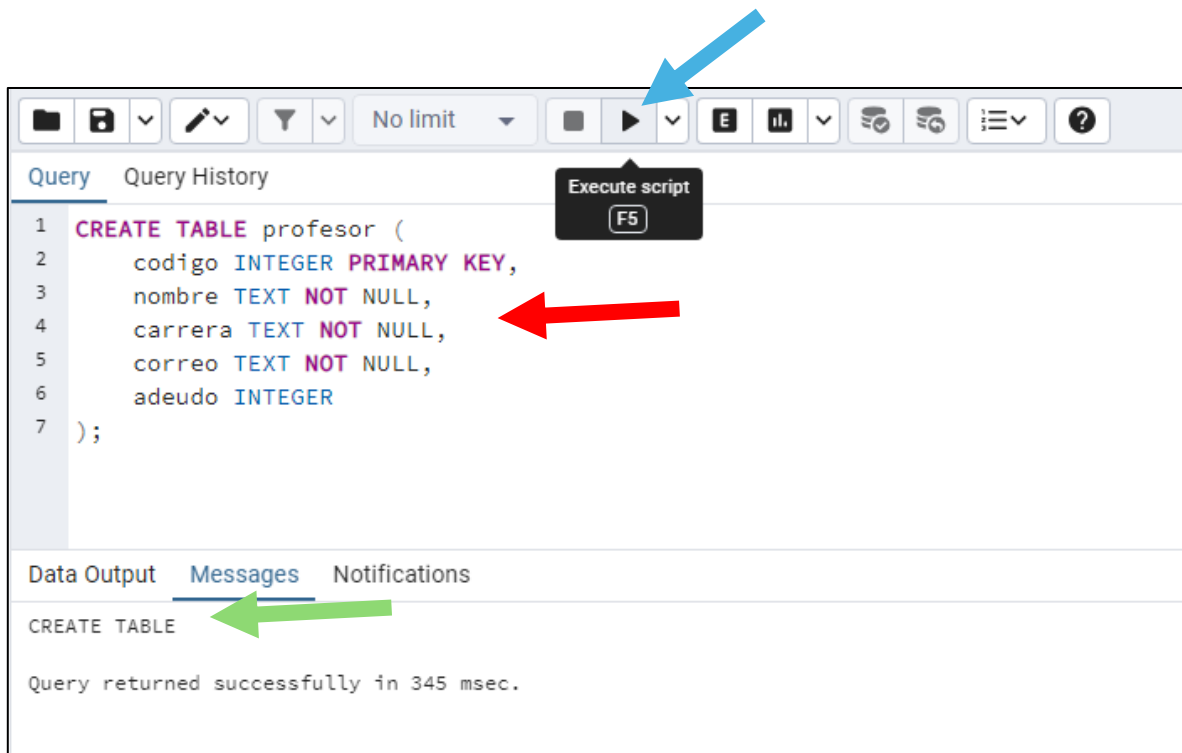
tenemos en esta los atributos codigo, nombre, carrera, correo, adeudo, siendo el atributo código la llave primaria, cada uno tiene su tipo de dato integer o text y todos son not null menos adeudo que puede ser null.



- Ahora crearemos la tabla de " profesor " mediante la siguiente sentencia SQL:

```
-CREATE TABLE profesor (  
    codigo INTEGER PRIMARY KEY,  
    nombre TEXT NOT NULL,  
    carrera TEXT NOT NULL,  
    correo TEXT NOT NULL,  
    adeudo INTEGER  
);
```

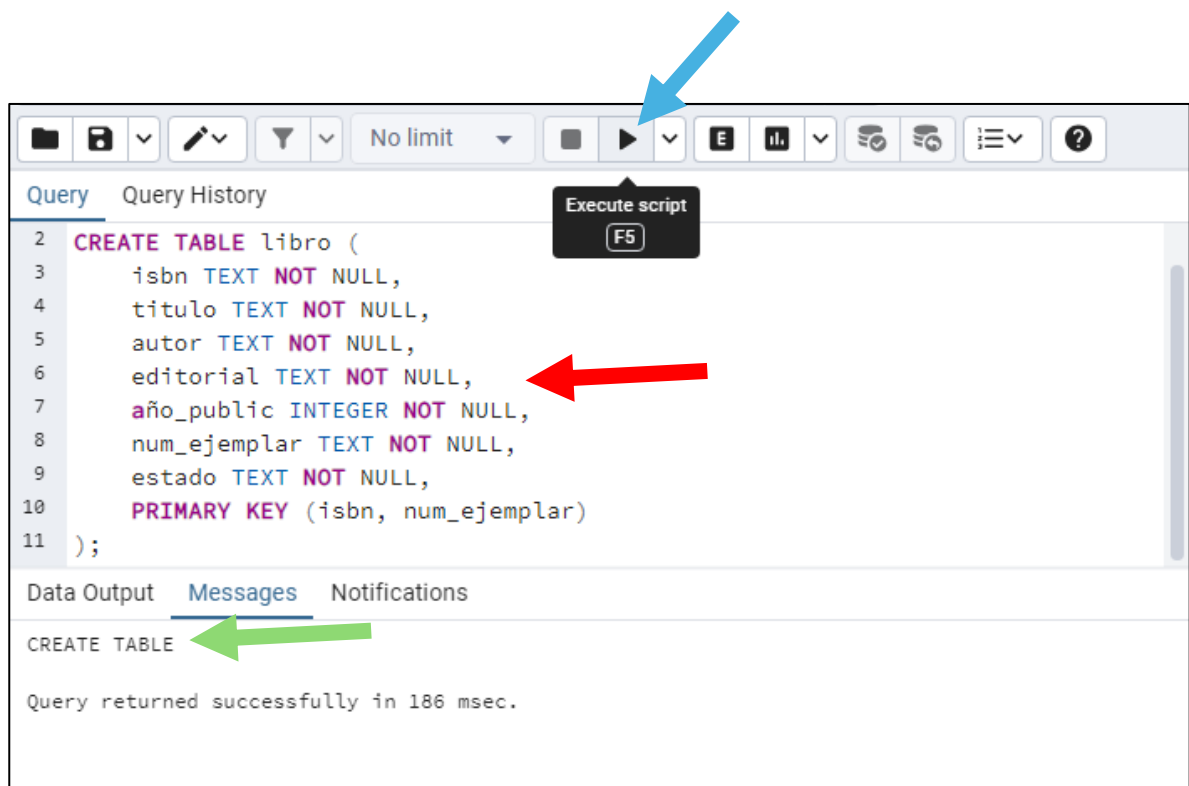
tenemos en esta los atributos codigo, nombre, carrera, correo, adeudo, siendo el atributo código la llave primaria, cada uno tiene su tipo de dato integer o text y todos son not null menos adeudo que puede ser null.



- Ahora crearemos la tabla de " libro " mediante la siguiente sentencia SQL:

