UNIVERSITÄT
BERN

Web Scraping & APIs

CAS in Applied Data Science

Martina Jakob and Sebastian Heinrich

26.08.2021

Learning Goals for this Course

- Working knowledge of data transmission on the internet
- Know the basics of web technologies, especially HTML

- Know how to download webpages with the Requests library
- Know how to parse HTML and extract content with BeautifulSoup and Pandas

- Working knowledge of APIs
- Know how to access the Wikipedia API as an example of an API

Obviously, it's not possible to learn everything about the internet, web technologies, web scraping and APIs in just a few hours. But we hope to provide you with enough basic understanding so that you can keep learning on your own.

Course Overview

- How the internet works
  - Computer networks: Clients and servers
  - Components of a website
  - A closer look at HTML

- Web scraping
  - Scraping pages with the requests package
  - Extracting data with BeautifulSoup
  - Extracting tables with Pandas

- APIs
  - What is an API?
  - Making API calls with the requests package
  - Using the Wikipedia API

What is Web Scraping?

UNIVERSITÄT BERN

"Web scraping, web harvesting, or web data extraction is data scraping used for extracting data from websites."

"Data scraping is a technique where a computer program extracts data from human-readable output coming from another program."

Source: Wikipedia

Why would we want to do Web Scraping?

UNIVERSITÄT BERN

Imagine you have a list of domain names of e.g., companies or organizations and you want to analyze or monitor their websites' content.

To achieve this, you need to:

- Download a possibly very large number of pages per domain
- Repeat this process, e.g., every month, day or even hour

To do this manually would be:

- very time-consuming
- error-prone
- simply not be feasible for vast amounts of pages

Web scraping allows you to retrieve the contents of web pages automatically.

A few things about the Internet and Web Technologies

UNIVERSITÄT BERN

Before we start practicing with Python we will look into a few key topics:

- Basics of data transmission over the internet
- Components of a web page
- Structure of the HTML language

You are able to do web scraping without much knowledge of how the internet and the web works. But we think it helps a lot to have at least a basic understanding of the underlying processes and technologies.

Computer Networks: Clients and Servers

UNIVERSITÄT BERN

You can think of the internet as a network that connects two types of computers: **clients** and **servers**.

The clients are the ones requesting the information. For example, your computer or your smartphone are clients.

Servers are where the information is stored. They are also computers (e.g. a single computer or a whole data center), but they are set up to store and deliver data for the clients.

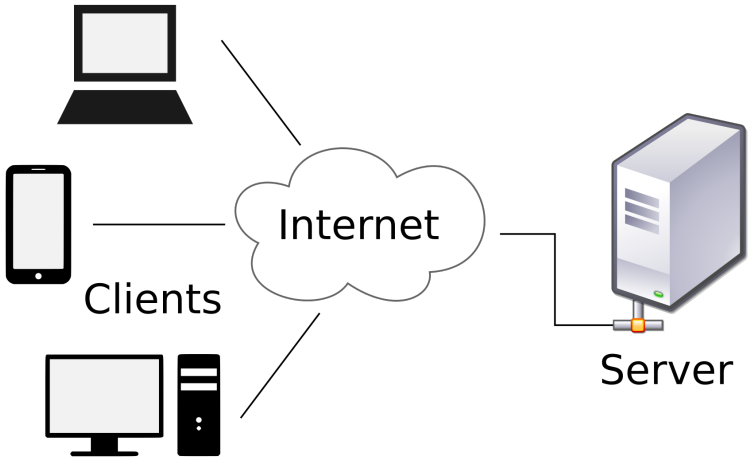


Image source: Wikipedia

What happens if you type an URL into your browser?

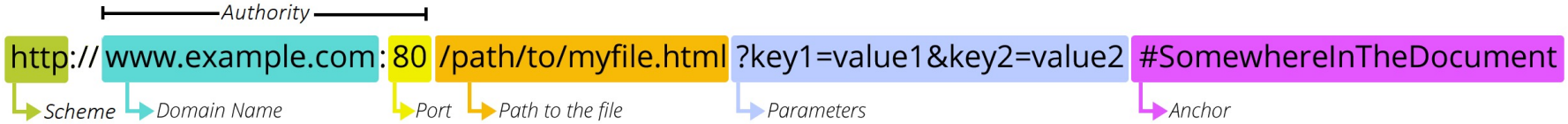
UNIVERSITÄT BERN

When you type www.google.ch into your browser, your computer (i.e. a client) is requesting information from the Google server with a HTTP request. But how does it know where to find this server?

