

# Warum ist Python so cool?

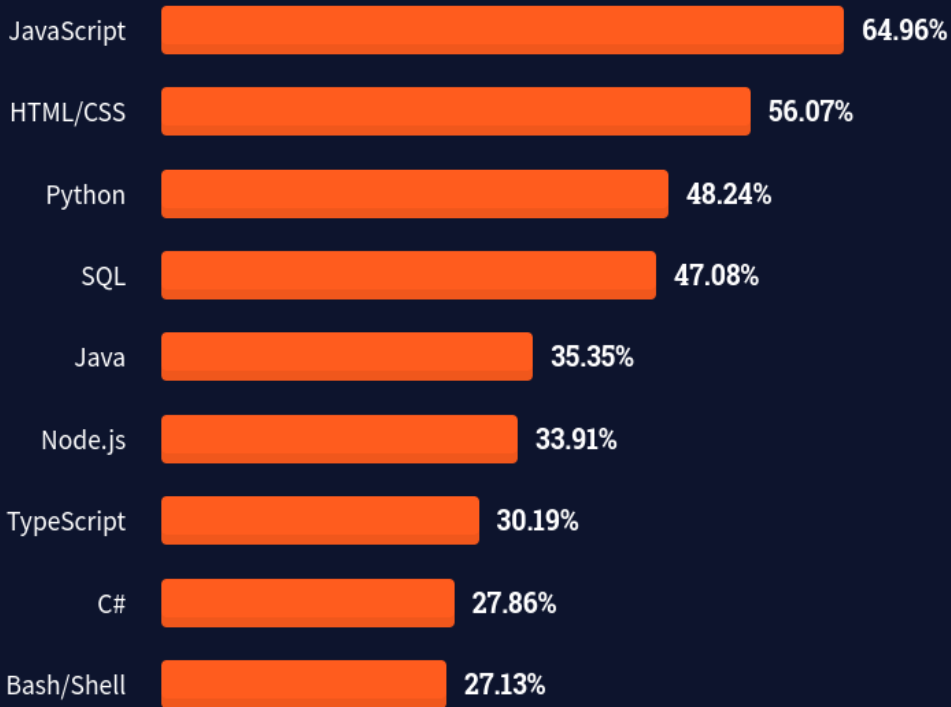
---

Lars Quentin

26.04.2022

# Wie beliebt ist Python?

---



Python



**48.24%**

SQL



**47.08%**

Java



**35.35%**

Node.js



**33.91%**

C#



**27.86%**

Bash/Shell



**27.13%**

Python



**48.24%**

Java



**35.35%**

Node.js



**33.91%**

C#



**27.86%**

**Was macht Python so besonders?**

---

**Plattformexklusivität?**

# Plattformexklusivität?

Nein!

