



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

Einführung in Python - Vorbereitung

Quelle: ÜK2 Datenorganisation

Zielsetzung - Vorbereitung



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Erste Einblicke in Daten, Strukturen &

1. Gefühl für den Umgang mit Daten und Dateien entwickeln

→ verschiedene Daten-Basen erlauben verschiedene Operationen und Analysen

Simple Datentypen

„E“

True

-3

10000

5.25

Komplexe Datentypen

„großer Baum“

09:15

(49.4186, 8.6769)

20.05.1997

```
{  
  „Hallo“: „Hello“,  
  „Tschüss“: „Bye“  
}
```

Simple Datentypen

„E“ → Char

Richtig → Boolean

-3 → signed Integer

10000 → unsigned Integer

5.25 → Float

Komplexe Datentypen

„großer Baum“ → String

09:15 → Time

(49.4186, 8.6769) → Point

20.05.1997 → Date

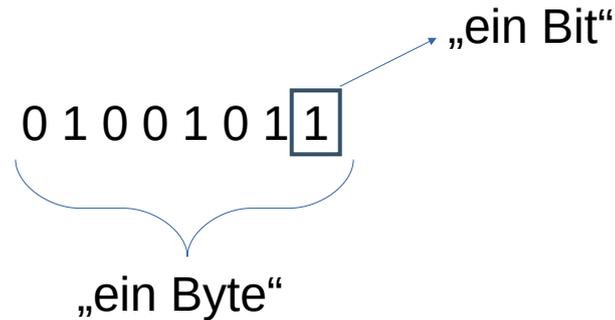
```
{  
  „Hallo“: „Hello“,  
  „Tschüss“: „Bye“  
} → Map/Dictionary
```

Datentypen aus Sicht des Computers



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Darstellung im Speicher des PCs nur in binär → 0 und 1
→ Daten brauchen Typen, die den Binär-Werten einen Sinn geben



- Was ist jetzt in diesem Byte gespeichert? Es könnte zum Beispiel eine Ganzzahl, oder ein Buchstabe sein.

Datentypen aus Sicht des Computers



- Darstellung im Speicher des PCs nur in binär → 0 und 1
→ Daten brauchen Typen, die den Binär-Werten einen Sinn geben

Int8: (8 Bit Ganzzahl)

$$\begin{array}{r} 0 * 2^7 = 0 \\ + 1 * 2^6 = 64 \\ + 0 * 2^5 = 0 \\ + 0 * 2^4 = 0 \\ + 1 * 2^3 = 8 \\ + 0 * 2^2 = 0 \\ + 1 * 2^1 = 2 \\ + 1 * 2^0 = 1 \\ \hline 75 \end{array}$$

0 1 0 0 1 0 1 1

ASCII Char: (Buchstabe)

Letter	ASCII Code	Binary
A	065	01000001
B	066	01000010
C	067	01000011
D	068	01000100
E	069	01000101
F	070	01000110
G	071	01000111
H	072	01001000
I	073	01001001
J	074	01001010
K	075	01001011
L	076	01001100
M	077	01001101
N	078	01001110

Datentypen



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Mit verschiedenen Datentypen muss verschieden umgegangen werden
- z.B. sind andere Rechenoperationen zulässig
 - $15^{\circ}\text{C} / 2 = ?$
 - „Baum“ * „Huhn“ = ?
- **Skalenniveaus** sind Kategorien, die uns eine Auskunft darüber geben
 - welche Merkmale unsere Daten aufweisen
 - wie die Daten analysiert werden können

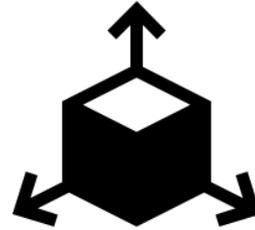
Skalenniveau



Skalenniveau	Beispiel	Eigenschaften	Rechenoperationen
Nominalskala	Gender, Straßennamen	Häufigkeit	=, ≠
Ordinalskala	Schulnoten, Schmerzgrade	Häufigkeit, Reihenfolge	=, ≠, >, <
Intervallskala	Temperatur °C	Häufigkeit, Reihenfolge, Abstand	=, ≠, >, <, +, -
Ratioskala	Alter, Distanz	Häufigkeit, Reihenfolge, Abstand, natürlicher Nullpunkt	=, ≠, >, <, +, -, *, /