- Every domain has a corresponding **IP (Internet Protocol) address**
- The **Domain Name System (DNS)** resolves IP addresses and domain names, kind of like a phone book
- With a **HTTP (Hypertext Transfer Protocol) request** the client is asking the server to deliver data (e.g. a webpage)

Anatomy of a URL

URLs are a central part to navigate the web. Here is a quick overview on the parts they may contain:



Source and more details:
https://developer.mozilla.org/en-US/docs/Learn/Common_questions/What_is_a_URL

HTTP (Hypertext Transfer Protocol) request

The HTTP request will be passed through several layers of protocols and eventually be transformed into signals that can be sent through the telecommunication infrastructure that spans the globe (e.g. fiber cables, satellites etc.).

```
GET / HTTP/2
Host: www.google.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:88.0) Gecko/20100101 Firefox/88.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
TE: Trailers
```

If everything worked well, the server will produce a HTTP response containing the contents of the page the client requested and send it back to your computer.

What are the components of a webpage?

UNIVERSITÄT BERN

Webpages are usually a combination of documents in three languages:

HTML – HyperText Markup Language: Structured contents of the page
- What are the headings of the page?
- What do the different paragraphs say?
- What images do we have?

CSS – Cascading Style Sheets: Styling of the page
- What font size and type should headings have?
- How should the paragraphs be styled?

JS – JavaScript: Interactivity of the page
- Show or hide more information with the click of a button
- Slide through a carousel of images

How does HTML work?

UNIVERSITÄT BERN

HTML Documents

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

HTML Tags

<html>	Opening tag of every HTML document
<head>	Document head
<title>	Page title (child of head tag)
<body>	Page content
<h1>	Section heading <p>: Paragraph
<p>	Paragraph
<a>	Link
<table>	Table

Tag Attributes

<a **href**="www.someaddress.com" **target**="_blank">
Click on this link

Web scraping in a few simple steps

UNIVERSITÄT BERN

requests to download HTML files

https://en.wikipedia.org/wiki/Cat

URL

```
<table class="infobox biota" style="text-align: left; width: 200px; font-size: 100%">…</table>
<p>
  The
  <b>cat</b>
  (
  <i>Felis catus</i>
  ) is a
  <a href="/wiki/Domestication" title="Domestication">domestic</a> event
  whitespace
  <a href="/wiki/Species" title="Species">species</a> event
  of small
  <a class="mw-redirect" href="/wiki/Carnivorous" title="Carnivorous">carnivorous</a> event
  whitespace
  <a href="/wiki/Mammal" title="Mammal">mammal</a> event
  .
  <sup id="cite_ref-Linnaeus1758_1-1" class="reference">…</sup> event
  <sup id="cite_ref-MSW3fc_2-1" class="reference">…</sup> event
  It is the only domesticated species in the family
  <a href="/wiki/Felidae" title="Felidae">Felidae</a> event
```

HTML file

BeautifulSoup to parse HTML and extract specific content

```
>> soup.find("h1")
<h1 id="firstHeading" class="firstHeading">
Cat</h1>
```

Content

UNIVERSITÄT BERN

We continue in the Jupyter Notebooks

Please open the file **1-Web_Scraping_Tutorial.ipynb** in Google Colab or your local Jupyter instance.

We start with the section **Scraping web pages with requests**.

What is an API?

UNIVERSITÄT BERN

"An application programming interface (API) is a connection between computers or between computer programs. It is a type of software interface, offering a service to other pieces of software."

Source: Wikipedia

APIs are a very broad concept, don't let this confuse your understanding of what we are actually doing with them in our context – it's not very complicated.

In the previous part, we learned how to automatically retrieve pages from the internet through web scraping. For many of the more popular websites such as Wikipedia, Youtube, Twitter or many newspapers, there is a more direct way to retrieve the information we need: through APIs.

Why would we want to use an API?

UNIVERSITÄT BERN

- Data is made accessible so that other applications/programs (for example your Python script) can work with it conveniently.
- APIS allow you to access content in a more structured way than scraping HTML pages. They are mostly well defined and documented sets of URLs.
- They return structured data that won't contain the presentational overhead that is required for a graphical user interface like a website.
- Allow to take more control over the content format you will receive.
- Allow requests not feasible with web scraping, like advanced text or geo search.

Accessing an API in a few simple steps

UNIVERSITÄT BERN

requests to download JSON files

https://en.wikipedia.org/wiki/Cat

URL

JSON file

Python's standard library to parse JSON and extract specific content

```
>> data["parse"]["title"]
'Cat'
```

Content

APIs and Web Scraping: What's the difference?

UNIVERSITÄT BERN

https://en.wikipedia.org/w/api.php?action=parse&page=Cat&format=json

URL

https://en.wikipedia.org/wiki/Cat

JSON file

requests for download

HTML file

Process JSON with Python standard library

```
>> data["parse"]["title"]
'Cat'
```

Process HTML with BeautifulSoup

```
>> soup.find("h1")
<h1 id="firstHeading" class="firstHeading">
Cat</h1>
```

UNIVERSITÄT BERN

We continue in the Jupyter Notebooks

Please open the file **2-APIs_Tutorial.ipynb** in Google Colab or your local Jupyter instance.

We start with the section **Making API requests with the requests library**.

UNIVERSITÄT BERN

Thanks for your attention!

If you have any questions, don’t hesitate to contact us:

martina.jakob@unibe.ch
heinrich@kof.ethz.ch