

# Web Scraping for Scientists: An Introduction with Python

This version: 2023-1

## Course instructor

**Name: Prof. Dr. Jens Förderer**

Room: L230, Heilbronn Campus

Tel.: +49 7131 264 18 802

Mail: [office.cdt@wi.tum.de](mailto:office.cdt@wi.tum.de)

<https://www.wi.tum.de/prof-dr-foerderer>

## Application procedure

Course will be held online

### Goal and target audience

Doctoral students and post-docs in business.

### Application process

Please send an email to the above stated address with a registration request that includes your name (see below) and your TUM eMail-address.

**Registration deadline: 21.04.2023**

## Prerequisites

During the course, we will apply the programming language Python to develop web scrapers.

Course participants are required to be *familiar* with the basic concepts of Python programming. You are *familiar* with Python if you have a Python environment set up and running, and are able to create lists, methods, conditional statements (if..else), and loops.

If you feel you cannot answer these questions, please take the following modules of the tutorial by W3Schools (<https://www.w3schools.com/python/default.asp>) before the course (requires approximately one day of work):

- Syntax
- Variables
- Data types
- Operators
- Lists
- If...Else
- While-Loops
- For-Loops
- Classes/Objects
- PIP

## Course aims

The course ...

- (1) makes participants familiar with the problem of collecting massive data from Internet sources,
- (2) guides participants to evaluate the costs and benefits of automating data collection,
- (3) introduces participants to the structure of web sites,
- (4) reviews the most effective approaches for collecting data from web sources,
- (5) provides hands-on implementations using Python, and
- (6) outlines ethical and methodological considerations.

## Course objectives

- Participants are familiar with the problem of collecting massive data from the Internet
- Participants can evaluate the costs and benefits of developing web scrapers
- Participants understand the structure of web sites and, from that, are able to derive the requirements for a web scraper
- Participants can program an own web scraper with Python
- Participants are aware of ethical and methodological considerations

## Preliminary schedule

Course will be held **online via Zoom**. Login details will be distributed after registration.

27.04.2023, 09:00-17:00: Day 1 (Fundamentals, HTML, Crawling, Fetching, Parsing)

28.04.2023, 09:00-17:00: Day 2 (Advanced Scraping, Methodological and Ethical Issues)

One Q&A meeting in May (date will be coordinated with the participants)

## Course procedures

The course will begin with two days of input by the instructor and discussions. Afterwards, participants work in groups on their assigned exercise. We will have a Q&A session to clarify questions regarding the exercise.

## Assessment

Form of assessment: group exercise (100%)

Participants will be assigned into groups and work on an exercise, which has to be completed and submitted in the weeks after the course introduction. The goal of the exercise is to develop a web scraper.

## References

Programming:

Chapagain, A. (2019). Hands-On Web Scraping With Python: Perform Advanced Scraping Operations Using Various Python Libraries And Tools Such As Selenium, Regex, And Others. Packt Publishing Ltd.

Mitchell, R. (2018). Web Scraping With Python: Collecting More Data From The Modern Web. O'Reilly Media, Inc.

#### Applications:

Foerderer, J., Lueker, N., Heinzl, A. (2021): And the Winner is ...? Platform Governance, Awards, and Complementors' Product Strategies. *Information Systems Research*, 32(4).

Foerderer, J. (2020): Interfirm Exchange and Innovation in Platform Ecosystems: Evidence from Apple's Worldwide Developers Conference, 66(10).

Foerderer, J., Kude, T., Mithas, S., Heinzl, A. (2018): Does Platform Owner's Entry Crowd Out Innovation? Evidence from Google Photos.

## Workload

3 ECTS (21 hours lectures, 90 hours total workload)