1

Examen Parcial 1

Heli Saul, Vásquez Gómez, 201700852 Escuela de Mecánica Eléctrica, Facultad de Ingeniería, Universidad de San Carlos de Guatemala

Resumen: a los resultados del examen parcial el cual consiste en una parte teórica y otra en una parte práctica, cumpliendo con todos los objetivos se logran visualizar los resultados requeridos en el documento final.

B. Objetivos

Objetivo General

1. Realiza el examen parcial

Objetivo especifico

- 1. Mostrar el código
- 2. Mostrar los resultados obtenidos

C. comandos utilizados

```
import psycopg2
import matplotlib.pyplot as plt
import pandas as pd
from prettytable import PrettyTable
import random
conn = psycopg2.connect(
  dbname='estudiantes',
  user='postgres',
  password='1234',
  host='localhost',
  port='5432'
cur = conn.cursor()
def opcion1():
  while True:
     print("Menu de Opciones:")
     print("1. Ingresar datos")
     print("2. Eliminar")
     print("3. Actualizar")
     print("4. Salir")
```

```
op = int(input("ingres opcion "))
     if op == 1:
            conn = psycopg2.connect(
              dbname='estudiantes',
              user='postgres',
              password='1234',
              host='localhost',
              port='5432'
            cur = conn.cursor()
            nombre = input("Ingrese su nombre : ")
            edad = int(input("Ingrese su edad : "))
            genero = input("ingrese su e; genero : ")
            direction = input("Ingrese su direction: ")
                    # Preparar la instrucción SQL para la
inserción
                      Ins1
                                   'INSERT
                                                INTO
                                                         datos
(nombre,edad,genero,direccion) VALUES (%s,%s,%s,%s,%s);' #
%s es un marcador de posición para el valor
                      Instruccion
                                             cur.mogrify(Ins1,
(nombre,edad,genero,direccion))
                      # Ejecutar la instrucción SQL
                      cur.execute(Instruccion)
                      # Confirmar la transacción
                      conn.commit()
            except psycopg2.Error as e:
                    print(f"Error durante la conexión a la DB.
Consulte el error: {e}")
            finally:
                 # Cerrar el cursor y la conexión
                   cur.close()
                   conn.close()
     elif op == 2:
       try:
          # Establecer la conexión a la base de datos
          conn = psycopg2.connect(
              dbname='estudiantes',
              user='postgres',
              password='1234',
              host='localhost',
              port='5432'
```

```
UPDATE datos
         cur = conn.cursor()
                                                                              SET edad = %s, genero = %s, direccion = %s
         # Obtener el nombre para identificar el registro a
                                                                              WHERE nombre = %s;
eliminar
         nombre a eliminar = input("Ingrese el nombre a
                                                                            # Ejecutar la actualización
eliminar: ")
                                                                            cur.execute(sql actualizacion,
                                                                                                                (nuevo edad,
         # Construir y ejecutar la sentencia DELETE
                                                                nueva genero,nuevo direccion,nombre estudiante))
         sentencia delete = "DELETE FROM datos WHERE
nombre = %s;"
                                                                            # Confirmar la transacción
         cur.execute(sentencia_delete, (nombre_a_eliminar,))
                                                                            conn.commit()
         # Confirmar la transacción
                                                                            print(f"Se ha actualizado la información del
         conn.commit()
                                                                estudiante '{nombre estudiante}'.")
         print("Registro eliminado exitosamente.")
                                                                       except psycopg2. Error as e:
       except psycopg2. Error as e:
                                                                          print(f"Error durante la conexión a la DB. Consulte el
         print(f"Error durante la conexión a la DB. Consulte el
                                                               error: {e}")
error: {e}")
                                                                         # Cerrar la conexión
       finally:
         # Cerrar el cursor y la conexión
                                                                         cur.close()
         cur.close()
                                                                         conn.close()
                                                                     elif op == 4:
         conn.close()
    elif op == 3:
                                                                       break
       conn = psycopg2.connect(
            dbname='estudiantes',
            user='postgres',
            password='1234',
            host='localhost',
            port='5432'
         )
       cur = conn.cursor()
       try:
         # Solicitar al usuario el nombre del producto
                                                                def opcion2():
         nombre estudiante = input("Ingrese el nombre del
estuante que desea actualizar: ")
                                                                  while True:
         # Verificar si el producto existe antes de continuar
         cur.execute("SELECT COUNT(*) FROM datos
                                                                     print("Menu de Opciones:")
WHERE nombre = %s;", (nombre estudiante,))
                                                                     print("1. Ingresar gasto")
         cantidad productos = cur.fetchone()[0]
                                                                     print("2. ver resumen")
                                                                     print("3. Actualizar")
         if cantidad productos == 0:
                                                                    print("4. Salir")
           print(f"No se encontró ningún estudiante con el
nombre '{nombre estudiante}'.")
                                                                    op = int(input("ingres opcion "))
         else:
            # Solicitar al usuario los nuevos valores
            nuevo edad = (input("Ingrese la nueva edad: "))
                                                                     if op == 1:
            nueva genero = (input("Ingrese el nuevo genero:
                                                                       conn = psycopg2.connect(
"))
                                                                         dbname='gasto',
            nuevo direccion = (input("Ingrese una nueva
                                                                          user='postgres',
                                                                          password='1234',
direccion: "))
                                                                          host='localhost',
            # Sentencia SQL de actualización
                                                                          port='5432'
            sql actualizacion = """
```

```
cur = conn.cursor()
                                                                        # Cerrar la conexión
       nombre gasto= input("nombre del gasto : ")
                                                                        cur.close()
       gasto = float(input("ingrese cantidad de gasto: "))
                                                                        conn.close()
       presupuesto = float(input("ingrese su presupuesto actual
                                                                     elif op == 3:
: "))
                                                                        conn = psycopg2.connect(
              # Preparar la instrucción SQL para la inserción
                                                                             dbname='gasto',
                Ins1
                               'INSERT
                                             INTO
                                                                             user='postgres',
                                                       gasto1
(nombregasto,cantidad,presupuesto) VALUES (%s,%s,%s);' #
                                                                             password='1234',
%s es un marcador de posición para el valor
                                                                             host='localhost',
                                                                            port='5432'
                 Instruccion
                                            cur.mogrify(Ins1,
                                                                        cur = conn.cursor()
(nombre gasto,gasto,presupuesto))
                                                                        try:
                 # Ejecutar la instrucción SQL
                                                                          # Solicitar al usuario el nombre del producto
                 cur.execute(Instruccion)
                                                                          nombre gasto= input("Ingrese el nombre del gasto
                                                                que desea actualizar: ")
                 # Confirmar la transacción
                 conn.commit()
                                                                          # Verificar si el producto existe antes de continuar
                                                                          cur.execute("SELECT COUNT(*) FROM gasto1
                                                                 WHERE nombregasto = %s;", (nombre gasto,))
       except psycopg2. Error as e:
              print(f''Error durante la conexión a la DB.
                                                                          cantidad productos = cur.fetchone()[0]
Consulte el error: {e}")
                                                                          if cantidad productos == 0:
                                                                            print(f"No se encontró ningún gasto con el nombre
       finally:
            # Cerrar el cursor y la conexión
                                                                '{nombre gasto}'.")
              cur.close()
                                                                          else:
              conn.close()
                                                                             # Solicitar al usuario los nuevos valores
     elif op == 2:
                                                                             nuevo cantidad = int(input("Ingrese la nueva
      # Parámetros de conexión a la base de datos
                                                                cantidad: "))
       conn = psycopg2.connect(
                                                                            nueva presupuesto = int(input("Ingrese el nuevo
         dbname='gasto',
                                                                presupesuto: "))
         user='postgres',
         password='1234',
         host='localhost',
                                                                             # Sentencia SQL de actualización
         port='5432'
                                                                             sql actualizacion = """
                                                                               UPDATE gasto1
                                                                               SET cantidad = %s, presupuesto = %s
       cur = conn.cursor()
                                                                               WHERE nombregasto = %s;
       # Nombre de la tabla que deseas consultar
       nombre tabla = "gasto1"
                                                                             # Ejecutar la actualización
       # Consulta SQL para obtener todos los datos de la tabla
                                                                             cur.execute(sql actualizacion,
                                                                                                              (nuevo cantidad,
       consulta sql = f"SELECT * FROM {nombre tabla};"
                                                                nueva presupuesto,nombre gasto))
       # Ejecutar la consulta
                                                                            # Confirmar la transacción
       cur.execute(consulta sql)
                                                                             conn.commit()
       # Obtener todos los resultados
                                                                            print(f"Se ha actualizado la información del gasto
       resultados = cur.fetchall()
                                                                '{nombre gasto}'.")
                                                                        except psycopg2.Error as e:
       # Mostrar los resultados
                                                                          print(f"Error durante la conexión a la DB. Consulte el
       for fila in resultados:
                                                                error: {e}")
         print(" id "," nombre "," cantidad ", " presupuesto ")
         print(" ",fila)
                                                                          # Cerrar la conexión
```

```
cur.close()
                                                                        finally:
         conn.close()
                                                                             # Cerrar el cursor y la conexión
     elif op == 4:
                                                                               cur.close()
                                                                               conn.close()
        break
                                                                      elif op == 2:
                                                                        try:
                                                                          # Establecer la conexión a la base de datos
def opcion3():
                                                                          conn = psycopg2.connect(
                                                                          dbname='produc',
                                                                          user='postgres',
                                                                          password='1234',
  while True:
                                                                          host='localhost',
     print("Menu de Opciones:")
                                                                          port='5432'
    print("1. Ingresar producto")
    print("2. Eliminar")
                                                                          cur = conn.cursor()
     print("3. Actualizar")
                                                                          # Obtener el nombre para identificar el registro a
     print("4. Salir")
                                                                eliminar
                                                                          nombre a eliminar = input("Ingrese el nombre del
                                                                producto a eliminar: ")
     op = int(input("ingres opcion "))
                                                                          # Construir y ejecutar la sentencia DELETE
                                                                          sentencia delete = "DELETE FROM productos
     if op == 1:
                                                                 WHERE nombre = %s;"
                                                                          cur.execute(sentencia_delete, (nombre_a_eliminar,))
       conn = psycopg2.connect(
         dbname='produc',
                                                                          # Confirmar la transacción
         user='postgres',
                                                                          conn.commit()
         password='1234',
                                                                          print("Registro eliminado exitosamente.")
         host='localhost',
         port='5432'
                                                                        except psycopg2.Error as e:
                                                                          print(f"Error durante la conexión a la DB. Consulte el
         )
       cur = conn.cursor()
                                                                error: {e}")
       nombre = input("Ingrese nombre del producto : ")
       precio = input("Ingrese el pprecio: ")
                                                                        finally:
                                                                          # Cerrar el cursor y la conexión
       cantidad = input("ingrese la cantidad : ")
                                                                          cur.close()
                                                                          conn.close()
                                                                     elif op == 3:
       try:
              # Preparar la instrucción SQL para la inserción
                 Ins1
                              'INSERT
                                          INTO
                                                    productos
(nombre,precio,cantidad) VALUES (%s,%s,%s);' # %s es un
                                                                        conn = psycopg2.connect(
                                                                             dbname='produc',
marcador de posición para el valor
                                                                             user='postgres',
                                                                             password='1234',
                 Instruccion
                                             cur.mogrify(Ins1,
                                                                             host='localhost',
(nombre,precio,cantidad))
                                                                             port='5432'
                 # Ejecutar la instrucción SQL
                                                                        cur = conn.cursor()
                 cur.execute(Instruccion)
                                                                        try:
                 # Confirmar la transacción
                                                                          # Solicitar al usuario el nombre del producto
                 conn.commit()
                                                                          nombre producto= input("Ingrese el nombre del
                                                                 producto que desea actualizar: ")
       except psycopg2. Error as e:
              print(f''Error durante la conexión a la DB.
                                                                          # Verificar si el producto existe antes de continuar
Consulte el error: {e}")
                                                                          cur.execute("SELECT COUNT(*) FROM productos
                                                                 WHERE nombre = %s;", (nombre producto,))
```

```
cantidad productos = cur.fetchone()[0]
                                                                          dbname='pedidos',
                                                                          user='postgres',
         if cantidad productos == 0:
                                                                          password='1234',
            print(f"No se encontró ningún gasto con el nombre
                                                                          host='localhost',
'{nombre producto}'.")
                                                                          port='5432'
         else:
            # Solicitar al usuario los nuevos valores
                                                                        cur = conn.cursor()
            nuevo precio = int(input("Ingrese el nuevo precio:
                                                                        nombre = input("Ingrese el nombre del solicitante : ")
"))
                                                                        nombre pedido = input("Ingrese nombre del pedido : ")
            nueva cantidad = int(input("Ingrese la nueva
cantidad: "))
                                                                        precio = input("ingrese el precio del pedido: ")
                                                                        cantidad = input("Ingrese la cantidad del pedido: ")
                                                                        try:
                                                                               # Preparar la instrucción SQL para la inserción
            # Sentencia SQL de actualización
            sql actualizacion = """
                                                                                                'INSERT
                                                                                                             INTO
                                                                                  Ins1
                                                                                                                        pedido
              UPDATE productos
                                                                 (nombre,nombrepedido,precio,cantidad)
                                                                                                                     VALUES
                                                                 (%s,%s,%s,%s);' # %s es un marcador de posición para el valor
              SET precio = %s, cantidad = %s
              WHERE nombre = %s;
                                                                                  Instruccion
                                                                                                             cur.mogrify(Ins1,
            # Ejecutar la actualización
                                                                 (nombre,nombre pedido,precio,cantidad))
            cur.execute(sql actualizacion,
                                               (nuevo precio,
nueva cantidad, nombre producto))
                                                                                  # Ejecutar la instrucción SQL
                                                                                  cur.execute(Instruccion)
            # Confirmar la transacción
            conn.commit()
                                                                                  # Confirmar la transacción
                                                                                  conn.commit()
            print(f"Se ha actualizado la información del
producto '{nombre producto}'.")
                                                                        except psycopg2.Error as e:
                                                                               print(f"Error durante la conexión a la DB.
       except psycopg2. Error as e:
                                                                 Consulte el error: {e}")
         print(f"Error durante la conexión a la DB. Consulte el
error: {e}")
                                                                        finally:
                                                                             # Cerrar el cursor y la conexión
       finally:
                                                                               cur.close()
         # Cerrar la conexión
                                                                               conn.close()
         cur.close()
                                                                     elif op == 2:
         conn.close()
                                                                        try:
     elif op == 4:
                                                                          # Establecer la conexión a la base de datos
       break
                                                                          conn = psycopg2.connect(
                                                                          dbname='pedidos',
def opcion4():
                                                                          user='postgres',
                                                                          password='1234',
                                                                          host='localhost',
  while True:
                                                                          port='5432'
     print("Menu de Opciones:")
                                                                          cur = conn.cursor()
     print("1. Ingresar datos")
                                                                          # Obtener el nombre para identificar el registro a
     print("2. Eliminar")
                                                                 eliminar
     print("3. Actualizar")
                                                                          nombre a eliminar = input("Ingrese el nombre a
     print("4. Salir")
                                                                eliminar: ")
                                                                          # Construir y ejecutar la sentencia DELETE
                                                                          sentencia delete = "DELETE FROM pedido WHERE
     op = int(input("ingres opcion "))
                                                                 nombre = %s;"
     if op == 1:
                                                                          cur.execute(sentencia delete, (nombre a eliminar,))
```

