Python

Thomas Schwarz, SJ

Why

- Universal, accessible language
  - Clear and simple syntax
    - Python philosophy: The frequent should be easy
    - Made for reading
    - Used for fast prototyping

- Excellent support community
  - Help for beginners and experts is easily available

Why

- Used in a variety of communities
  - Python allows easy development of modules
    - Numpy to allow scientific calculations
    - Keras for deep-learning
    - Turtle to teach children
    - Pyplot etc. for producing article-quality graphics
    - …
- Has become a “glue” language for AI and Data Science

Why

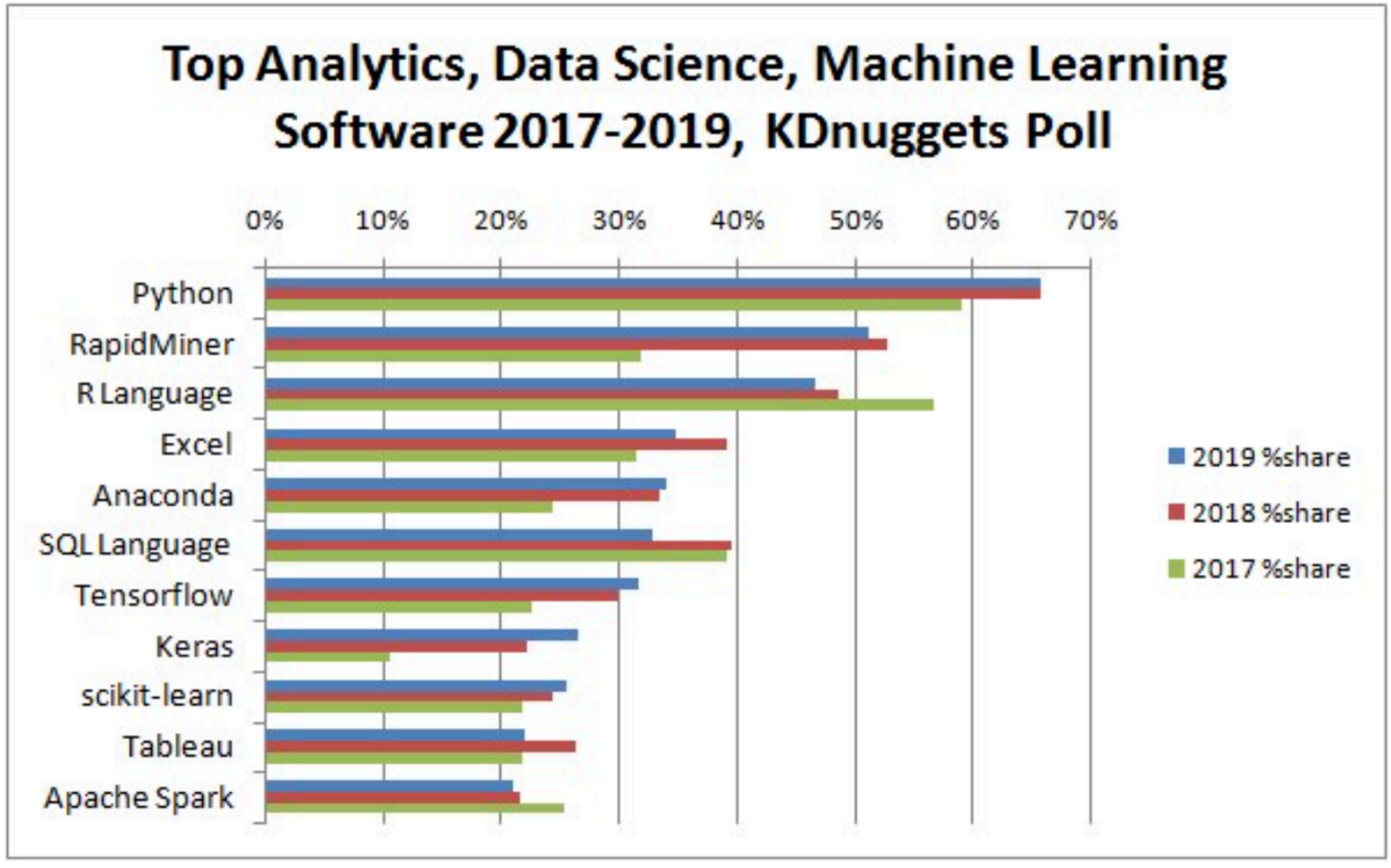


Fig 1: KDnuggets Analytics/Data Science 2019 Software Poll: top tools in 2019, and their share in the 2017, 2018 polls