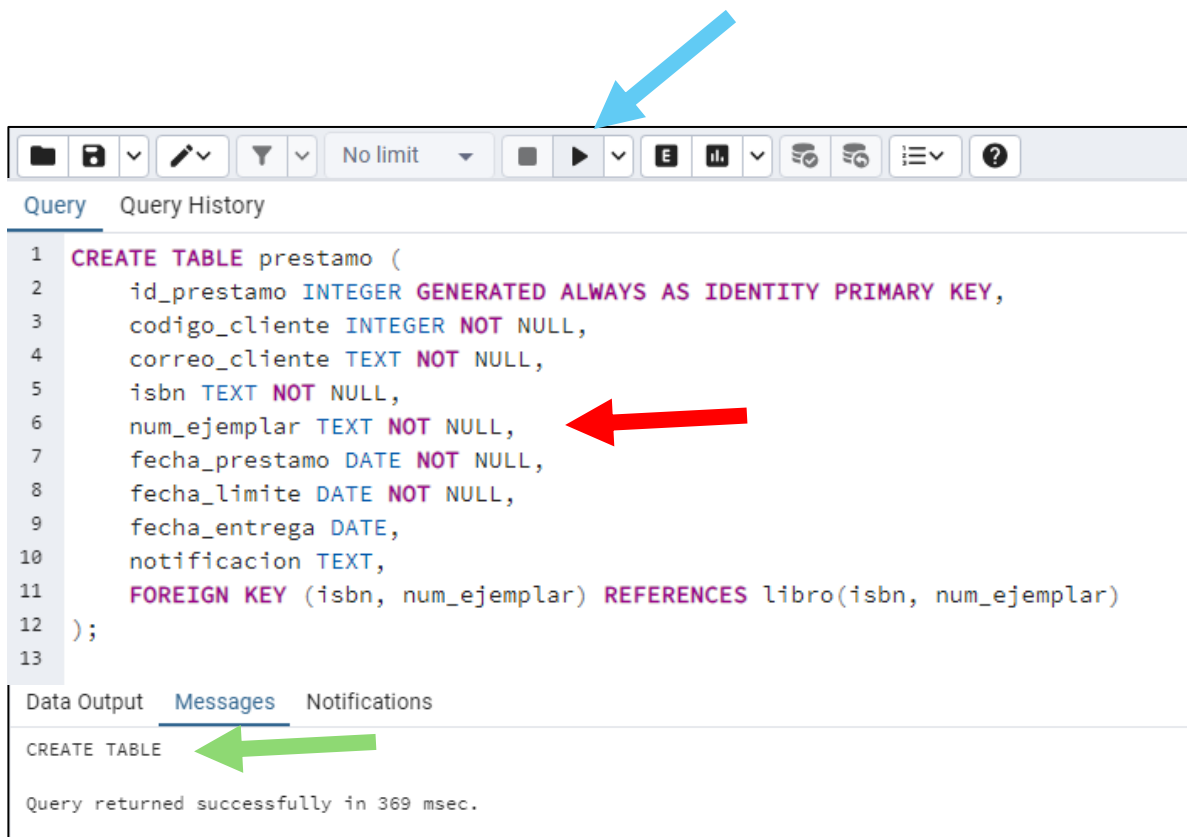
```
CREATE TABLE libro (  
    isbn TEXT NOT NULL,  
    titulo TEXT NOT NULL,  
    autor TEXT NOT NULL,  
    editorial TEXT NOT NULL,  
    año_public INTEGER NOT NULL,  
    num_ejemplar TEXT NOT NULL,  
    estado TEXT NOT NULL,  
    PRIMARY KEY (isbn, num_ejemplar)  
);
```

tenemos en esta los atributos isbn, titulo, autor, editorial, año_public, num_ejemplar, estado, siendo los atributos isbn y num_ejemplar la llave primaria, cada uno tiene su tipo de dato integer o text y todos son not null



- Ahora crearemos la tabla de " prestamo " mediante la siguiente sentencia SQL:

```
CREATE TABLE prestamo (  
id_prestamo INTEGER GENERATED ALWAYS AS IDENTITY PRIMARY KEY,  
codigo_cliente INTEGER NOT NULL,  
correo_cliente TEXT NOT NULL,  
isbn TEXT NOT NULL,  
num_ejemplar TEXT NOT NULL,  
fecha_prestamo DATE NOT NULL,  
fecha_limite DATE NOT NULL,  
fecha_entrega DATE,  
    notificacion TEXT,  
FOREIGN KEY (isbn, num_ejemplar) REFERENCES libro(isbn, num_ejemplar)  
);
```



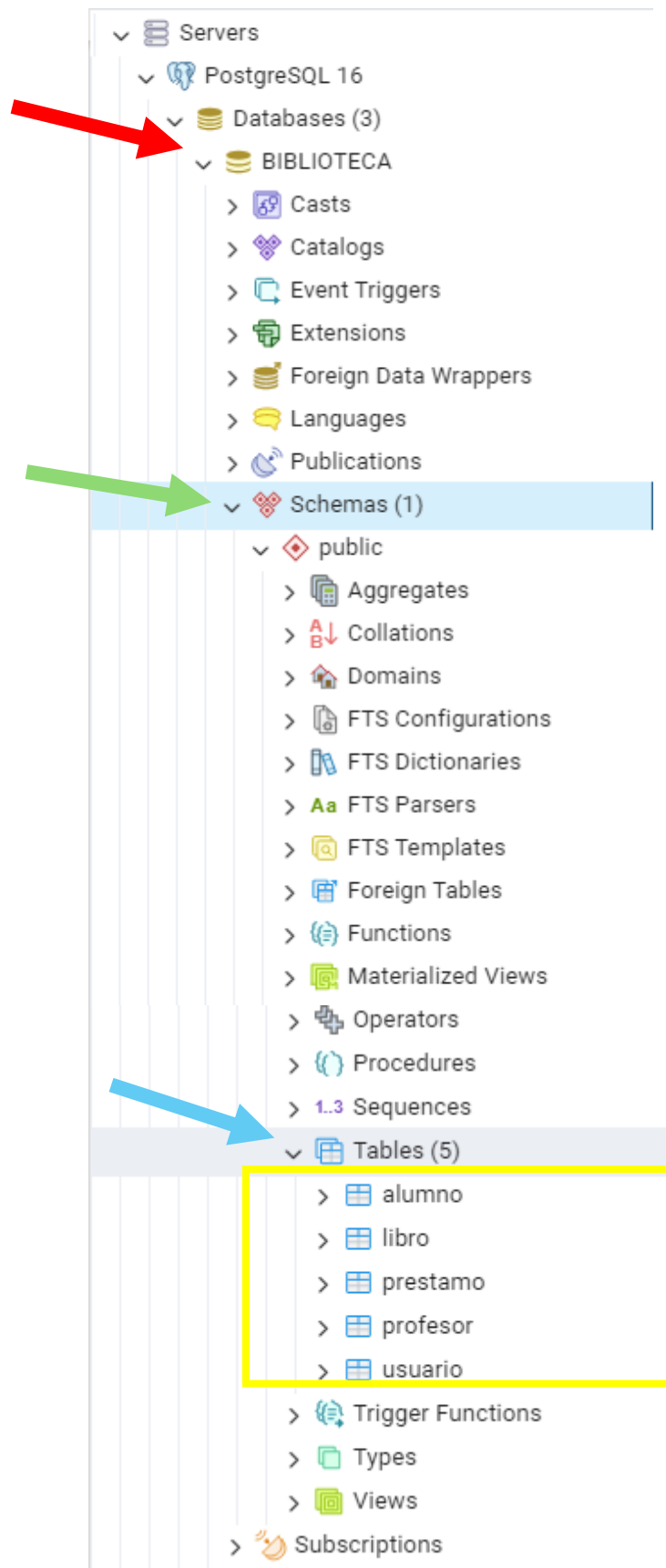
The screenshot shows a database query editor interface. At the top, there is a toolbar with various icons. A blue arrow points to the 'Execute' button (a play icon). Below the toolbar, the 'Query' tab is active, displaying a SQL statement to create a table named 'prestamo'. The statement includes columns for 'id_prestamo' (INTEGER GENERATED ALWAYS AS IDENTITY PRIMARY KEY), 'codigo_cliente' (INTEGER NOT NULL), 'correo_cliente' (TEXT NOT NULL), 'isbn' (TEXT NOT NULL), 'num_ejemplar' (TEXT NOT NULL), 'fecha_prestamo' (DATE NOT NULL), 'fecha_limite' (DATE NOT NULL), 'fecha_entrega' (DATE), and 'notificacion' (TEXT). A foreign key constraint is defined on 'isbn' and 'num_ejemplar' referencing the 'libro' table. A red arrow points to the 'num_ejemplar' column. Below the query, the 'Messages' tab is active, showing the message 'CREATE TABLE' and 'Query returned successfully in 369 msec.' A green arrow points to the 'CREATE TABLE' message.

```
1 CREATE TABLE prestamo (  
2     id_prestamo INTEGER GENERATED ALWAYS AS IDENTITY PRIMARY KEY,  
3     codigo_cliente INTEGER NOT NULL,  
4     correo_cliente TEXT NOT NULL,  
5     isbn TEXT NOT NULL,  
6     num_ejemplar TEXT NOT NULL,  
7     fecha_prestamo DATE NOT NULL,  
8     fecha_limite DATE NOT NULL,  
9     fecha_entrega DATE,  
10    notificacion TEXT,  
11    FOREIGN KEY (isbn, num_ejemplar) REFERENCES libro(isbn, num_ejemplar)  
12 );  
13
```

CREATE TABLE

Query returned successfully in 369 msec.

