

MANEJO DE EXCEPCIONES

1. Por un lado, podemos ser nosotros quien lancemos las excepciones:

También podemos ser nosotros los que levantemos o lancemos una excepción. Volviendo a los ejemplos usados en el apartado anterior, **podemos ser nosotros los que levantemos** `ZeroDivisionError` o `NameError` usando `raise`. La sintaxis es muy fácil.

```
raise ZeroDivisionError("Información de la excepción")
raise NameError("Información de la excepción")
```

2. Por otro lado, podemos capturarlas.

En este ejemplo vemos el uso de los bloques `try`, `except`, `else` y `finally` todos juntos.

```
try:
    # Se fuerza excepción
    x = 2/0
except:
    print("Entra en except, ha ocurrido una excepción")
else:
    print("Entra en el else, no ha ocurrido ninguna excepción")
finally:
    print("Entra en finally, se ejecuta el bloque finally")
```

Except, excepción genérica. El único problema, que no se va a saber el tipo de error. Podemos sustituir por la excepción última que aparece en el documento, que nos permite mostrar el tipo de excepción.

Except ValueError: típica excepción que se utiliza cuando se ha introducido un valor que no es el correcto .Ej: metemos una letra y deberíamos haber metido un número.

Ej:

```
...
while continuar:

    try:
        filas = int(input("Introduce el número de filas: "))
        columnas = int(input("Introduce el número de columnas: "))
        continuar = False
    except ValueError:
        print("Por favor, introduce un número entero válido.")
...

```

Except `TypeError`

```
try:
    d = 2 + "Hola"

except TypeError:
    print("Problema de tipos!")
```

Except `ZeroDivisionError`: Podríamos tratar ese error manualmente de la siguiente forma

```
a = 5
b = 0
# A través de esta comprobación prevenimos que se divida entre cero.
if b!=0:
    print(a/b)
else:
    print("No se puede dividir!")
    # raise ZeroDivisionError("Información de la excepción")
```

Pero lo ideal es usar la excepción:

```
a = 5; b = 0

try:
    c = a/b
except ZeroDivisionError:
    print("No se ha podido realizar la división")
```

Except `OSError`:

```
# Se intenta abrir un fichero y se captura una posible excepción
try:
    with open('fichero.txt') as file:
        read_data = file.read()
# Capturamos una excepción concreta
except OSError:
    print('OSError. No se pudo abrir')
```

Except `Exception` as `ex`

```
try:
    d = 2 + "Hola" # Si comentas esto entra en ZeroDivisionError
except Exception as ex:
    print("Ha habido una excepción", type(ex))

# Lo que devolverá será lo siguiente: Ha habido una excepción <class 'TypeError'>
```