- 5 out of 10 top skills listed in the KDnuggets poll are related to Python

What

- What are the goals of this class?
- What are we going to cover?

What

- Learn programming the *Python* way
  - Python is terse, clear, incorporates support for different programming styles
    - Imperative programming
    - Functional programming
    - Object Oriented programming
    - Event-driven programming: TkInter
- Pythonesk: Doing things the Python way

How

- You learn programming by programming
  - Activity driven class
  - Collaboration is encouraged
    - Ideal environment:
      - Pairwise programming:
        - One student is the driver
          - Types the program
        - Other student is the navigator
          - Tells the driver what to do
        - And hopefully forces both to discuss and explain

How to get Python

- Python uses major and minor release numberings
- Current: Python 3.10
  - Python 2.7 is still being used, but its use is fading
    - MacOS comes with Python 2.7.18
- We will use Python 3.x

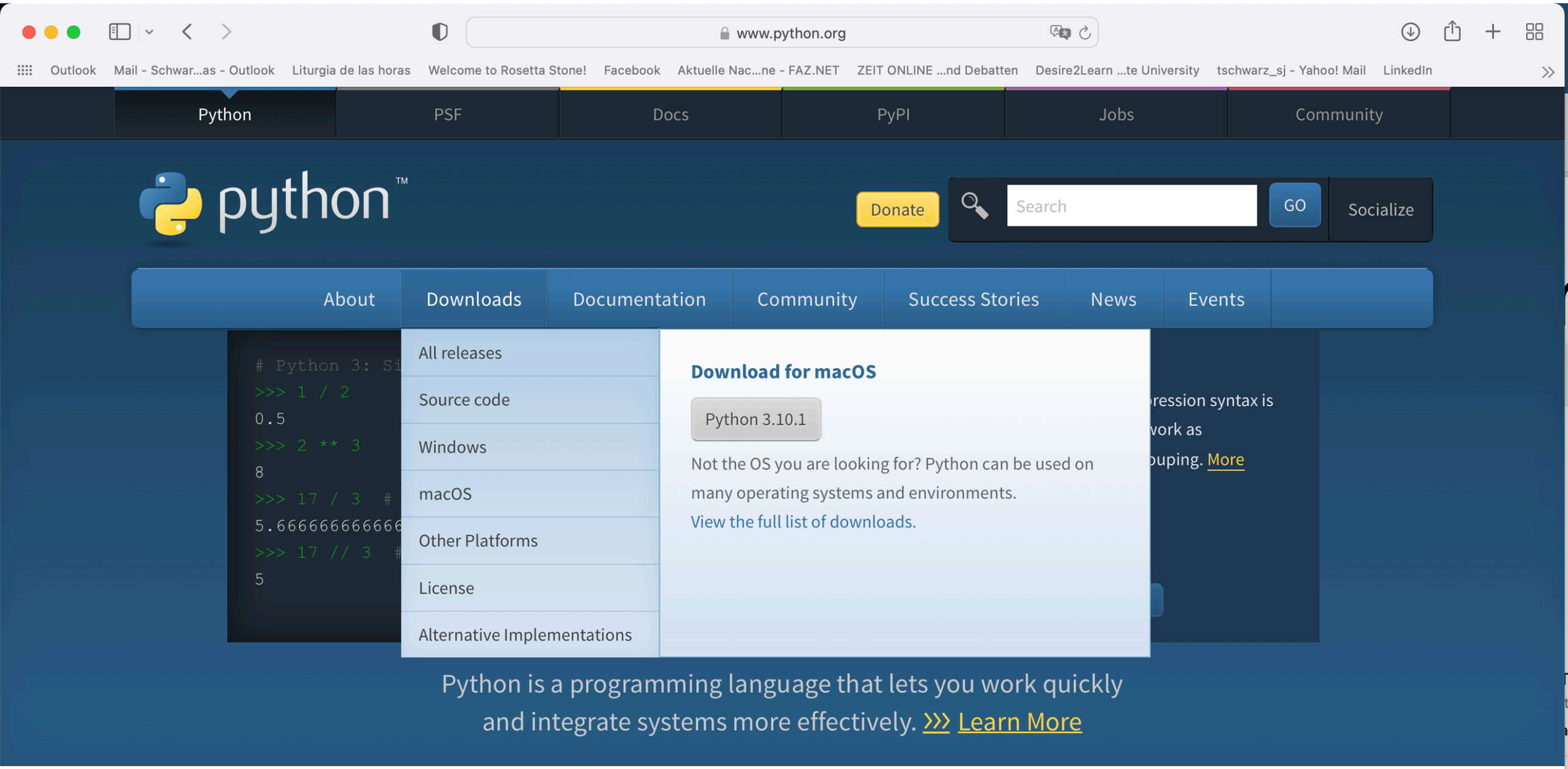
How to get Python

- Data Scientists like Anaconda
  - Comes with all usual data science modules pre-installed
- Standard Python
  - from python.org
  - Comes with pip, the Python packet installer
  - **We use this one**
  - Has an interface called IDLE

How to get Python

- Follow instruction at the website
- Download according to OS
- **WinOS Important:** When given the opportunity, **add Python to Path**
    - You can set the path yourself, but this makes some tasks easier