- Para comprobar que tenemos las tablas en la base de datos del lado izquierdo donde tenemos la base de datos existen diferentes apartados, lo que vamos a hacer será ir a Schemas presionamos esta.
- nos vamos a el apartado Tables y de igual manera presionamos este apartado.
- en este apartado deberá de aparecernos todas las tablas que creamos



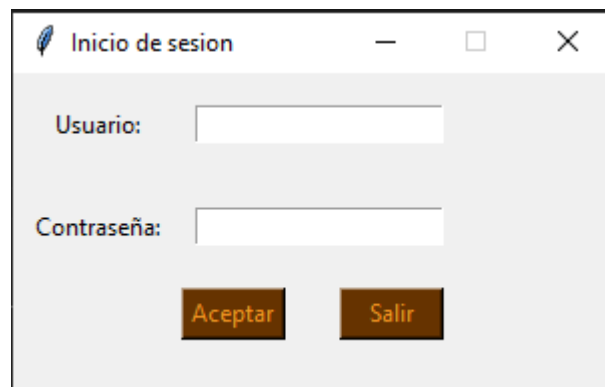
- Ahora anteriormente se creó una interfaz gráfica utilizando el lenguaje de programación Python y el entorno de programación fue por medio de Visual Studio Code.
- En los siguientes apartados mostraremos como conectamos la base de datos situada en pgAdmin a la interfaz creada en Python, importamos la librería psycopg2 y enseguida escribimos el código para la conexión que fue escribir el usuario en este caso postgres el password el host con su puerto y el nombre de la base de datos que en nuestro ejemplo fue BIBLIOTECA.

```

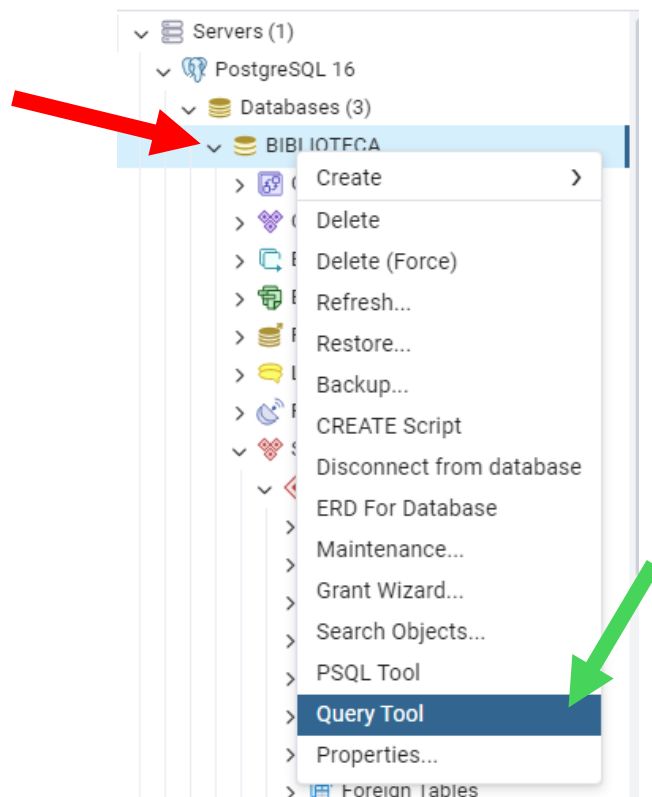
15 import psycopg2    #Extension de python para conectar con postgresql
16 from tkinter import messagebox
17 from envioCorreo import enviar_mensaje_notifiacion, enviar_mensaje_multa
18 import datetime
19 from datetime import timedelta
20
21 def establecer_conexion():
22     try:
23         conexion = psycopg2.connect(#Usamos el metodo de psycopg2 y le pasamos los parametros
necesarios
24         user="postgres",          #Primer parametro: Nuestro usuario de postgresql
25         password="ADMIN",         #Segundo parametro: Contraseña de nuestro server
26         host="127.0.0.1",         #Tercer parametro: Host de nuestro server
27         port="5432",             #Cuarto parametro: Puerto de nuestro server
28         database="BIBLIOTECA"     #Quinto parametro: Nombre de la base de datos
29     )
30     return conexion
31 except (Exception, psycopg2.Error):    #Exception por si la conexion falla
32     messagebox.showerror("Error de conexion", "No se pudo conectar a la base de datos.")
#Notifica error

```

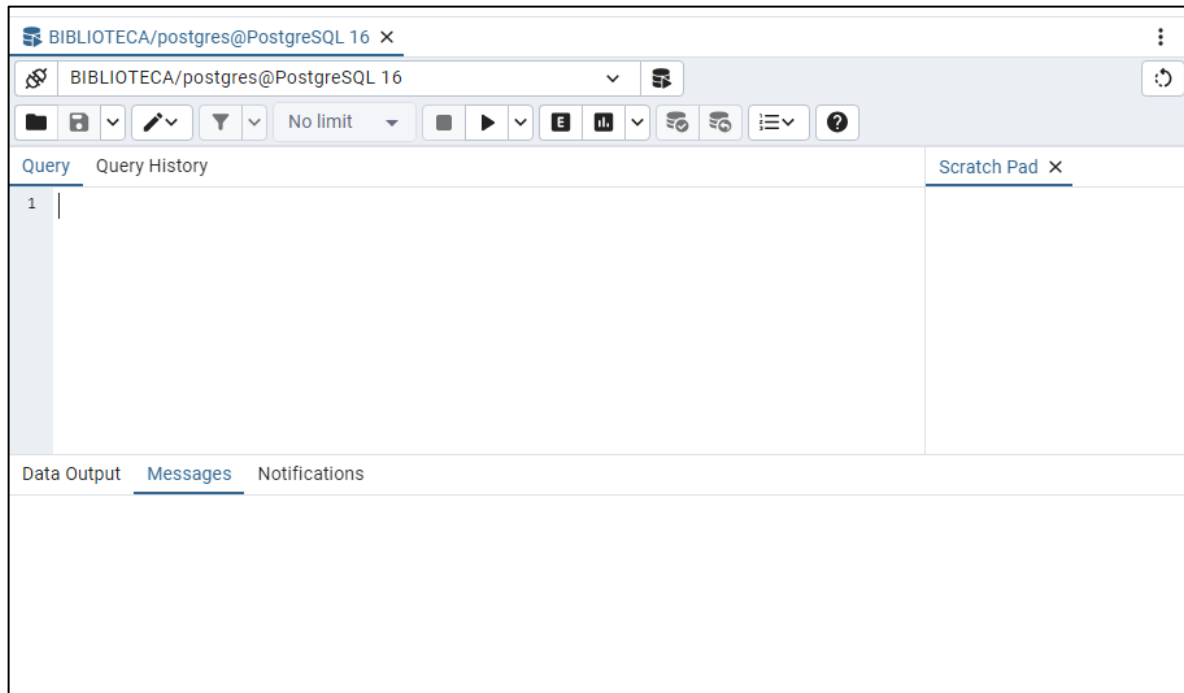
- En el siguiente apartado mostraremos como se ve la interfaz al correr el código que es la primera pestaña de validar las credenciales



- Para poder iniciar sesión en la interfaz ocupamos en la base de datos agregar a los usuarios en este caso agregaremos los usuarios de ADMIN con contraseña ADMIN y EMPLEADO con contraseña EMPLEADO, para esto en PgAdmin4 daremos clic derecho en el nombre de la base de datos " BIBLIOTECA" y daremos clic en Query Tool.



- Del lado derecho nos aparecerá el apartado donde podremos escribir sentencias SQL y en este caso escribiremos una sentencia para la creación de nuestros usuarios



- Comenzaremos con la creación de nuestro usuario ADMIN:
INSERT INTO usuario (nombre, telefono, privilegio, contraseña)
VALUES ('ADMIN', 331320, 'Administrador', 'ADMIN');
esta seria la sentencia que escribiríamos para la creación de este usuario.

