



UNIVERSIDAD DE COLIMA

Christian David Sánchez Sánchez

Walter Alexander Mata López

Ejecuciones de programas

24 de octubre de 2023

Fundamentos de Programación

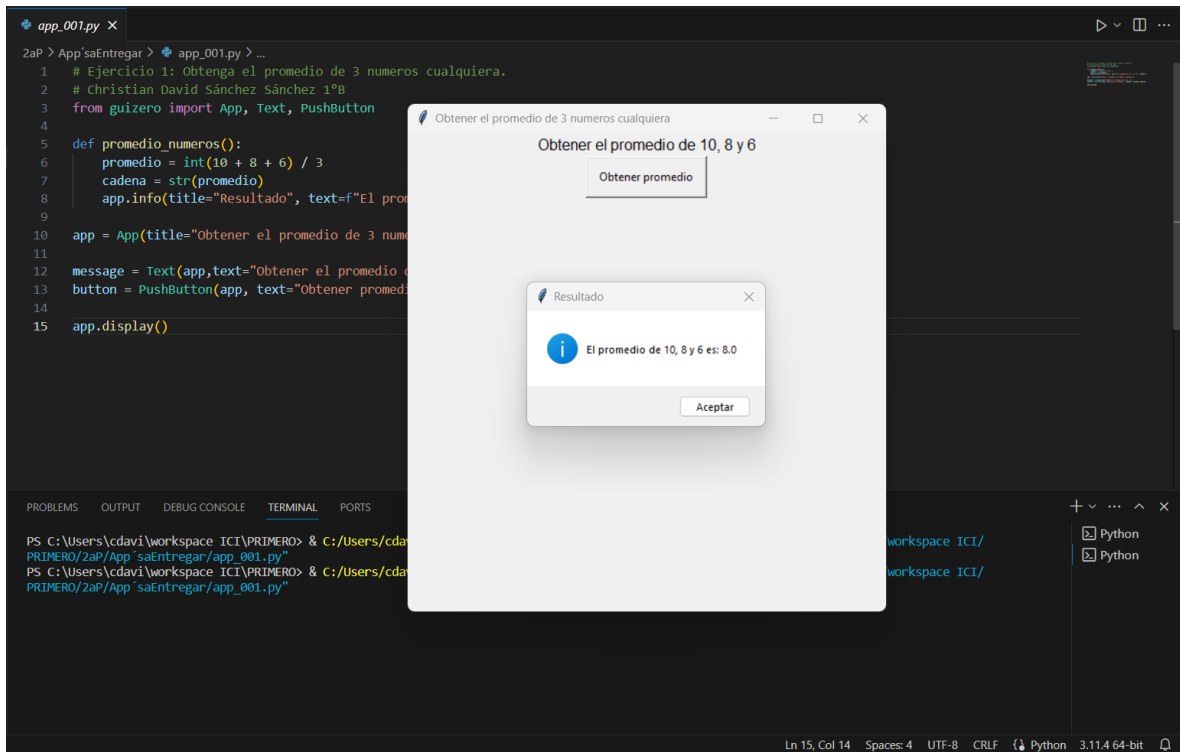
Ingeniería en Computación Inteligente

Universidad de Colima

1°B

Ejercicio 1: Obtenga el promedio de 3 números cualquiera.

En este programa obtenemos el promedio de 3 números que, como el enunciado no dice que tienen que leerse del teclado, ya están establecido. Al presionar el botón nuestro programa llama a una función que hace la suma de 10, 8 y 6 y el resultado lo divide entre 3 resultándonos 8 de promedio.



```
app_001.py X
2aP > App'saEntregar > app_001.py > ...
1 # Ejercicio 1: Obtenga el promedio de 3 numeros cualquiera.
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text, PushButton
4
5 def promedio_numeros():
6     promedio = int(10 + 8 + 6) / 3
7     cadena = str(promedio)
8     app.info(title="Resultado", text=f"El promedio de 10, 8 y 6 es: {cadena}")
9
10 app = App(title="Obtener el promedio de 3 numeros cualquiera")
11
12 message = Text(app, text="Obtener el promedio de 3 numeros cualquiera")
13 button = PushButton(app, text="Obtener promedio", on_click=promedio_numeros)
14
15 app.display()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\cdavi\workspace ICI\PRIMERO> & C:/Users/cdavi/OneDrive/Desktop/PRIMERO/2aP/App 'saEntregar/app_001.py'

PS C:\Users\cdavi\workspace ICI\PRIMERO> & C:/Users/cdavi/OneDrive/Desktop/PRIMERO/2aP/App 'saEntregar/app_001.py'

workspace ICI/ Python

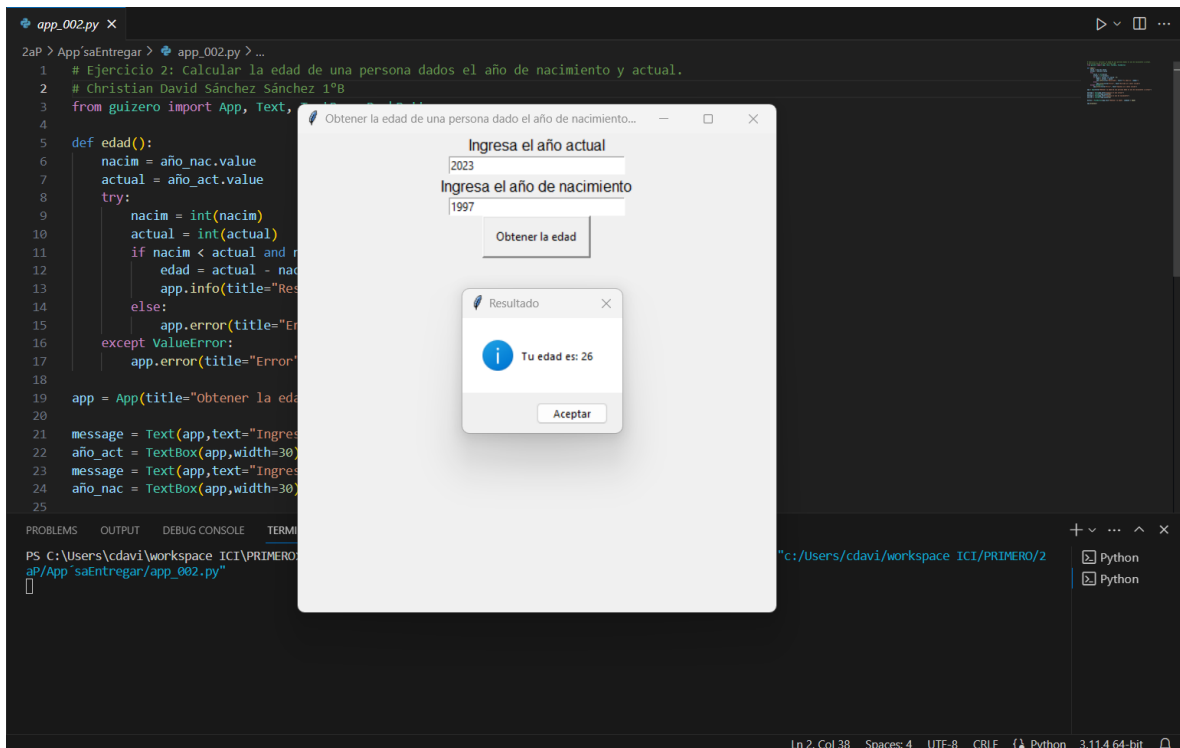
workspace ICI/ Python

Ln 15, Col 14 Spaces: 4 UTF-8 CRLF Python 3.11.4 64-bit

Ejercicio 2: Calcular la edad de una persona dados el año de nacimiento y el año actual.

En esta app o programa calculamos la edad de una persona mediante el año actual y de nacimiento que el usuario ingresa. Con la función “try” hacemos que todo lo que esté dentro de esta y con el “except” lo que no, cumpliendo la función de un if en Python. Después, tenemos que hacer casting a nuestras variables de año de nacimiento y año actual ya que, al tratarse de años, no pueden ser flotantes por lógica. Verificamos que el año de nacimiento sea menor al año actual para que el resultado sea lógico y que el año de nacimiento sea mayor a 0.

En caso de que el año de nacimiento sea mayor a 0 y menor al año actual ingresado, entonces se realiza la operación entre estas dos variables y se imprimirá el resultado en una ventana emergente indicándonos la edad de la persona.



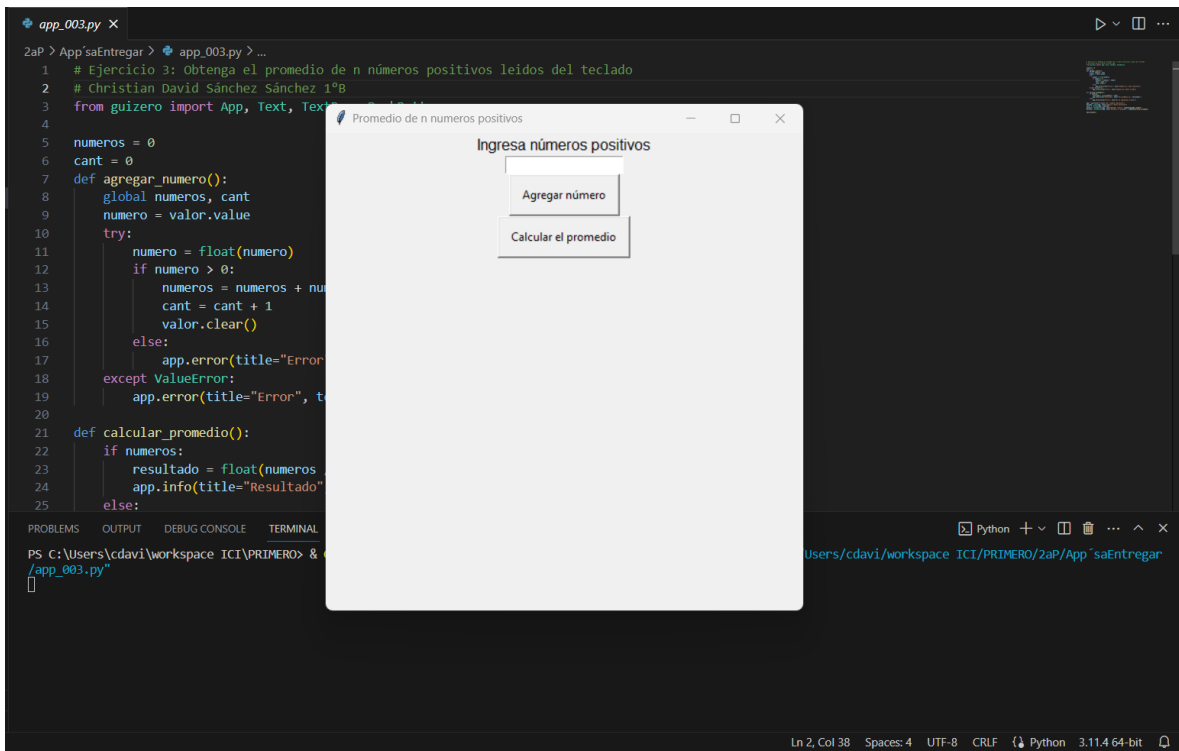
Ejercicio 3: Obtenga el promedio de n números positivos leídos del teclado.

En este programa obtenemos el promedio de n números positivos que el usuario ingresa. Al ser de n números, utilizamos dos botones de los cuales uno es para agregar el número deseado y el otro para obtener el promedio

Al momento de querer ingresar un número negativo no nos va a dejar agregarlo, cumpliendo con uno de los requerimientos del enunciado del problema. Si el número es positivo, entonces si lo agregará.

Cuando presionamos el botón de “Calcular el promedio” y no hemos ingresado un número, la condición if no permite que se obtenga el promedio y lanza un mensaje de error en una ventana emergente. Esto se debe a que la condición solo es verdadera si existe la variable “numeros” la cual se crea al agregar un número.

Cuando ingresamos los números deseados y presionamos el botón para calcular el promedio, nos arrojará el promedio en una ventana emergente, resolviendo nuestro problema.



