Intro to programming 3

Henri Vandendriessche
henri.vandendriessche@ens.fr

2023-10-03

Terminal cheat sheet reminder

- Bash commands to navigate directories
  - Print Working Directory. Print the path of the current directory

```
pwd
```

  - List all files of the current directory

```
ls folder
```

  - Moving into folder1 and subfolder2 at once.

```
cd folder1/subfolder2
```

  - Moving out of a directory

```
cd ..
```

  - Going back and forth in the directory tree

```
cd ../../folder1/subfolder1
```

  - Going back to the root directory

```
cd ~
```

- “**Tab**” to use the auto-completion
- **Ctrl + C** to stop a program execution
- “**Upper arrow**” to see last commands
- Many more bash commands to use. . .

So far

- Python
- Data types:
  - integer
  - float
  - string
  - boolean
- **If**, **For** and **While** loops:
  - syntax
  - indentation
- Data collections:
  - list
  - tuple
  - set
  - dictionary

Today

- Python standard library
- Random numbers and number choices
- Exercises

If you want to practice a bit

- You can find exercises online for example on W3schools.com
  - https://www.w3schools.com/python/exercise.asp

Python standard library 1/3

- Python's standard library is incredibly extensive, offering a wide range of facilities, as indicated by the comprehensive table of contents listed below:

Python standard library 1/3

- Python's standard library is incredibly extensive, offering a wide range of facilities, as indicated by the comprehensive table of contents listed below:
    - https://docs.python.org/3/library/index.html

Python standard library 1/3

- Python's standard library is incredibly extensive, offering a wide range of facilities, as indicated by the comprehensive table of contents listed below:
  - https://docs.python.org/3/library/index.html
- The "Python library" encompasses various types of components.

Python standard library 1/3

- Python’s standard library is incredibly extensive, offering a wide range of facilities, as indicated by the comprehensive table of contents listed below:
  - https://docs.python.org/3/library/index.html
- The “Python library” encompasses various types of components.
- It includes data types that are typically considered part of the core of a language, such as numbers and lists.

Python standard library 1/3

- Python's standard library is incredibly extensive, offering a wide range of facilities, as indicated by the comprehensive table of contents listed below:
    - https://docs.python.org/3/library/index.html

- The "Python library" encompasses various types of components.

- It includes data types that are typically considered part of the core of a language, such as numbers and lists.

- The library also contains built-in functions and exceptions—objects that can be used by all Python code without the need for an import statement. While some of these are defined by the core language, many are not essential for the core semantics and are only described in this documentation.

Python standard library 1/3

- Python's standard library is incredibly extensive, offering a wide range of facilities, as indicated by the comprehensive table of contents listed below:
    - https://docs.python.org/3/library/index.html
- The "Python library" encompasses various types of components.
- It includes data types that are typically considered part of the core of a language, such as numbers and lists.
- The library also contains built-in functions and exceptions—objects that can be used by all Python code without the need for an import statement. While some of these are defined by the core language, many are not essential for the core semantics and are only described in this documentation.
- This manual, available at (https://docs.python.org/3/library/index.html) is organized "from the inside out." It first describes the built-in functions, data types, and exceptions and finally covers the modules, grouped in chapters of related modules.

Python standard library 2/3

- One of Python's greatest strengths is its extensive standard library.

Python standard library 2/3

- One of Python’s greatest strengths is its extensive standard library.
- Python supports many internet protocols.

Python standard library 2/3

- One of Python's greatest strengths is its extensive standard library.
- Python supports many internet protocols.
- Python includes modules for working with relational databases.

Python standard library 2/3

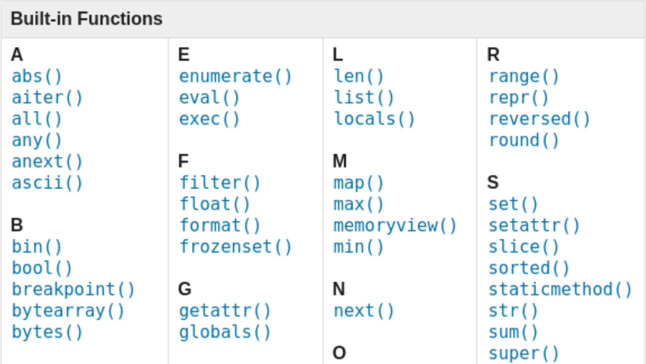
- One of Python's greatest strengths is its extensive standard library.
- Python supports many internet protocols.
- Python includes modules for working with relational databases.
- Python also offers modules for creating graphical user interfaces.

