

Clase 13. Excepciones

Ejercicios: <https://docs.hektorprofe.net/python/errores-y-excepciones/ejercicios/>

Ejercicio

Escribir una función que reciba dos números (ingresados por el usuario) y a partir de las operaciones especificadas en un archivo (cuyo nombre deberá ser provisto como parámetro), realice los cálculos y devuelva el resultado final de cada línea (resultado que se obtiene operando los dos números recibidos). El archivo contendrá por línea una operación aritmética sencilla; por ej:

```
+  
/  
**
```

Nota: investigar la función de Python llamada eval.

En caso de producirse un error durante la lectura de una línea del archivo, se debe lanzar una excepción propia que informe número de línea, operación problemática y motivo del error.

Una posible solución:

```
class OperationError(SyntaxError):  
    def __init__(self,op,linea):  
        self.op = op  
        self.linea = linea  
    def __str__(self):  
        return f"Linea: {self.linea} Operacion incorrecta: {self.op}"  
  
def operacion(num1,num2,archivo):  
    try:  
        with open(archivo,'r') as archivo:  
            cont_linea = 0  
            for linea in archivo:  
                cont_linea += 1  
                _op = linea.replace("\n","")  
                operacion = f"{num1} {_op} {num2}"  
                try:  
                    print(f"Linea: {cont_linea} Operacion: {operacion} = {eval(operacion)}")  
                except SyntaxError as e:  
                    raise OperationError(operacion,cont_linea)  
  
            except FileNotFoundError as e:  
                print("El archivo no existe")  
  
    try:  
        operacion(1,2,"operaciones.txt")  
    except OperationError as e:  
        print(e)
```