Assignment 3:
Web Client and Server (HTTP Protocol)

- This assignment will be introduced in a class meeting on November 08, 2016 at 11:30 in seminar room 0.124.
- The groups will present the solution of this assignment on November 15, 2016 at 11:30.
- You may work on the assignments in groups of up to three people (if possible, keep the same groups as in the previous assignments).
- All members of a group have to show up together for the grading of the assignment. For the grading, each group will need to have the source code ready for the discussion as well as the running implementation to be presented by the group. Moreover, each group member might be asked questions about the solution.
- If you have questions, send me an email to adnan.tariq@ipvs.uni-stuttgart.de.

Task 1 – Web Client

In this task, a simple web client shall be implemented. Given a URL, the web client retrieves a web page (HTML or plain text format) or an image (JPG format) from a web server using the HTTP protocol and stores it as a file on the local hard disk (you don't have to implement a GUI for displaying the content). The typical interaction between your client and the web server looks as follows:

1. The client reads the URL of the web page or image to be fetched from the server from the command line (using, for instance, `System.in`).
2. The client opens a connection (stream socket) to the web-server.
3. The client sends an HTTP request to the server according to the HTTP standard.
4. The server responds with an HTTP reply.
5. The client reads and interprets the response and stores the file on its local hard disk.

Use the Java programming language for your implementation. Don't use a class of the Java class library such as `URLConnection` that implements the HTTP interaction! Instead implement the HTTP interaction yourself by sending a suitable HTTP request via the stream socket and interpreting the HTTP response from the server.

Your client should be able to deal with errors such as a non-existing file at the web-server. In case of an error, output a message with the HTTP error code on the command line (`System.out`).

Your client only needs to support the content types HTML, plain text, and JPG images.

You can use HTTP version 1.0 or 1.1.

(Task continued on next page ➔)

Test your client with real web servers using the following URLs:

- http://www.priloc.de/
- http://www.priloc.de/img/priloc.jpg
- http://www.ipvs.uni-stuttgart.de/abteilungen/vs/abteilung/mitarbeiter/john.doe/de
- http://www.google.com/

Task 2 – Web Server

In this task, a simple web server shall be implemented. This server only has to implement the HTTP method for retrieving web pages from the server (not for posting data to the server, etc.). If the server receives such a request, it fetches the requested file from its local hard disk and sends it back to the client using a suitable HTTP response. If the file does not exist, the server should respond with a suitable HTTP error message. For all other requests, the server should respond with the error "(functionality) not Implemented" according to the HTTP protocol.

Again, you are to implement the HTTP interaction yourself using socket programming and by implementing the necessary subset of the HTTP protocol (without using high-level classes implementing the HTTP interaction).

Your server should be able to handle HTML pages and JPG images that are stored in a special folder (the "root" directory of your web server).

To test your server, fetch an HTML file and an image file (JPG) from your server using a common web browser such as Firefox, Internet Explorer, Google Chrome, etc. Additionally, you may also use the client from Task 1.

Note: The HTTP header can contain many different options. It is not necessary to include every possible option, but only the essential ones.