UNIVERSITÄT BIELEFELD
Faculty of Technology

Programming

Introduction

Daniel Dörr

Faculty of Technology, Bielefeld University

UNIVERSITÄT BIELEFELD

Who am I?

- Dr. Daniel Dörr

  email: ddoerr@cebitec.uni-bielefeld.de
  office: UHG U10-132
- Post-doc in the research group “Genome Data Science” headed by Prof. Dr. Alexander Schönhuth

  `https://gds.techfak.uni-bielefeld.de`

Course syllabus

Part 1

- Programming basics and terminology
- Introduction to Python

Part 2

- Scientific Programming
- Data Science with Python

Organizational matters

What is Programming?

Overview of Python

Python Basics

What is a programming language?

- Natural vs. programming language
- Human-readable vs. machine-readable

Syntax and semantics

Syntax

Symbols, words, sentences

Semantics

Meaning behind symbols, words, and sentences

Syntax and semantics

Syntax

Symbols, words, sentences

e.g. English:

- Words: `He`, `She`, `It`, `Program`, ...
- Sentence grammar rule:
  Subject + Verb + Object

`She loves Python` ✔

`The house table the cup` ✘

`The table reads the cup` ✔

Semantics

Meaning behind symbols, words, and sentences

Syntax and semantics

Syntax

Symbols, words, sentences

e.g. English:

- Words: `He`, `She`, `It`, `Program`, ...
- Sentence grammar rule:
  Subject + Verb + Object

`She loves Python` ✔

`The house table the cup` ✘

`The table reads the cup` ✔

Semantics

Meaning behind symbols, words, and sentences

`She loves Python` ✔

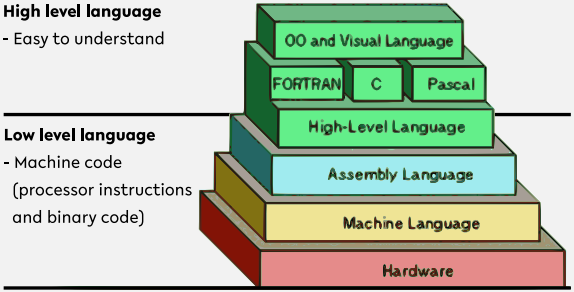
`The table reads the cup` ✘

Programming languages …

- are formal languages with unambiguous context-free grammars,

  syntactic ambiguity:
  “Tom hit the man with a stick.”

- offer different levels of abstraction,



Source: https://thebittheories.com

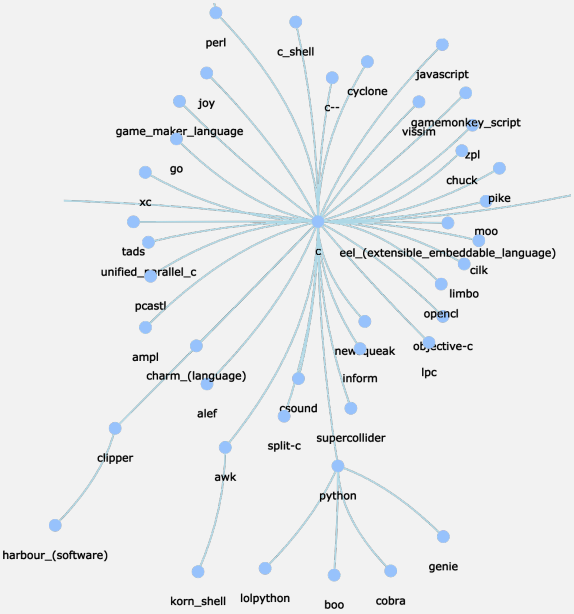
UNIVERSITÄT BIELEFELD

Programming languages …

- are formal languages with unambiguous context-free grammars,

  syntactic ambiguity:
  “Tom hit the man with a stick.”

- offer different levels of abstraction,
- change over time,
- inspire new generations of languages.