```
app_003.py X
2aP > App'saEntregar > app_003.py > ...
1 # Ejercicio 3: Obtenga el promedio de n números positivos leídos del teclado
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text, Textbox, Button
4
5 numeros = 0
6 cant = 0
7 def agregar_numero():
8     global numeros, cant
9     numero = valor.value
10    try:
11        numero = float(numero)
12        if numero > 0:
13            numeros = numeros + numero
14            cant = cant + 1
15            valor.clear()
16        else:
17            app.error(title="Error", text="No se puede agregar un número negativo")
18    except ValueError:
19        app.error(title="Error", text="No se puede agregar un número que no sea un número")
20
21 def calcular_promedio():
22     if numeros:
23         resultado = float(numeros / cant)
24         app.info(title="Resultado", text=f"El promedio es: {resultado}")
25     else:
26         app.error(title="Error", text="No se puede calcular el promedio si no se han ingresado números")
27
28 app = App(title="Promedio de n numeros positivos")
29 valor = Textbox(app, text="Ingresa números positivos")
30 boton_agregar = Button(app, text="Agregar número")
31 boton_calcular = Button(app, text="Calcular el promedio")
32 valor.bind(on_text_enter=agregar_numero)
33 boton_agregar.bind(on_click=agregar_numero)
34 boton_calcular.bind(on_click=calcular_promedio)
35 app.run()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\cdavi\workspace\ICI\PRIMERO\2aP\App'saEntregar> python app_003.py

Ln 2, Col 38 Spaces: 4 UTF-8 CRLF Python 3.11.4 64-bit

Ejercicio 4: Obtener la edad promedio de n personas preguntándoles su año de nacimiento y asumiendo que el año actual es 2023.

En este programa vamos a obtener la edad promedio de n personas al preguntar solo el año de nacimiento ya que asumimos que es 2023. Al nosotros preguntar el año de nacimiento, vamos a convertir el valor ingresado a un entero usando casting. Si el número ingresado es flotante, aparecerá una ventana emergente diciendo que ingresemos un número válido.

Si el usuario ingresa un número negativo o mayor a 2023 entonces aparecerá una ventana emergente de error informando que el valor no es el adecuado.

Si se ingresa un año de nacimiento adecuado y pasa todas las condicionales, entonces se hace la resta al 2023 y se acumula, a la vez que un contador de personas incrementa su valor uno.

Al presionar el botón de “Obtener promedio” si no hemos ingresado algún valor nos va a dar un mensaje de error ya que hay una condicional de que la variable edades debe de ser mayor a 0 y esa se incrementa al agregar un número. De otro modo, nos arrojará el resultado, ósea el promedio de las edades.

```
app_004.py X
2aP > App'saEntregar > app_004.py > ...
1 # Ejercicio 4: Obtener la edad promedio de n personas preguntando su año de nacimiento y asumiendo que es 2023.
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text, Textbox, Button
4
5 personas = 0
6 edades = 0
7
8 def agregar_año():
9     global personas, edades
10    año = años.value
11    try:
12        año = int(año)
13        if año < 2023 and año >
14            edad = 2023 - año
15            personas = personas
16            edades = edades + ed
17            años.clear()
18    else:
19        app.error(title="Err
20    except ValueError:
21        app.error(title="Error",
22
23 def promedio():
24     if edades > 0:
25         promedio = float(edades/

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\cdavi\workspace ICI\PRIMERO>
/app_004.py"
```

Obtener la edad promedio de n personas preguntando su a...

