Libraries and Sockets

February 6, 2020

What we will cover today

- Libraries? What are they and how we use them in Python
- Exploring the socket library
- A cool project with socket

Libraries

A library is a collection of code that can contain functions, objects and variables which we wish to use in different programs. Libraries make our life easier because it's code we can use in our programs quickly without having to "re-inventing the wheel".

Example of library

A very common example of a library is the library `math` which contains a lot of helper functions to do math.

```
import math

print(math.sqrt(2))
```

To import a library we use the keywork `import` followed with the name of the library we want to use. Once a library is imported we can use it by calling the name of the library `.` whatever function/variable/object we want.

Selective import

Some libraries are very big and sometimes we only wish to import a certain thing from them. We can import a specific thing from a library without loading the whole thing like this:

```
from math import sqrt

print(sqrt(2))
```

We use the keyword `from` followed by the name of the library, then `import` and whatever we want to import from the library. In this case we import the function `sqrt`, which can be used by simply calling it. If we want to import multiple things from a library we simply separate them with `,` or if we want to import everything without having to use the name of the library every time we can use this : `from math import *`

Renaming libraries

If we are feeling very lazy we can rename libraries so we don't have to use the name everytime we want to use something:

```
import math as m

print(m.sqrt(2))
```

We use the keyword `as`.

Socket library

The socket library is a library used for anything regarding low-level networking with python. This is how we import it:

```
import socket
```

Today we will be creating a simple client-server application which allows us to send data to each others when we are on the same network

Working with interactive Python

To play around with socket we will use the python3 interactive shell which you can access by running `python3` on your terminal. On this shell you can write and run python code quickly without having to create a script.

First of all choose a friend to work with and find out each others local ip address.

The server

```
>>> import socket
>>> s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
>>> s.bind(("0.0.0.0", 4444))
>>> s.listen()
>>> conn, addr = s.accept()
>>> print("Connected with", addr)
>>> data = conn.recv(1024)
>>> print("Received:", str(data))
>>> conn.sendall(b'Your reply')
```

The client

```
>>> import socket
>>> s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
>>> s.connect(("friend.ip.0.0", 4444))
>>> s.sendall(b'Your message')
>>> data = s.recv(1024)
>>> print("Received:", str(data))
```

Challenge

Your challenge for today is to create a more interactive program to communicate with your friend. Right now we can only send one fixed message and one fixed reply, how can we improve this?