conn = psycopg2.connect(

```
# Confirmar la transacción
                                                                             # Confirmar la transacción
         conn.commit()
                                                                             conn.commit()
         print("Registro eliminado exitosamente.")
                                                                             print(f"Se ha actualizado la información del pedido
       except psycopg2. Error as e:
                                                                 '{nombre pedido}'.")
         print(f"Error durante la conexión a la DB. Consulte el
error: {e}")
                                                                        except psycopg2. Error as e:
                                                                           print(f"Error durante la conexión a la DB. Consulte el
       finally:
                                                                 error: {e}")
         # Cerrar el cursor y la conexión
         cur.close()
                                                                        finally:
                                                                           # Cerrar la conexión
         conn.close()
     elif op == 3:
                                                                           cur.close()
                                                                           conn.close()
       conn = psycopg2.connect(
                                                                      elif op == 4:
            dbname='pedidos',
            user='postgres',
                                                                        break
            password='1234',
            host='localhost',
                                                                 def opcion5():
            port='5432'
                                                                    while True:
       cur = conn.cursor()
                                                                      print("Menu de Opciones:")
       try:
         # Solicitar al usuario el nombre del producto
                                                                      print("1. Ingresar venta")
         nombre pedido= input("Ingrese el nombre que desea
                                                                      print("2. Informe")
actualizar: ")
                                                                      print("3. Tendencia")
                                                                      print("4. Analisis")
                                                                      print("5. Salir")
         # Verificar si el producto existe antes de continuar
         cur.execute("SELECT COUNT(*) FROM pedido
WHERE nombre = %s;", (nombre pedido,))
         cantidad_productos = cur.fetchone()[0]
                                                                      op = int(input("ingres opcion : "))
         if cantidad productos == 0:
                                                                      if op == 1:
            print(f"No se encontró ningún gasto con el nombre
                                                                        conn = psycopg2.connect(
'{nombre pedido}'.")
                                                                           dbname='ventas',
                                                                           user='postgres',
         else:
            # Solicitar al usuario los nuevos valores
                                                                           password='1234',
            nuevo pedido = (input("Ingrese el nuevo nombre
                                                                           host='localhost',
                                                                           port='5432'
del pedido: "))
            nueva precio = int(input("Ingrese el nuevo precio:
                                                                        )
"))
                                                                        cur = conn.cursor()
                                                                        nombre gasto= input("nombre venta: ")
            nueva cantidad = int(input("Ingrese la nueva
cantidad: "))
                                                                        gasto = float(input("ingrese cantidad : "))
                                                                        try:
            # Sentencia SQL de actualización
                                                                                # Preparar la instrucción SQL para la inserción
            sql actualizacion = """
                                                                                                  'INSERT
                                                                                                               INTO
                                                                                  Ins1
                                                                                                                          venta
              UPDATE pedido
                                                                 (nombre,cantidad) VALUES (%s,%s);' # %s es un marcador de
              SET nombrepedido = %s, precio = %s, cantidad
                                                                 posición para el valor
= \frac{\%}{5}
               WHERE nombre = %s;
                                                                                  Instruccion
                                                                                                              cur.mogrify(Ins1,
                                                                 (nombre gasto,gasto))
            # Ejecutar la actualización
            cur.execute(sql actualizacion,
                                               (nuevo pedido,
                                                                                  # Ejecutar la instrucción SQL
nueva precio,nueva cantidad,nombre pedido))
                                                                                  cur.execute(Instruccion)
```

```
# Confirmar la transacción
                                                                 venta:"
                 conn.commit()
                                                                           cur.execute(consulta sql)
       except psycopg2. Error as e:
                                                                           # Obtener resultados
              print(f"Error durante la conexión a la DB.
                                                                           resultados = cur.fetchall()
Consulte el error: {e}")
                                                                           # Extraer columnas para el gráfico
       finally:
                                                                           columna x, columna y = zip(*resultados)
            # Cerrar el cursor y la conexión
                                                                           # Graficar los datos
              cur.close()
              conn.close()
                                                                           plt.plot(columna x,
                                                                                                    columna y,
                                                                                                                     marker='o',
    elif op == 2:
                                                                 linestyle='-')
      # Parámetros de conexión a la base de datos
                                                                           plt.title("Tendencia de ventas")
       conn = psycopg2.connect(
                                                                           plt.xlabel("Nombre de la Venta")
         dbname='ventas',
                                                                           plt.ylabel("Cantidad de ventas")
         user='postgres',
         password='1234',
                                                                           # Mostrar el gráfico
                                                                           plt.show()
         host='localhost',
         port='5432'
                                                                           # Cerrar la conexión
                                                                           cur.close()
       cur = conn.cursor()
                                                                           conn.close()
       # Nombre de la tabla que deseas consultar
       nombre tabla = "venta"
                                                                      elif op ==4:
       # Consulta SQL para obtener todos los datos de la tabla
       consulta sql = f"SELECT * FROM {nombre tabla};"
                                                                        conn = psycopg2.connect(
                                                                           dbname='ventas',
       # Ejecutar la consulta
                                                                           user='postgres',
       cur.execute(consulta sql)
                                                                           password='1234',
                                                                           host='localhost',
       # Obtener todos los resultados
                                                                           port='5432'
       resultados = cur.fetchall()
                                                                        cur = conn.cursor()
       # Mostrar los resultados
                                                                        # Consulta SQL para obtener datos de la tabla
                                                                        consulta sql = "SELECT * FROM venta;"
       for fila in resultados:
                                                                        df = pd.read sql query(consulta sql, conn)
         print(" id "," nombre "," cantidad ")
         print(" ",fila)
                                                                        # Convertir la columna 'cantidad' a tipo numérico y
                                                                 manejar NaN
       # Cerrar la conexión
                                                                        df['cantidad']
                                                                                                   pd.to_numeric(df['cantidad'],
       cur.close()
                                                                 errors='coerce')
       conn.close()
    elif op == 3:
                                                                        # Eliminar filas con valores nulos
                                                                        df.dropna(subset=['cantidad'], inplace=True)
         conn = psycopg2.connect(
                                                                        # Realizar análisis de datos (ejemplo: obtener
         dbname='ventas',
                                                                 estadísticas descriptivas)
                                                                        analisis descriptivo = df.describe()
         user='postgres',
         password='1234'.
         host='localhost',
                                                                        # Mostrar estadísticas descriptivas
         port='5432'
                                                                        print("Estadísticas Descriptivas:")
                                                                        print(analisis descriptivo)
         cur = conn.cursor()
                                                                        # Visualizar los datos (ejemplo: histograma)
                                                                        df['cantidad'].plot(kind='hist',
                                                                                                                       bins=10,
         # Consulta SQL para obtener datos de la tabla
                                                                 title="Histograma")
         consulta sql = "SELECT nombre, cantidad FROM
                                                                        plt.show()
```

```
# Cerrar la conexión
                                                                                   # Ejecutar la instrucción SQL
                                                                                   cur.execute(Instruccion)
       conn.close()
                                                                                   # Confirmar la transacción
                                                                                   conn.commit()
     elif op == 5:
        break
                                                                                   nombre sensor2= input("nombre : ")
                                                                                   dato2 = float(input("datos: "))
                                                                                 # Preparar la instrucción SQL para la inserción
                                                                                   Ins2 = 'INSERT INTO sensor2 (ptc,datos)
                                                                VALUES (%s,%s);' # %s es un marcador de posición para el
def opcion6():
                                                                valor
  while True:
                                                                                   Instruccion2
                                                                                                             cur.mogrify(Ins2,
                                                                (nombre sensor2,dato2))
     print("Menu de Opciones:")
     print("1. Ingresar datos de los senores")
                                                                                    # Ejecutar la instrucción SQL
     print("2. PTC")
    print("3. PT100")
                                                                                   cur.execute(Instruccion2)
    print("4. PT100")
    print("5. Gratica y Tabla ")
                                                                                   # Confirmar la transacción
     print("6. Salir")
                                                                                   conn.commit()
     op = int(input("ingres opcion : "))
                                                                                   nombre sensor3= input("nombre : ")
                                                                                   dato3 = float(input("datos: "))
     if op == 1:
         conn = psycopg2.connect(
                                                                                 # Preparar la instrucción SQL para la inserción
            dbname='sensores',
            user='postgres',
                                                                                   Ins3
                                                                                                'INSERT
                                                                                                             INTO
                                                                                                                      sensor3
            password='1234',
                                                                (pt100,datos) VALUES (%s,%s);' # %s es un marcador de
            host='localhost',
                                                                posición para el valor
            port='5432'
         cur = conn.cursor()
         nombre sensor= input("nombre : ")
                                                                                   Instruccion3
                                                                                                             cur.mogrify(Ins3,
         dato = float(input("datos : "))
                                                                (nombre sensor3,dato3))
         try:
                 # Preparar la instrucción SQL para la inserción
                                                                                   # Ejecutar la instrucción SQL
                   Ins1 = 'INSERT INTO sensor (ntc,datos)
                                                                                   cur.execute(Instruccion3)
VALUES (%s,%s);' # %s es un marcador de posición para el
valor
                                                                                   # Confirmar la transacción
                                                                                   conn.commit()
                                                                          except psycopg2.Error as e:
                   Instruccion
                                            cur.mogrify(Ins1,
                                                                                 print(f"Error durante la conexión a la DB.
(nombre sensor,dato))
                                                                Consulte el error: {e}")
```

```
finally:
                                                                              port='5432'
              # Cerrar el cursor y la conexión
                 cur.close()
                                                                         cur = conn.cursor()
                 conn.close()
                                                                         consulta sql = "SELECT * FROM sensor2;"
                                                                         df = pd.read sql query(consulta sql, conn)
     elif op == 2:
       conn = psycopg2.connect(
                                                                         # Convertir la columna 'cantidad' a tipo numérico y
            dbname='sensores',
                                                                 manejar NaN
            user='postgres',
                                                                         df['datos'] = pd.to numeric(df['datos'], errors='coerce')
            password='1234',
            host='localhost',
                                                                         # Eliminar filas con valores nulos
            port='5432'
                                                                         df.dropna(subset=['datos'], inplace=True)
                                                                         # Realizar análisis de datos (ejemplo: obtener
       cur = conn.cursor()
                                                                 estadísticas descriptivas)
       consulta sql = "SELECT * FROM sensor;"
                                                                         analisis descriptivo = df.describe()
       df = pd.read sql query(consulta sql, conn)
                                                                         # Mostrar estadísticas descriptivas
       # Convertir la columna 'cantidad' a tipo numérico y
                                                                         print("Estadísticas Descriptivas:")
manejar NaN
                                                                         print(analisis descriptivo)
       df['datos'] = pd.to numeric(df['datos'], errors='coerce')
                                                                         # Visualizar los datos (ejemplo: gráfico de dispersión)
                                                                         plt.scatter(df.index, df['datos'])
       # Eliminar filas con valores nulos
       df.dropna(subset=['datos'], inplace=True)
                                                                         plt.title("Gráfico de Dispersión")
                                                                         plt.xlabel("Índice de Datos")
       # Realizar análisis de datos (ejemplo: obtener
                                                                         plt.ylabel("Cantidad")
estadísticas descriptivas)
                                                                         plt.show()
       analisis descriptivo = df.describe()
                                                                         # Cerrar la conexión
       # Mostrar estadísticas descriptivas
                                                                         conn.close()
       print("Estadísticas Descriptivas:")
       print(analisis descriptivo)
                                                                      elif op ==4:
       # Visualizar los datos (ejemplo: gráfico de dispersión)
       plt.scatter(df.index, df['datos'])
                                                                         conn = psycopg2.connect(
       plt.title("Gráfico de Dispersión")
                                                                              dbname='sensores',
       plt.xlabel("Índice de Datos")
                                                                              user='postgres',
       plt.ylabel("Cantidad")
                                                                              password='1234',
       plt.show()
                                                                              host='localhost',
                                                                              port='5432'
                                                                         cur = conn.cursor()
                                                                         consulta sql = "SELECT * FROM sensor3;"
       # Mostrar los datos como una tabla utilizando prettytable
                                                                         df = pd.read sql query(consulta sql, conn)
                                                                         # Convertir la columna 'cantidad' a tipo numérico y
       # Cerrar la conexión
                                                                 manejar NaN
                                                                         df['datos'] = pd.to numeric(df['datos'], errors='coerce')
       conn.close()
                                                                         # Eliminar filas con valores nulos
     elif op == 3:
                                                                         df.dropna(subset=['datos'], inplace=True)
       conn = psycopg2.connect(
            dbname='sensores',
                                                                         # Realizar análisis de datos (ejemplo: obtener
            user='postgres',
                                                                 estadísticas descriptivas)
            password='1234',
                                                                         analisis descriptivo = df.describe()
            host='localhost',
```

```
# Mostrar estadísticas descriptivas
  print("Estadísticas Descriptivas:")
  print(analisis descriptivo)
                                                            def opcion7():
  # Visualizar los datos (ejemplo: gráfico de dispersión)
  plt.scatter(df.index, df['datos'])
                                                                 while True:
  plt.title("Gráfico de Dispersión")
  plt.xlabel("Índice de Datos")
                                                                   buscar = input("Buscar Gusto : ")
  plt.ylabel("Cantidad")
  plt.show()
                                                                   if buscar == "accion":
  # Cerrar la conexión
  conn.close()
                                                                      conn = psycopg2.connect(
                                                                        dbname='peliculas',
elif op == 5:
                                                                        user='postgres',
                                                                        password='1234',
                                                                        host='localhost',
  conn = psycopg2.connect(
                                                                        port='5432'
       dbname='sensores',
       user='postgres',
       password='1234',
                                                                      cur = conn.cursor()
                                                                      consulta sql = "SELECT * FROM pelicula;"
       host='localhost',
       port='5432'
                                                                      df = pd.read sql query(consulta sql, conn)
    )
  cur = conn.cursor()
  consulta sql = "SELECT * FROM sensor;"
  df = pd.read_sql_query(consulta_sql, conn)
                                                                      # Mostrar los datos como una tabla utilizando pandas
                                                                      print("Peliculas de accion:")
  consulta sql2 = "SELECT * FROM sensor2;"
                                                                      print(df)
  df2 = pd.read sql query(consulta sql2, conn)
  consulta sql3 = "SELECT * FROM sensor3;"
                                                                      # Cerrar la conexión
  df3 = pd.read sql query(consulta sql3, conn)
                                                                      conn.close()
  # Mostrar los datos como una tabla utilizando pandas
                                                                   elif buscar == "comedia":
  print("Datos de la Tabla NTC:")
  print(df)
                                                                      conn = psycopg2.connect(
  print("Datos de la Tabla PTC:")
                                                                        dbname='peliculas',
                                                                        user='postgres',
  print(df2)
  print("Datos de la Tabla PT100:")
                                                                        password='1234',
  print(df3)
                                                                        host='localhost',
                                                                        port='5432'
  # Cerrar la conexión
  conn.close()
                                                                      cur = conn.cursor()
                                                                      consulta sql = "SELECT * FROM pelicula3;"
                                                                      df = pd.read sql query(consulta sql, conn)
elif op == 6:
   break
                                                                      # Mostrar los datos como una tabla utilizando pandas
                                                                      print("peliculas de comedia:")
                                                                      print(df)
                                                                      # Cerrar la conexión
                                                                      conn.close()
                                                                   elif buscar == "miedo":
```