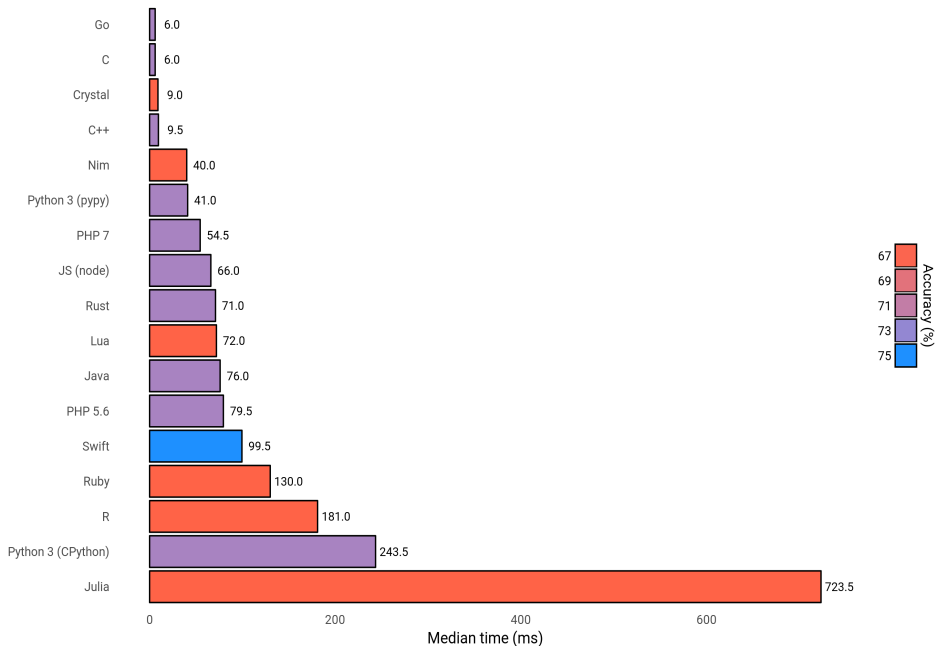


**Performance?**

# Speed comparison of various programming languages

Method: calculating  $\pi$  through the Leibniz formula x times

Languages



# Performance?

lol

**Libraries?**

# Was macht Python so besonders?

Also... was ist es nun?

# Was macht Python so besonders?

Also... was ist es nun?

- Plattformexklusivität

# Was macht Python so besonders?

Also... was ist es nun?

- Plattformexklusivität
- Performance

# Was macht Python so besonders?

Also... was ist es nun?

- Plattformexklusivität
- Performance
- Revolutionäre Paradigmen?



# Was macht Python so besonders?

Also... was ist es nun?

- Plattformexklusivität
- Performance
- Revolutionäre Paradigmen?
- Einfach bugfreie Programme zu schreiben?

# Was macht Python so besonders?

Also... was ist es nun?

- ~~Plattformexklusivität~~
- Performance
- ~~Revolutionäre Paradigmen?~~
- ~~Einfach bugfreie Programme zu schreiben?~~
- Besondere Sprachfeatures für ML, Webdev, Web3, Embedded...

# Was macht Python so besonders?

Also... was ist es nun?

- ~~Plattformexklusivität~~
- ~~Performance~~
- ~~Revolutionäre Paradigmen?~~
- ~~Einfach bugfreie Programme zu schreiben?~~
- ~~Besondere Sprachfeatures für ML, Webdev, Web3, Embedded...~~

???????

**Python ist langweilig.**

# **Python ist langweilig.**

Und langweilig ist simpel.

# Aufgabenstellung

## Gegeben:

Eine Textdatei mit kommaseparierten Ganzzahlen

## Aufgabe:

- Lies die Datei ein
- Berechne die Summe
- Berechne die Anzahl der Elemente
- Gebe den Mittelwert aus

**Wieso ist Python so gut?**

# Wieso ist Python so gut?

Vieles!



# **Zugänglichkeit für Nicht-Programmierer**

---

# Jupyter Notebooks

- Webbasierte Pythonumgebung
- Interaktiv
- Grafisch
- Kostenloses online benutzbar
  - Google Colab
  - Jupyter GWDG
- Open Source, BSD-3 lizenziert
- Später mehr



Was braucht man für Vorwissen?

### Was braucht man für Vorwissen?

- Typensysteme? Integergrößen?

### Was braucht man für Vorwissen?

- Typensysteme? Integergrößen?
- Kommandozeile?

### Was braucht man für Vorwissen?

- Typensysteme? Integergrößen?
- Kommandozeile?
- Compiler?

### Was braucht man für Vorwissen?

- Typensysteme? Integergrößen?
- Kommandozeile?
- Compiler?
- Speicher?

### Was braucht man für Vorwissen?

- Typensysteme? Integergrößen?
- Kommandozeile?
- Compiler?
- Speicher?
- Build Systems?



### Was braucht man für Vorwissen?

- Typensysteme? Integergrößen?
- Kommandozeile?
- Compiler?
- Speicher?
- Build Systems?
- `reinterpret_cast<bool(__stdcall*)(const char*, const char*)>?`

### Was braucht man für Vorwissen?

- Typensysteme? Integergrößen?
- Kommandozeile?
- Compiler?
- Speicher?
- Build Systems?
- `reinterpret_cast<bool(__stdcall*)(const char*, const char*)>?`

Englischkenntnisse reichen zum lesen.

Einfach Notebook auf und tippen.

