



Ingeniería Electrónica
Proyectos Computacionales Aplicados a Ingeniería Electrónica
Ing. José Anibal Silva de Los Angeles
Aux. Fernando Mardoqueo Paxtor Sam

CARNÉ:	201807371	FECHA:	24/02/2022
NOMBRE:	Cristian Noé González Márquez		
REGISTRO ADMIN.			

Primer Parcial

Serie II

Ejercicio 1)

```
import psycopg2

while True:

    try:

        conexion = psycopg2.connect(
            host = "localhost",
            port = "5432",
            user = "postgres",
            password = "12345abc",
            dbname = "Tarea_Preparatoria"
        )

        print("""
        1) Abrir programa
        2) Ver historial""")

        nn1 = int(input("Seleccione una opcion: "))

        if nn1==1:
            print("Introduzca su fecha de nacimiento")
            print("Ejemplo:  Dia: 5   Mes: 2   Año: 2000")

            dia1 = int(input("Dia: "))
            mes1 = int(input("Mes: "))
            year1 = int(input("Año: "))
```

```

        dia2 = 24 #coloque la fecha de hoy
        mes2 = 2  # pero se puede modificar para recibir la fecha de
nacimiento y la fecha en la que se esta haciendo la consulta
        year2 = 2022

    Edad = year2-year1

    x1=0
    x2=0

    if mes1<mes2:
        print("Esta persona ya cumplio años")
        x1=Edad
        print("Esta persona tiene ", x1, " años")
        x2 = "Esta pesona ya cumplio años"

    elif (mes1==mes2) & (dia1<=dia2):
        print("Esta persona ya cumplio años")
        x1=Edad
        print("Esta persona tiene ", x1," años")
        x2 = "Esta persona ya cumplio años"

    elif (mes1>mes2):
        print("Esta persona aun no cumple años")
        x1=Edad-1
        print("Esta persona tiene ",x1, " años")
        x2 = "Esta persona aun no cumple años"
    elif (mes1==mes2)&(dia1>dia2):
        print("Esta persona aun no cumple años")
        x1=Edad-1
        print("Esta persona tiene ", x1," años")
        x2 = "Esta persona aun no cumple años"

    cursor = conexion.cursor()
    cursor.execute("INSERT INTO ejer1(dia,mes,ano,edad,celebro)
VALUES(%s,%s,%s,%s,%s);",(dia1,mes1,year1,x1,x2))
    conexion.commit()
    cursor.close()
    conexion.close()

    elif nn1==2:
        cursor = conexion.cursor()
        SQL = "SELECT*FROM ejer1;"

```

```

        cursor.execute(SQL)
        valores = cursor.fetchall()
        print(valores)
        cursor.close()
        conexion.close()
except:
    print("Error: Revise la informacion que ha proporcionado")

```

Resultado:

```

1) Abrir programa
2) Ver historial
Seleccione una opcion: 1
Introduzca su fecha de nacimiento
Ejemplo: Dia: 5 Mes: 2 Año: 2000
Dia: 23
Mes: 2
Año: 2002
Esta persona ya cumple años
Esta persona tiene 20 años

1) Abrir programa
2) Ver historial
Seleccione una opcion: 26

1) Abrir programa
2) Ver historial
Seleccione una opcion: 1
Introduzca su fecha de nacimiento
Ejemplo: Dia: 5 Mes: 2 Año: 2000
Dia: 26
Mes: 2
Año: 1995
Esta persona aun no cumple años
Esta persona tiene 26 años

1) Abrir programa
2) Ver historial
Seleccione una opcion: 2
[('15', '11', '1998', '24', 'Esta persona aun no cumple años'), ('15', '11', '1998', '23', 'Esta persona aun no cumple años'), ('23', '2', '2002', '20', 'Esta persona ya cumple años'), ('26', '2', '1995', '26', 'Esta persona aun no cumple años')]

1) Abrir programa

```

Base de Datos:

```
--create table ejer1(
    --dia varchar(20),
    --mes varchar(20),
    --ano varchar(20),
    --edad varchar(20),
    --celebro varchar(50))

select * from ejer1;
```