consulta sql = "SELECT * FROM empresa;"

```
dbname='peliculas',
                                                                         df = pd.read sql query(consulta sql, conn)
            user='postgres',
                                                                         # Convertir la columna 'cantidad' a tipo numérico y
            password='1234',
            host='localhost',
                                                                 manejar NaN
            port='5432'
                                                                         df['datos'] = pd.to numeric(df['datos'], errors='coerce')
                                                                         # Eliminar filas con valores nulos
          cur = conn.cursor()
          consulta sql = "SELECT * FROM pelicula2;"
                                                                         df.dropna(subset=['datos'], inplace=True)
          df = pd.read_sql_query(consulta_sql, conn)
                                                                         # Realizar análisis de datos (ejemplo: obtener
                                                                 estadísticas descriptivas)
                                                                         analisis descriptivo = df.describe()
          # Mostrar los datos como una tabla utilizando pandas
          print("peliculas de miedo:")
                                                                         # Mostrar estadísticas descriptivas
          print(df)
                                                                         print("Estadísticas Descriptivas:")
                                                                         print(analisis_descriptivo)
          # Cerrar la conexión
                                                                         # Visualizar los datos (ejemplo: gráfico de dispersión)
          conn.close()
                                                                         plt.scatter(df.index, df['datos'])
                                                                         plt.title("Gráfico de Dispersión")
                                                                         plt.xlabel("Índice de Datos")
                                                                         plt.ylabel("Cantidad")
       elif buscar == "salir":
                                                                         plt.show()
           break
                                                                         # Mostrar los datos como una tabla utilizando prettytable
def opcion8():
   while True:
                                                                         # Cerrar la conexión
                                                                         conn.close()
     print("Menu de Opciones:")
     print("1. Google")
                                                                      elif op == 2:
     print("2. Cementos Progresos")
                                                                         conn = psycopg2.connect(
     print("3. Intel")
                                                                              dbname='empresas',
     print("4. Gratica y Tabla ")
                                                                              user='postgres',
    print("5. Salir")
                                                                              password='1234',
                                                                              host='localhost',
                                                                              port='5432'
     op = int(input("ingres opcion : "))
                                                                           )
                                                                         cur = conn.cursor()
                                                                         consulta sql = "SELECT * FROM empresa2;"
                                                                         df = pd.read sql query(consulta sql, conn)
     if op == 1:
       conn = psycopg2.connect(
                                                                         # Convertir la columna 'cantidad' a tipo numérico y
            dbname='empresas',
                                                                 manejar NaN
                                                                         df['datos'] = pd.to_numeric(df['datos'], errors='coerce')
            user='postgres',
            password='1234',
            host='localhost',
                                                                         # Eliminar filas con valores nulos
            port='5432'
                                                                         df.dropna(subset=['datos'], inplace=True)
                                                                         # Realizar análisis de datos (ejemplo: obtener
       cur = conn.cursor()
                                                                 estadísticas descriptivas)
```