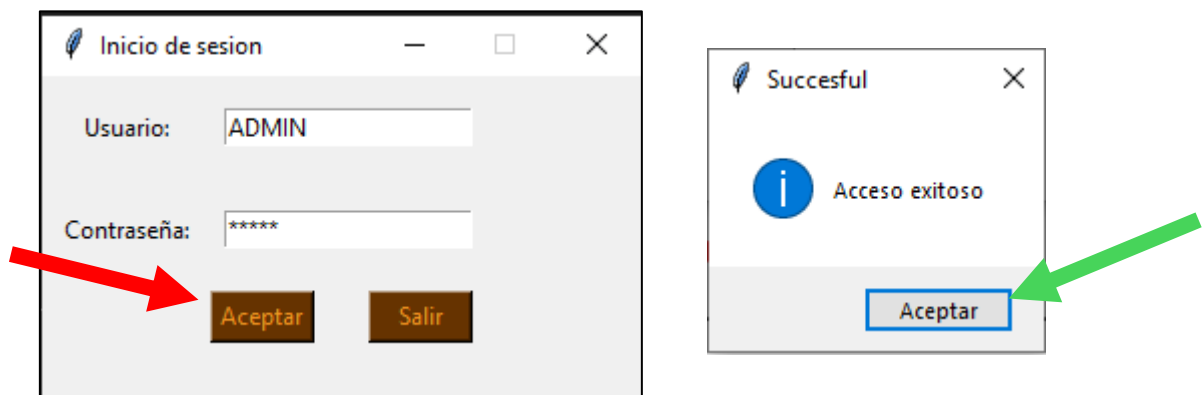


- Ahora para la creación del usuario EMPLEADO será de la siguiente manera:
INSERT INTO usuario (nombre, telefono, privilegio, contraseña)
VALUES ('EMPLEADO', 331325, 'Empleado', 'EMPLEADO');

esta sería la sentencia que escribiríamos para la creación de este usuario.

Query	Query History
1	INSERT INTO usuario (nombre, telefono, privilegio, contraseña)
2	VALUES ('EMPLEADO', 331325, 'Empleado', 'EMPLEADO');
3	

- Después de darle en correr y ver que se crearon correctamente lo que realizaremos es ir a nuestro visual studio code que es donde tenemos nuestro código y volver a correrlo, ahora en la interfaz pondremos el usuario y contraseña del ADMIN para entrar por primera vez, después de poner los datos presionaremos el botón de aceptar para que nos lance una alerta de “Acceso exitoso” de igual manera presionaremos Aceptar.



- Ahora bien, nos abrirá una nueva pestaña siendo esta la pestaña principal que puede ver el ADMIN siendo la siguiente:



- Tenemos diferentes opciones en la interfaz como podremos ver tiene en la parte superior un apartado donde viene Alumnos, Profesores, Libros, Prestamos y Salir, en este momento insertaremos los Alumnos de nuestra sección D03 de la clase de Base de Datos
- Para esto presionaremos en el apartado donde dice Alumnos de la parte superior con esto se nos desplegaran 2 opciones mas siendo estas Altas y consultas



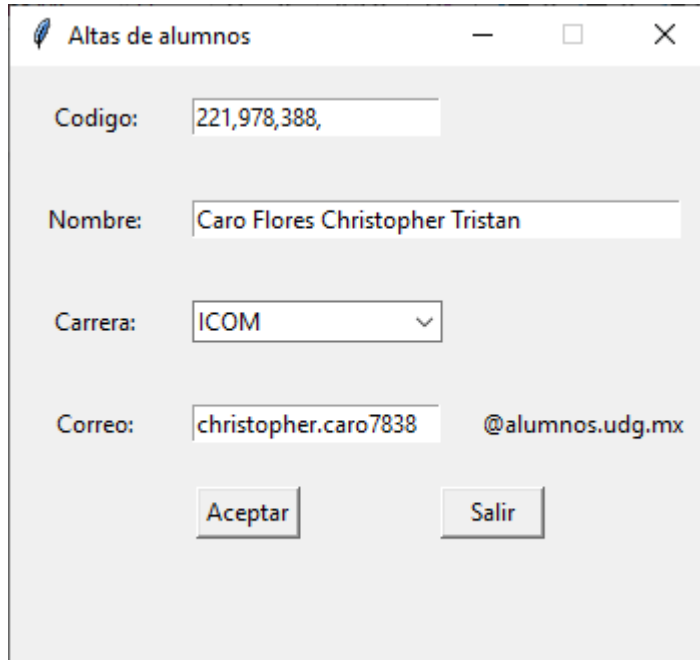
- Para poder registrar a los Alumnos debemos de presionar el apartado de Altas



- Nos abrirá una nueva pestaña con un formulario para realizar el registro de los alumnos.

The image shows a web form titled 'Altas de alumnos' in a window. The form contains the following fields: 'Codigo:' with a text input field; 'Nombre:' with a text input field; 'Carrera:' with a dropdown menu; and 'Correo:' with a text input field followed by the text '@alumnos.udg.mx'. At the bottom of the form, there are two buttons: 'Aceptar' and 'Salir'.

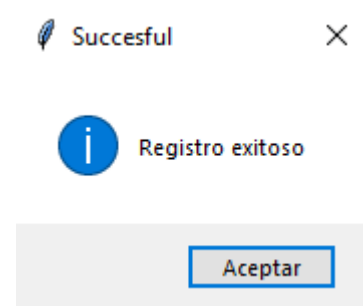
- Ahora bien, llenaremos el formulario de registro con alguno de los alumnos, despues de llenar el formulario presionaremos el botón de Aceptar para poder registrar al alumno (esto se hace por cada alumno que se desea registrar)



A screenshot of a web application window titled "Altas de alumnos". The window contains a registration form with the following fields and values:

- Codigo:** 221,978,388,
- Nombre:** Caro Flores Christopher Tristan
- Carrera:** ICOM (selected from a dropdown menu)
- Correo:** christopher.caro7838 @alumnos.udg.mx

At the bottom of the form are two buttons: "Aceptar" and "Salir".



- El administrador elegirá en el menú Alumnos, luego la opción consultas generales para mostrar la información de todos los alumnos previamente registrados, tiene el modo de buscar a un alumno individualmente.



Consulta de alumnos

Index	Código	Nombre	Carrera	Correo	Adeudo
1	216757047	ARECHIGA ROJAS ROMAN ANTONIO	ICOM	roman.arechiga@alumnos.udg.mx	None
2	222361619	BAILON BADILLO EMMANUEL	ICOM	emmanuel.bailon6161@alumnos.udg.mx	None
3	221978388	CARO FLORES CHRISTOPHER TRISTAN	ICOM	christopher.car07838@alumnos.udg.mx	None
4	218529823	CHAVEZ SAUCEDO BRAIN JESUS	ICOM	brain.chavez@alumnos.udg.mx	None
5	219246728	DE ALBA VELARDE CHRISTIAN MOISES	ICOM	christian.dealba2467@alumnos.udg.mx	None
6	218569116	DIAZ LOPEZ JUAN ENRIQUE	ICOM	juan.diaz5691@alumnos.udg.mx	None
7	221978256	FIERROS CASAS DIEGO ARMANDO	ICOM	diego.fierros7825@alumnos.udg.mx	None
8	219091902	FLORES GRANERO MARIO ZAHIT	ICOM	mario.flores0919@alumnos.udg.mx	None
9	218620774	GARCIA GUTIERREZ JUAN JOSE	ICOM	juan.garcia6207@alumnos.udg.mx	None
10	218419149	GARCIA VALDES EDMUNDO	ICOM	edmundo.garcia4191@alumnos.udg.mx	None
11	222362372	GODINEZ FLORES JONATAN	ICOM	jonatan.godinez6237@alumnos.udg.mx	None
12	217471988	GOMEZ DIAZ ALAN	ICOM	alan.gomez4719@alumnos.udg.mx	None
13	219248402	HERNANDEZ LOPEZ JESUS YAHIR	ICOM	jesus.hernandez2484@alumnos.udg.mx	None
14	219548635	JIMENEZ MAGAÑA SERGIO	ICOM	sergio.jimenez5486@alumnos.udg.mx	None
15	218465841	LOPEZ TORRES RODRIGO ALONSO	ICOM	rodrigo.lopez4658@alumnos.udg.mx	None
16	218507056	MEZA DIAZ VICTOR URIEL	ICOM	victor.meza5070@alumnos.udg.mx	None
17	217509144	NAVARRO RODRIGUEZ DAMIAN	ICOM	damian.navarro5091@alumnos.udg.mx	None
18	219129985	OCHOA LEOS ANDREA GUADALUPE	ICOM	andrea.ochoa1299@alumnos.udg.mx	None
19	218518198	PEREZ DE LA TORRE KEVIN EFREN	ICOM	kevin.perez5181@alumnos.udg.mx	None
20	221977578	RAMOS ROMERO OSCAR DAVID	ICOM	oscar.ramos7757@alumnos.udg.mx	None
21	222361821	RIVERA HARO DARIO	ICOM	dario.rivera6182@alumnos.udg.mx	None
22	216650684	ROMERO ALVAREZ MIRIAM GUADALUPE	ICOM	miriam.romero6506@alumnos.udg.mx	None
23	218111497	SALAZAR ORNELAS OSCAR ALEJANDRO	ICOM	oscar.salazar1114@alumnos.udg.mx	None
24	218564947	ACOSTA SEVILLA EDER ISAAC	ICOM	eder.acosta5649@alumnos.udg.mx	None
25	218798093	ALVAREZ CORTES EDSON ARMANDO	ICOM	edson.alvarez7980@alumnos.udg.mx	None

