	FORMATO EVALUACIÓN		Código: DC-FO-058 Versión: 1
	Macroproceso: Docencia	Proceso: Desarrollo de la actividad académica	

Tipo de evaluación: Parcial: ☒ Seguimiento: ☐ Final: ☐

Estudiante: Yan Frank Ríos López Código: _____
Asignatura: LENGUAJE DE PROGRAMACIÓN Programa: INGENIERÍA INFORMÁTICA
Docente: FEIBERT ALIRIO GUZMÁN PÉREZ Fecha: 23/10/2024

1. Selección Múltiple: El siguiente código implementa una calculadora para realizar operaciones básicas como suma, resta, multiplicación y división. Sin embargo, queremos agregar una nueva operación que calcule el **módulo** (el resto de la división entre dos números).

- a. ¿Cuál de las siguientes opciones modificaría correctamente el código para incluir esta nueva funcionalidad?

```
function realizarOperacion(operacion) {
    var num1 =
parseFloat(document.getElementById("numero
1").value);
    var num2 =
parseFloat(document.getElementById("numero
2").value);
    var resultado;

    switch (operacion) {
        case "suma":
            resultado = num1 + num2;
            break;
        case "resta":
            resultado = num1 - num2;
            break;
        case "multiplicacion":
            resultado = num1 * num2;
            break;
        case "division":
            if (num2 !== 0) {
                resultado = num1 / num2;
            } else {
                resultado = "Error: No se
puede dividir por cero";
            }
            break;
        // Completa la operación de módulo
    }
}
```

```
document.getElementById("resultado").inner
Text = resultado;
```

}
* Indica el nombre LP trabajado: JavaScript

A) **case "modulo":**
 resultado = num1 % num2;
 break;

B) **case "residuo":**
 resultado = num1 % num2;
 break;

C) **case "mod":**
 resultado = num1 / num2;
 break;

D) **case "resto":**
 resultado = num1 - num2;
 break;

2. Verdadero o Falso con Justificación: El código de la calculadora verifica si el segundo número es cero antes de realizar la división. Si es cero, se muestra un mensaje de error en lugar de intentar la operación.


- a. El siguiente código evita correctamente que se divida por cero al manejar el caso de la división.

```
case "division":
    if (num2 !== 0) {
        resultado = num1 / num2;
    } else {
        resultado = "Error: No se puede
dividir por cero";
    }
    break;
```

Respuesta: Verdad.

Justificación: Esto es verdad debido a que si el num2 es diferente de 0, sí se podrá llevar a cabo la división, pero sino, nos mostrará un error.

3. Ejercicio Práctico: Tienes el siguiente código incompleto que intenta implementar una operación para **eleva el primer número al exponente del**

	FORMATO EVALUACIÓN		Código: DC-FO-058
	Macroproceso: Docencia	Proceso: Desarrollo de la actividad académica	Versión: 1

segundo número (es decir, $\text{num1}^{\text{num2}}$). Completa los espacios vacíos para que la funcionalidad de la potencia funcione correctamente cuando se presione el botón correspondiente.

```
function potencia() {
    realizarOperacion('Potencia');
}

function realizarOperacion(operacion) {
    var num1 =
parseFloat(document.getElementById("numero
1").value);
    var num2 =
parseFloat(document.getElementById("numero
2").value);
    var resultado;

    switch (operacion) {
        case "suma":
            resultado = num1 + num2;
            break;
        case "resta":
            resultado = num1 - num2;
            break;
        case "multiplicacion":
            resultado = num1 * num2;
            break;
        case "division":
            if (num2 !== 0) {
                resultado = num1 / num2;
            } else {
                resultado = "Error: No se
puede dividir por cero";
            }
            break;
        case "potencia":
            resultado = Math.pow(num1,
num2);
            break;
        default:
            resultado = "Operación no
válida";
    }

    document.getElementById("resultado").innerHTML
    Text = resultado;
}
```

Tareas:

1. Completa los espacios vacíos en el código.

2. Verifica que el código compile correctamente y que la operación de potencia funcione al ingresar los valores.

4. Selección Múltiple: El siguiente código implementa una calculadora con funciones básicas. Sin embargo, queremos agregar una nueva operación que calcule la **potencia** de un número.

```
from tkinter import Tk, Label, Entry,
Button, StringVar, messagebox


def realizar_operacion(operacion):
    try:
        num1 = float(entry_num1.get())
        num2 = float(entry_num2.get())
        resultado = ""

        if operacion == "suma":
            resultado = num1 + num2
        elif operacion == "resta":
            resultado = num1 - num2
        elif operacion ==
"multiplicacion":
            resultado = num1 * num2
        elif operacion == "division":
            resultado = num1 / num2 # Se
debe manejar la división por cero
# Falta implementar potencia aquí

        label_resultado.config(text="Resultado: "
+ str(resultado))
    except ValueError:
        messagebox.showerror("Error", "Por
favor, ingresa números válidos.")
    except ZeroDivisionError:
        messagebox.showerror("Error",
"División por cero no permitida.")

# Configuración de la interfaz gráfica
root = Tk()
root.title("Calculadora")
label_num1 = Label(root, text="Número 1:")
label_num1.pack()
entry_num1 = Entry(root)
entry_num1.pack()

label_num2 = Label(root, text="Número 2:")
label_num2.pack()
```