# Zugänglichkeit für Programmierer

---

Paradigmenagnostisch

Riesige Standard Library

# Standard Library

Was hat die Standard Library alles?

# Standard Library

Was hat die Standard Library alles?

- Parser: JSON, CSV, XML

Was hat die Standard Library alles?

- Parser: JSON, CSV, XML
- Protokolle: HTTP, FTP, POP3, IMAP4, SMTP



Was hat die Standard Library alles?

- Parser: JSON, CSV, XML
- Protokolle: HTTP, FTP, POP3, IMAP4, SMTP
- Grafische Interfaces: tkinter

Was hat die Standard Library alles?

- Parser: JSON, CSV, XML
- Protokolle: HTTP, FTP, POP3, IMAP4, SMTP
- Grafische Interfaces: tkinter
- Reguläre Ausdrücke

Was hat die Standard Library alles?

- Parser: JSON, CSV, XML
- Protokolle: HTTP, FTP, POP3, IMAP4, SMTP
- Grafische Interfaces: tkinter
- Reguläre Ausdrücke
- SQL(ite)-Interop

# Standard Library

Was hat die Standard Library alles?

- Parser: JSON, CSV, XML
- Protokolle: HTTP, FTP, POP3, IMAP4, SMTP
- Grafische Interfaces: tkinter
- Reguläre Ausdrücke
- SQL(ite)-Interop
- Logging Library

# Standard Library

Was hat die Standard Library alles?

- Parser: JSON, CSV, XML
- Protokolle: HTTP, FTP, POP3, IMAP4, SMTP
- Grafische Interfaces: tkinter
- Reguläre Ausdrücke
- SQL(ite)-Interop
- Logging Library

POSIX-Tools, SSL-Wrapper, CURSES, GZIP/BZIP2/TAR  
compression...

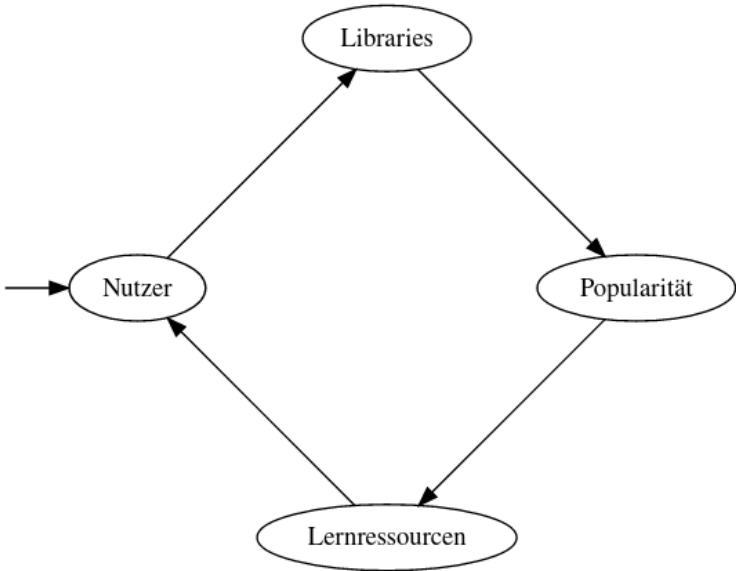
# Die Entwickler werden erwachsen

...und entwerfen nun ihre eigenen Libraries.

# Compound Interest

---

# Compound Interest



**Endresultat:** Es gibt Libraries/Bindings für alles.



**“das Reichtum an Paketen und libs lässt über mögliche Nachteile der Sprache hinwegsehen, weil 50% der use cases eh nur import + 3 Zeilen eigener Code sind”**

– David Eipper, 2022

## **Bonus: Wichtige Bibliotheken**

---

- Skripting / Administration
- GUI-Programmierung
- Data Science
- Web-Development