Tarea_Preparatoria/postgres@PostgreSQL 12 ▾					
Data Output					
	dia character varying (20)	mes character varying (20)	ano character varying (20)	edad character varying (20)	celebro character varying (50)
1	15	11	1998	24	Esta persona aun no cumple años
2	15	11	1998	23	Esta persona aun no cumple años
3	23	2	2002	20	Esta persona ya cumplio años
4	26	2	1995	26	Esta persona aun no cumple años

Ejercicio 2:

```
import psycopg2

while True:
    try:

        conexion = psycopg2.connect(
            host = "localhost",
            port = "5432",
            user = "postgres",
            password = "12345abc",
            dbname = "Tarea_Preparatoria"
        )

        print("""
        1) Abrir programa
        2) Ver el historial""")

        nn1 = int(input("Seleccione una opcion: "))

        if nn1==1:
```

```

a = int(input("Primer angulo: "))
b = int(input("Segundo angulo: "))

c = 180-a-b
print("El angulo es: ",c)

cursor = conexion.cursor()
cursor.execute("INSERT INTO
ejer2(primer_angulo,segundo_angulo,angulo_resultante)
VALUES(%s,%s,%s);",(a,b,c))
conexion.commit()
cursor.close()
conexion.close()

elif nn1==2:
    cursor = conexion.cursor()
    SQL = "SELECT*FROM ejer2;"
    cursor.execute(SQL)
    valores = cursor.fetchall()
    print(valores)
    cursor.close()
    conexion.close()
except:
    print("Error: Posiblemente introdujo un caracter no numerico")

```

Resultado:

```

    1) Abrir programa
    2) Ver el historial
Seleccione una opcion: 1
Primer angulo: 60
Segundo angulo: 30
El angulo es: 90

    1) Abrir programa
    2) Ver el historial
Seleccione una opcion: 1
Primer angulo: 95
Segundo angulo: 25
El angulo es: 60

    1) Abrir programa
    2) Ver el historial
Seleccione una opcion: 90

    1) Abrir programa
    2) Ver el historial
Seleccione una opcion: 1
Primer angulo: 45
Segundo angulo: 90
El angulo es: 45

    1) Abrir programa
    2) Ver el historial
Seleccione una opcion: 2
[('60', '30', '90'), ('95', '25', '60'), ('45', '90', '45')]

    1) Abrir programa
    2) Ver el historial
Seleccione una opcion: █

```

Base de datos:

```

--create table ejer2(
--  primer_angulo varchar(20),
--  segundo_angulo varchar(20),
--  angulo_resultante varchar(20))

select * from ejer2;|

```

	primer_angulo character varying (20)	segundo_angulo character varying (20)	angulo_resultante character varying (20)
1	60	30	90
2	95	25	60
3	45	90	45

Ejercicio 3:

#Realice un programa que ingrese Un número entre 1 y 999 e indique por cuantas unidades,
decenas y centenas está formado el número.

```
import psycopg2

while True:

    try:
        conexion = psycopg2.connect(
            host = "localhost",
            port = "5432",
            user = "postgres",
            password = "12345abc",
            dbname = "Tarea_Preparatoria"
        )

        print("""
        1) Ejecutar Programa
        2) Ver el historial""")

        nn1 = int(input("Seleccione un opcion: "))

        if nn1==1:
            numero = int(input ("Ingrese un valor entre 1 y 999  "))
```

```

        centenas=(numero%1000-numero%100)//100
        decenas=(numero%100-numero%10)//10
        unidades=numero%10
        print ("Valor de centenas: " , centenas)
        print ("Valor de decenas: " , decenas)
        print ("Valor de unidades: ", unidades)
        print ()

        cursor = conexion.cursor()
        cursor.execute("INSERT INTO
ejer3(numero,centenas,decenas,unidades)
VALUES(%s,%s,%s,%s);",(numero,centenas,decenas,unidades))
        conexion.commit()
        cursor.close()
        conexion.close()

    elif nn1==2:
        cursor = conexion.cursor()
        SQL = "SELECT*FROM ejer3;"
        cursor.execute(SQL)
        valores = cursor.fetchall()
        print(valores)
        cursor.close()
        conexion.close()
except:
    print("Error: Posiblemete introdujo un caracter no numerico")