Ingresa tu año de nacimiento

Agregar año

Obtener promedio

Resultado

El promedio es: 21.333333333333332

Aceptar

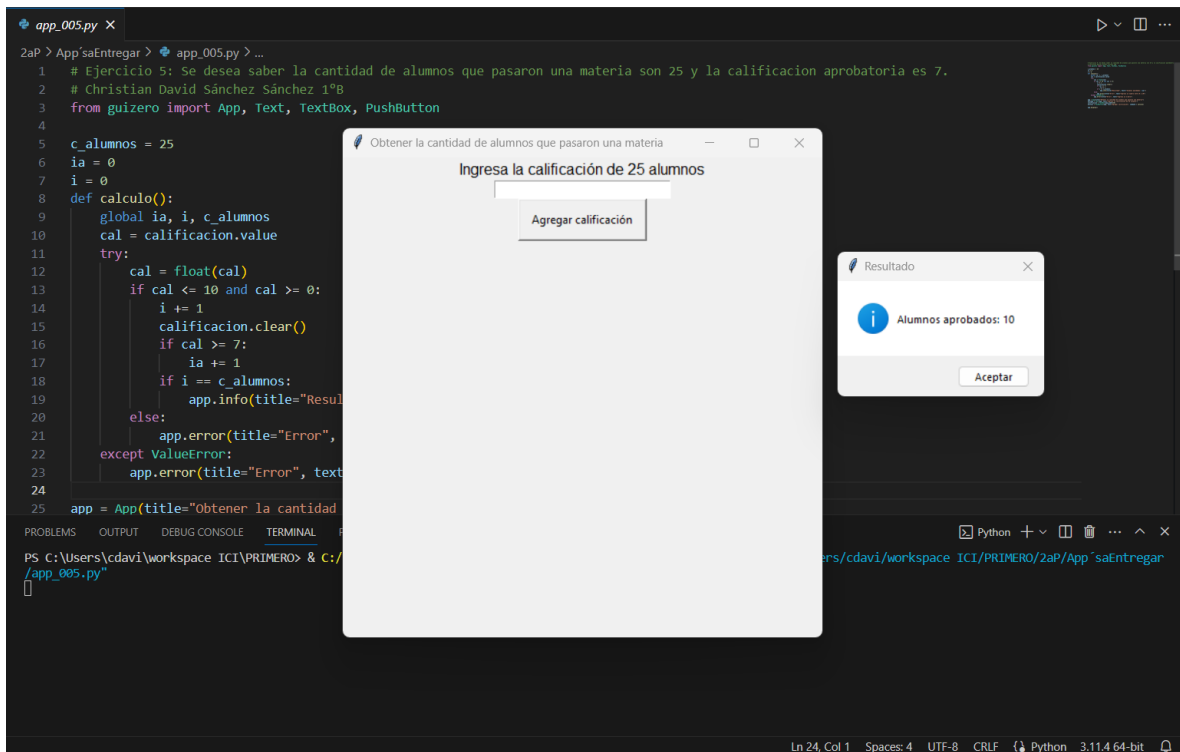
Python 3.11.4 64-bit

Ejercicio 5: Se desea saber la cantidad de alumnos que pasaron una materia. Son 25 y la calificación aprobatoria es 7.

En este programa vamos a ver, de 25 alumnos, cuantos aprobaron una materia si se pasa con 7. Para eso vamos a pedir calificaciones 25 veces y vamos a ir almacenando en un contador a las personas con 7 o más calificación.

Si el usuario ingresa una calificación menor a 0 o mayor a 10 entonces nos aparecerá una ventana emergente de error explicando que la calificación ingresada debe de estar en esos rangos.

Para saber si se ingresaron 25 calificaciones exactamente, al final tenemos una comparación de el contador con los alumnos, variables definidas con sus respectivos valores anteriormente. Así, cuando el contador llega a 25 nos imprime en una ventana emergente la cantidad de alumnos aprobados en total. Por lógica el resto estarán reprobados.



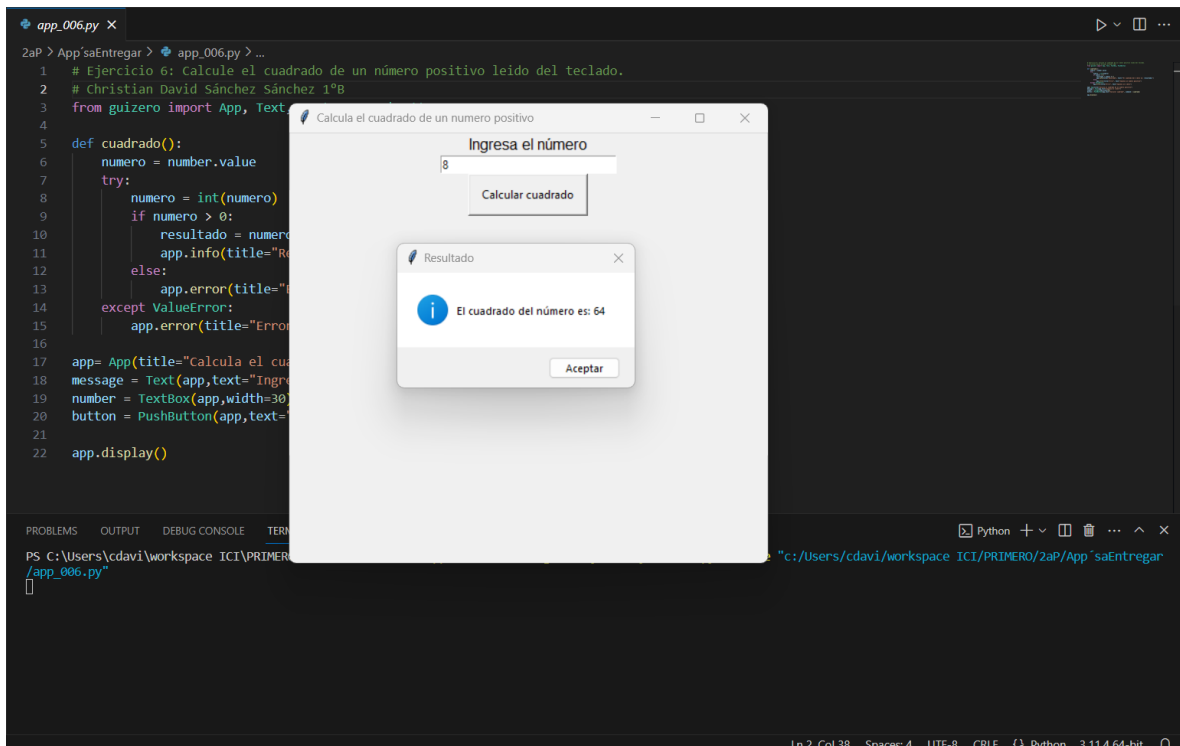
```
1 # Ejercicio 5: Se desea saber la cantidad de alumnos que pasaron una materia son 25 y la calificacion aprobatoria es 7.
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text, TextBox, PushButton
4
5 c_alumnos = 25
6 ia = 0
7 i = 0
8 def calculo():
9     global ia, i, c_alumnos
10    cal = calificacion.value
11    try:
12        cal = float(cal)
13        if cal <= 10 and cal >= 0:
14            i += 1
15            calificacion.clear()
16            if cal >= 7:
17                ia += 1
18            if i == c_alumnos:
19                app.info(title="Resultado", text="Alumnos aprobados: {}".format(ia))
20        else:
21            app.error(title="Error", text="La calificación debe estar entre 0 y 10.")
22    except ValueError:
23        app.error(title="Error", text="La calificación debe ser un número.")
24
25 app = App(title="Obtener la cantidad de alumnos que pasaron una materia")
26 calificacion = TextBox(app, label="Ingresa la calificación de 25 alumnos")
27 boton = PushButton(app, text="Agregar calificación", command=calculo)
28 app.run()
```

Ejercicio 6: Calcula el cuadrado de un número positivo leído del teclado.

En este programa calcularemos el cuadrado de un número. El usuario tiene que ingresar un número en la caja de texto, después de eso presionar el botón para calcular el cuadrado del número deseado.

Si el número es flotante o negativo, nos va a tirar un mensaje de error mencionando que el valor ingresado no es valido y que ingresemos otro.

Cuando el número ingresado sea positivo entonces se realiza el calculo y se guarda en una variable. Seguido de esto se imprime en una ventana emergente el resultado de elevar el número deseado al cuadrado.



```
app_006.py
2aP > App'saEntregar > app_006.py > ...
1 # Ejercicio 6: Calcule el cuadrado de un número positivo leído del teclado.
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text
4
5 def cuadrado():
6     numero = number.value
7     try:
8         numero = int(numero)
9         if numero > 0:
10             resultado = numero**2
11             app.info(title="Resultado", text="El cuadrado del número es: " + str(resultado))
12         else:
13             app.error(title="Error", text="El valor ingresado no es válido. Ingrese un número positivo.")
14     except ValueError:
15         app.error(title="Error", text="El valor ingresado no es válido. Ingrese un número positivo.")
16
17 app = App(title="Calcula el cuadrado de un numero positivo")
18 message = Text(app, text="Ingresa el número")
19 number = TextBox(app, width=30)
20 button = PushButton(app, text="Calcular cuadrado")
21
22 app.display()
```

