



Aprenda a programar con Python 3

Errores y excepciones

Carolina Mañoso, Ángel P. de Madrid y Miguel Romero

Índice

◆ Errores

◆ Excepciones

◆ Referencias



Errores

- ◆ Los errores de sintaxis son errores que lanza el intérprete cuando no reconoce el código escrito.
 - El interprete muestra donde se produce el error y una descripción del mismo:

```
>>>print("hola)
```

```
SyntaxError: EOL while scanning string literal
```

```
>>>while True print("hola")
```

```
SyntaxError: invalid syntax
```

Excepciones (1/5)

◆ Las excepciones son errores detectados en tiempo de ejecución.

- El mensaje de error indica el tipo de excepción y una explicación:

```
>>>5/0
```

```
Traceback (most recent call last):
```

```
File "<pyshell#1>", line 1, in <module>
```

```
5/0
```

```
ZeroDivisionError: division by zero ➡
```

```
>>>"a"+2
```

```
Traceback (most recent call last):
```

```
File "<pyshell#2>", line 1, in <module>
```

```
"a"+2
```

```
TypeError: Can't convert 'int' object to str implicitly ➡
```

```
>>>3+x
```

```
Traceback (most recent call last):
```

```
File "<pyshell#9>", line 1, in <module>
```

```
3+x
```

```
NameError: name 'x' is not defined ➡
```

Excepciones (2/5)

◆ Si sabemos que una parte del código que usamos puede dar una excepción, se debe programar que hacer en el caso de que ocurra. Para ello se usa `try-except` para capturar y tratar las excepciones:

- `try`: define el fragmento de código en el que creemos que podría producirse la excepción.
- `except`: permite indicar el tratamiento que se llevará a cabo de producirse una excepción.

```
try:
```

```
    codigo donde podría haber un problema
```

```
except type_of_error:
```

```
    codigo a ejecutar si hay un error
```

```
try:
```

```
    f = file("archivo.txt")
```

```
except:
```

```
    print("el archivo no existe")
```

Excepciones (3/5)

- Para encontrar el tipo de error, se provoca la excepción:

```
Traceback (most recent call last):
```

```
File "<pyshell#0>", line 1, in <module>
```

```
    num = int(input("enter a number: "))
```

```
ValueError: invalid literal for int() with base 10:'hola'
```

- Una declaración `try` puede tener mas de un `except` para tener manejadores para distintas excepciones. Sólo uno será ejecutado.
- El último `except` puede omitir el nombrar el tipo de excepción, para servir de comodín.
- Esta construcción puede tener un bloque `else` opcional después de los `except` que se ejecutará si no se genera excepción.

Excepciones (4/5)

```
try:
    x = int(input("introduzca un numero "))
except ValueError:
    print("eso no es un número")
except:
    print("error inesperado")
else:
    print("muy bien")
```

Excepciones (5/5)

- `try` tiene la clausula opcional `finally` que se ejecuta siempre antes de salir de la declaración, con la intención de realizar tareas de limpieza.

```
def dividir(x,y):  
    try:  
        result = x / y  
    except ZeroDivisionError:  
        print("division por cero!!!")  
    else:  
        print("El resultado es", result)  
    finally:  
        print("ejecuto finally")
```

```
dividir(6,3)
```

```
dividir(6,0)
```



Referencias

- ◆ La documentación oficial, *Python Tutorial Release 3.5.2*. Guido van Rossum and the Python development team. Python Software Foundation. Junio 25, 2016.
<https://docs.python.org/3/download.html>
- ◆ *Learning Python 5th edition*. Mark Lutz. O'Reilly 2013
- ◆ *Practical Programming: An Introduction to computer Science using Python 3*. P. Gries, J. Campbell & J. Montoyo. Edited by Lynn Beighley. 2nd Edition. The Pragmatic Bookshelf. 2013
- ◆ *Learning Python with Raspberry Pi*. Alex Bradbury, Ben Everard. Wiley. 2014
- ◆ Tutorial: *Python 3 para impacientes*. Antonio Suárez Jiménez. 2014
<http://python-para-impacientes.blogspot.com.es/p/indice.html>
- ◆ *Raspberry Pi, User Guide*. Eben Upton, Gareth Halfacree. Wiley. 2012

Aviso



Aprenda a programar con Python 3 by C. Mañoso, A. P. de Madrid, M. Romero is licensed under a [Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional License](https://creativecommons.org/licenses/by-nc-sa/4.0/).

Esta colección de transparencias se distribuye con fines meramente docentes.

Todas las marcas comerciales y nombres propios de sistemas operativos, programas, hardware, etc. que aparecen en el texto son marcas registradas propiedad de sus respectivas compañías u organizaciones.