

1.1 Recap last Lecture

- Daten haben eine Geschichte
 - Geschichte der Gesellschaft

1.2 Outline

- ziemlich trockener Einstieg in Python. Notwendig für

2.2 Python

2.3 Python is ...

- Python vs. R
 - Python breiter einsetzbar
 - meiste Tools für beide Sprachen

2.4 How to learn programming?

- Es wäre gelogen zu sagen, Programmieren sei einfach
- Welt des Programmierens ist riesig
 - es gibt mehr möglichkeiten/einschränkungen als ich zeigen kann
- vieles wird nicht unmittelbar klar
 - lernen aus Fehlern, wenn sie passieren
- Programmbeispiele sind möglichst kurz / einfach geschrieben
- Es gibt auch Positives
 - unmittelbares feedback
 - tolles Gefühl, wenns klappt
 - selbständiges Lernen is gut möglich

3.4 Wheting your Appetite

4.4 Programming Concepts and Python Syntax

4.5 Variables

5.5 define variables

6.5 combine variables

7.5 show content of variable

- Variablen sind wie Schachteln oder ein Tupperware
 - etwas (Objekte) reinstecken
 - bei Gebrauch hervorholen
 - Zuweisung/Zugriff
- Namenskonvention von Variablen: Kleinschreibung, English, underscore statt space

7.6 Data Types

- es gibt verschiedene Aufbewahrungsboxen
- Namen unabhängig von Programmiersprache
- je nach Typ andere Eigenschaften

- der Typ ist immer implizit (dynamisch), nicht angeben bei Zuweisung
- Liste ist eine grosse Box, die kleine Boxen aufnehmen kann

7.7 Data Type Conversion

8.7 check the type

9.7 convert types (similar for other types)

10.7 easiest way to use a number in a text

- Analogie: nur gleiche Schachteln können kombiniert werden

10.8 Confusing Equal-Sign

11.8 assign a value to a variable

12.8 compare two values if they are identical

- = mit ungewohnter Funktion
- = Zuweisungsoperator
 - in Schule gelernt: Entsprechung linker + rechter Seite
 - R nutzt <-
- == Vergleichsoperator

12.9 Comments

13.9 single line comment

13.10 Visual Studio Code

- demonstration
 - script
 - variable explorer
 - tab for completion
 - check for WSL and tabnine support

13.11 In-class: Run your first Python Program I

13.12 In-class: Run your first Python Program I

13.13 Iterations

14.13 iterate over each element

- Anwendungsfall
 - für viele Objekte das gleiche machen
- Objekt innerhalb loop bearbeiten
- sentence + token sind Variablen
 - token ist temporäre Variable (Inhalt ändert in loop)

14.14 Conditionals

- Anwendungsfall
 - Aktion abhängig machen von Variableninhalt
- Frage: Was wird ausgegeben?
- Funktionen für Länge und Print

14.15 Indentation matters!

- shift + tab für unindent
- tab wird umgewandelt in 4 spaces

14.16 Methods

15.16 split at whitespace

16.16 check the variables

17.16 add something to a list

18.16 concatenate elements to string

- jedes Objekt/Variable stellt Methoden zur Verfügung

18.17 Functions and Arguments

19.17 define a new function

- Anwendungsfall
 - Code strukturieren, Redundanzen vermeiden
- zwei Teile: definieren + aufrufen
- Funktion mit vordefinierten Argumenten

19.18 Indexing

20.18 element at position X

21.18 elements of subsequence [start:end]

22.18 elements of subsequence backwards

- Anwendungsfall
 - nur ein Teil der Objekte auswählen

22.19 Errors

- produce NameError, TypeError

22.20 Modules/Packages

- Standing on the shoulders of giants
- alles auf GitHub

22.21 NLP Packages

22.22 Mini-Project

- welche Daten das ist das wichtigste
- start on 19 May in class

22.23 In-class: Exercises I

22.24 In-class: Exercises II

22.25 Resources

23.25 Questions?

23.26