Calcula el cuadrado de un numero positivo

Ingresa el número

8

Calcular cuadrado

Resultado

El cuadrado del número es: 64

Aceptar

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

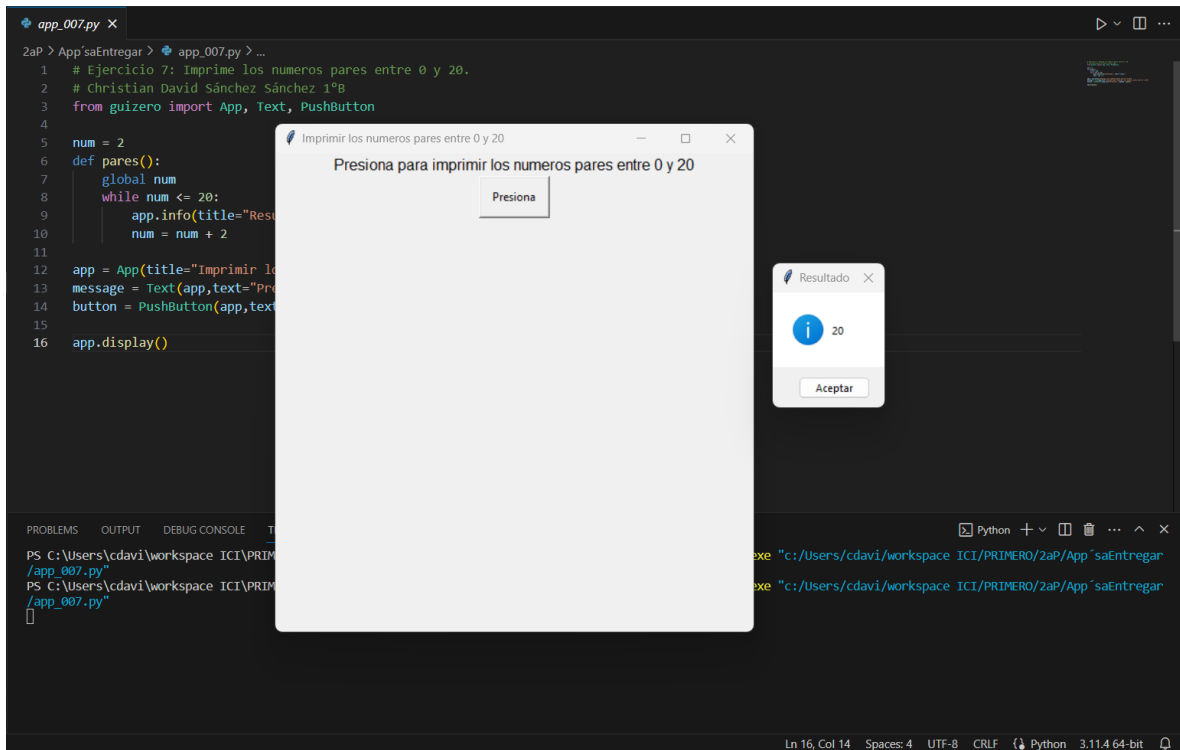
PS C:\Users\cdavi\workspace ICI\PRIMERO\2aP\App'saEntregar> python /app_006.py

Ln 2, Col 38 Spaces: 4 UTF-8 CRLF Python 3.11.4 64-bit

Ejercicio 7: Imprimir los números pares entre 0 y 20.

En este programa vamos a imprimir los números pares entre 0 y 20. Aquí no vamos a necesitar ingresar valores ya que estos son irrelevantes para el propósito que tiene el problema.

Al presionar el botón que hay, comenzarán a salir ventanas emergentes una por una con los números pares que existen entre 0 y 20. Al llegar al 20 y cerrar esa ventana, se termina la ejecución del programa.

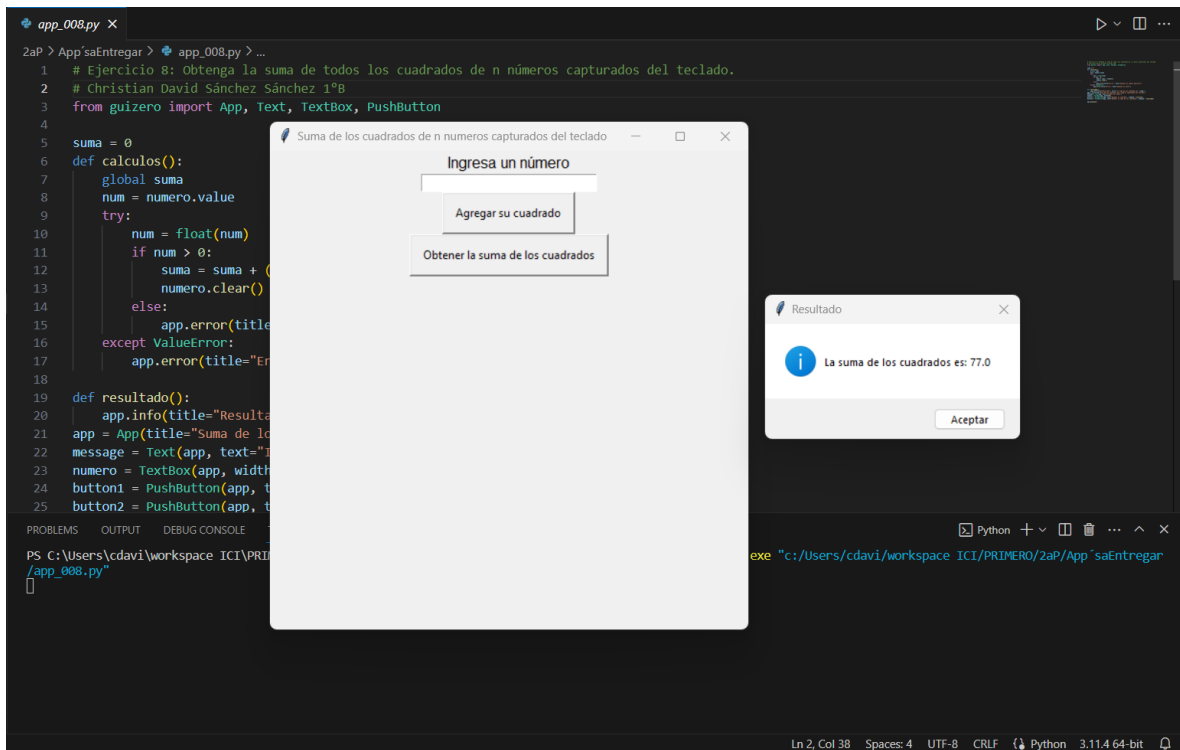


Ejercicio 8: Obtenga la suma de todos los cuadrados de n números capturados del teclado.

En este programa vamos a obtener la suma de los cuadrados de los números que el usuario ingrese. Lo que hace es que toma el número que el usuario ingresa y le saca el cuadrado, acumulándolo.

Si el usuario ingresa un número negativo, nos aparecerá un mensaje de error indicando que el número no es válido. Cuando se ingrese uno positivo, se le va a sacar el cuadrado y ese número se va a almacenar en un acumulador, para que después un contador aumente uno ya que esta cuenta la cantidad de números ingresados.

Al presionar el botón de “Obtener la suma” entonces se nos va a imprimir el total de lo que la variable que almacena o acumulador alcanzó, siendo esta la suma de los cuadrados de los números que ingresamos.