	FORMATO EVALUACIÓN		Código: DC-FO-058 Versión: 1
	Macroproceso: Docencia	Proceso: Desarrollo de la actividad académica	

```

entry_num2 = Entry(root)
entry_num2.pack()

label_resultado = Label(root,
text="Resultado:")
label_resultado.pack()

boton_suma = Button(root, text="Sumar",
command=lambda:
realizar_operacion("suma"))
boton_suma.pack()

boton_resta = Button(root, text="Restar",
command=lambda:
realizar_operacion("resta"))
boton_resta.pack()

boton_multiplicacion = Button(root,
text="Multiplicar", command=lambda:
realizar_operacion("multiplicacion"))
boton_multiplicacion.pack()

boton_division = Button(root,
text="Dividir", command=lambda:
realizar_operacion("division"))
boton_division.pack()

# Aquí se deben agregar botones para la
nueva operación de potencia
root.mainloop()

```

* Indica el nombre del LP trabajado: Python con Tkinter

a. ¿Cuál de las siguientes opciones completa correctamente la implementación de la operación de potencia?

A) `elif operacion == "potencia":`
`resultado = num1 ^ num2`

B) `elif operacion == "potencia":`
`resultado = num1 ** num2`

C) `elif operacion == "potencia":`
`resultado = num1 * num2 * num2`

D) `elif operacion == "potencia":`
`resultado = num1 + num2`

b. El código actual de la calculadora es capaz de manejar divisiones por cero sin detener el programa, gracias al uso de la excepción `ZeroDivisionError`.

Pregunta: ¿Es verdadero o falso?

Respuesta: Verdadero

Justificación: Verdadero, debido a que luego del Try al tener el except `ZeroDivisionError` que este protege el código de detenerse cuando la división es entre 0.

5. Ejercicio Práctico: A continuación, se muestra una sección del código que debe ser modificada para incluir una nueva función que maneje la verificación de la división por cero.

```


def realizar_operacion(operacion):
    try:
        num1 = float(entry_num1.get())
        num2 = float(entry_num2.get())
        resultado = ""

        if operacion == "suma":
            resultado = num1 + num2
        elif operacion == "resta":
            resultado = num1 - num2
        elif operacion ==
"multiplicacion":
            resultado = num1 * num2
        elif operacion == "division":
            resultado = num1 / num2 # Se
debe manejar la división por cero
# Falta implementar potencia aquí

label_resultado.config(text="Resultado: "
+ str(resultado))
    except ValueError:
        messagebox.showerror("Error", "Por
favor, ingresa números válidos.")
    except ZeroDivisionError:
        messagebox.showerror("Error",
"División por cero no permitida.")

```

a. Implementa una función llamada `verificar_division` que lanza una excepción `ZeroDivisionError` si el segundo número es cero. Luego, utiliza esta función en las operaciones de división y potencia.

	FORMATO EVALUACIÓN		Código: DC-FO-058 Versión: 1
	Macroproceso: Docencia	Proceso: Desarrollo de la actividad académica	

TENER PRESENTE: RESUMEN DE CAMBIOS:

a.1. Agregar la Función verificar_division:

- Esta función comprueba si num2 es igual a cero y lanza una excepción ZeroDivisionError si lo es.

a.2. Modificar la Lógica de las Operaciones:

- En la operación de **división** (operacion == "division"), se llama a verificar_division(num2) antes de realizar la división.

- En la operación de **potencia** (operacion == "potencia"), también se llama a verificar_division(num2) antes de calcular la potencia.

c. Los códigos deben ser entregados de forma funcional.

Estructura Primer código:

1. **Crea una carpeta:** Crea una carpeta en tu sistema con el nombre que prefieras (por ejemplo, "calculadora").
2. **Guarda los archivos:**
 - Guarda el archivo HTML como index.html. Suministrado por el docente
 - Guarda el archivo CSS como styles.css. Suministrado por el docente
 - Guarda el archivo JavaScript como script.js. Creado por el estudiante
3. **Abre el archivo HTML:** Haz doble clic en el archivo index.html y se abrirá en tu navegador.
4. **Prueba la calculadora:** Ahora podrás ingresar los dos números, seleccionar la operación que desees realizar, y el resultado se mostrará en la parte inferior.

Con estos tres archivos, la calculadora estará completamente funcional en tu navegador.

Estructura segundo y tercer código:

- Como son similares solo adjunta uno donde emplees lo trabajado en .py
- No olvidar adjuntar el documento en PDF y el .py del ejercicio de ciencia de datos.

Tener presente: Crea un repo en tu GitHub con el nombre Parcial_2_Nombre_Apellido y me compartes el repo con README bien organizado.

Es deber del estudiante realizar las tareas universitarias con honradez y veracidad (Artículo 38 del Reglamento Estudiantil para pregrado). Por lo anterior, es falta disciplinaria toda modalidad de plagio o fraude en actividades, trabajos y evaluaciones académicas y acarrea la aplicación del Régimen Disciplinario (Capítulo XIX).

Debe entregar el parcial solo marcando la respuesta correcta y en Teams adjuntar la evidencia de lo trabajado.

Observaciones:

- Durante el examen no se resuelven dudas.
- La práctica evaluativa debe tener consignado el valor de cada una de las preguntas.