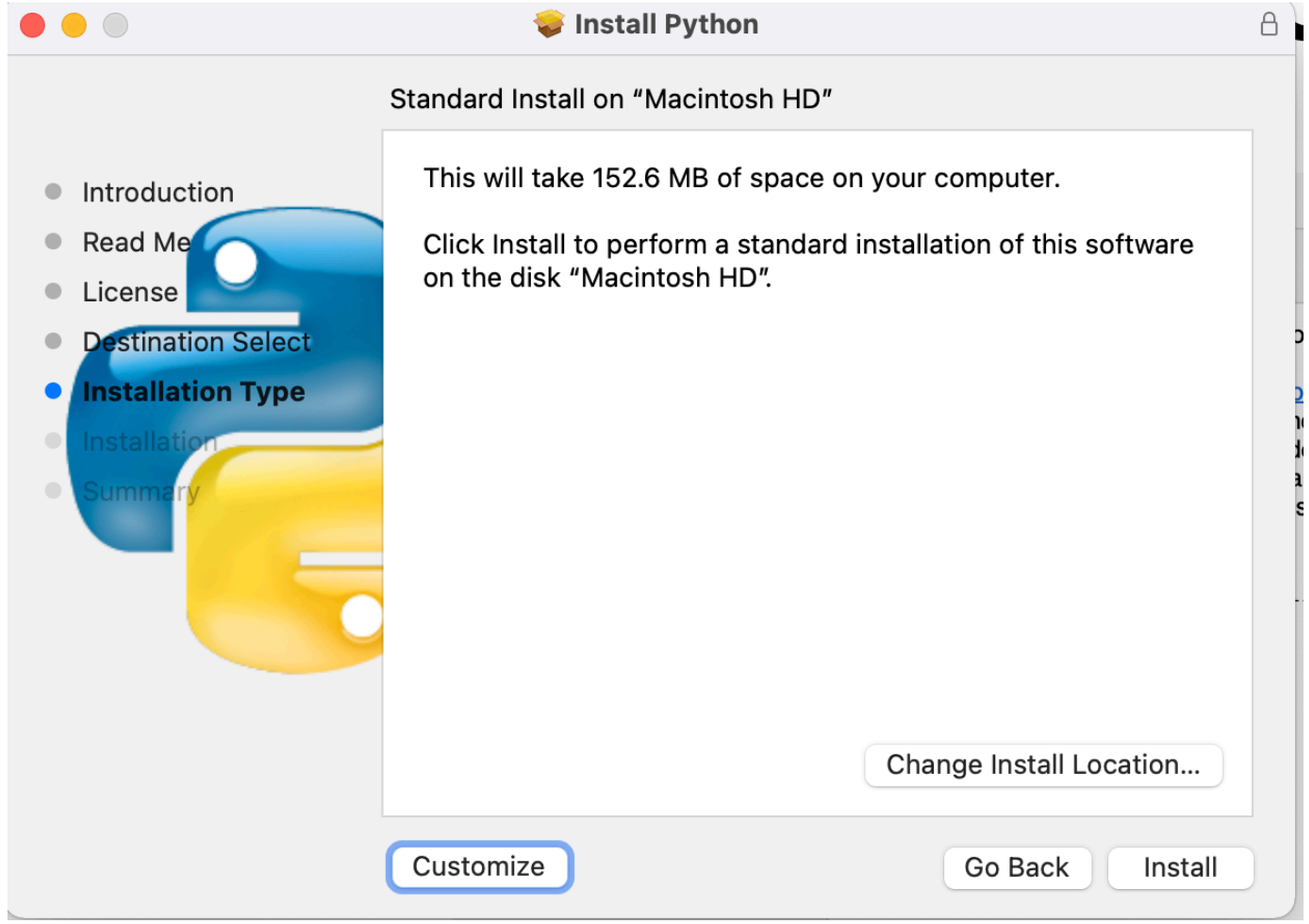
Installation



Installation

Installation

- Use the default location

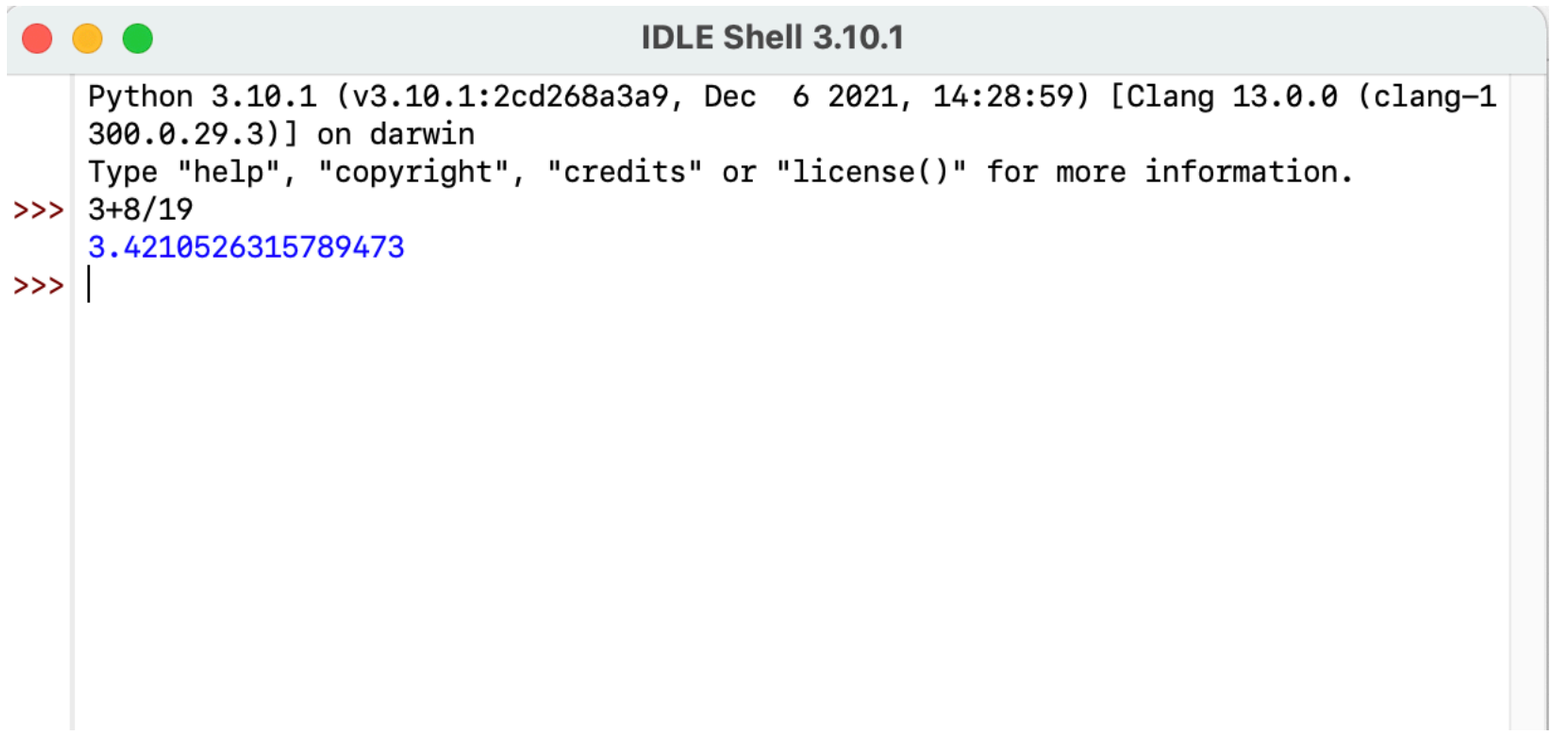


Installation

- After installation:
  - Open a command prompt / shell / terminal and type in
    - `python3 --version`
      - WinOS: If this does not work, you might want to change the path to include the python directory
      - MacOS: `python --version` gives Python 2.7.18
- `python3, python3.10 ...` allows you to access different python versions

Installation

- For this class, we will use the IDLE interface
  - IDLE is in the Python folder
  - You might want it in your taskbar
  - You can use IDLE as a desktop calculator



Python is an Interpreted Language

- Code is executed in the CPU
  - Instructions are in machine language
    - Depending on the architecture

Python is an Interpreted Language

- Code is executed in the CPU
  - Instructions are in machine language
    - Machine languages differ between architectures
  - Early programming done directly into machine language
  - Replaced by programming in Assembly
    - An assembly instruction is a more readable version of a machine language instruction