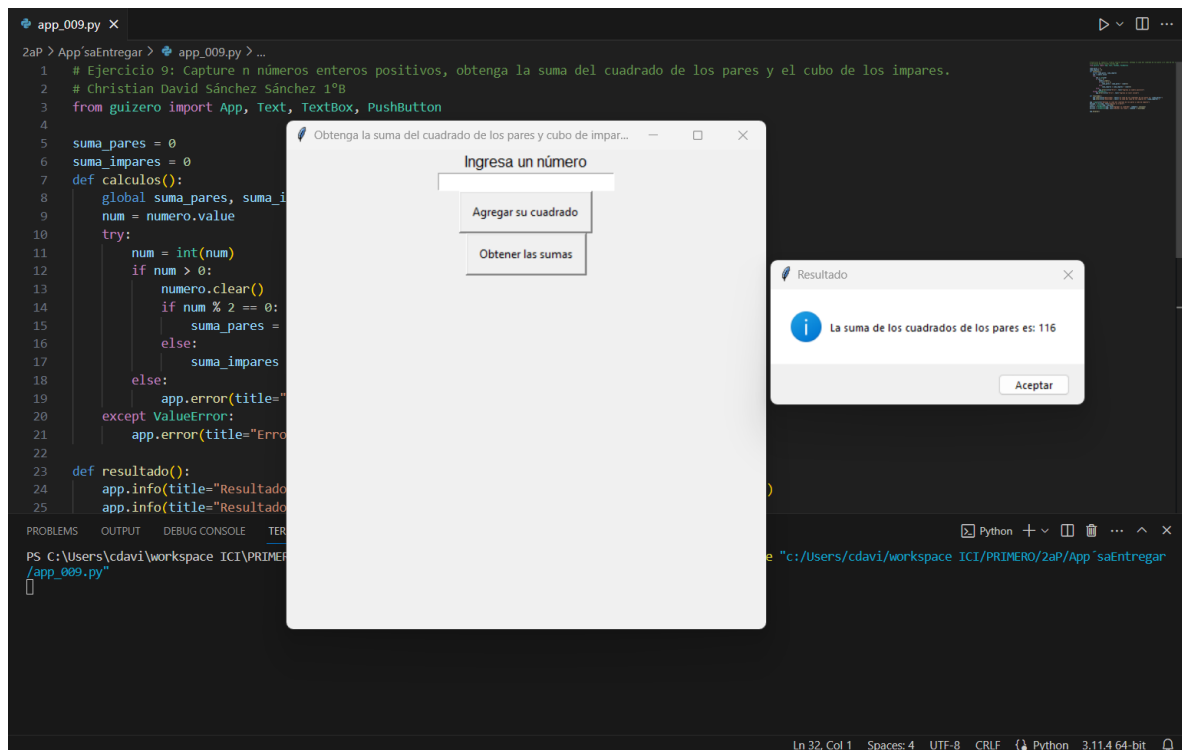
Ejercicio 9: Capture n números enteros positivos, obtenga la suma del cuadrado de los pares y el cubo de los impares.

Este programa calcula la suma del cuadrado de los números pares y el cubo de los números impares leídos del teclado. En este se nos presentan dos botones, el de “Agregar número” y “Obtener la suma”.

Al momento de querer agregar un número, si el valor ingresado es no es mayor a 0 entonces aparecerá un mensaje de error el cual nos dirá que el valor no es correcto. De ser mayor, entonces va a revisar si el número es divisible entre dos.

Si el valor es divisible entre dos entonces va a sacar su cuadrado y almacenarlo en un acumulador. Si no lo es, sacará su cubo y lo almacenará en otro acumulador.

Al presionar “Obtener las sumas” se nos abrirán dos ventanas emergentes, una de ellas nos imprimirá la suma de los cuadrados y la otra la suma de los cubos.



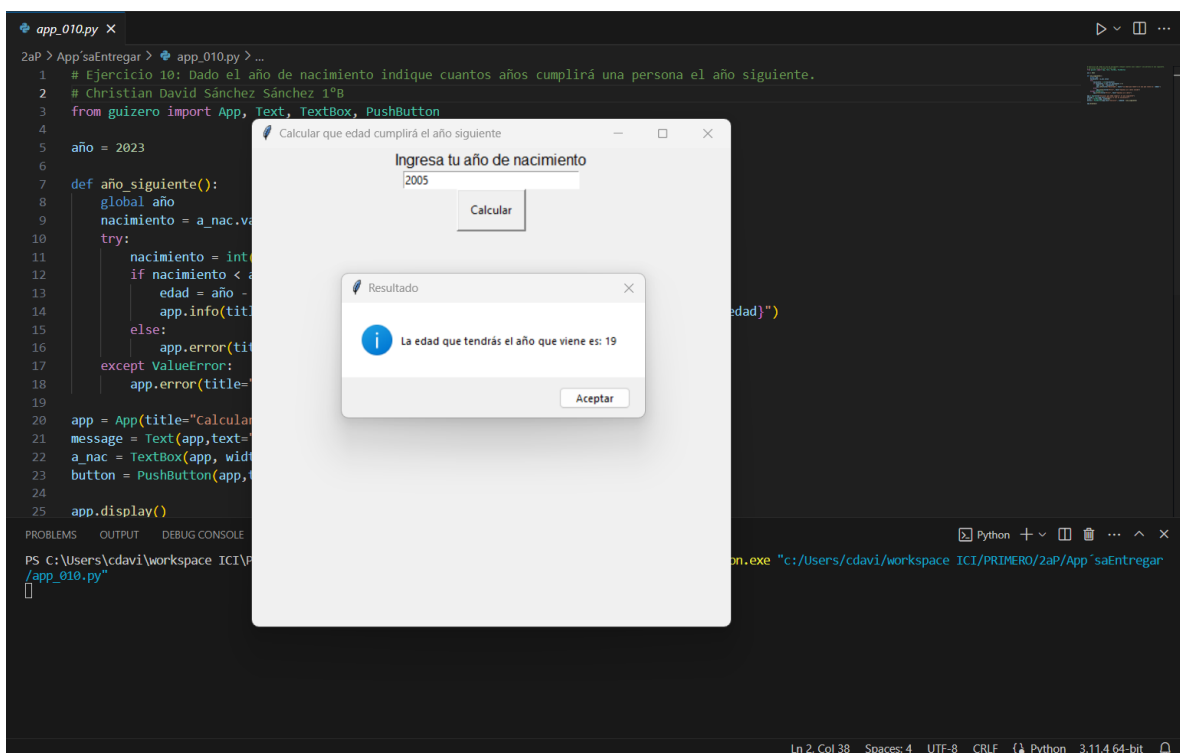
```
app_009.py X
ZaP > App'saEntregar > app_009.py > ...
1 # Ejercicio 9: Capture n números enteros positivos, obtenga la suma del cuadrado de los pares y el cubo de los impares.
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text, TextBox, PushButton
4
5 suma_pares = 0
6 suma_impares = 0
7 def calculos():
8     global suma_pares, suma_i
9     num = numero.value
10    try:
11        num = int(num)
12        if num > 0:
13            numero.clear()
14            if num % 2 == 0:
15                suma_pares =
16            else:
17                suma_impares
18        else:
19            app.error(title="
20    except ValueError:
21        app.error(title="Erro
22
23 def resultado():
24     app.info(title="Resultado
25     app.info(title="Resultado
26
27 PROBLEMS OUTPUT DEBUG CONSOLE TER
PS C:\Users\cdavi\workspace ICI\PRIME
/app_009.py"
Python 3.11.4 64-bit
```

Ejercicio 10: Indique, dado el año de nacimiento, cuantos años va a cumplir una persona el siguiente año.

En este programa vamos a indicar que edad cumplirá una persona el año que viene. Para esto, definimos el año 2023 en una variable para que, si queremos cambiar el año, solo tengamos que cambiar ese valor.

Si el usuario ingresa un año de nacimiento menor a 0 o mayor a 2023 entonces aparecerá una ventana emergente de error y tendremos que ingresar un valor nuevamente. Cuando el valor sea positivo, entonces al 2023 le vamos a restar este valor y le vamos a sumar 1, para que la edad que nos arroje sea la edad que se va a cumplir un año después.

Al final, en una ventana emergente imprimimos la edad que la persona cumplirá.



