

## Práctica 3

Trabaje los problemas en archivos por separado de acuerdo al problema planteado.  
**Para todos los casos, emplee manejo de errores y funciones según corresponda.**

### Manejo de Errores:

#### Problema 1:



Implemente un programa que solicite al usuario una fracción, con formato X/Y, donde cada uno de X e Y es un número entero, y luego muestra, como un porcentaje redondeado al número entero más cercano, donde se indicará la cantidad de combustible en el tanque. Se debe tener en cuenta los siguientes casos:

- Colocar E en caso X/Y sea menor a 1% del total
- Colocar F en caso X/Y sea mayor a 99%.
- En otro caso, devolver el valor en porcentaje %

También debe tomar en cuenta los siguientes casos:

- X y Y deben ser números enteros
- X debe ser mayor o igual a Y, y Y != 0

De no cumplirse estos casos, se debe volver a preguntar al usuario. Asegúrese de detectar cualquier excepción como ValueError o ZeroDivisionError.

#### Ejemplos:

- Input: 4/0 Acción: Volver a preguntar al usuario dada la excepción ZeroDivisionError
- Input 1.5/3 Acción: Error dado que solo se permiten números enteros ValueError
- Input 5 / 4 Acción: Volver a preguntar al usuario
- Input 3 / 4 Output: 75%
- Input 4/4: Output F

**Nota:** Le será de utilidad aplicar

```
try:
    ...
except ValueError:
    ...
except ZeroDivisionError:
    ...
```

#### Problema 2:

Cree un programa que solicite al usuario una lista de calificaciones separadas por comas. Divida la cadena en calificaciones individuales y almacénelas en una lista para luego convertir cada calificación en un entero. Deberá utilizar una sentencia try/except para informar al usuario cuando los valores introducidos no puedan ser convertidos debido a un error de tipeo o formato. (Los métodos de cadena le serán de utilidad)

## **POO:**

Material Complementario POO [link](#)

### **Problema 3:**

Definir una clase llamada "CIRCULO" la cual contenga un atributo inicializador **radio**. La clase "CIRCULO" debe tener un método que puede calcular el área en utilizando el atributo radio.

### **Problema 4:**

Definir una clase llamada "RECTANGULO" que puede ser construida por los atributos largo y ancho. La clase "RECTANGULO" debe tener un método que puede calcular el área utilizando los atributos de la clase.

### **Problema 5:**

Cree una clase Alumno e inicialícela con el nombre y el número de registro. Haga los métodos para:

1. **Display** - Debe mostrar toda la información del estudiante (nombre y número de registro).
2. **setAge** - Debe asignar la edad al estudiante
3. **setNota** - Debe asignar las notas al estudiante.

### **Problema 6:**

Modulo3/Ejercicios/Ejercicio2

Implementación clase Punto según las indicaciones del ejercicio

## **Módulos y Paquetes:**

### **Problema 7:**

Desarrollar un módulo que contenga las siguientes funciones:

- Que genere 20 números enteros aleatorios entre 0 y 100 y devuelva una lista.
- Mostrar la lista obtenida por pantalla.
- Ordenar los valores de la lista y mostrarla por pantalla.

Luego crea un script main.py en el mismo directorio en el que deberás importar el módulo y ejecutar las funciones.

Nota: utilizar el módulo "[random](#)" para generar un número aleatorio.

**Problema 8:**

Cree un módulo llamado `operaciones.py` el cual contendrá 4 funciones para realizar una suma, una resta, un producto y una división entre dos números. Todas ellas devolverán el resultado.

En las funciones del módulo deberá de haber tratamiento e invocación manual de errores para evitar que se quede bloqueada una funcionalidad, esto incluye:

- En caso de que se envíen valores a las funciones que no sean números, deberá aparecer un mensaje que informe Error: Tipo de dato no válido.
- En caso de realizar una división por cero, deberá aparecer un mensaje que informe Error: No es posible dividir entre cero.

Una vez creado el módulo, crea un script `calculos.py` en el mismo directorio en el que deberás importar el módulo y ejecutar las funciones.