Mehr dazu in der Statistikvorlesung

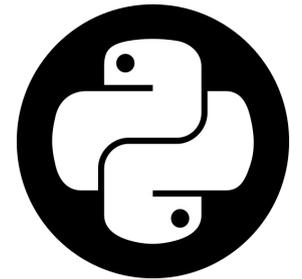
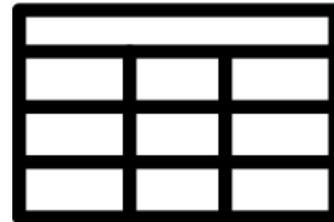
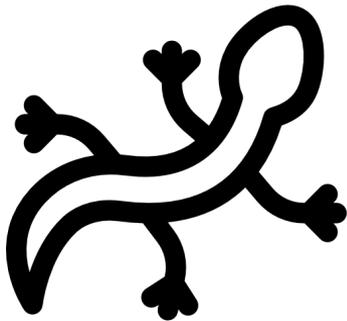
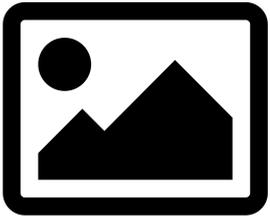


UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386



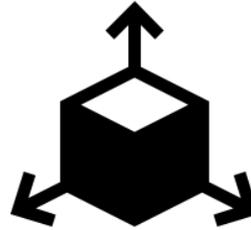
Datei-Typen

Welche Arten kennen Sie bereits?





.txt → Textdatei



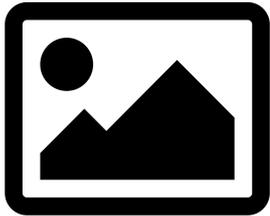
.xyz → Punktdatei



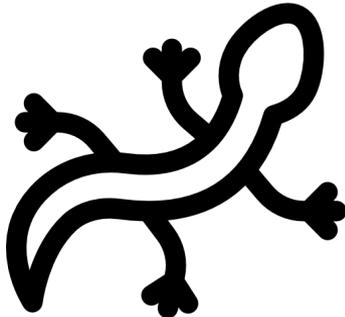
UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386



.zip → Komprimiert

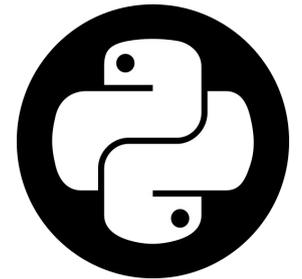


.jpeg → Bilddatei



.exe → Executable/Ausführbare Datei

.csv → Tabelle



.py → Python-Code

Dateien



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Speicherort für Daten
 - Datei-Typ teilt mit welche Daten erwartbar sind - sowohl dem PC als auch den Nutzenden
 - oft Speicherung gleichartiger Daten
 - z.B. Zeichen in einer .txt-Datei
 - Punktkoordinaten in einer .xyz Datei
 - Bilddaten in einer .jpg Datei
 - ...
- Verschiedene Dateitypen halten Daten mit verschiedenen Datentypen

Dateityp - Unterschiede



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Daten lassen sich häufig in verschiedenen Dateitypen speichern – Unterschiede zwischen diesen bestehen z.B.
 - Leicht für Menschen versus leicht für Computer lesbar
 - Komprimiert versus unkomprimiert
 - Struktureller Aufbau innerhalb einer Datei, z.B. tabellarisch versus objektorientiert
 - Datencontainer versus ausführbare Datei

Metadaten



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

→ Daten über Daten

- z.B. was kann man mit einem Datensatz machen → vgl. Datei-Typ
- Aber auch präziser e.g.
 - mit welchem Gerät wurde ein Datensatz aufgenommen?
 - Wann wurde etwas zum letzten mal editiert?
 - Welche Präzision hat der Sensor?
 - Welche Größe in Metern hat ein Pixel in einem GeoTIFF?
- Meist an Dateien gebunden

➔ auch wieder wichtig um einzuschätzen was mit einer Datei oder einem Datensatz möglich ist

Binär-Dezimal Umrechnung



- Binär zu Dezimal

- 1101 \rightarrow
- 1001
- 0011

$$\begin{array}{r} 1 * 2^3 = 8 \\ + 1 * 2^2 = 4 \\ + 0 * 2^1 = 0 \\ + 1 * 2^0 = 1 \\ \hline 13 \end{array}$$

Finden Sie nun selbstständig heraus:

- Wie Daten mit dem Datentyp 'float' gespeichert werden.
- Was der Unterschied zwischen 'signed' und 'unsigned' Integers ist
- Was der Unterschied zwischen ASCII und UTF-8 ist (und was aktueller ist).