```

Resultado:


```

Ingrese un valor entre 1 y 999    23
Valor de centenas:  0
Valor de decenas:   2
Valor de unidades:  3

1) Ejecutar Programa
2) Ver el historial
Seleccione un opcion: 1
Ingrese un valor entre 1 y 999    872
Valor de centenas:  8
Valor de decenas:   7
Valor de unidades:  2

1) Ejecutar Programa
2) Ver el historial
Seleccione un opcion: 59

1) Ejecutar Programa
2) Ver el historial
Seleccione un opcion: 998

1) Ejecutar Programa
2) Ver el historial
Seleccione un opcion: 2
[('478', '4', '7', '8'), ('23', '0', '2', '3'), ('872', '8', '7', '2')]

1) Ejecutar Programa
2) Ver el historial
Seleccione un opcion: █

```

Base de datos:

```

-- create table ejer3(
--     numero varchar(20),
--     centenas varchar(20),
--     decenas varchar(20),
--     unidades varchar(20))

select * from ejer3;

```

Tarea_Preparatoria/postgres@PostgreSQL 12 ▾				
Data Output				
	numero character varying (20)	centenas character varying (20)	decenas character varying (20)	unidades character varying (20)
1	478	4	7	8
2	23	0	2	3
3	872	8	7	2

Ejercicio 4:

La Reglas del juego. Debes lanzar un par de dados. Si la suma de las
caras es un 8, ganas. Si sale 7, pierdes. Si no hasalido,
ni 8, ni 7, puedes seguir lanzando. Si sale 8 ganas, pero
si en algún otro lanzamiento sale 7, pierdes.

```
import psycopg2
import random

while True:
    try:
        conexion = psycopg2.connect(
            host = "localhost",
            port = "5432",
            user = "postgres",
            password = "12345abc",
            dbname = "Tarea_Preparatoria"
        )

        print("""
        1) Jugar
        2) Ver historial""")

        nn1 =int(input("Seleccione un opcion: "))

        if nn1==1:
            x = random.randrange(6)
            y = random.randrange(6)

            z = x+y
```

```

        w=0
        pulsador = input("Presione la letra j y luego enter para tira
los dados: ")
        if pulsador=="j":
            if z==8:
                print("Dado 1: ",x)
                print("Dado 2: ",y)
                print("Felicidades a gando el juego obtuvo un: ",z)

            elif z==7:
                print("Dado 1: ",x )
                print("Dado 2: ",y)
                print("Has perdido el juego")
            else:
                print("Dado 1: ",x)
                print("Dado 2: ",y)
                print("Vuelva a jugar")

        cursor = conexion.cursor()
        cursor.execute("INSERT INTO ejer4(dado1,dado2,resultado)
VALUES(%s,%s,%s);",(x,y,z))
        conexion.commit()
        cursor.close()
        conexion.close()

    if nn1==2:
        cursor = conexion.cursor()
        SQL = "SELECT*FROM ejer4;"
        cursor.execute(SQL)
        valores = cursor.fetchall()
        print(valores)
        cursor.close()
        conexion.close()
except:
    print("Se produjo un error, se recomienda reiniciar el juego")

```

Resultado:

```

        2) Ver historial
Selecione un opcion: 1
Presione la letra j y luego enter para tira los dados: j
Dado 1: 1
Dado 2: 2
Vuelva a jugar

        1) Jugar
        2) Ver historial
Selecione un opcion: 1
Presione la letra j y luego enter para tira los dados: j
Dado 1: 0
Dado 2: 3
Vuelva a jugar

        1) Jugar
        2) Ver historial
Selecione un opcion: 1
Presione la letra j y luego enter para tira los dados: j
Dado 1: 3
Dado 2: 1
Vuelva a jugar

        1) Jugar
        2) Ver historial
Selecione un opcion: █

