Programming fundamentals with Python

Pepe García

2020-04-20

Programming fundamentals with Python

https://slides.com/pepegar/pfp-17/live

Plan for today

Error handling

Exceptions

Files

Errors

What errors have you found in Python?

Errors

object NoneType does not have...

KeyError

TypeError: ... expects 2 positional arguments, 1 given

ArithmeticError (division by zero)

Exceptions

Exceptions are detailed errors that happen in Python. There are lots of them:

Exceptions

```
>>> 0 / 0
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ZeroDivisionError: division by zero
```

Exceptions

```
>>> my_dictionary = {"a": 3}
>>> my_dictionary["b"]
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
KeyError: 'b'
```

Exceptions

```
>>> open("potato.txt")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
FileNotFoundError: [Errno 2] No such file or directory: 'potato.txt'
```

Exceptions

```
>>> assert 3 == 4
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AssertionError
```

Exceptions

There are two important parts to exceptions

They can be raised

They can be handled

Raising exceptions

```
raise Exception("message")
```

Raising exceptions

raising exceptions

assert

Handling exceptions

For handling exceptions we use a try-except block:

```
>>> try:
...     raise Exception("this is the message")
... except Exception:
...     print("potato")
...
potato
```

Handling exceptions

Create a program that asks the user for their age and tells their age in 10 years.

The age should be a number

The age should be positive

Reading files

```
file = open("file_path")

for line in file:
    #do something with line
    pass
```

Reading files

Create a function that counts the number of lines in a file .

Control errors that may occur if the file doesn't exist!

Exercises

Exercise 1

Modify your ecommerce example so, instead of returning None when two services of the same type happen on checkout, raises an exception

Exercise 1

Create a function that reads through a file and prints all the lines in uppercase.

be sure to control exceptions that may occur here, such as the file not existing

Exercise 3

Create a function to **copy files**.

The function must receive two names (origin and destination, for example), and copy the contents of origin into destination.

You'll need to **investigate** how to write files for this exercise