- Dezimal zu Binär

- 9 \rightarrow
- 14
- 4

$$\begin{array}{r} 9 \% 2 \rightarrow 4 \quad \text{Rest } 1 \\ 4 \% 2 \rightarrow 2 \quad \text{Rest } 0 \\ 2 \% 2 \rightarrow 1 \quad \text{Rest } 0 \\ 1 \% 2 \rightarrow 0 \quad \text{Rest } 1 \end{array}$$

„%“ Operator

\rightarrow Ergebnis ist Rest aus Division

Skalenniveau

- IQ
- Nationalität
- Bundesland
- Temperatur (K)
- Erdbebenstärke (Richterskala)

Nominalskala



Ordinalskala



Intervallskala



Ratioskala



Skalenniveau



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Daten in welchem Skalenniveau müssen erhoben werden um folgende Fragen zu beantworten?
 - Wird das Neuenheimer Feld als grün wahrgenommen?
 - Wie viele Quadratmeter Grünfläche gibt es im Neuenheimer Feld?

Welche Skalenniveaus haben Characters, Strings, (signed) Integers, (unsigned) Integers und Floats in Python?



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

Datei-Systeme und Konsolenprogramme

Zielsetzung



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Gefühl für den Umgang mit Dateisystemen, Pfadnamen, der Konsole und git entwickeln
- ➔ Schlecht gepflegte Dateisysteme sind unübersichtlich und führen zu vermeidbaren Fehlern; insbesondere bei internationaler FOSS-Software wie z.B. QGIS
- ➔ Beherrschung der Konsole für viele Tools in der Forschung nützlich

Ordnerstrukturen



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Gute Struktur erleichtert den Überblick über Datei- und Datenmengen erheblich
- Ähnliche/Verwandte Dateien in Ordner gruppieren

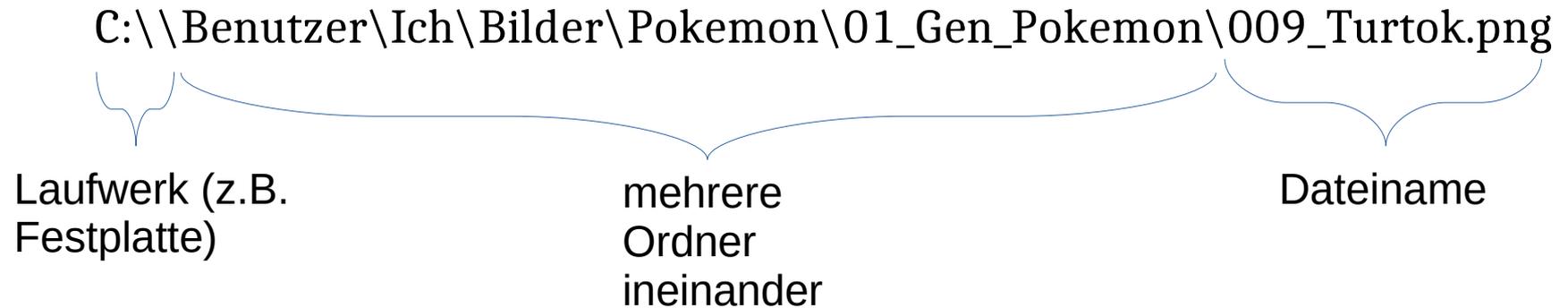
Best Practices zur Ordner- und Dateibenennung:

- Vermeidung von Sonderzeichen (insb. in Windows!)
- Vermeidung von Leerzeichen (!!)
- So kurz wie möglich, so lang wie nötig
- Start des Namens zur Sortierung verwenden (z.B. 2013_11_06_Abrechnung)
- Namensschema vor Sammlung von Daten ausdenken!

Datei-Pfade



- Aus dem Ablageort einer Datei resultiert deren absoluter **Datei-Pfad**
- Z.B. bei Windows:



➔ Identifiziert Speicherort eindeutig

Kommandozeile



- Öffnen mit: Win + R, dann CMD eingeben und Enter drücken
- Textbasierte Kommandos können Programme ausführen und Daten/Dateien manipulieren

- Wichtiges Werkzeug in der Forschung

➔ Viele Programme nur hier ausführbar

Z.B. TRAIN von Prof. Menzel

Warum?

