

U 5.1 Java Sockets: HTTP-Server (single-thread)

Implementieren Sie einen HTTP-Server, der Anfragen von Clients in einem einzigen Thread behandelt. Der Einfachheit halber soll sich die Funktionalität des Servers auf HTTP-Version 0.9 (<http://www.w3.org/Protocols/HTTP/AsImplemented.html>) stützen, die auf das Ausliefern einzelner Dateien beschränkt ist.

Der Server soll über Kommandozeilenparameter konfiguriert werden. Dabei müssen sowohl TCP-Port als auch Basisverzeichnis (document root) des HTTP-Servers beim Start angegeben werden.

Hinweise:

- Beachten Sie, dass diese ursprüngliche HTTP-Version keine Möglichkeit bietet, Fehlermeldungen an den Client zu übertragen. Signalisieren Sie daher einen Fehler einfach dadurch, dass der Server nichts ausliefert und die Verbindung schließt.
- Sollte ein angegebenes Dokument nicht existieren, so schließt der Server einfach die Verbindung, ohne Daten auszuliefern.
- Sollte ein Verzeichnis angefordert werden, so soll der Server die in diesem Verzeichnis befindliche Datei index.html zurückgeben. Ist diese nicht vorhanden, so wird wiederum nichts an den Client retourniert.
- Die Beschreibung (<http://www.w3.org/Protocols/HTTP/AsImplemented.html>) geht davon aus, dass nur Text zurückgeliefert wird. Die aktuellen Browser unterstützen aber auch, dass binäre Daten wie z.B. Bilder ausgeliefert werden.

Im Moodle-Kurs befinden sich ein paar Testdaten (documentRoot.zip), mit denen Sie Ihren Server mit einem Browser Ihrer Wahl testen können.

U 5.2 Java Sockets: HTTP-Server (multi-thread)

Erweitern Sie die Funktionalität des Servers aus Ü 5.1 wie folgt. Anstatt die Anfragen von Clients seriell in einem Thread zu behandeln, soll jede Anfrage durch einen neuen Thread behandelt werden.

Weiters ist der Server durch eine Logging-Komponente zu erweitern. Jeder Zugriff auf den Server (Zeitpunkt, Request, IP und Port des Clients) soll dabei mitprotokolliert werden. Dieses Logging soll zentral durch einen eigenen Thread realisiert werden, der in Abständen von 5 Sekunden das Zugriffsprotokoll auf die Konsole schreibt. Die Zugriffe müssen daher mit geeigneten Mechanismen von den behandelnden Threads an diesen Logging-Thread kommuniziert werden. Sorgen Sie für eine geeignete Synchronisation der Threads.

Beispiel-Ausgabe:

```
2011-09-26 15:40:10 GET /index.html 127.0.0.1 44239
2011-09-26 15:40:10 GET /images/logo.gif 127.0.0.1 44240
2011-09-26 15:40:10 GET /images/TechnikErleben.png 127.0.0.1 44241
```