Buscar Salir

- Ahora que comprobamos que se registraron bien los alumnos podremos pasar a lo siguiente que será registrar a los profesores para ello presionaremos el apartado que diga Profesores y presionaremos Altas



- Al presionar este apartado nos abrirá una pestaña igual que la de alumnos con un formulario para poder dar de alta a los profesores

Altas de profesores

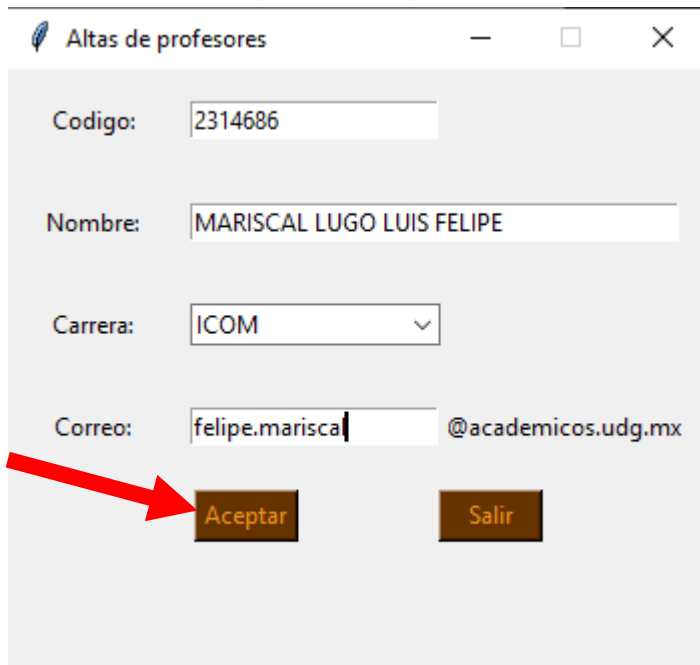
Codigo:

Nombre:

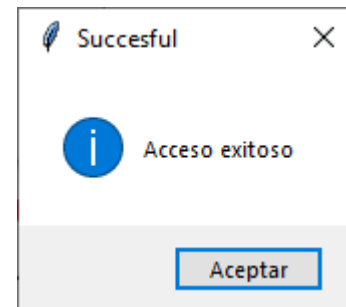
Carrera:

Correo: @academicos.udg.mx

- Ahora registraremos a profesores llenando el formulario con los datos y de igual manera al llenar los datos presionaremos en el botón de aceptar para registrar al profesor (debemos de hacer el formulario con cada uno de los profesores)



The screenshot shows a web form titled "Altas de profesores". It contains the following fields: "Codigo:" with the value "2314686", "Nombre:" with the value "MARISCAL LUGO LUIS FELIPE", "Carrera:" with a dropdown menu showing "ICOM", and "Correo:" with the value "felipe.mariscal@academicos.udg.mx". At the bottom of the form, there are two buttons: "Aceptar" and "Salir". A red arrow points to the "Aceptar" button.



- Para que el Admin pueda revisar que se registraron de manera correcta los profesores volverá al menú y presionara la opción consultar en Profesores de esta manera le aparecerá la tabla de los profesores registrados




Altas de profesores

Consultas profesor							
Index	Código	Nombre	Carrera	Correo	Adeudo		
1	2934568	GUERRRO CARLOS ANA JAZMIN	ICOM	ana.guerrero@academicos.udg.mx	None		
2	2890231	RAMIREZ LUGO FRANCISCO JOSUE	ICOM	josue.ramirez@academicos.udg.mx	None		
3	2790234	LUGO JAIME MIGUEL	ICOM	miguel.lugo@academicos.udg.mx	None		
4	2467893	RODRIGUEZ PIMIENTA CRISTINA	ICOM	cristina.rodriguez@academicos.udg.mx	None		
5	2314686	MARISCAL LUGO LUIS FELIPE	ICOM	felipe.mariscal@academicos.udg.mx	None		

- Ahora pasaremos a la inserción (ALTAS) de libros en el menú presionaremos el apartado de Libros y despues Altas para que nos aparezca un formulario para poder llenar con los daos del libro y empezarlos a registrar-




Altas de libros
—
□
×

ISBN:

Título:

Autor:

Editorial:

Publicacion:

Cantidad:

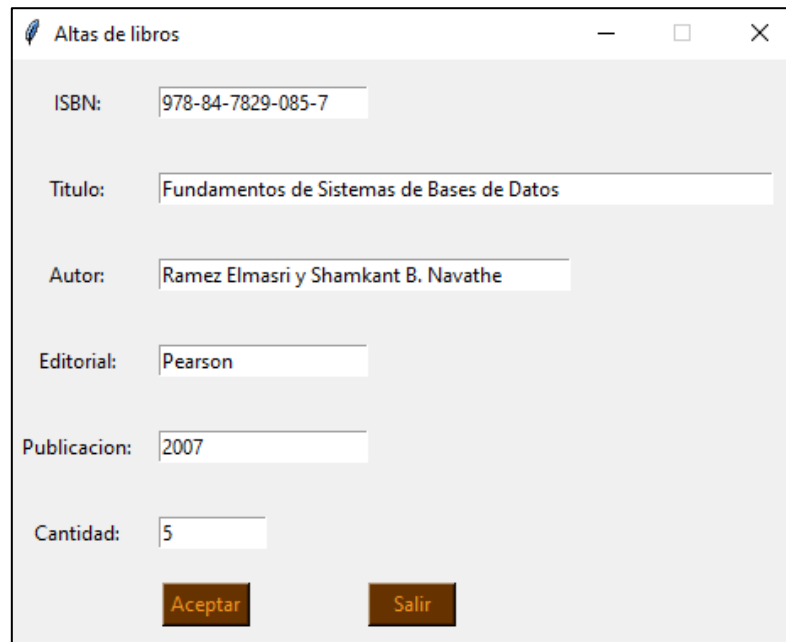
Aceptar

Salir

- Ahora ingresaremos los libros que nos proporcionaron

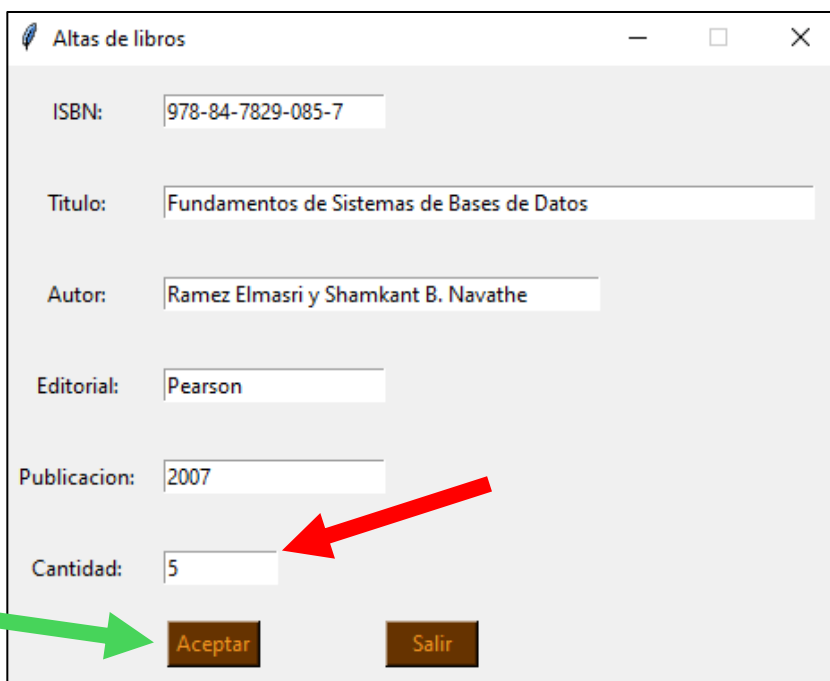
LIBRO					
ISBN	TITULO	AUTOR	EDITORIAL	AÑO_PUBLIC	EJEMPLAR
978-84-7829-085-7	Fundamentos de Sistemas de Bases de Datos	Ramez Elmasri y Shamkant B. Navathe	Pearson	2007	1
978-84-7829-085-7	Fundamentos de Sistemas de Bases de Datos	Ramez Elmasri y Shamkant B. Navathe	Pearson	2007	2
978-84-7829-085-7	Fundamentos de Sistemas de Bases de Datos	Ramez Elmasri y Shamkant B. Navathe	Pearson	2007	3
978-84-7829-085-7	Fundamentos de Sistemas de Bases de Datos	Ramez Elmasri y Shamkant B. Navathe	Pearson	2007	4
978-84-7829-085-7	Fundamentos de Sistemas de Bases de Datos	Ramez Elmasri y Shamkant B. Navathe	Pearson	2007	5
970-613-461-1	Bases de datos, Modelos, lenguajes, diseño	James L. Johnson	Oxford	2000	1
970-613-461-1	Bases de datos, Modelos, lenguajes, diseño	James L. Johnson	Oxford	2000	2
970-613-461-1	Bases de datos, Modelos, lenguajes, diseño	James L. Johnson	Oxford	2000	3
970-613-461-1	Bases de datos, Modelos, lenguajes, diseño	James L. Johnson	Oxford	2000	4
970-613-461-1	Bases de datos, Modelos, lenguajes, diseño	James L. Johnson	Oxford	2000	5
978-84-291-4396-6	Sistemas de bases de datos	Paul Beynon-Davies	Reverte	2014	1
978-84-291-4396-6	Sistemas de bases de datos	Paul Beynon-Davies	Reverte	2014	2
978-84-291-4396-6	Sistemas de bases de datos	Paul Beynon-Davies	Reverte	2014	3
978-84-291-4396-6	Sistemas de bases de datos	Paul Beynon-Davies	Reverte	2014	4
978-84-291-4396-6	Sistemas de bases de datos	Paul Beynon-Davies	Reverte	2014	5
10-607-481-618-2	Bases de datos, diseño, implementación y administración	Carlos Coronel, Steven Morris, Peter Rob	CENGAGE Learning	2011	1
10-607-481-618-2	Bases de datos, diseño, implementación y administración	Carlos Coronel, Steven Morris, Peter Rob	CENGAGE Learning	2011	2
10-607-481-618-2	Bases de datos, diseño, implementación y administración	Carlos Coronel, Steven Morris, Peter Rob	CENGAGE Learning	2011	3
10-607-481-618-2	Bases de datos, diseño, implementación y administración	Carlos Coronel, Steven Morris, Peter Rob	CENGAGE Learning	2011	4
10-607-481-618-2	Bases de datos, diseño, implementación y administración	Carlos Coronel, Steven Morris, Peter Rob	CENGAGE Learning	2011	5

- Ahora insertaremos los datos de algún libro y lo daremos de alta

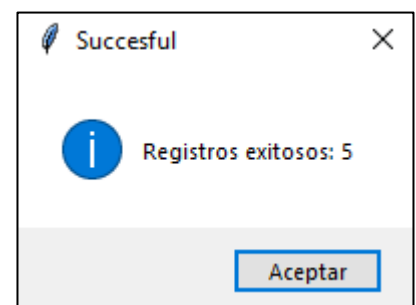


A screenshot of a web application window titled "Altas de libros". It contains several input fields for book registration: ISBN (978-84-7829-085-7), Titulo (Fundamentos de Sistemas de Bases de Datos), Autor (Ramez Elmasri y Shamkant B. Navathe), Editorial (Pearson), Publicacion (2007), and Cantidad (5). At the bottom, there are two buttons: "Aceptar" and "Salir".

- Podemos observar que aparece un apartado de cantidad en este apartado podemos ingresar la cantidad de ejemplares del mismo libro para no estar ingresando libro por libro y sea más fácil poner la cantidad de ejemplares del mismo libro, al tener los datos ingresados podremos registrar los libros dándole en aceptar y apareciendo que registramos 5 con éxito



A screenshot of the same "Altas de libros" form. A red arrow points to the "Cantidad" field, which contains the value "5". A green arrow points to the "Aceptar" button at the bottom left.



- Para poder consultar los libros registrados vamos al menú y en el apartado de libros presionaremos consultas para que nos muestre una nueva pestaña con la tabla de todos los registros de libros, tiene forma de buscar a los libros en particular, ingresando en el buscado el ISBN



- Ahora pasaremos a la parte de prestamos para esto registraremos 2 prestamos uno para, pasaremos al menú y presionaremos en el apartado de prestamos y le daremos clic en altas



- Al presionar en Altas se nos abrirá una pestaña con un formulario para realizar los prestamos (los préstamos por default están para los alumnos)

Altas de prestamos

☒ PROFESOR

Codigo alumno:

Codigo profesor:

Correo:

ISBN:

Ejemplares:

☐ MANUAL

Fecha prestamo: 5/5/24

Fecha limite: 05/12/24

Aceptar Salir

- Como por default los prestamos están para alumnos, nosotros ocupamos realizar unos para profesores, lo que haremos será presionar el cuadrado de la parte superior que dice PROFESOR para que el formulario cambie para realizar prestamos de profesores

Altas de prestamos

☒ PROFESOR

Codigo alumno:

Codigo profesor:

Correo:

ISBN:

Ejemplares:

☐ MANUAL

Fecha prestamo: 5/5/24

Fecha limite: 05/12/24

Aceptar Salir

- Ahora bien, llenamos los datos para realizar el pedido del maestro Mariscal y de la maestra Cristina despues de los datos daremos clic en aceptar para realizar el pedido

Altas de prestamos

☒ PROFESOR

Codigo alumno:

Codigo profesor:

Correo:

ISBN:

Ejemplares:

☒ MANUAL

Fecha prestamo:

Fecha limite:

Successful

Registro exitoso

Altas de prestamos

☒ PROFESOR

Codigo alumno:

Codigo profesor:

Correo:

ISBN:

Ejemplares:

☒ MANUAL

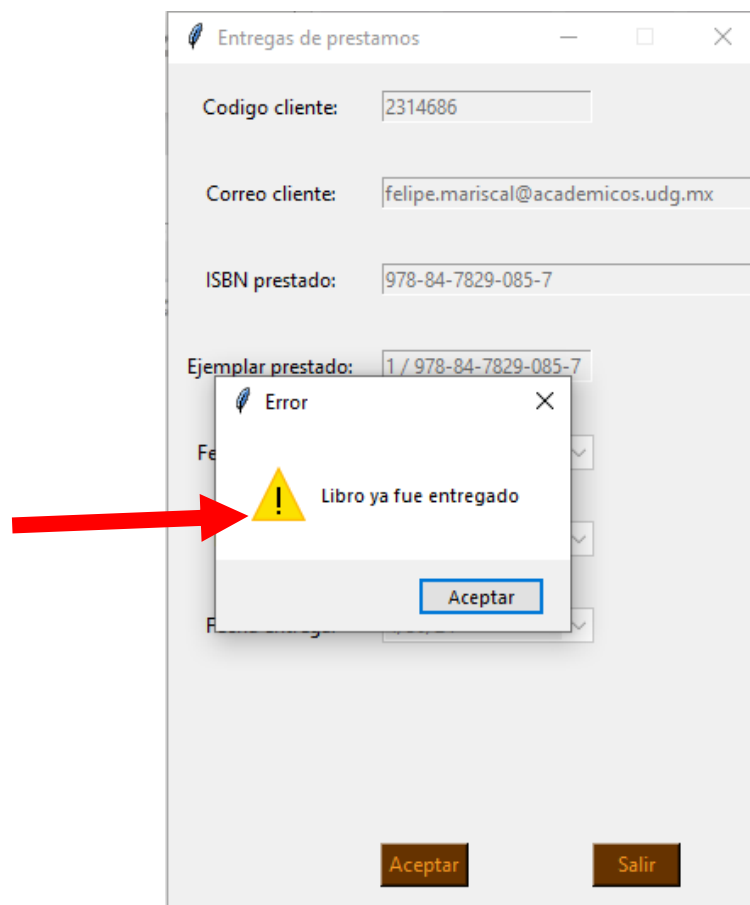
Fecha prestamo:

Fecha limite:

Successful

Registro exitoso

- Si por alguna razón por error realizamos un préstamo que ya existe nos aparecería un error indicando que el préstamo ya fue realizado



- Para consultar los préstamos que se realizaron nos pasamos al menú principal y en el apartado de prestamos le damos en consultas.