conn = psycopg2.connect(

```
conn = psycopg2.connect(
       # Mostrar estadísticas descriptivas
                                                                              dbname='empresas',
       print("Estadísticas Descriptivas:")
                                                                              user='postgres',
       print(analisis descriptivo)
                                                                              password='1234',
                                                                              host='localhost',
                                                                              port='5432'
       # Visualizar los datos (ejemplo: gráfico de dispersión)
       plt.scatter(df.index, df['datos'])
                                                                           )
       plt.title("Gráfico de Dispersión")
                                                                         cur = conn.cursor()
       plt.xlabel("Índice de Datos")
                                                                         consulta sql = "SELECT * FROM empresa;"
       plt.ylabel("Cantidad")
                                                                         df = pd.read_sql_query(consulta_sql, conn)
       plt.show()
                                                                         consulta sql2 = "SELECT * FROM empresa2;"
       # Cerrar la conexión
                                                                         df2 = pd.read sql query(consulta sql2, conn)
       conn.close()
                                                                         consulta sql3 = "SELECT * FROM empresa3;"
     elif op ==3:
                                                                         df3 = pd.read sql query(consulta sql3, conn)
                                                                         # Mostrar los datos como una tabla utilizando pandas
       conn = psycopg2.connect(
                                                                         print("Datos de la Tabla Google:")
            dbname='empresas',
                                                                         print(df)
            user='postgres',
                                                                         print("Datos de la Tabla Cementos progresos:")
            password='1234',
            host='localhost',
                                                                         print("Datos de la Tabla INTEL:")
            port='5432'
                                                                         print(df3)
                                                                         # Cerrar la conexión
       cur = conn.cursor()
                                                                         conn.close()
       consulta sql = "SELECT * FROM empresa3;"
       df = pd.read sql query(consulta sql, conn)
       # Convertir la columna 'cantidad' a tipo numérico y
                                                                      elif op == 5:
manejar NaN
       df['datos'] = pd.to_numeric(df['datos'], errors='coerce')
                                                                          break
                                                                 def opcion9():
       # Eliminar filas con valores nulos
       df.dropna(subset=['datos'], inplace=True)
       # Realizar análisis de datos (ejemplo: obtener
                                                                    while True:
estadísticas descriptivas)
       analisis descriptivo = df.describe()
                                                                      print("Menu de Opciones:")
                                                                      print("1. Ingresar productos")
                                                                      print("2. actualizar")
       # Mostrar estadísticas descriptivas
       print("Estadísticas Descriptivas:")
                                                                      print("3. vender")
       print(analisis descriptivo)
                                                                      print("4. Generar informe de ventas en tabla")
                                                                      print("5. Salir")
       # Visualizar los datos (ejemplo: gráfico de dispersión)
       plt.scatter(df.index, df['datos'])
       plt.title("Gráfico de Dispersión")
                                                                      op = int(input("ingres opcion "))
       plt.xlabel("Índice de Datos")
       plt.ylabel("Cantidad")
                                                                      if op == 1:
       plt.show()
                                                                         nombre = input("Ingrese su producto: ")
                                                                         precio = input("Ingrese su precio : ")
       # Cerrar la conexión
       conn.close()
                                                                         cantidad = input("ingrese su cantidad : ")
                                                                         conn = psycopg2.connect(
     elif op == 4:
                                                                              dbname='problema9',
```

