

Informatik I: Einführung in die Programmierung

21. Das WWW befragen

Albert-Ludwigs-Universität Freiburg



Bernhard Nebel

13.01.2015

1 Motivation



Motivation
Webseiten
und HTML
Das urllib-
Paket

13.01.2015

B. Nebel – Info I

3 / 17

Motivation



Motivation
Webseiten
und HTML
Das urllib-
Paket

- Oft braucht ein Programm Informationen, die es im **WWW** einfach zu finden gibt.
- Dazu müsste man bloß kurz eine Webseite aufrufen und ein Detail nachschlagen.
- Zum Beispiel wollen wir die **aktuelle Temperatur** wissen.
- Könnte das nicht ein kleines Skript für uns tun?
- Auf **<http://www.wetteronline.de>** findet man die aktuelle Temperatur ziemlich weit oben auf der Seite.

13.01.2015

B. Nebel – Info I

4 / 17

2 Webseiten und HTML



Motivation
Webseiten
und HTML
Das urllib-
Paket

13.01.2015

B. Nebel – Info I

6 / 17

- Alle Webseiten bestehen aus Texten (und Bildern) mit **HTML**-Formatanweisungen (*Hypertext markup language*).
- Die HTML-Anweisungen beschreiben, wie bestimmte Textteile **erscheinen** sollen.
- HTML-Formatanweisungen kommen normalerweise in **Paaren**, z.B. `<h1>` und `</h1>` für Überschriften.
- Generell wird eine öffnende Markierung `<mark>` durch eine schließende Markierung abgeschlossen: `</mark>`.
- Bei der öffnenden Markierung werden oft noch weitere Attribute angegeben, z.B. `<table border="2">`.
- Außerdem können die Dateien weitere Formatanweisungen (**CSS**) und aktive Komponenten (**JavaScript**) enthalten.
- Eine gute Einführung findet sich z.B. auf <http://de.selfhtml.org/>.

Motivation
Webseiten
und HTML
Das urllib-
Paket

HTML page

```
<!DOCTYPE html> <!-- kann hier auch mehr stehen -->
<html> <!-- Jede HTML-Seite beginnt damit -->
<head> <!-- leitet Head-Sektion ein -->
<meta ...>
</head>
<body> <!-- hier nach folgt der Seitentext -->
... <!-- der verschiedene Markierungen nutzt -->
</body>
</html>
```

Motivation
Webseiten
und HTML
Das urllib-
Paket

- Man kann sich den **Quellcode** der Webseite anschauen.
- Normalerweise findet man schnell ein **Pattern**, das zutreffend ist.
- Schauen wir uns den Quellcode der <http://www.wetteronline.de/freiburg>-Seite an.
- Seite anwählen, dann rechts klicken und **Quelltext anschauen** wählen; ggfs. Text vorher markieren.
- Nach dem Text **suchen**.
- Pattern konstruieren!

Motivation
Webseiten
und HTML
Das urllib-
Paket

- Am besten nach **id=...name** schauen, da diese eindeutig auf der HTML-Seite sind.
 - Bei uns ist folgende Zeile **relevant**:
`<div id="current-weather"> ...`
 - **Möglicher** regulärer Ausdruck:
`r'<div[^>]*id="current-weather">\s*<span>aktuell</span><span class="temperature tooltip gt0">(\d+)°C</span>'`
 - ... zumindest solange sich nichts ändert ...
 - ... zund die Temperaturen über Null liegen ...
 - Aber wie kommen wir an die **Webseite**?
- **urllib**

Motivation
Webseiten
und HTML
Das urllib-
Paket

3 Das urllib-Paket



Motivation
Webseiten
und HTML
Das urllib-
Paket

Das urllib-Paket



Motivation
Webseiten
und HTML
Das urllib-
Paket

- Das **urllib-Paket** bietet komfortable Schnittstellen, um auf Ressourcen im WWW zuzugreifen.
- Das Paket enthält mehrere Module:
 - **urllib.request**: Enthält Funktionen und Klassen zum Zugriff auf Ressourcen im Internet.
 - **urllib.parse**: Unterstützt das Parsen von URLs (*Universal Resource Locators*).
- Die wichtigsten Funktionen aus `urllib.request` ist:
 - `urlopen(url, data=None, timeout=*, cafile=None, capath=None, cadefault=False)`: Stellt ein Datei-ähnliches Objekt zur Verfügung. `url` ist die URL, auf die zugegriffen werden soll; `data` sind zusätzliche Daten, die bei einer Anfrage geschickt werden; `timeout` ist ein optionaler Parameter für eine obere Zeitschranke. Die anderen Parameter sind für Zertifikate (bei HTTPS).
 - Nach `urlopen` kann man auf dem resultierenden Objekt `read`-Methoden anwenden und erhält **bytes** zurück.

Eine Webseite anschauen



Motivation
Webseiten
und HTML
Das urllib-
Paket

```
wetter.py
from urllib.request import urlopen

showlines = 10
remotefile = urlopen("http://www.wetteronline.de/")
# method to get info about connection
print(remotefile.info())
# read all lines
remotedata = remotefile.readlines()
remotefile.close()
for line in remotedata[:showlines]:
    print(line)
```

Browser vortäuschen



Motivation
Webseiten
und HTML
Das urllib-
Paket

- So funktionierte es jedenfalls letztes Jahr.
- Heute kommt eine Fehlermeldung "HTTP Error 403: Forbidden"
- Webseitenbetreiber mögen keine Zugriffe über Skripte (s.u.).
- Vortäuschung falscher Tatsachen:

```
wetter1.py
...
from urllib.request import Request
req = Request(url="http://www.wetteronline.de/",
    data=b'None', headers={'User-Agent': 'Mozilla/5.0 \
(Windows NT 6.1; WOW64; rv:12.0) Gecko/20100101 \
Firefox/12.0'})
remotefile = urlopen(req)
...
```

- Das täuscht einen Firefox-Browser vor.

temperature.py

```
import re
...

remotedata = remotefile.read().decode('utf8')
remotefile.close()
rx = re.compile(r'<div[^>]*id="current-weather">\s*&nbsp;\s*<span>aktuell</span>\s*<span class=\s*"temperature tooltip gt0">(\d+)&deg;C</span>',
re.I+re.M)

print("Die Temperatur beträgt zur Zeit",
      rx.search(remotedata).group(1),
      "Grad Celsius")
```

Motivation

Webseiten
und HTML

Das urllib-
Paket

- Auf diese Weise, die man *Scraping* nennt, kann man beliebige interessante Informationen von Webseiten sammeln und z.B. per E-Mail verschicken.
- Zum Beispiel: Was gibt es heute in der Mensa?
- Aber **Vorsicht**:
 - Webdesigner ändern gerne öfter mal das **Seitenlayout**.
 - Seitenbetreiber lieben das Scraping nicht, speziell wenn es zu **starker Belastung des Webserver**s führt.
 - Das umfangreiche **Kopieren** und auf eigener Webseite zur Verfügung stellen ist im Übrigen Missbrauch!
 - Manchmal gibt es Maßnahmen gegen den Zugriff durch Skripte.
- Manche Seitenbetreiber bieten auch **Webservices** an, über die man dann per definierter Schnittstelle maschinenlesbar Daten bekommen kann (Beispiel: der Wetterservice www.wunderground.com).

Motivation

Webseiten
und HTML

Das urllib-
Paket