- Después de darle clic en consultas nos mandara a una pestaña que nos mostrara los pedidos que hayamos echo en este caso solo deben de aparecer 2 y solo de maestros

Index	ID prestamo	Codigo cliente	Correo cliente	ISBN	N° Ejemplar	Fecha prestamo	Fecha limite	Fecha entrega
1	3	2314686	felipe.mariscal@academicos.udg.mx	978-84-7829-085-7	1 / 978-84-7829-085-7	2024-04-22	2024-04-29	None
2	4	2467893	cristina.rodriguez@academicos.udg.	978-84-291-4396-6	1 / 978-84-291-4396-6	2024-04-22	2024-04-29	None

- Como podrás observar la fecha de entrega aparece como NONE esto quiere decir que no se a echo ninguna entrega para poder realizar la entrega debemos de dar clic en alguna de las tuplas (filas) para poder editar la fecha de entrega

Index	ID prestamo	Codigo cliente	Correo cliente	ISBN	N° Ejemplar	Fecha prestamo	Fecha limite	Fecha entrega
1	3	2314686	felipe.mariscal@academicos.udg.mx	978-84-7829-085-7	1 / 978-84-7829-085-7	2024-04-22	2024-04-29	None
2	4	2467893	cristina.rodriguez@academicos.udg.	978-84-291-4396-6	1 / 978-84-291-4396-6	2024-04-22	2024-04-29	2024-04-29

- Cuando le damos clic en la tupla que queremos editar para poder hacer la entrega del préstamo en este caso utilizaremos el ejemplo con la maestra Cristina nos abrirá una pestaña nueva y para poder realizar el pedido presionaremos el botón de aceptar

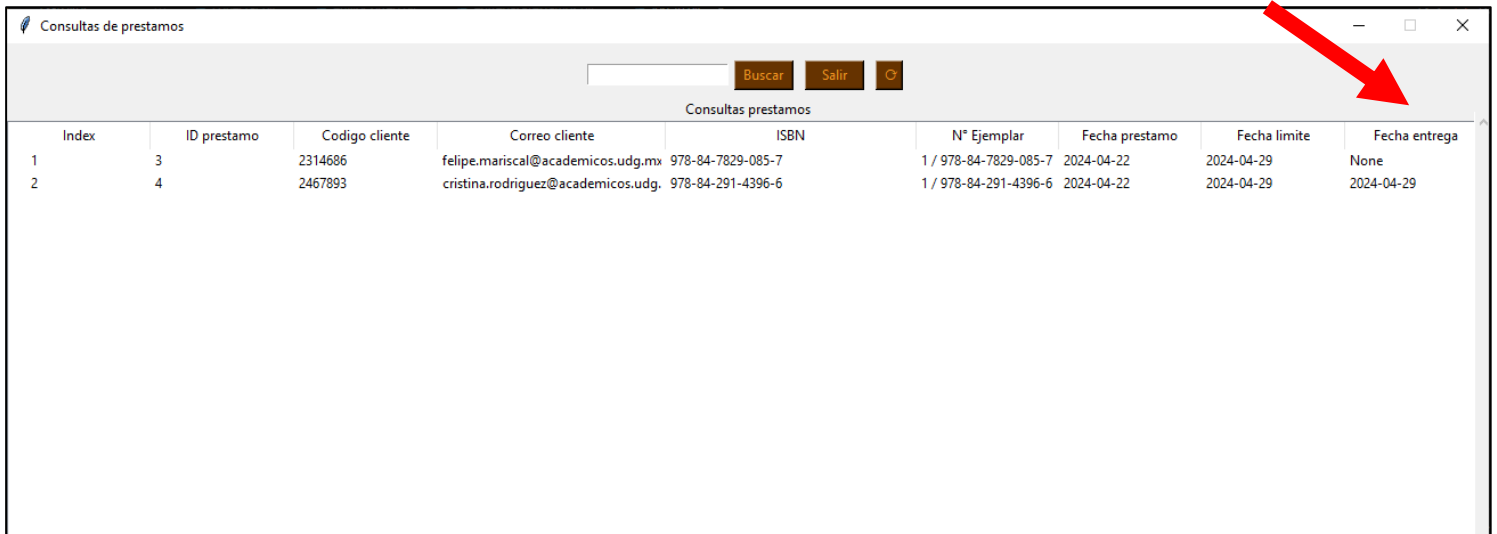
The image shows a web application interface for managing loan deliveries. The main window, titled "Entregas de prestamos", contains a form with the following fields:

- Codigo cliente: 2467893
- Correo cliente: cristina.rodriguez@academicos.udg.mx
- ISBN prestado: 978-84-291-4396-6
- Ejemplar prestado: 1 / 978-84-291-4396-6
- Fecha prestamo: 4/22/24 (dropdown menu)
- Fecha limite: 04/29/24 (dropdown menu)
- Fecha entrega: 04/29/24 (dropdown menu)

At the bottom of the form are two buttons: "Aceptar" (Accept) and "Salir" (Exit). A red arrow points to the "Aceptar" button.

To the right of the main form is a smaller window titled "Successful" with a close button (X). It displays a blue information icon and the text "Entrega exitosa" (Successful delivery). At the bottom of this window is a button labeled "Aceptar".

- Ahora bien si regresamos a donde se encuentra la tabla podemos observar que en el apartado de la fecha de entrega aparece ya la fecha de entrega de la maestra Cristina mientras que la del profe Mariscal aparece como NONE



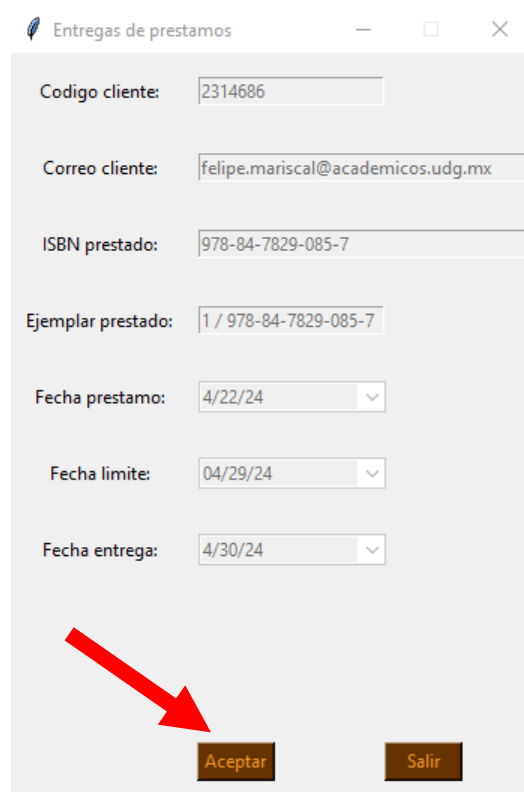
Consultas de prestamos

Buscar Salir

Consultas prestamos

Index	ID prestamo	Codigo cliente	Correo cliente	ISBN	N° Ejemplar	Fecha prestamo	Fecha limite	Fecha entrega
1	3	2314686	felipe.mariscal@academicos.udg.mx	978-84-7829-085-7	1 / 978-84-7829-085-7	2024-04-22	2024-04-29	None
2	4	2467893	cristina.rodriguez@academicos.udg.	978-84-291-4396-6	1 / 978-84-291-4396-6	2024-04-22	2024-04-29	2024-04-29

- Ahora para la entrega del libro del profe Mariscal realizaremos una entrega tardía para que se cobre una multa, llenamos los datos y le damos en aceptar



