

In []:

```
#Gestión de los errores try - except
```

```
def suma(a, b="3")  
    resultado = a + b  
    return resultado
```

```
s = suma(a=2)  
s
```

In [3]:

```
try:  
    resultado = a + b  
  
except Exception as error: #Esto muestra el error pero sigue ejecutando las siguientes li  
neas  
    print(error)  
  
print("hola")
```

```
name 'a' is not defined  
hola
```

In [8]:

```
def suma(a, b="3"):  
  
    try:  
        resultado = a + b  
  
    except:  
        print ("Ha ocurrido un error al sumar")  
        a = int(a)  
        b = int(b)  
        resultado = a + b  
  
    return resultado  
  
s = suma(a=2)
```

```
Ha ocurrido un error al sumar
```

In [13]:

```
def minicalculadora(a, b, operador, DEBUG=0):  
  
    try:  
        a = int(a)  
        b = int(b)  
        if operador == "+":  
            return a + b  
        elif operador == "/":  
            return a / b  
  
    except Exception as error:  
        if DEBUG != 0:  
            print(error)  
        print("Ha ocurrido un error, solo se admiten números")  
  
s = input("Escribe un operador")  
a = input("Escribe un numero")  
b = input("Escribe otro numero")  
  
minicalculadora(a=a, b=b, operador=s)
```

Ha ocurrido un error, solo se admiten números

In []:

```
#Assert
```

```
def mostrar_dinero_banco(password):  
    assert (1==2), "mensaje" #Se utiliza para que algo tan crítico como una contraseña no  
    se introduce correctamente, ocja y pare todo los posterior.  
    return 82748274278
```

In []:

```
#Diferencia entre assert y try, el except.
```

In [14]:

```
try:  
    assert (1==2), "Error 2 no es 1"  
  
except Exception as error:  
    print("error")  
    print ("Ha ocurrido un error")
```

error
Ha ocurrido un error

In []:

```
#Archivos .py
```