- GUI (Graphic User Interface) kostet Zeit und Geld
- Für kleine Programme reicht Text als In- und Output völlig aus

```
Eingabeaufforderung
mv          Move or rename a file, a directory, or a symlink
reset      Reset current HEAD to the specified state
rm         Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
bisect     Use binary search to find the commit that introduced a bug
grep      Print lines matching a pattern
log       Show commit logs
show     Show various types of objects
status   Show the working tree status

grow, mark and tweak your common history
branch   List, create, or delete branches
checkout Switch branches or restore working tree files
commit  Record changes to the repository
diff    Show changes between commits, commit and working tree, etc
merge   Join two or more development histories together
rebase  Reapply commits on top of another base tip
tag     Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
fetch      Download objects and refs from another repository
pull      Fetch from and integrate with another repository or a local branch
push     Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.

C:\Users\jocho>
```

P.S.: An sich ist die Kommandozeile nur eine nicht schön graphisch dargestelltes Programm. So lange Sie hier keine Administratoren Rechte haben (müssen Sie aktiv entscheiden), kann hier genauso wenig passieren wie in einem ganz normalen Datei-Explorer und Sie können recht frei aus- und rumprobieren.

Kommandozeile - Aufbau



```
C:\> Eingabeaufforderung  
Microsoft Windows [Version 10.0.19045.3570]  
(c) Microsoft Corporation. Alle Rechte vorbehalten.  
C:\Users\jocho> cd ./Documents
```

./ → von dem aktuellen Pfad aus
../ → von dem Parent (ein Ordner obendrüber) aus

1) 2) 3)

- 1) Aktueller Pfad → Hier wird das Kommando ausgeführt
- 2) Kommando → Was getan werden soll; hier: den Pfad wechseln
- 3) Übergabeparameter → Weitere Eingabe in das Programm; hier: zu diesem Ort wechseln

➔ Zusammen: Wechsel von C:\Users\jocho zu C:\Users\jocho\Documents

Kommandozeile - Grundkommandos



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

Befehl	Ausgeschrieben	Funktion
dir	dir ectory	Zeigt an was sich im aktuellen Pfad/Ordner befindet
cd	c hange d irectory	Wechselt das aktuelle Verzeichnis
del	d el	Lösche eine Datei
mkdir	m ake d irectory	Erstelle einen Ordner
rmdir	r emove d irectory	Lösche einen Ordner
help	... h elp	Hilfe zu einem Kommando, z.B. help rmdir
color	... c olor	Wechselt Farbe und Hintergrundfarbe der Kommandozeile; Empfehlung: color a

Kommandozeile – Zusätzliche Eingaben



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Viele Kommandos können feinjustiert werden
- Mit „Flaggen“ (flags) z.B.

rmdir /s Pokemonbilder → /s löscht dann auch alle Dateien und Subordner, sonst müsste der Ordner „Pokemonbilder“ leer sein

- Mit optionalen Übergabeparameter z.B.

git commit -m „Version2.0“ → -m hängt „Version2.0“ als **Message** an den commit.
Was ein commit ist, kommt gleich

→ In anderen Kommandozeilen wird meist statt „/“, „--“ oder „-“ genutzt



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

Praxis

Konsole



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

- Startet die Konsole
- In welchem Pfad startet die Konsole? Welche Ordner oder Dateien sind hier zu finden?
- Wechsel in der Konsole zu dem Desktop Pfad
- Erstelle hier einen Ordner „InToPy“
- Wechsel in diesen Ordner und erstelle dort den Ordner „Vorbereitung“

P.S.: Löschen Sie den Ordner danach wieder vom Desktop, navigieren Sie in der Commandline in den richtigen Ordner, bspw. Ihres Semesters, und erstellen Sie den Ordner hier erneut. Alternativ können Sie den Ordner auch 'nur' bewegen (move).

Der richtige Befehl ist gefunden, aber die genaue Nutzung unklar? → help hilft!

E.g. help rmdir

Algorithmisches Denken



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

Algorithmus

→ endliche Folge an Anweisungen die zu einem Ziel führen

Heißt konkret?

→ komplexe Fragestellungen zu kleinen einfach ausführbaren Paketen schnüren

Vorteile:

- Strukturierte Art und Weise ein Problem anzugehen → nächster Schritt immer klar
- Kleine Fragen mit der Suchleiste des Vertrauens einfacher lösbar

P.S.: Algorithmisches Denken ist eine der Grundlagen von Programmierung. Eine weitere Beschäftigung und Vertrautmachung ist überaus sinnvoll. Schauen Sie daher auch gern bspw. hier vorbei für mehr Informationen: <https://medium.com/tech-based-teaching/if-curious-then-learn-a-brief-intro-to-algorithmic-thinking-ba683bf44994> und nutzen Sie Google um ein wenig mehr zu erfahren.