Source: http://svalver.github.io/Proglang/paradigms.html

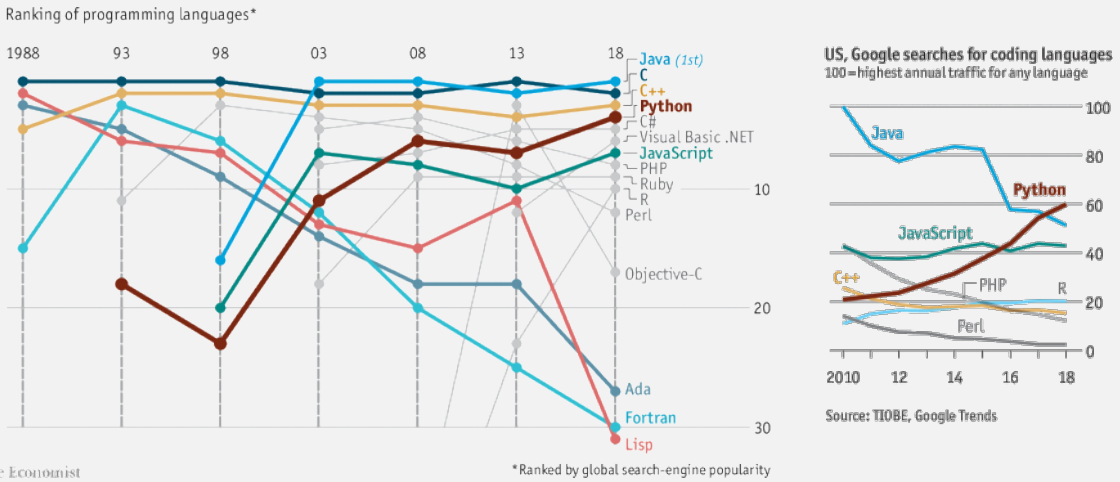
Programming paradigms

Many different general paradigms (notable excerpts):

- Imperative – *Do this, then do that!*
  - Procedural (C)
  - Object-oriented (C++, C#, Java)
- Declarative – *I want this, I want that!*
  - Logic (Prolog)
  - Functional (Haskell, Lisp)
- Mixed (Python, R)

There are also special-purpose languages (not necessarily considered “programming” languages), e.g. *LaTeX*, *HTML*, *XML*.

Language popularity



Source: Python is becoming the world's most popular coding language - The Economist (2018)

Quiz

- *Syntactic* or *semantic* ambiguity?
  - “Milk drinkers are turning to powder.”
  - “Stolen painting found by tree.”
  - “She went to her house, and so did Jane.”
- *True* or *false*?
  - “All context-free grammars are unambiguous.”
  - “Assembly language is a low level language.”
  - “Functional programming is a form of imperative programming.”

Quiz

- *Syntactic* or *semantic* ambiguity?
  - “Milk drinkers are turning to powder.” syntactic
  - “Stolen painting found by tree.” syntactic
  - “She went to her house, and so did Jane.” semantic
- *True* or *false*?
  - “All context-free grammars are unambiguous.” false
  - “Assembly language is a low level language.” true
  - “Functional programming is a form of imperative programming.” false

Organizational matters

What is Programming?

Overview of Python

Python Basics

Python

Originally developed by Guido van Rossum in the late 1980s.

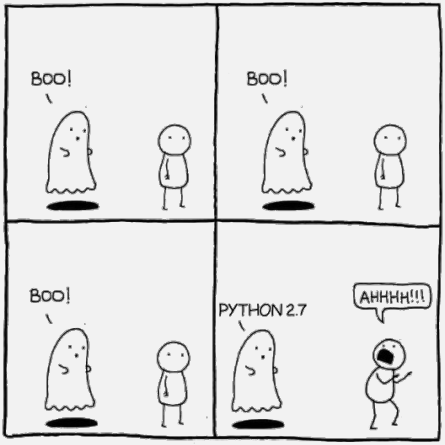
- Open-source and actively maintained
- Applicable to a wide range of applications
- Extremely popular in the data science community

But: There are alternative programming languages. Make sure to use the right one for the task.

Guido van Rossum, source: `https://gvanrossum.github.io`, ©Michael Cavotta, license: CC BY-NC-ND 4.0

Which Python version?

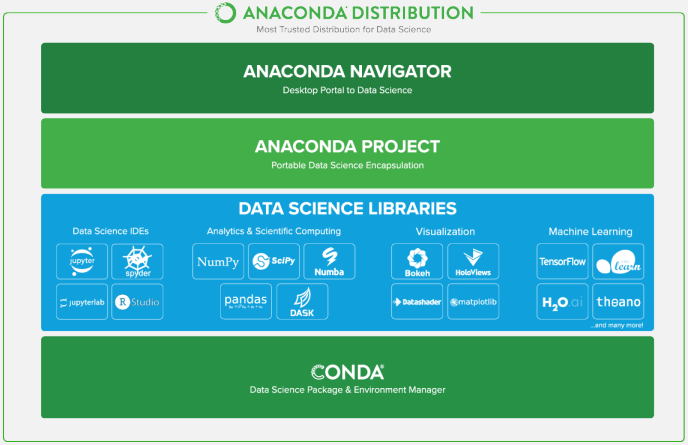
- Python 2: still common, although no longer maintained
- Python 3: modernized, *backwards-incompatible* version of the language