analisis descriptivo = df.describe()

```
user='postgres',
                                                               precio: "))
            password='1234',
                                                                           nueva cantidad = int(input("Ingrese la nueva
           host='localhost',
                                                               cantidad: "))
           port='5432'
                                                                           # Sentencia SQL de actualización
                                                                           sql actualizacion = """
       cur = conn.cursor()
                                                                              UPDATE productos
                                                                              SET precio = %s, cantidad = %s
       try:
              # Preparar la instrucción SQL para la inserción
                                                                              WHERE nombre = %s;
                            'INSERT
                                         INTO
                                                   productos
(nombre,precio,cantidad) VALUES (%s,%s,%s);' # %s es un
marcador de posición para el valor
                                                                           # Ejecutar la actualización
                                                                           cur.execute(sql actualizacion,
                                                                                                              (nuevo precio,
                                                               nueva cantidad, nombre producto))
                Instruccion
                                            cur.mogrify(Ins1,
(nombre,precio,cantidad))
                                                                           # Confirmar la transacción
                                                                           conn.commit()
                # Ejecutar la instrucción SQL
                cur.execute(Instruccion)
                                                                           print(f"Se ha actualizado la información del
                                                               producto '{nombre producto}'.")
                # Confirmar la transacción
                conn.commit()
                                                                      except psycopg2.Error as e:
                                                                         print(f"Error durante la conexión a la DB. Consulte el
       except psycopg2. Error as e:
                                                               error: {e}")
              print(f"Error durante la conexión a la DB.
Consulte el error: {e}")
                                                                      finally:
                                                                         # Cerrar la conexión
       finally:
                                                                         cur.close()
           # Cerrar el cursor y la conexión
                                                                         conn.close()
              cur.close()
              conn.close()
    elif op == 2:
       conn = psycopg2.connect(
            dbname='problema9',
           user='postgres',
           password='1234',
            host='localhost',
           port='5432'
       cur = conn.cursor()
                                                                    elif op == 3:
                                                                      conn = psycopg2.connect(
       try:
         # Solicitar al usuario el nombre del producto
                                                                           dbname='problema9',
         nombre producto = input("Ingrese el nombre del
                                                                           user='postgres',
                                                                           password='1234',
producto que desea actualizar: ")
                                                                           host='localhost',
         # Verificar si el producto existe antes de continuar
                                                                           port='5432'
         cur.execute("SELECT COUNT(*) FROM productos
WHERE nombre = %s;", (nombre producto,))
                                                                      cur = conn.cursor()
         cantidad productos = cur.fetchone()[0]
                                                                      try:
         if cantidad productos == 0:
                                                                         # Solicitar al usuario el nombre del producto
           print(f"No se encontró ningún producto con el
                                                                         nombre producto = input("Ingrese el nombre del
nombre '{nombre producto}'.")
                                                               producto que desea vender: ")
         else:
            # Solicitar al usuario los nuevos valores
                                                                         # Verificar si el producto existe antes de continuar
            nuevo precio = float(input("Ingrese el nuevo
                                                                         cur.execute("SELECT COUNT(*) FROM productos
```

```
WHERE nombre = %s;", (nombre producto,))
                                                                      # Mostrar los datos como una tabla utilizando pandas
         cantidad productos = cur.fetchone()[0]
                                                                      print("Informes de ventas:")
                                                                      print(df)
         if cantidad productos == 0:
           print(f"No se encontró ningún producto con el
nombre '{nombre producto}'.")
                                                                      # Cerrar la conexión
                                                                      conn.close()
                                                                    elif op == 5:
           # Solicitar al usuario la cantidad que desea vender
            cantidad a vender = int(input("Ingrese la cantidad
que desea vender: "))
                                                                      break
           # Verificar si hay suficiente cantidad en el
                                                               def opcion10():
inventario
           cur.execute("SELECT cantidad FROM productos
WHERE nombre = %s;", (nombre producto,))
            cantidad actual = int(cur.fetchone()[0])
                                                                   while True:
           if cantidad a vender > cantidad actual:
              print(f"No hay suficiente cantidad en inventario
                                                                      buscar = input("Informacion del producto : ")
para vender {cantidad a vender} unidades.")
           else:
              # Actualizar la cantidad en el inventario
                                                                      if buscar == "block":
              nueva cantidad
                                =
                                       cantidad actual
cantidad a vender
              cur.execute("UPDATE productos SET cantidad
                                                                        conn = psycopg2.connect(
          WHERE nombre = %s;", (nueva cantidad,
                                                                           dbname='problema10',
                                                                           user='postgres',
nombre producto))
                                                                           password='1234',
                                                                           host='localhost',
              # Confirmar la transacción
                                                                           port='5432'
              conn.commit()
              print(f"Se han vendido {cantidad a vender}
                                                                         cur = conn.cursor()
unidades del producto '{nombre producto}'. Nuevo inventario:
{nueva cantidad} unidades.")
                                                                           # Consulta SQL para obtener la cantidad de arena
                                                                           consulta sql = "SELECT cantidad FROM
       except psycopg2. Error as e:
                                                               optimizar WHERE nombre = 'arena';"
         print(f"Error durante la conexión a la DB. Consulte el
error: {e}")
                                                                           # Ejecutar la consulta
                                                                           cur.execute(consulta sql)
                                                                           cantidad arena str = cur.fetchone()[0]
       finally:
         # Cerrar la conexión
         cur.close()
                                                                           # Convertir la cantidad de arena a entero
                                                                           cantidad arena = int(cantidad arena str)
         conn.close()
    elif op == 4:
                                                                           # Calcular la cantidad de bloques que se pueden
                                                               fabricar
       conn = psycopg2.connect(
                                                                           bloques disponibles = cantidad arena // 2 #
                                                               Usamos "//" para la división entera
            dbname='problema9',
           user='postgres',
            password='1234'.
                                                                           # Mostrar el resultado
                                                                           print(f"Con {cantidad arena} unidades de arena,
            host='localhost',
           port='5432'
                                                               puedes fabricar {bloques disponibles} bloques.")
       cur = conn.cursor()
                                                                        except psycopg2.Error as e:
       consulta sql = "SELECT * FROM productos;"
                                                                           print(f"Error durante la conexión a la base de datos.
                                                               Consulte el error: {e}")
       df = pd.read sql query(consulta sql, conn)
                                                                         finally:
```

```
# Cerrar el cursor y la conexión
                                                                              # Ejecutar la consulta
            cur.close()
                                                                              cur.execute(consulta sql)
            conn.close()
                                                                              # Obtener los resultados
                                                                              resultados = cur.fetchall()
       elif buscar == "salir":
                                                                              if resultados:
                                                                                for resultado in resultados:
                                                                                  print(f'Artista:
                                                                                                       {resultado[1]}\nCanción:
           break
def opcion11():
                                                                  {resultado[2]}\nLetra:\n{resultado[3]}\n")
   while True:
                                                                                print(f"No se encontraron resultados para el
     print("Menu de Opciones:")
                                                                 artista {artista buscar}.")
     print("1. Desplegar lista de canciones")
    print("2. Buscar por artita ")
                                                                           except psycopg2.Error as e:
    print("3. Buscar por cancion")
                                                                              print(f"Error durante la conexión a la base de datos.
     print("4. salir")
                                                                 Consulte el error: {e}")
                                                                           finally:
                                                                              # Cerrar el cursor y la conexión
     op = int(input("ingres opcion : "))
                                                                              cur.close()
                                                                              conn.close()
     if op == 1:
       conn = psycopg2.connect(
            dbname='canciones',
            user='postgres',
            password='1234',
            host='localhost',
            port='5432'
         )
       cur = conn.cursor()
       consulta sql = "SELECT * FROM tablacancion;"
       df = pd.read sql query(consulta sql, conn)
       # Mostrar los datos como una tabla utilizando pandas
                                                                      elif op == 3:
       print("Lista de canciones :")
                                                                           conn = psycopg2.connect(
       print(df)
                                                                                dbname='canciones',
                                                                                user='postgres',
                                                                                password='1234',
       # Cerrar la conexión
                                                                                host='localhost',
       conn.close()
     elif op == 2:
                                                                                port='5432'
         conn = psycopg2.connect(
                                                                           cur = conn.cursor()
              dbname='canciones',
                                                                           # Solicitar al usuario que ingrese el nombre de la
              user='postgres',
                                                                 canción
              password='1234',
              host='localhost',
              port='5432'
                                                                           cancion buscar = input('Ingrese el nombre de la
                                                                 canción: ')
         cur = conn.cursor()
         artista buscar = input('Ingrese el nombre del artista: ')
                                                                             # Consulta SQL para obtener información de la
                                                                 canción
                                                                             consulta sql = f"SELECT * FROM tablacancion
            # Consulta SQL para obtener información del
                                                                 WHERE cancion = '{cancion buscar}';"
artista
            consulta sql = f"SELECT * FROM tablacancion
WHERE artista = '{artista buscar}';"
                                                                             # Ejecutar la consulta
                                                                              cur.execute(consulta sql)
```

```
puntaje objetivo = 5
            # Obtener los resultados
            resultados = cur.fetchall()
                                                                         # Baraja aleatoriamente las preguntas
                                                                         random.shuffle(preguntas respuestas)
            if resultados:
                                                                         # Presenta preguntas hasta que el usuario alcance el
               for resultado in resultados:
                 print(f"Artista:
                                      {resultado[1]}\nCanción:
                                                                  puntaje objetivo
\{resultado[2]\}\nLetra:\n\{resultado[3]\}\n"\}
                                                                         for pregunta, respuesta in preguntas respuestas:
                                                                            print(f"Pregunta: {pregunta}")
            else:
               print(f"No se encontraron resultados para la
                                                                            respuesta usuario = input("Respuesta (Verdadero o
canción {cancion buscar}.")
                                                                  Falso): ").lower()
          except psycopg2.Error as e:
                                                                            if respuesta usuario == respuesta.lower():
            print(f"Error durante la conexión a la base de datos.
                                                                               print(" Ganaste 1 punto.")
Consulte el error: {e}")
                                                                               puntaje += 1
                                                                            else:
                                                                              print("Incorrecto. Perdistes puntos.")
          finally:
            # Cerrar el cursor y la conexión
            cur.close()
                                                                            print(f"puntaje actual: {puntaje}")
            conn.close()
                                                                            if puntaje == puntaje objetivo:
     elif op == 4:
                                                                              print(f"Felicidades,
                                                                                                                 puntaje
                                                                                                                               es
                                                                  {puntaje objetivo} puntos.")
       break
                                                                              break
def opcion12():
                                                                         # Cierra la conexión a la base de datos
                                                                         cur.close()
  while True:
                                                                         conn.close()
                                                                       elif op == 2:
     print("Menu de Opciones:")
                                                                          print(" ")
     print("1. Jugar")
     print("2. Instruciones ")
                                                                          print(" Las instrcciones son : \n se presentaran una serie
     print("3. Ver Preguntas")
                                                                  de pregunatas \n cada pregunta si aciertas sumaras un punto \n
                                                                  de lo contrario perderas 1 punto tienes 3 vidas")
    print("4. salir")
                                                                          print(" ")
                                                                       elif op == 3:
                                                                         conn = psycopg2.connect(
     op = int(input("ingres opcion : "))
                                                                                 dbname='problema12',
                                                                                 user='postgres',
                                                                                 password='1234',
     if op == 1:
                                                                                 host='localhost',
                                                                                 port='5432'
       conn = psycopg2.connect(
               dbname='problema12',
                                                                         cur = conn.cursor()
               user='postgres',
               password='1234',
                                                                         consulta sql = "SELECT * FROM juego;"
               host='localhost',
                                                                         df = pd.read sql query(consulta sql, conn)
               port='5432'
                                                                         # Mostrar los datos como una tabla utilizando pandas
                                                                         print("Lista de preguntas :")
       cur = conn.cursor()
                                                                         print(df)
       # Realiza una consulta para obtener preguntas y
                                                                       elif op == 4:
respuestas
       cur.execute("SELECT pregunta, respuesta FROM
                                                                          break
juego;")
       preguntas respuestas = cur.fetchall()
       # Inicializa el puntaje del usuario
       puntaje = 0
                                                                  while True:
```