Entregas de prestamos

Codigo cliente: 2314686

Correo cliente: felipe.mariscal@academicos.udg.mx

ISBN prestado: 978-84-7829-085-7

Ejemplar prestado: 1 / 978-84-7829-085-7

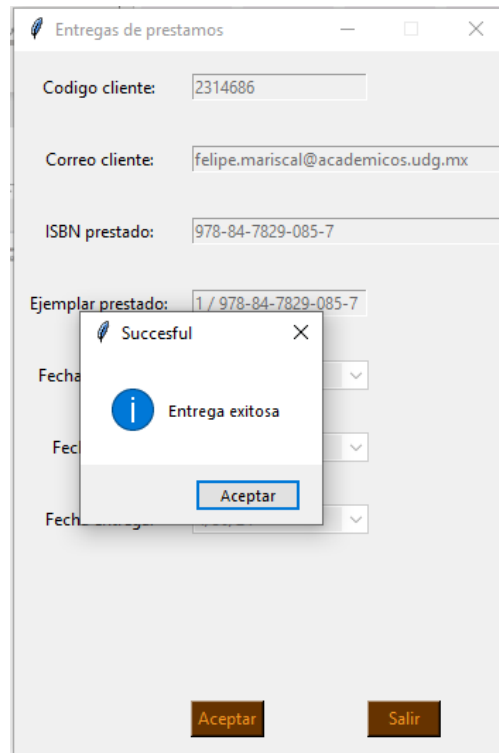
Fecha prestamo: 4/22/24

Fecha limite: 04/29/24

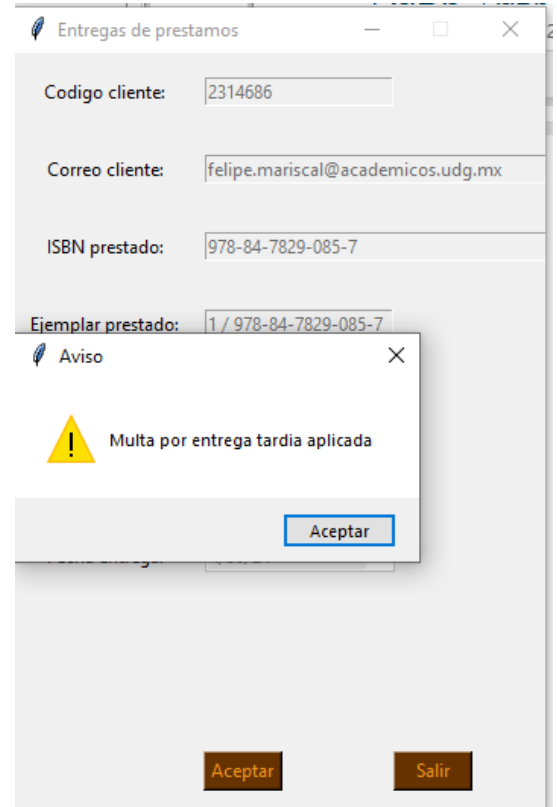
Fecha entrega: 4/30/24

Aceptar Salir

- Al darle en aceptar como fue una entrega tardía nos aparecerá una pestaña emergente que indica que se realizó la entrega con éxito.



- Despues de que aparezca esa pestaña emergente nos aparecerá otra que nos indicara que se aplicó una multa por la entrega tardía



- Al realizar esto le llega un correo con un formato de una ficha de pago y el adeudo que obtuvo por la entrega tardía,
- Ahora bien si volvemos a la tabla de las consultas nos aparecerá el préstamo del profesor Mariscal con fecha de entrega

Consultas prestamos									
Index	ID prestamo	Codigo cliente	Correo cliente	ISBN	N° Ejemplar	Fecha prestamo	Fecha limite	Fecha entrega	
1	4	2467893	cristina.rodriguez@academicos.udg.	978-84-291-4396-6	1 / 978-84-291-4396-6	2024-04-22	2024-04-29	2024-04-29	
2	3	2314686	felipe.mariscal@academicos.udg.mx	978-84-7829-085-7	1 / 978-84-7829-085-7	2024-04-22	2024-04-29	2024-04-30	

- para el registro de los prestamos se creo el siguiente Código que se presentara a continuación:

```

15 import tkinter as tk
16 from elementos import LabelEntryFrame, BotonText, SeleccionadorFecha
17 from tkinter import messagebox
18 from conectar import obtener_datos_cliente, entrega_libro, obtener_dato, enviar_notificacion
19 from datetime import timedelta
20
21 def
registrar(salidaCodigoIsbn, salidaEjemplaresPrestado, entradaFechaEntrega, id_prestamo, salidaFechaLimite, salidaCodigoCliente):
22     codigo_cliente = salidaCodigoCliente.get_entry() #Obtener el codigo del cliente del entry
23
24     isbn = salidaCodigoIsbn.get_entry() #Obtener el ISBN del entry
25
26     ejemplar = salidaEjemplaresPrestado.get_entry() #Obtener el numero del ejemplar del entry
27
28     fecha_entrega_sinFormato = entradaFechaEntrega.get_fecha() #Obtener la fecha de entrega
directo del DateEntry
29
30     fecha_entrega = fecha_entrega_sinFormato.strftime("%Y/%m/%d") #Formatear la fecha para
su almacenamiento en la base
31
32     fecha_limite_sinFormato = salidaFechaLimite.get_fecha() #Obtener la fecha limite de entrega
directo del DateEntry
33
34     if fecha_entrega_sinFormato > fecha_limite_sinFormato: #Verificar que la fecha de entrega sea
menor a la fecha limite
35         multa = verificar_entrega(fecha_entrega_sinFormato, fecha_limite_sinFormato) #Si es mayor
se llama a la funcion "verificar_entrega", se le pasan los argumentos y se le asigna el valor retornado a
multa
36     else:

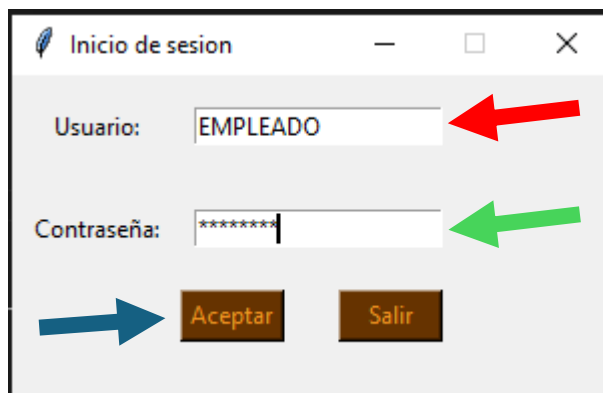
```

```

37     multa = 0 #Si es menor , se le asigna valor de 0 a multa
38
39     verificarEstado = obtener_dato(isbn,ejemplar,"estado","libro") #Se llama a la funcion
"obtener_dato" para buscar el estado del libro y se le asigna a verificarEstado
40
41     if verificarEstado == "No disponible": #Verificar que el estado sea "No disponible"
42         if entrega_libro(isbn,ejemplar,fecha_entrega,id_prestamo,multa,codigo_cliente) == True:
#Se intenta la entrega del libro
43             entradaFechaEntrega.deshabilitar() #Asegurar que entradaFechaEntrega esta deshabilitado
44         else:
45             messagebox.showwarning("Error", "Libro ya fue entregado") #Si no , se avisa que el libro ya
fue entregado

```

- con esto terminamos de explicar la funcionalidad del proyecto, pero como se abran dado cuenta solo realizamos el proceso por medio del ADMIN ahora para hacerlo con EMPLEADO correremos de nuevo el programa e iniciaremos sesión como EMPLEADO (daremos en ACEPTAR)



- Como podemos observar en la interfaz parece ser que no tiene nada de raro y esta normal pero si nos vamos a las opciones de Alumnos, Profesores, Libros podemos ver que la opción de Altas se encuentra fuera de servicio sin poder acceder a ella solamente a consultas



- Lo que sucede es que tenemos una jerarquía de privilegios entre lo que puede hacer un empleado y lo que puede hacer un ADMIN entonces lo que puede

hacer el EMPLEADO es que puede consultar todas las tablas y el si puede disponer de todas las funciones de prestamos ya que será el tipo encargado de la realización de los prestamos al igual que llevar el control de las devoluciones



- Finalmente tenemos el botón de salir que lo que ocurrirá es que terminara la interfaz de la biblioteca