```




Base de datos:

```

32
33 --create table ejer4(
34 --  dado1 varchar(20),
35 --  dado2 varchar(20),
36 --  resultado varchar(20))
37
38 select * from ejer4;
39
--

```

Data Output

	 dado1 character varying (20)	 dado2 character varying (20)	 resultado character varying (20)	
1	2	2	4	
2	3	4	7	
3	1	2	3	
4	0	3	3	
5	3	1	4	

Ejercicio 1:

```
1  clc;clear;
2
3  try
4      pkg load database
5      conn = pq_connect(setdbopts('dbname','Tarea_Preparatoria','host','localhost','port','5432','user','postgres','password','12345abc'));
6
7      fprintf('Introduzca su edad \n');
8      fprintf('Ejemplo : Dia: 15 Mes: 14 Año: 1998 \n ');
9      dial = input('Ingrese el dia de su cumpleaños: ');
10     mes1 = input('Ingrese el numero de su mes de nacimiento: ');
11     year1 = input('Ingrese el año en que nacio: ');
12     printf('Usted nacio el dia %i del mes %i y año %i \n', dial, mes1, year1);
13     edad=2022-year1;
14
15     if (mes1<2)
16         printf('Usted ya cumplio %i años este año', edad);
17         concat= strcat(' Usted nacio el dia ', num2str(dial), ' del mes ', num2str(mes1), ' y año ',num2str(year1));
18         concat1= strcat(' Ya cumplio ', num2str(edad), ' años');
19         N = pq_exec_params(conn,"insert into ejer1(dia, mes,ano,edad,celebro) values ($1,$2,$3,$4,$5);",(dial,mes1,year1,edad,'Ya cumplio'));
20     elseif (mes1==2)&(dia<=24)
21         printf('Usted ya cumplio %i años este año', edad);
22         concat= strcat(' Usted nacio el dia ', num2str(dial), ' del mes ', num2str(mes1), ' y año ',num2str(year1));
23         concat1= strcat(' Ya cumplio ', num2str(edad), ' años');
24         N = pq_exec_params(conn,"insert into ejer1(dia, mes,ano,edad,celebro) values ($1,$2,$3,$4,$5);",(dial,mes1,year1,edad,'Ya cumplio'));
25
26     elseif (mes1>2)
27         printf('Usted ya no ha cumplido %i años', edad);
28         concat= strcat(' Usted nacio el dia ', num2str(dial), ' del mes ', num2str(mes1), ' y año ',num2str(year1));
29
30     N = pq_exec_params(conn,"insert into ejer1(dia, mes,ano,edad,celebro) values ($1,$2,$3,$4,$5);",(dial,mes1,year1,edad,'Ya cumplio'));
31
32     elseif (mes1==2)&(dia>24)
33         printf('Usted ya no ha cumplido %i años', edad);
34         concat= strcat(' Usted nacio el dia ', num2str(dial), ' del mes ', num2str(mes1), ' y año ',num2str(year1));
35         concat1= strcat(' Ya cumplio ', num2str(edad), ' años');
36         N = pq_exec_params(conn,"insert into ejer1(dia, mes,ano,edad,celebro) values ($1,$2,$3,$4,$5);",(dial,mes1,year1,edad,'Ya no ha cumplido
37
38     endif
39
40
41
42
43
44     catch
45         printf("Ha ocurrido un error")
46     end_try_catch
```

Resultado:

```
Introduzca su edad
Ejemplo : Dia: 15 Mes: 14 Año: 1998
Ingrese el dia de su cumpleaños: 5
Ingrese el numero de su mes de nacimiento: 1
Ingrese el año en que nacio: 1991
Usted nacio el dia 5 del mes 1 y año 1991
Usted ya cumplio 31 años >>
```

Base de datos:

Data Output					
	dia character varying (20)	mes character varying (20)	ano character varying (20)	edad character varying (20)	celebro character varying (50)
1	15	11	1998	24	Esta persona aun no cumple años
2	15	11	1998	23	Esta persona aun no cumple años
3	23	2	2002	20	Esta persona ya cumplio años
4	26	2	1995	26	Esta persona aun no cumple años
5	15	5	2066	-45	Esta persona aun no cumple años
6	12	1	2001	21	Ya cumplio
7	15	7	1995	27	Ya no ha cumplido
8	19	1	2013	9	Ya cumplio
9	5	1	1991	31	Ya cumplio