exit()

```
print("Menu de Opciones:")
  print("1.Programa de registro de estudiantes")
  print("2.Programa de seguimiento de presupuesto personal
gastos")
  print("3.Programa de gestión de inventario")
  print("4.Programa de seguimiento de pedidos")
  print("5.Programa de monitoreo de ventas")
  print("6.Programa de análisis de datos de sensores")
  print("7.Sistema de recomendación de películas")
  print("8.Programa de análisis financiero")
  print("9.Sistema de gestión de inventario")
  print("10.Sistema de planificación de producción")
  print("11. Buscador de canciones")
  print("12. Concurso")
  print("13. Salir")
  selection = int(input("ingrese una option : "))
  if selection == 1:
     opcion1()
  elif seleccion == 2:
    opcion2()
  elif seleccion == 3:
     opcion3()
  elif seleccion == 4:
     opcion4()
  elif seleccion == 5:
     opcion5()
  elif seleccion == 6:
     opcion6()
  elif seleccion == 7:
     opcion7()
  elif seleccion == 8:
     opcion8()
  elif seleccion == 9:
     opcion9()
  elif seleccion == 10:
     opcion10()
  elif seleccion == 11:
     opcion11()
  elif seleccion == 12:
     opcion12()
  elif seleccion == 13:
```