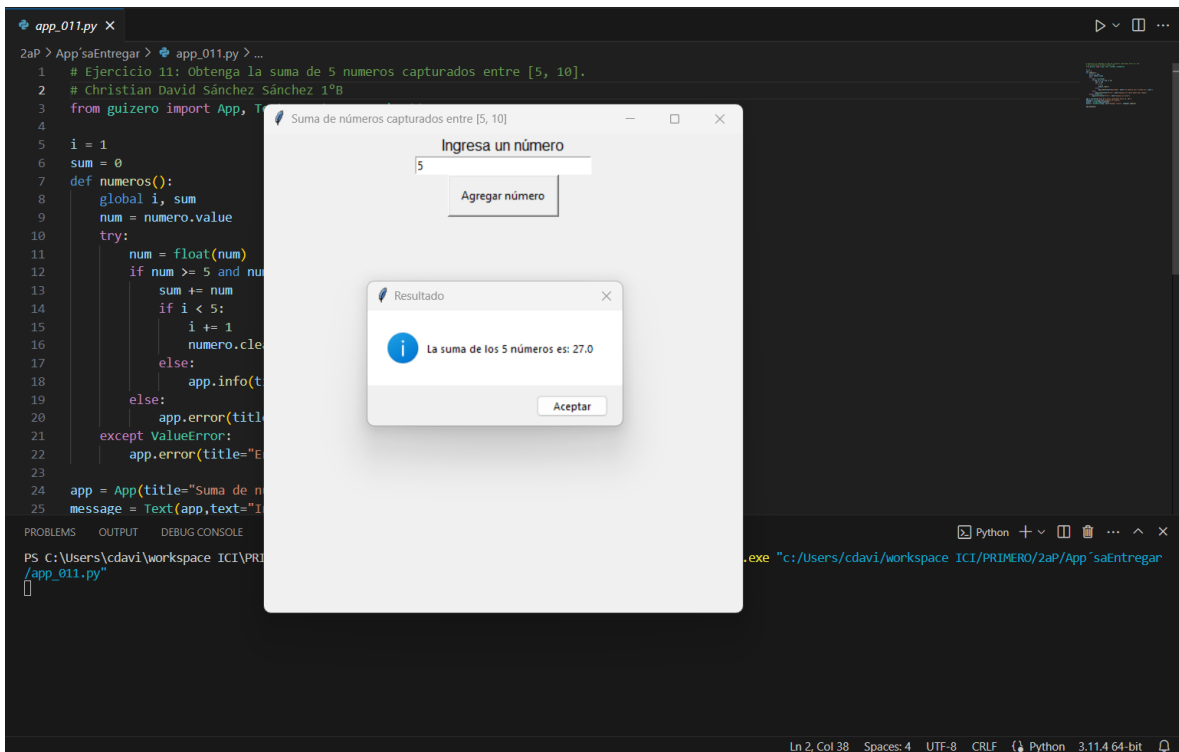
```
1 # Ejercicio 10: Dado el año de nacimiento indique cuantos años cumplirá una persona el año siguiente.
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text, TextBox, PushButton
4
5 año = 2023
6
7 def año_siguiente():
8     global año
9     nacimiento = a_nac.value
10    try:
11        nacimiento = int(nacimiento)
12        if nacimiento < 0 or nacimiento > año:
13            edad = año - nacimiento
14            app.info(title="Resultado", text=f"La edad que tendrás el año que viene es: {edad}")
15        else:
16            app.error(title="Error", text="El año de nacimiento debe ser mayor a 0 y menor a 2023.")
17    except ValueError:
18        app.error(title="Error", text="El año de nacimiento debe ser un número.")
19
20 app = App(title="Calcula que edad cumplirá el año siguiente")
21 message = Text(app, text="Ingresa tu año de nacimiento")
22 a_nac = TextBox(app, width=20)
23 button = PushButton(app, text="Calcular", on_click=año_siguiente)
24
25 app.display()
```

Ejercicio 11: Obtenga la suma de 5 números capturados entre [5 y 10] inclusive.

En este programa vamos a calcular la suma de 5 números leídos del teclado entre 5 y 10. Para esto el usuario tendrá que ingresar una cantidad dentro del rango y darle al botón “Agregar número”.

Aquí dentro, se revisará que el número ingresado sea mayor o igual a 5 y menor o igual a 10, si no es así entonces nos aparecerá una ventana de error indicando que el número ingresado no es válido. Pero de otro modo, entonces lo acumulará y va a contar que ya agregamos un número. Así pasará sucesivamente mientras los números ingresados sean positivos.

Al llegar al quinto número ingresado, nos abrirá una ventana emergente en la cual se nos indicará la suma de los 5 números positivos ingresados respetando todas las condiciones.

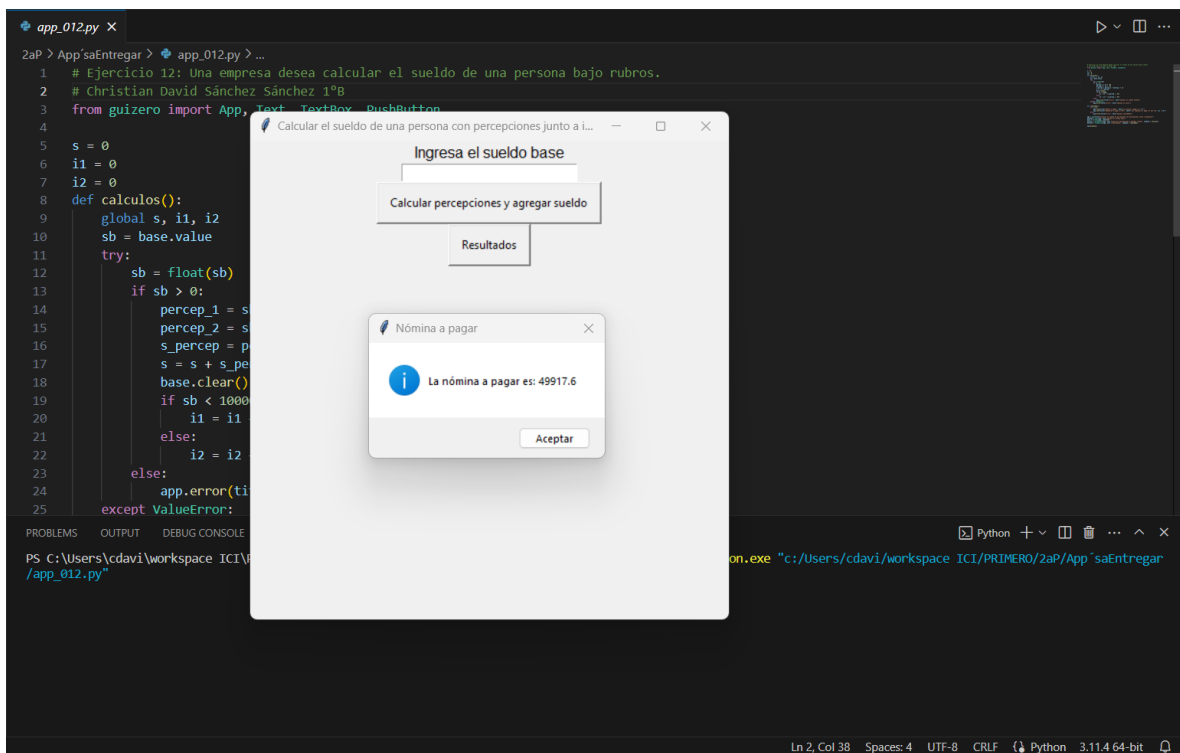


Ejercicio 12: Una empresa desea calcular el sueldo total de una persona bajo los siguientes rubros: Percepciones, el sueldo base, 5% de canasta básica, 3% de prima de antigüedad y deducciones si el salario base excede los 10,000 pesos el ISR es de 30% y menos es de 20%. Indicar el total de la nómina a pagar y los impuestos que el patrón pagará al SAT.

En este programa vamos a calcular el total de la nómina a pagar de n empleados y los impuestos a pagar al SAT de una empresa con base a los sueldos.

Cuando se ingrese un sueldo negativo entonces nos mandará un mensaje de error indicando que el sueldo debe de ser mayor a 0. De otro modo entonces calculará todas las percepciones mencionadas y almacenará el sueldo con percepciones en una variable para después revisar si el sueldo base es menor a 10,000. De ser así, se calcula el impuesto con la multiplicación del sueldo con percepciones por el 20% ya que es menor y se almacena en un acumulador de impuesto. De otro modo, ocurre lo mismo, pero se almacena en un acumulador de impuesto distinto.