Ejercicio 2:

```
1  clr;clear;
2
3  try
4      fprintf('Determinar el angulo restante de un triangulo \n');
5      a = input('Ingrese el primer angulo: ');
6      b = input('Ingrese el segundo angulo: ');
7      c = 180-a-b;
8      pkg load database
9      conn = pg_connect(setdbopts('dbname','Tarea_Preparatoria','host','localhost','port','5432','user','postgres','password','12345abc'));
10
11      printf('El angulo es igual a %i \n', c);
12      concat=strcmp('El angulo flatante es c=', num2str(c));
13      N = pq_exec_params(conn,"insert into ejer2(primer_angulo, segundo_angulo, angulo_resultante) values($1,$2,$3);",(a,b,c));
14
15  catch
16      printf("Ha ocurrido un error \n");
17  end_try_catch
```

Resultado:

```
× Ventana de comandos
Determinar el angulo restante de un triangulo
Ingrese el primer angulo: 45
Ingrese el segundo angulo: 66
El angulo es igual a 69
>> |
```

Ejercicio 3:

```
1 clc;clear;
2
3 try
4     fprintf('Determinar cuantas unidades, decenas y centenas tiene el numero \n');
5     nl = input('Ingrese un numero entre 0 y 999: ');
6
7
8     pkg load database
9     conn = pq_connect(setdbopts('dbname','Tarea_Preparatoria','host','localhost','port','5432','user','postgres','password','12345abc'))
10
11     if (nl<=999)&&(nl>=0)
12         c=fix(nl/100);
13         d=fix((nl -c*100)/10);
14         u=fix(nl-c*100-d*10);
15
16         printf('El numero %i tiene: \n', nl);
17         printf('Centenas = %i \n ',c);
18         printf('Decenas = %i \n', d);
19         printf('unidades = %i \n', u);
20         N = pq_exec_params(conn,"insert into ejer3(numero, centenas,decenas,unidades) values($1,$2,$3,$4);",(nl,c,d,u));
21     else
22         printf('Por favor ingrese un numero entre 0 y 999');
23     endif
24 catch
25     printf("Ha ocurrido un error \n");
26 end_try_catch
```

Resultado:

```
× Ventana de comandos
Determinar cuantas unidades, decenas y centenas tiene el numero
Ingrese un numero entre 0 y 999: 222
conn = <PGconn object>
El numero 222 tiene:
Centenas = 2
Decenas = 2
unidades = 2
>> |
```

Base de datos:

Data Output					Scratch
	numero character varying (20)	centenas character varying (20)	decenas character varying (20)	unidades character varying (20)	
1	478	4	7	8	
2	23	0	2	3	
3	872	8	7	2	
4	378	3	7	8	
5	222	2	2	2	

Ejercicio 4: Este ya no lo pude terminar

```
3 try
4   pkg_load_database
5   conn = pq_connect(setdbopts('dbname','Tarea_Preparatoria','host','localhost','port','5432','user','postgres','password','12345abc'));
6
7   x1 = randperm(6)
8   x2 = randperm(6)
9
10  x3 = x1+x2
11
12  if x3==8
13    print('Ha ganado el juego')
14
15  elseif x3==7
16    print('Ha perdido el juego')
17
18  else
19    print('Vuelva a jugar otra vez')
20
21  endif
22  N = pq_exec_params(conn,"insert into ejer4(dado1, dado2,resultado) values($1,$2,$3):", {x1,x2,x3,edad});
23
24
25  N=pq_exec_params(conn,'select * from ejer4:')
26
27
28 catch
29   print('Ocurrio un error, reinicie el juego')
```

Este es el enlace al repositorio:

<https://github.com/CristianGonzalez76/Primer-Parcial.git>

Enlace al drive:

https://drive.google.com/drive/folders/1M8pQ_tTETgVbWTggolFNee1aH2Zynhwd?usp=sharing