

Arbeitsblatt: Schlechte Witze

Im Unterricht haben Sie gelernt wie man Konsolen-I/O programmiert und wie man Dateien lesen und schreiben kann. In diesem Arbeitsblatt lernen Sie, wie man aus einem Haskell Programm Internetdienste über HTTP ([Hypertext Transfer Protocol](#)) abfragen kann.

HTTP ist ein Protokoll zur Übertragung von Daten über ein Netzwerk. Es wird hauptsächlich eingesetzt um Webseiten aus dem Internet in einen Webbrowser zu laden. Die Kommunikation basiert dabei auf einem einfachen Anfrage-Antwort-Modell (Request-Response). Der Browser fordert eine Seite von einem Server an und bekommt als Antwort das HTML mit dem darzustellenden Inhalt.

Dieses Modell lässt sich aber nicht nur für HTML Internetseiten verwenden, sondern kann ganz allgemein verwendet werden um auf Netzwerkdienste zuzugreifen. Viele Internetdienstleister bieten Netzwerkschnittstellen an um programmatischen Zugriff auf die Dienste zu gewähren.

The Internet Chuck Norris Database z.B. hat einen Dienst um Chuck Norris Sprüche/ Witze abzufragen. Die Sprüche können auf einen beliebigen Namen personalisiert werden. Das API ist beschrieben unter <http://www.icndb.com/api/>. So können Sie z.B. einen Spruch über Ihren fprog Dozenten abholen:

```
http://api.icndb.com/jokes/random?limitTo=[nerdy]&firstName=Daniel&lastName=Kroeni
```

Sie können diese URL mit dem Browser ausprobieren. Das Resultat ist dann keine schöne Webseite sondern ein JSON Object.

```
{
  "type": "success",
  "value": { "id": 478,
             "joke": "Daniel Kroeni can instantiate an abstract class.",
             "categories": ["nerdy"]
           }
}
```

In diesem Arbeitsblatt programmieren Sie nun ein Haskell Programm um beklopfte Sprüche abzuholen. Hinweis: Die eckigen Klammern '[' und ']' in der URL müssen durch %5B und %5D ersetzt werden.

Damit die HTTP Library verfügbar ist, wurde diese als Abhängigkeit in der WS-Webclient.cabal Datei deklariert. Wechseln Sie in das Verzeichnis WS_Webclient. Folgende Kommandos sind nun nützlich:

Start von GHCi (wobei alle abhängigen Bibliotheken zur Verfügung stehen)

```
> cabal repl
```

Bauen und ausführen der main Funktion in der Datei Main.hs und Übergabe von zwei Kommandozeilenargumenten (arg0 und arg1)

```
> cabal run webclient arg0 arg1
```

Durch den Import von `Network.HTTP` stehen die notwendigen Funktionen zur Verfügung (die Typsignaturen sind vereinfacht dargestellt):

-- Erzeugt aus einer URL (String) eine Anfrage.

`getRequest :: String -> Request`

-- Nimmt eine Anfrage und erzeugt ein I/O Programm, das eine Antwort liefert.

`simpleHTTP :: Request -> IO Response`

-- Nimmt eine Antwort und liefert deren Hauptinhalt

`getResponseBody :: Response -> IO String`

Aufgabe:

Implementieren Sie ein Programm, das einen weichen Spruch von der ICNDB lädt. Vor- und Nachname soll als Programmargument beim Aufruf mitgegeben werden können.