## Scripting

- Subprocess
- Shlex
- OS

## **Scripting**

- Subprocess
- Shlex
- OS

## **GUI-Programmierung**

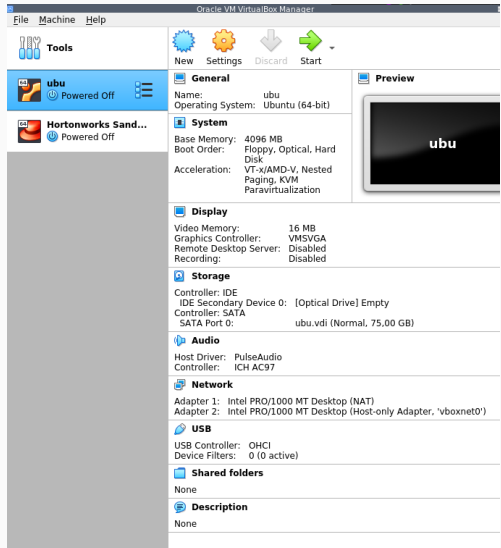
- tkinter
- **PyQT5**
- pyimgui

## Scripting

- Subprocess
- Shlex
- OS

## GUI-Programmierung

- tkinter
- PyQt5
- pyimgui



# Data Science

## **Data Science**

- **Anaconda**



## **Data Science**

- **Anaconda**
- **numpy + scipy**

## **Data Science**

- **Anaconda**
- **numpy + scipy**
- **matplotlib**

## **Data Science**

- **Anaconda**
- **numpy + scipy**
- **matplotlib**
- **pandas**

## Data Science

- **Anaconda**
- **numpy + scipy**
- **matplotlib**
- **pandas**
- **dask**

## Data Science

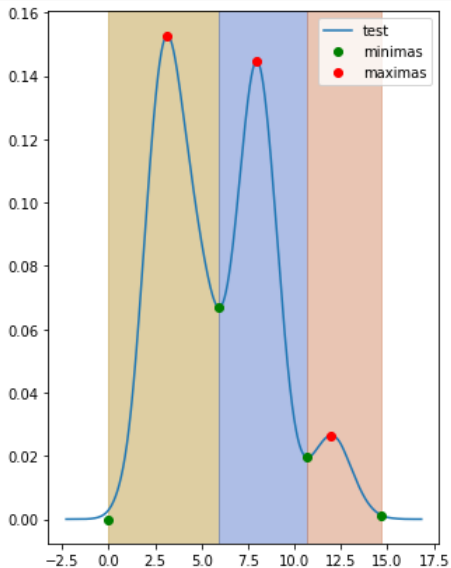
- **Anaconda**
- **numpy + scipy**
- **matplotlib**
- **pandas**
- **dask**
- **scikit-learn**

## Data Science

- **Anaconda**
- **numpy + scipy**
- **matplotlib**
- **pandas**
- **dask**
- **scikit-learn**
- **keras**

## Data Science

- Anaconda
- numpy + scipy
- matplotlib
- pandas
- dask
- scikit-learn
- keras
- BeautifulSoup



# Web Development



## Web Development

- Flask

## Web Development

- Flask
  - Pinterest

## Web Development

- **Flask**
  - Pinterest
- **Django**

## Web Development

- **Flask**

- Pinterest

- **Django**

- Youtube
  - Instagram
  - Dropbox, Spotify, Mozilla, NASA...

## Web Development

- **Flask**
  - Pinterest
- **Django**
  - Youtube
  - Instagram
  - Dropbox, Spotify, Mozilla, NASA...
- **FastAPI**

## Web Development

- **Flask**

- Pinterest

- **Django**

- Youtube
- Instagram
- Dropbox, Spotify, Mozilla, NASA...

- **FastAPI**

- Microsoft
- Netflix
- Uber

## Web Development

- **Flask**

- Pinterest

- **Django**

- Youtube
- Instagram
- Dropbox, Spotify, Mozilla, NASA...

- **FastAPI**

- Microsoft
- Netflix
- Uber

- **Pelican**

## Web Development

- **Flask**

- Pinterest

- **Django**

- Youtube
- Instagram
- Dropbox, Spotify, Mozilla, NASA...

- **FastAPI**

- Microsoft
- Netflix
- Uber

- **Pelican**

- Django-REST
- Bottle.py



**Danke für eure  
Aufmerksamkeit!**

Gleich gehts weiter mit Teil 2 über den Code.