Python standard library 2/3

- One of Python's greatest strengths is its extensive standard library.
- Python supports many internet protocols.
- Python includes modules for working with relational databases.
- Python also offers modules for creating graphical user interfaces.
- Python comes with a wide range of built-in functions. You can find a list of these functions here (https://docs.python.org/3/library/functions.html)

Python standard library 2/3

- One of Python's greatest strengths is its extensive standard library.
- Python supports many internet protocols.
- Python includes modules for working with relational databases.
- Python also offers modules for creating graphical user interfaces.
- Python comes with a wide range of built-in functions. You can find a list of these functions here (https://docs.python.org/3/library/functions.html)



Built-in Functions

A
abs()
aiter()
all()
any()
anext()
ascii()

B
bin()
bool()
breakpoint()
bytearray()
bytes()

E
enumerate()
eval()
exec()

F
filter()
float()
format()
frozenset()

G
getattr()
globals()

L
len()
list()
locals()

M
map()
max()
memoryview()
min()

N
next()

O

R
range()
repr()
reversed()
round()

S
set()
setattr()
slice()
sorted()
staticmethod()
str()
sum()
super()

Python standard library 3/3

- In addition to the standard library, Python Package Index (PyPI), the official repository for third-party Python software, contains over 485 000 packages (as of September 2023)
- Example of famous third-party Python packages on PyPi:
  - Pandas (data analysis/datascience)
  - Matplotlib (data exploration and vizualization)
  - Seaborn (high end graphics and drawing)
  - Scikit-learn (data analysis, machine learning)

Python module - example of **Random** 1/4

-Python includes a multitude of modules in its standard library, covering a wide range of subjects and problem domains, including network operations, text processing, mathematics, file and directory access, cryptography, and more.

https://docs.python.org/3/library/index.html

- The standard library includes a dedicated module for random (pseudo-random) number generation.

https://docs.python.org/3/library/random.html

Python module - example of **Random** 2/4

- There are several ways to import a python module

```
import random # import random
int_list =[1,2,3]
random.shuffle(int_list)# from that object you have to access all the functions
print(int_list)
```

```
## [3, 2, 1]
```

```
import random as rand # import random using a custom local name
rand.shuffle(int_list) # from that object you have to access all the functions
print(int_list)
```

```
## [2, 3, 1]
```

```
from random import shuffle,randint,choice # import only needed function
shuffle(int_list) # use the function directly without object before
print(int_list)
```

```
## [1, 3, 2]
```

```
from random import * # import all the functions bundled inside Random at once
shuffle(int_list)
print(int_list)
```

Python module - example of **Random** 3/4

```
from random import *

print(randint(1, 100))    # Pick a random integer between 1 and 100.
```

```
## 96
```

```
print(uniform(1, 100))    # Pick a random float between 1 and 100.
```

```
## 45.511995740238255
```

```
# prints a random value from the list
list1 = [1, 2, 3, 4, 5, 6]
print(choice(list1))
```

```
## 6
```

```
items = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
y = sample(items, 4)    # Pick 4 random items from the list
print(y)
```

```
## [8, 3, 9, 5]
```

Python module - example of **Random** 4/4

```
# using randrange() to generate in range from 20
# to 50. The last parameter 3 is step size to skip
# three numbers when selecting.
print("A random number from range is : ", end="")
```

```
## A random number from range is :
```

```
print(randrange(20, 50, 3))
```

```
## 41
```

Exercises

- Exercise 1: Lottery pick. Generate 100 random lottery tickets (one ticket is a sequence of 5 digits) and pick one winner out of it.
- Exercise 2: write a program that generates a random 10 character long password including 6 letters with 2 of them uppercase, 1 digit and 1 special symbol.
- Exercise 3: Monte Carlo estimation of Pi: one way to estimate the value of the pi is to generate a large number of random points in the unit square and see how many fall within the unit circle; their proportion is an estimate of the area of the circle. See https://academo.org/demos/estimating-pi-monte-carlo. Implement the proposed algorithm to estimate the value of pi.
- Exercise 4: Write a program that prints the first N rows of Pascal's triangle (see https://www.youtube.com/watch?v=XMriWTvPXHI).