C. Resultados imágenes

```
PS C:\Users\sergi\OneDrive\Desktop\examen parcial> & C:/Users/sergi/App
Menu de Opciones:
1.Programa de registro de estudiantes
2.Programa de seguimiento de presupuesto personal gastos
3. Programa de gestión de inventario
4. Programa de seguimiento de pedidos
5.Programa de monitoreo de ventas
6. Programa de análisis de datos de sensores
7. Sistema de recomendación de películas
8. Programa de análisis financiero
9. Sistema de gestión de inventario
10. Sistema de planificación de producción
11. Buscador de canciones
12. Concurso
13. Salir
ingrese una opcion : 1
Menu de Opciones:
1. Ingresar datos
2. Eliminar
3. Actualizar
4. Salir
ingres opcion 1
Ingrese su nombre : saul
Ingrese su edad : 45
ingrese su e¿genero : hombre
Ingrese su direccion: joyas
```

```
Ingrese su direccion: joyas
Menu de Opciones:
1. Ingresar datos
2. Eliminar
3. Actualizar
4. Salir
ingres opcion 4
Menu de Opciones:
1.Programa de registro de estudiantes
2.Programa de seguimiento de presupuesto personal gastos
3.Programa de gestión de inventario
4.Programa de seguimiento de pedidos
5. Programa de monitoreo de ventas
6.Programa de análisis de datos de sensores
7. Sistema de recomendación de películas
8.Programa de análisis financiero
9.Sistema de gestión de inventario
10.Sistema de planificación de producción
11. Buscador de canciones
12. Concurso
13. Salir
ingrese una opcion : 2
Menu de Opciones:
1. Ingresar gasto
2. ver resumen
3. Actualizar
4. Salir
ingres opcion 1
nombre del gasto : cemento
ingrese cantidad de gasto: 23
ingrese su presupuesto actual : 1234
Menu de Opciones:
1. Ingresar gasto
2. ver resumen
3. Actualizar
4. Salir
ingres opcion 4
Menu de Opciones:
1.Programa de registro de estudiantes
2.Programa de seguimiento de presupuesto personal gastos
3.Programa de gestión de inventario
4.Programa de seguimiento de pedidos
5.Programa de monitoreo de ventas
6.Programa de análisis de datos de sensores
7. Sistema de recomendación de películas
8.Programa de análisis financiero
9.Sistema de gestión de inventario
10.Sistema de planificación de producción
11. Buscador de canciones
12. Concurso
13. Salir
ingrese una opcion : 3
Menu de Opciones:
1. Ingresar producto
2. Eliminar
3. Actualizar
4. Salir
ingres opcion 1
Ingrese nombre del producto : carne
Ingrese el pprecio: 45
ingrese la cantidad : 67
 Menu de Opciones:

    Ingresar producto
    Eliminar
    Actualizar

 4. Salir
```

```
Menu de Opciones:
1. Ingresar producto
2. Eliminar
3. Actualizar
4. Salir
ingres opcion 4
Menu de Opciones:
1. Programa de registro de estudiantes
2. Programa de seguimiento de presupuesto personal gastos
3. Programa de seguimiento de periodico
3. Programa de seguimiento de pedidos
5. Programa de monitoreo de ventas
6. Programa de análisis de datos de sensores
7. Sistema de recomendación de películas
8. Programa de análisis financiero
9. Sistema de gestión de inventario
10. Sistema de planificación de producción
11. Buscador de canciones
12. Concurso
13. Salir
ingrese una opcion : 4
Menu de Opciones:
1. Ingrese and para de sensores
2. Eliminar
3. Actualizar
4. Salir
ingrese opcion 1
Ingrese el nombre del solicitante : heli
Ingrese nombre del pedido: 1tx
ingrese el precio del pedido: 22
```

```
4.Programa de seguimiento de pedidos
5.Programa de monitoreo de ventas
                   6.Programa de análisis de datos de sensores
                7.Sistema de recomendación de películas
8.Programa de análisis financiero
                9.Sistema de gestión de inventario
10.Sistema de planificación de producción
                   11. Buscador de canciones
                   13. Salir
                ingrese una opcion : 5
Menu de Opciones:

    Ingresar venta
    Informe

                3. Tendencia
                4. Analisis
5. Salir
                ingres opcion : 1
nombre venta : carner
ingrese cantidad : 3
                   Menu de Opciones:
                   1. Ingresar venta
                   2. Informe
                   3. Tendencia
                4. Analisis
                5. Salir
ingres opcion : □
             PROBLEMS OUTPUT DEBUCCONSCI.

6. Programa de análisis de datos de .7.5istema de recomendación de palíci.

8. Programa de análisis financiero 9.5istema de gestión de inventario 10.5istema de gestión de inventario 11. Buscador de canciones 12. Concurso 13. Salir larguese una opcion : 6 | Memor de Opciones: 10. Programa debos de los senores 2. PTC 5. PT
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 Gráfico de Dispersión
                                                                                                                                                                                                                                                                                                                                                                           50
4. PTR00
4. PTR00
5. Gratica y Tabla
6. Salita
6. Salita y Tabla
6. Salita y Tabla
6. Salita y Tabla
6. Salita y Tabla
7. Salita y Tabla
7. Salita y Tabla
8. Salita y Tabla
8
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            2 3
indice de Datos
                                                                                                                                                                                                                                                                                                                             ☆←→ + Q = □
13. Salir
ingrese una opcion : 7
Buscar Gusto : accion
Cilbuers\sengi\(\text{One}\) involved:
Users\sengi\(\text{One}\) involved:
ase consider using \(\text{SQLALchemy}\).

df = pd.read.sql_query\(\text{consulta}\)_sql, conn)
Peliculas de accion:

peliculas de accion:

                                          Culas de accione

de la monomore accion el padrino accion el padrino accion el padrino accion 3 El Padrino 3 accion el padrino accion el padrino el padrin
                   5.Programa de monitoreo de venta
6.Programa de análisis de datos
7.Sistema de recomendación de pe
8.Programa de análisis financier
9.Sistema de gestión de inventar
10.Sistema de planificación de p
                       9.Sistema de gestion de
10.Sistema de planifica
11. Buscador de cancion
12. Concurso
13. Salir
ingrese una opcion : 8
Menu de Opciones:
1. Google
2. Cementos Progresos
3. Intel
4. Gratica y Tabla
5. Salir
                                                                                                                                                                                                                                                                                                                             80
                                                                                                                                                                                                                                                                                               Cantidad
                1d datos
count 17.000000 17.000000
mean 9.000000 51.000000
std 5.049752 30.242768
min 1.000000 23.000000
25% 5.000000 23.000000
50% 9.000000 47.000000
75% 13.000000 78.000000
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           10
                                                                                                                                                                                                                                                                                  # ← → | + Q ± | B
```

Menu de Opciones:

1.Programa de registro de estudiantes

2.Programa de seguimiento de presupuesto personal gastos 3.Programa de gestión de inventario

```
1.Programa de registro de estudiantes
           2.Programa de seguimiento de presupuesto personal gastos
           3. Programa de gestión de inventario
            4.Programa de seguimiento de pedidos
          5. Programa de monitoreo de ventas
          6.Programa de análisis de datos de sensores
7.Sistema de recomendación de películas
           8. Programa de análisis financiero
           9. Sistema de gestión de inventario
          10.Sistema de planificación de producción
11. Buscador de canciones
          12. Concurso
13. Salir
          ingrese una opcion : 9
Menu de Opciones:
          1. Ingresar productos
           2. actualizar
          3. vender
            4. Generar informe de ventas en tabla
          5. Salir
            ingres opcion 1
          Ingrese su producto : pollo
Ingrese su precio : 34
           ingrese su cantidad : 23
           Menu de Opciones:
          1. Ingresar productos

    actualizar
    vender

          4. Generar informe de ventas en tabla
          5. Salir
          ingres opcion [
         A. Generar informe de ventas en tabla
5. Salir
ingrese sopcion 1
Ingrese su producto : pollo
Ingrese su precio : 34
ingrese su cantidad : 23
Menu de Orciones:
          Menu de Opciones:
         1. Ingresar productos
2. actualizar
3. vender
4. Generar informe de ventas en tabla
          5. Salir
         ingres opcion 5
        ingres opcion 5
Menu de Opciones:

1.Programa de registro de estudiantes
2.Programa de seguimiento de presupuesto personal gastos
3.Programa de gestión de inventario
4.Programa de seguimiento de pedidos
5.Programa de monitoreo de ventas
6.Programa de análisis de datos de sensores
7.Sistema de recomendación de películas
8.Programa de análisis financiero
9.Sistema de gestión de inventario
10.Sistema de planificación de producción
11. Buscador de canciones
12. Concursos
          12. Concurso
13. Salir
         ingrese una opcion : 10
Informacion del producto : block
Con 12 unidades de arena, puedes fabricar 6 bloques.
Informacion del producto : []
1. Desplegar lists de canciones
2. Buscar por artita
3. Buscar por carcion
4. salir
Ingres opcion: 1
c:VUsers\sergil\noePrive\Desktop\examen parcial\main.py:1496: UserWarning: pandas only supports SQLAlchemy co
ase consider using SQLAlchemy.

df = pd.read.sql_query(consulta_sql, conn)
Lista de canciones:
    id artista
0 1 Ricardo Arjona anor lettra de cancion
1 2 Ricardo Arjona Hi novia se me esta poniendo vieja lettra de cancion
2 3 Ricardo Arjona Fie quiero lettra de cancion
3 4 Ricardo Arjona Si tu no existienas lettra de cancion
4 5 Ricardo Arjona Si tu no existienas lettra de cancion
5 6 Alejandro Sanz
6 7 Alejandro Sanz
7 8 Alejandro Sanz
8 9 Alejandro Sanz
9 10 Alejandro Sanz
9 10 Alejandro Sanz
10 Lettra de cancion
11 12 Laura Pausini
11 12 Laura Pausini
12 Laura Pausini
13 14 Laura Pausini
14 Laura Pausini
15 Lesulearu Jista de cancion
16 Nenu de Opciones:
1 Nesolearu Jista de cancions
16 Nenu de Opciones:
1 Nesolearu Jista de cancions
   1. Desplegar lista de canciones
            de Opciones:
   Menu de Opciones:

1. Desplegar lista de canciones

2. Buscar por artita

3. Buscar por cancion
```

