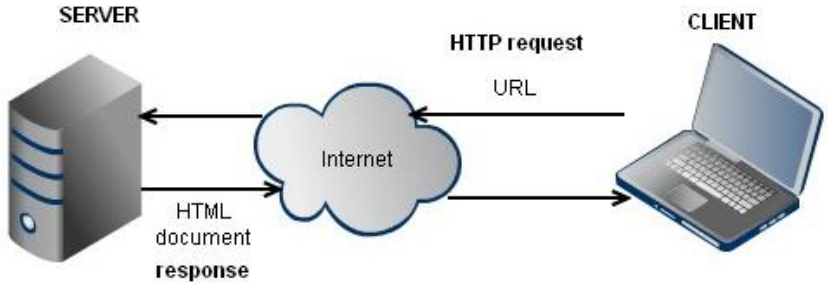
Lesson 16 – Introduction to Web Servers

Introduction to Web servers:

- Read: https://www.fullstackpython.com/web-servers.html (up to **Building web servers**)



Welcome to Flask:

- Take a quick look at this micro-web-framework for Python: http://flask.pocoo.org/

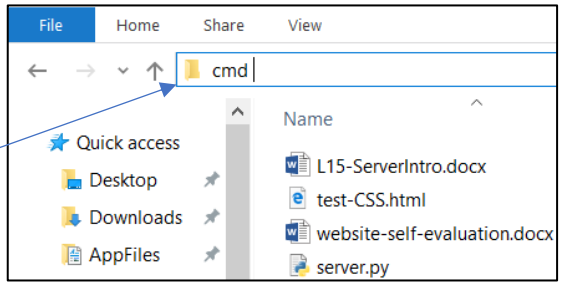
Flask
web development,
one drop at a time

Task 1:

- Establish a simple Flask web-server on your computer.

Instructions:

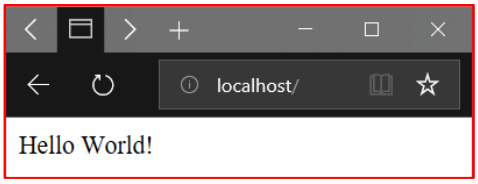
0. Make sure that the Flask Python module is installed:
   - Open command line by pressing ⊞ key, writing **cmd** and pressing enter.
   - Run command >>> `pip install Flask`
   - Close the command line.
1. **Download** the script **server.py** from NAS and **open** it in Notepad++ (or any other editor).
2. Open the **folder** with the downloaded script.
3. Open the **command line** (from the same folder as the script is located) by typing **cmd** into the explorer **address bar** and press enter.



4. Run the command >>> `python server.py`

```
E:\Data\Documents\L15-Server Intro>server.py
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 362-398-215
 * Running on http://0.0.0.0:80/ (Press CTRL+C to quit)
127.0.0.1 - - [17/Jan/2019 20:53:13] "GET / HTTP/1.1" 200 -
```

5. Open the **web browser** at an address http://localhost:80/
6. If everything goes well, you should see a **Hello World** message.



Task 2:

- Enable the server to serve **arbitrary HTML code**.

Instructions:

1. Look at the script **code** in Notepad++. Try to change the line `return "Hello World!"` to some valid HTML code such as:
   `return "<a href='https://gymes.edupage.org/'>GYMES</a>"`
2. **Save** the modified script file.
3. Return to the **command line** and press **CTRL+C to** kill the server.
4. **Restart** the server via command >>> `python server.py`
5. Refresh the **web browser** at an address http://localhost:80/
6. If everything goes well, you should see a **link to our school website**.

Task 3:

- Enable the server to serve content **at arbitrary URLs**.

Instructions:

1. Read about URL routing at http://flask.pocoo.org/docs/1.0/quickstart/#routing
   (up to **Variable Rules**)
2. Try to create these subpages:
   - "Hello subpage 1!" at URL http://localhost:80/subpage1
   - "Hello subpage 2!" at URL http://localhost:80/subpage2
   - GYMES website *inside an iframe* at URL http://localhost:80/gymes

---

Optional:

- Try to connect to your server from another computer.

Instructions:

1. Find out the IP address of the server PC by using command >>> `ipconfig`
2. Disable computer firewall or add an exception to allow incoming outside connections to the Flask app on port :80 which is used to handle HTTP communication.
3. Open the browser on the client PC and connect to http://SERVER.IP.ADDRESS:80/