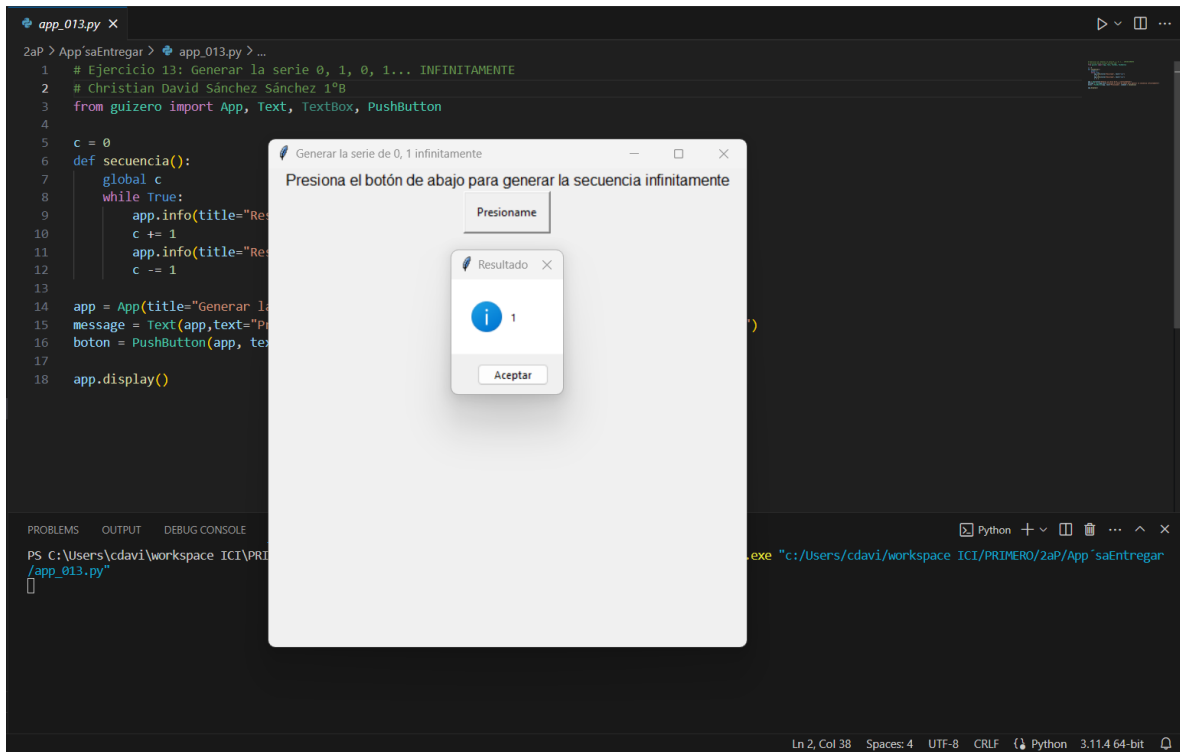
Cuando ya hayamos ingresado los sueldos deseados, damos click al botón “Resultados” y se nos abrirán dos ventanas emergentes. Una de ellas nos dirá la nómina a pagar ya con todos los cálculos realizados y la otra nos dirá la suma de los impuestos de los sueldos menores y mayores a 10,000, siendo el impuesto total que el patrón tendrá que pagar al SAT.



Ejercicio 13: Generar la serie 0, 1, 0, 1, 0, 1... INFINITAMENTE

En este programa imprimiremos una secuencia de 0's y 1's de forma infinita. Al nosotros presionar el botón que dice "Presioname", vamos a entrar a un ciclo while para el cual no hay salida.

Se imprimirá el contador, ahora con valor 0, y después se le aumenta 1 al contador y se imprime 1, se resta y se imprime 0 y así sucesivamente hasta el infinito ya que no le pusimos un fin. Todo esto sucede en distintas ventanas emergentes.



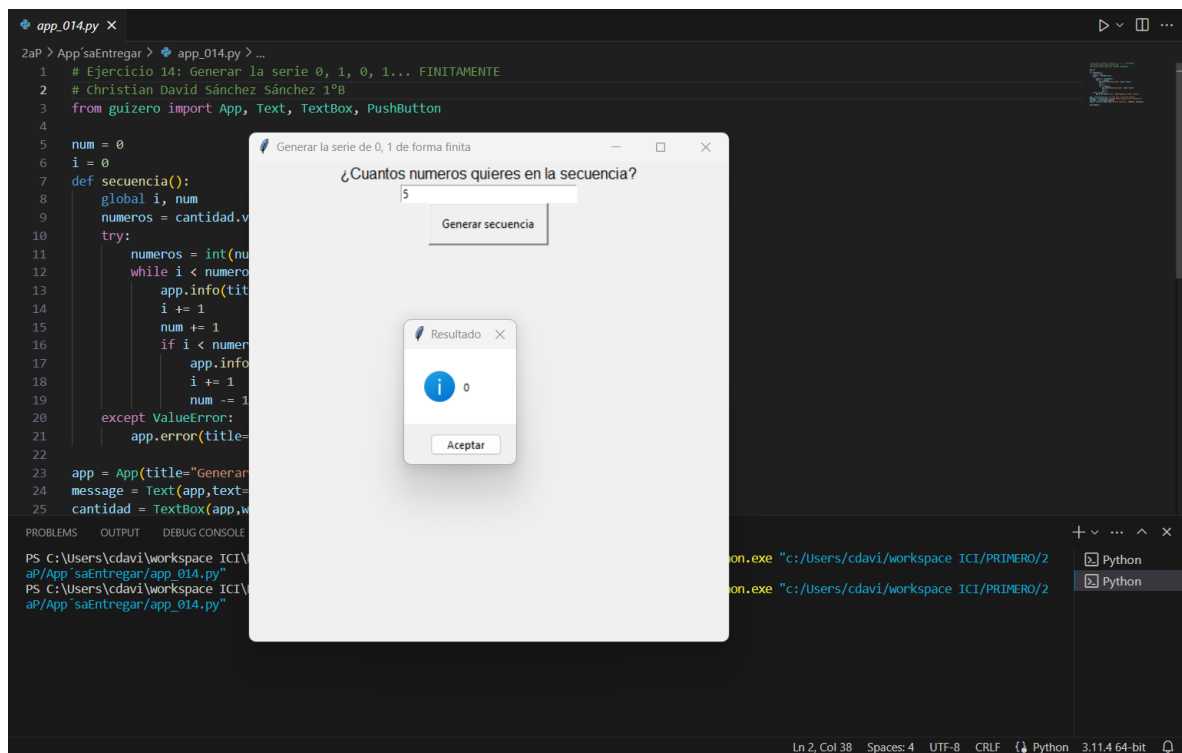
Ejercicio 14: Generar la serie 0, 1, 0, 1, 0, 1... FINITAMENTE

En este programa, a diferencia del anterior, vamos a imprimir una secuencia de 0's y 1's, pero de forma finita.

Para esto tenemos que ingresar la cantidad de números que queremos en el cuadro de texto y presionar el botón "Presióname". Si el número ingresado es negativo entonces nos dará un error indicándonos que el número ingresado debe de ser valido para poder generar la secuencia.

Si el número es positivo, entonces entra a un ciclo while que se repetirá mientras el contador sea menor al número ingresado. Se imprimirá el valor del contador de 0's y 1's, que por el momento es cero, y se sumará uno al igual que al contador para después volver a imprimir el valor de este contador de 0's y 1's que ahora vale uno, volviendo a crecer el valor del contador normal y así sucesivamente hasta que el while no se cumpla.

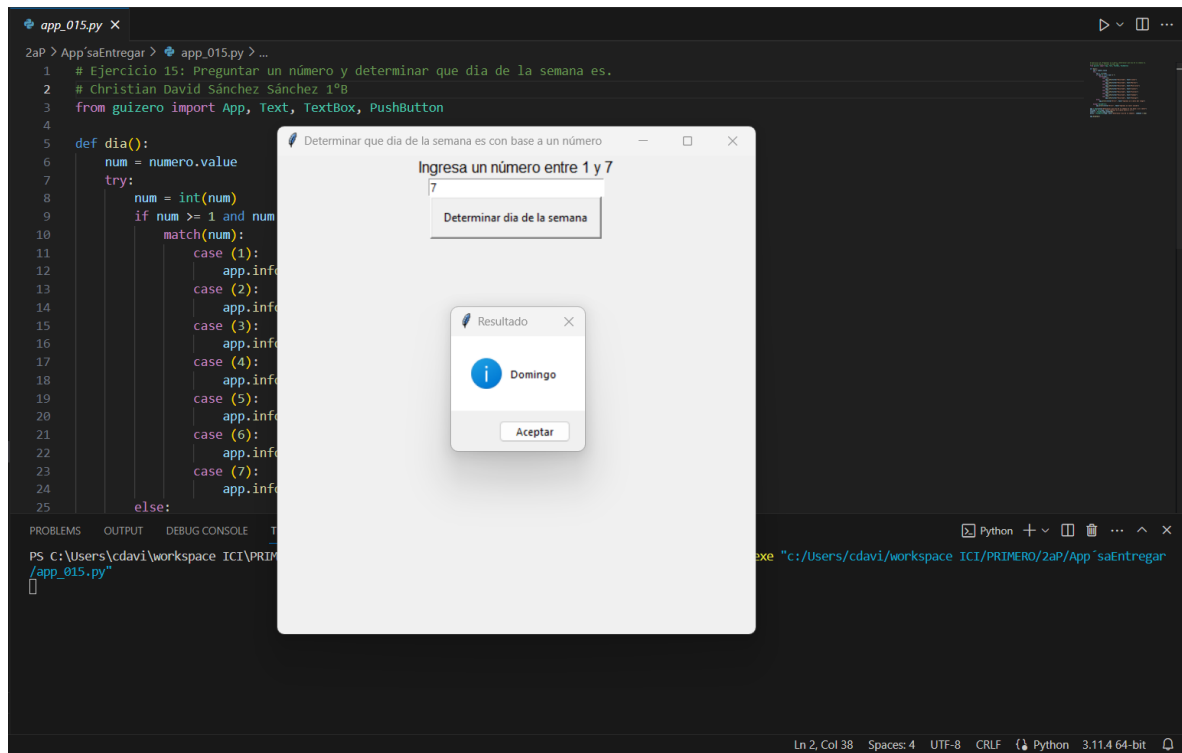
Cuando el while no se cumpla entonces ya se habrán impreso n veces los numeros 0 y 1.