4. salir ingres opcion : [

```
13 14 Laura Pausini
14 15 Laura Pausini
Menu de Opciones:
1. Desplegan lista de canciones
2. Buscar por artita
3. Buscar por cancion
4. salír
                                                                                                                                                                       Sino lettra de cancion
                                                                                                                                                                                  Yo lettra de cancion
  4. salir
 ingres opcion : 4
Menu de Opciones:
Menu de Ópciones:

1.Programa de registro de estudiantes

2.Programa de seguimiento de presupuesto personal gastos

3.Programa de seguimiento de presupuesto personal gastos

3.Programa de seguimiento de pedidos

5.Programa de monitoreo de ventas

6.Programa de análisis de datos de sensores

7.Sistema de recomendación de películas

8.Programa de análisis financiero

9.Sistema de gastión de inventario

10.Sistema de planificación de producción

11. Buscador de canciones

12. Concursos
  12. Concurso
13. Salir
 ingrese una opcion : 12
Menu de Opciones:
1. Jugar
2. Instruciones
  3. Ver Preguntas
  4. salir
  ingres opcion : 1
Pregunta: El mundo es redondo ?
Respuesta (Verdadero o Falso):

    Jugar
    Instruciones

 3. Ver Preguntas
4. salir
ingres opcion : 1
 Pregunta: El mar es salado?
Respuesta (Verdadero o Falso): verdadero
Ganaste 1 punto.
  puntaje actual: 1
 Pregunta: las plantas necesitan agua?
Respuesta (Verdadero o Falso): falso
Incorrecto. Perdistes puntos.
puntaje actual: 1
 puntaje actual: 1
Pregunta: El mundo es redondo ?
Respuesta (Verdadero o Falso): falso
Incorrecto. Perdistes puntos.
puntaje actual: 1
puntaje actual: 1
Pregunta: los gatos comen ratones?
Respuesta (Verdadero o Falso): falso
Incorrecto. Perdistes puntos.
puntaje actual: 1
Pregunta: los pericos vuelan?
Respuesta (Verdadero o Falso): falso
Incorrecto. Perdistes puntos.
puntaje actual: 1
Menu de Orciones:
   Menu de Opciones:

    Jugar
    Instruciones
    Ver Preguntas
  4. salir
 ingres opcion : [
```

Nombre: Heli Saul Vasquez Gomez

Carne: 201700852 CU1: 2852625480106

Proyectos Apicados Serie I

pregonta 2	pregunata 14	pregon +9 26
ell a) save()	p11 a) mean()	a) CREATE TABLE
Pregunta 4 Pll a) Sgrt()	pregenta 16 Pl/a) round()	Pregenta 28 C) INSERT MTO
pregunta 6 pll a) round()	pregenta 18	Pregenta 30 PI) a) UPDATE
pregenta 8 pl) a) mean()	pregunta 20	pregenta 32 P// c) MITZICENSE
pregunta 10	P// a) save() pregunta 22	pregunta 34
pll b) xcom()	P// c) \$592	a) CREATE DATABASE
Pregunta 12 P// a/ 5,2e()	G) p// 11	pregon 1a 36 p/l b)

Escaneado con Cam

Pregenta 38 P// a) Pregenta 40 Pregenta 42 PP// a) Pregenta 42 PP// a) Pregenta 44 P// d) Pregenta 48 P// b) Pregenta 48 P// a) Pregenta 48 P// a) Pregenta 48 P// a) Pregenta 50 d/ d.)

Serie 2

Pregenta 2	
	pregonta 12
P// Es un campo o coppinto de	-
Campos que posce dos	#// se there dos matrices
propie dodes.	AYC
NOTE OF THE PARTY	A = [1,3]
pregonta 4	C = [3, 9]
R/l se utiliza	moitipica = A+C
DETETE FROM nombedela taba;	
pregunta 6	Pregonta 14
ell se utiliza para filitar los	fll se otiliza ol Comardo mean ()
resultados de una Consulta.	
pregunta 8	pregenta 16
regenter	Pl/ se calala
ploced lawriu alder and bosed	conel comando
en les resultados de una	sart()
Consulta.	pregenta 18
pregunta 10	F// se sealiza otilizando
Ell Eson bloque de codigo	el romando maxi)
que realiza una torea	pregenta 20
	P// (ore) romando rand()

Escaneado con Cam

```
]

||f(exist('OCTAVE_VERSION', 'builtin') -=0)
||pkg load sigmal;
-end
||opeion = 0;
||while opeion -=5|
||disp('1.Grabar')
||disp('2.Graficar')
||disp('3.Graficar')
||disp('4.Graficar densidad')
||disp('5.Salir')
||opeion = input('Ingrese su elección: ');
||awitch opeion
opcion = input('Ingrese su elección: ');
]switch opcion
case !
% Grabación de audio
]!ry
duracion = input('Ingrese la duración de la grabación en segundos: ');
disp('Comenzando la grabación...');
recOb) = audiorecorder;
recordblocking(recOb), duracion);
disp('Grabación finalizada.');
data = getaudiodata(recOb);
audiowrite ('audio.waw', data, recOb).SampleRate);
disp('Archivo de audio grabado correctamente.');
catch
  disp('Error al grabar el audio.');
  end
case 2
 ltry
[data, fs] = audioread('audio.wav');
    sound(data, fs);
catch
     disp('Error al reproducir el audio.');
 ]try
[data, fs] = audioread('audio.wav');
 [cata, fs] = audioread('audio.wav');
tiempo = linspace(0, length(data)/fs, length(data));
plot(tiempo, data);
xlabel('Tiempo (s)');
ylabel('Amplitud');
tite('Audio');
catch
 disp('Error al graficar el audio.');
 case 4
% Graficando espectro de frecuencia
Tiempo (s)
```

```
>> tarea5
1.Grabar
2.Reproducir
3.Graficar
4.Graficar
4.Graficar
4.Graficar
1.Ingrese su elección: 1
1.Ingrese su elección: 1
1.Ingrese la duración de la grabación en segundos: 10
Comenzando la grabación...
Grabación finalizada.
Archivo de audio grabado correctamente.
1.Grabar
2.Reproducir
3.Graficar
4.Graficar densidad
5.Salir
1.Ingrese su elección: 2
1.Grabar
2.Reproducir
3.Graficar
4.Graficar
4.Graficar
5.Salir
Ingrese su elección: 1
1.Grabar
2.Reproducir
3.Graficar
4.Graficar
4.Graficar densidad
5.Salir
Ingrese su elección: 1
```

conclusiones

D.

- Se logro realizar todos los programas en el lenguaje python
- 2. Se logro ingresar toda la información en la base de datos postgres Pgadmin
- 3. Se ejecutaron de manera correcta todos los programas

E. Código

https://github.com/helisaul/Proyec com apli AIE.git