      - (In first approximation)
    - Some code is still written in assembly (e.g. malware, …)

Python is an Interpreted Language

- Higher Level Languages HLL (e.g. Fortran, C, …)
  - Instruction corresponds to several assembly language instruction
  - Program is somewhat independent of machine architecture
  - A program called compiler translates HLL code into assembly language code
  - Another program called assembler translates assembly into machine language
  - Result is “executable code”

Python is an Interpreted Language

- With better architectures
  - Can write executable code for (almost) all architectures
  - By programming a *virtual machine*
    - A virtual machine is a computer program that simulates a simple computer
  - This program takes a string of bytes as its input: the byte code
  - When running and executing a Python program:
    - Python translates the “source instructions” into Python byte code, which is then “executed” / interpreted by the virtual machine

Python is an Interpreted Language

- Disadvantage of interpreted languages:
  - Cannot make best use of the underlying architecture
- Advantages of interpreted languages:
  - For each platform, need to optimize only the virtual machine
  - Same code / script can run on any platform with a virtual machine

Python is an Interpreted Language

- Python virtual machines are usually written in C
  - Cython
- And sometimes in Java
  - Jython
- For optimized code:
  - Python can use modules developed in C / Java
  - Can be called from within Python

Use of Python

- Python has a simple, but powerful syntax
  - Good first programming language
- Python is ideal for *fast prototyping*
  - Developing Python code is faster than for other languages
  - If Python performance is not enough, can use the Python code as a blue print to develop in a high-performance language like C or C++

Use of Python

- “Glue” language for machine learning
  - Many machine learning algorithms use heavy duty numerical algorithms
  - But users are usually not programmers with years of experience
  - Use a python interface
    - Modules are implemented in C
  - This is a common phenomenon

Use of Python

- Numerical algorithms : numpy and scipy
- Statistics
- Web frameworks: django, flask, …
- Web scraping
- Neural Networking Pytorch, Keras
- Data Science : data cleaning, statistics, ML
- Visualization : pyplot
- …