source: https://www.reddit.com/r/ProgrammerHumor/comments/91vtas/python_27/

Development environment: Anaconda

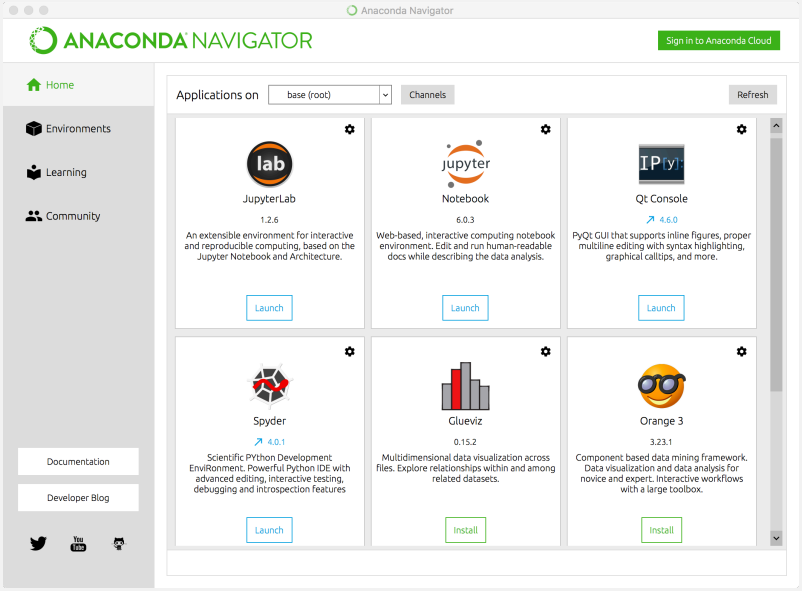
Python Data Science Distribution



Download Anaconda

https://www.anaconda.com/distribution#download-section

Anaconda navigator



Quiz

- *True* or *false*?
  - “Python has been developed for data science analysis.”
  - “Python is the only language used in data science analysis.”
  - “The university has bought Python licenses for this course.”

Quiz

- *True* or *false*?
  - “Python has been developed for data science analysis.” false
  - “Python is the only language used in data science analysis.” false
  - “The university has bought Python licenses for this course.” false

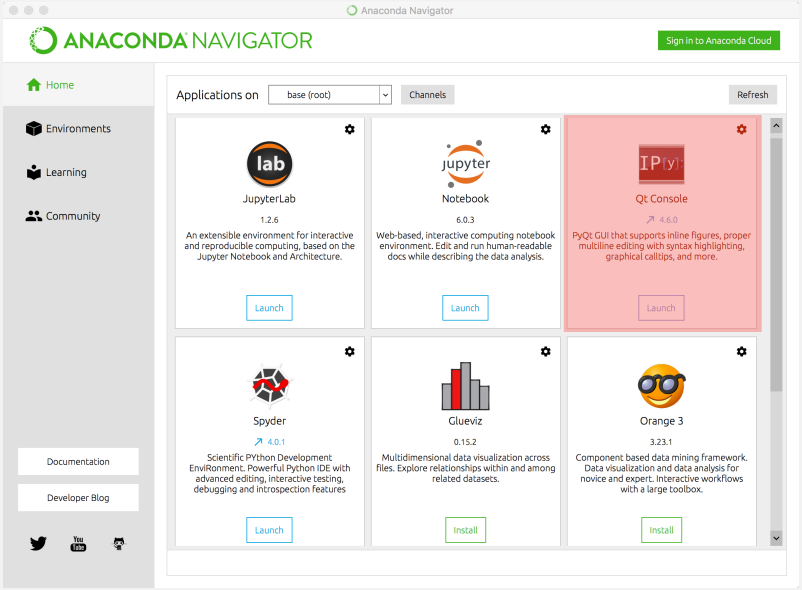
Organizational matters

What is Programming?