Ejercicio 15: Preguntar un número y determinar qué día de la semana es.

En este programa con base a un número ingresado vamos a determinar el día de la semana. Para ello vamos a ingresar un valor en la caja de texto y presionar el botón “Determinar día de la semana”.

Si el número ingresado no es mayor o igual a 1 y menor o igual a 7 entonces nos saltará un mensaje de error que indica que tenemos que ingresar un numero que se encuentre dentro del rango. De otro modo, entonces el valor que está en el rango entrará al switch y se ubicará a si mismo, para así imprimir el día al que este corresponde.



```
app_015.py X
2aP > App'saEntregar > app_015.py > ...
1 # Ejercicio 15: Preguntar un número y determinar que día de la semana es.
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text, TextBox, PushButton
4
5 def dia():
6     num = numero.value
7     try:
8         num = int(num)
9         if num >= 1 and num <= 7:
10             match(num):
11                 case (1):
12                     app.info("Lunes")
13                 case (2):
14                     app.info("Martes")
15                 case (3):
16                     app.info("Miércoles")
17                 case (4):
18                     app.info("Jueves")
19                 case (5):
20                     app.info("Viernes")
21                 case (6):
22                     app.info("Sábado")
23                 case (7):
24                     app.info("Domingo")
25             else:
26                 app.info("Número fuera de rango")
27
28 app = App(title="Determinar que día de la semana es con base a un número")
29 texto = Text(app, "Ingresa un número entre 1 y 7")
30 boton = PushButton(app, "Determinar día de la semana")
31 boton.when_clicked = dia
32 app.run()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\cdavi\workspace ICI\PRIMERO\2aP\App'saEntregar> python /app_015.py

Python 3.11.4 64-bit

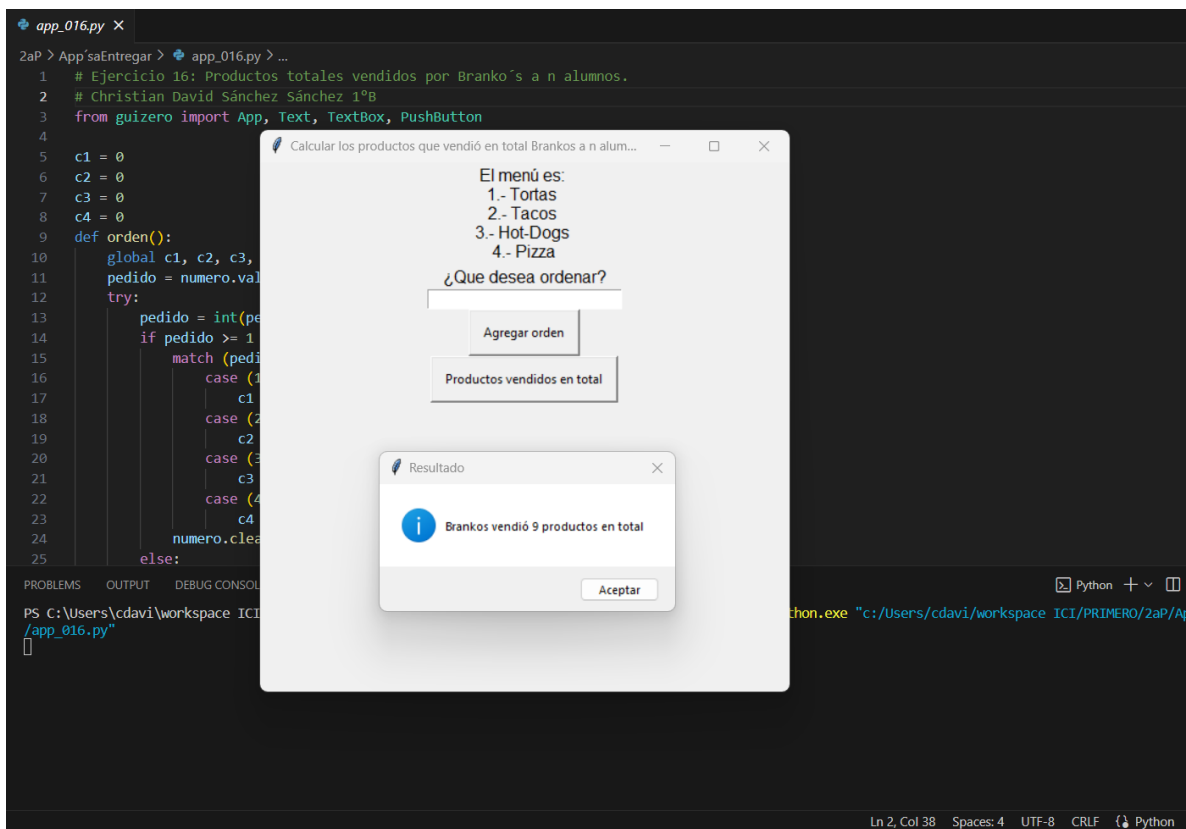
Ejercicio 16: La tienda “Brankos” debe vender productos a n alumnos, ofrecen: tortas, tacos, hotdogs y pizzas. Imprime la cantidad de productos vendidos en total.

En este programa vamos a calcular cuantos productos vendió en total la tienda Brankos.

Para esto, en la caja de texto vamos a ingresar el número correspondiente a lo que vamos a pedir. Si se ingresa un número que no sea mayor o igual a 1 y menor o igual a 4 entonces nos saltará un mensaje de error el cual nos va a indicar que el número que ingresemos tiene que estar en este rango.

Una vez ingresado un número adecuado este entrará en el match y se ubicará donde le corresponde, aumentando uno en el contador que le toca. Así estará mientras estemos agregando números válidos.

Cuando queramos obtener el total de productos, entonces vamos a dar click en el botón “Productos vendidos en total” y se nos abrirá una ventana emergente la cual nos indicará el total de productos vendidos de cualquier cosa del menú.



```
app_016.py X
2aP > App'saEntregar > app_016.py > ...
1 # Ejercicio 16: Productos totales vendidos por Branko's a n alumnos.
2 # Christian David Sánchez Sánchez 1ºB
3 from guizero import App, Text, TextBox, PushButton
4
5 c1 = 0
6 c2 = 0
7 c3 = 0
8 c4 = 0
9 def orden():
10     global c1, c2, c3,
11     pedido = numero.val
12     try:
13         pedido = int(pe
14         if pedido >= 1
15         match (pedi
16             case (1
17                 c1
18             case (2
19                 c2
20             case (3
21                 c3
22             case (4
23                 c4
24             numero.clea
25     else:
26
PROBLEMS OUTPUT DEBUG CONSOLE
PS C:\Users\cdavi\workspace ICI
/app_016.py"
Python.exe "c:/Users/cdavi/workspace ICI/PRIMERO/2aP/A
```

Ejercicio 17: Crea el menú de una calculadora funcional que pueda sumar, restar, multiplicar y dividir al seleccionar el tipo de operación y dar 2 números.

En este programa vamos a crear una calculadora funcional de dos valores.

En las primeras cajas de texto vamos a ingresar los valores con los que queremos hacer operaciones, pueden ser de cualquier tipo ya que esta calculadora funciona con positivos, negativos y flotantes.

En la tercera caja de texto se ingresará un valor del uno al 4 ya que son las operaciones disponibles, si se ingresa un número fuera de ese rango entonces nos saltará un mensaje de error indicándonos que el número debe de estar dentro del rango para realizar la operación. Una vez esté dentro del rango, el número ingresado entrará al switch y tomará el camino que le corresponde.

Dependiendo el número ingresado será la operación que nos va a dar como resultado en una ventana emergente.