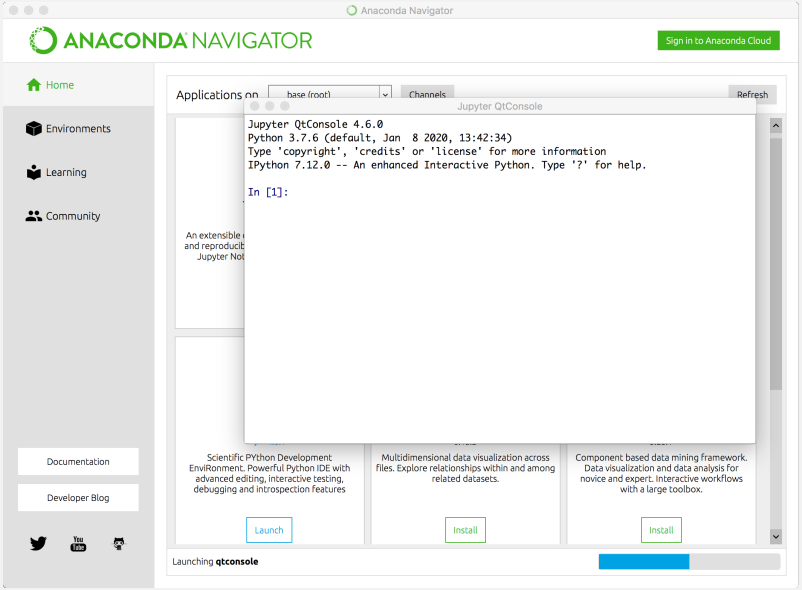
Overview of Python

Python Basics

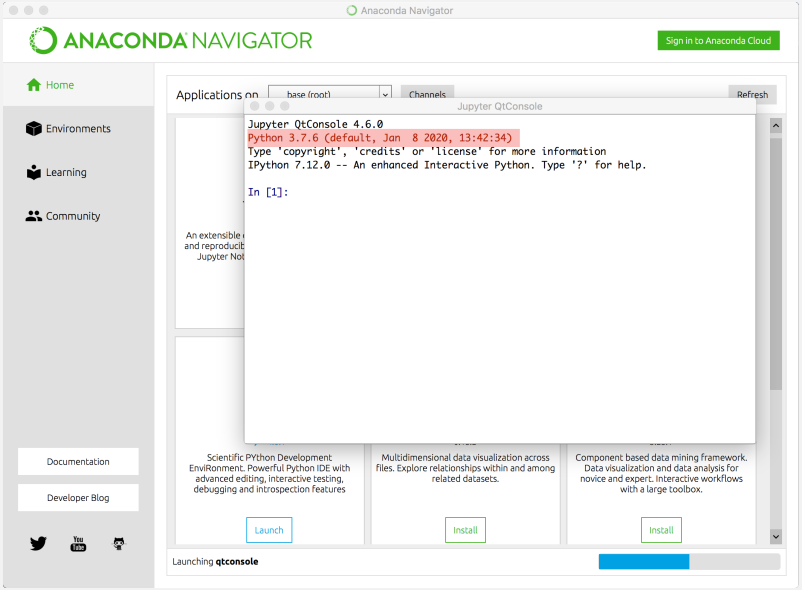
Launching Python



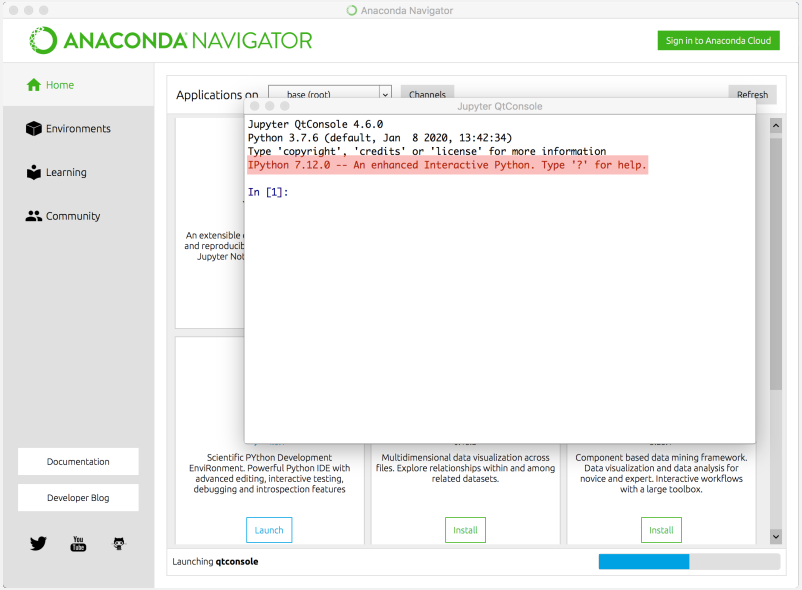
Launching Python



Launching Python



Launching Python



Arithmetic in Python

Numeric types:

- Integer: `int` `42`
- Real valued numbers: `float` `42.0`
- Complex numbers: `complex` `42+0j`

Operators

- Addition and subtraction `+ -`
- Multiplication and division `* / // %`
- Exponentiation `**`

Variables

Variable assignment

- `a = 42`
- `b = a - 6.0`

`type(«name of the variable»)`: returns type of variable

Quiz

- What are types numeric types of the following calculations?
  - `type(42 / 3)`
  - `type(42 // 3)`
  - `type(3.14 + 2.71+8j)`
  - `type(42 // 3.14)`
  - ```
    a = 1
    a - 10 * 1.0
    type(a)
    ```
    ```

Quiz

- What are types numeric types of the following calculations?
  - `type(42 / 3)` — `float`
  - `type(42 // 3)` — `int`
  - `type(3.14 + 2.71+8j)` — `complex`
  - `type(42 // 3.14)` — `float`
  - ```
    a = 1
    a - 10 * 1.0
    type(a)
    ```
    — `int`
    ```

Recap

Summary

- Course logistics
- Introduction to Programming
- First steps in Python