Python

ELYSIAN EDU

Week 1
Python Basic

Basic Python Programming

Python for ML

Python

ELYSIAN EDU
AI LEARNING CENTER

Table of contents

ELYSIAN EDU
AI LEARNING CENTER

01

Introduction to Python

Python

Python

ELYSIAN EDU
AI LEARNING CENTER

1.1. What is Python ?

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development (server-side),
- Software development,
- Data science
- Machine learning

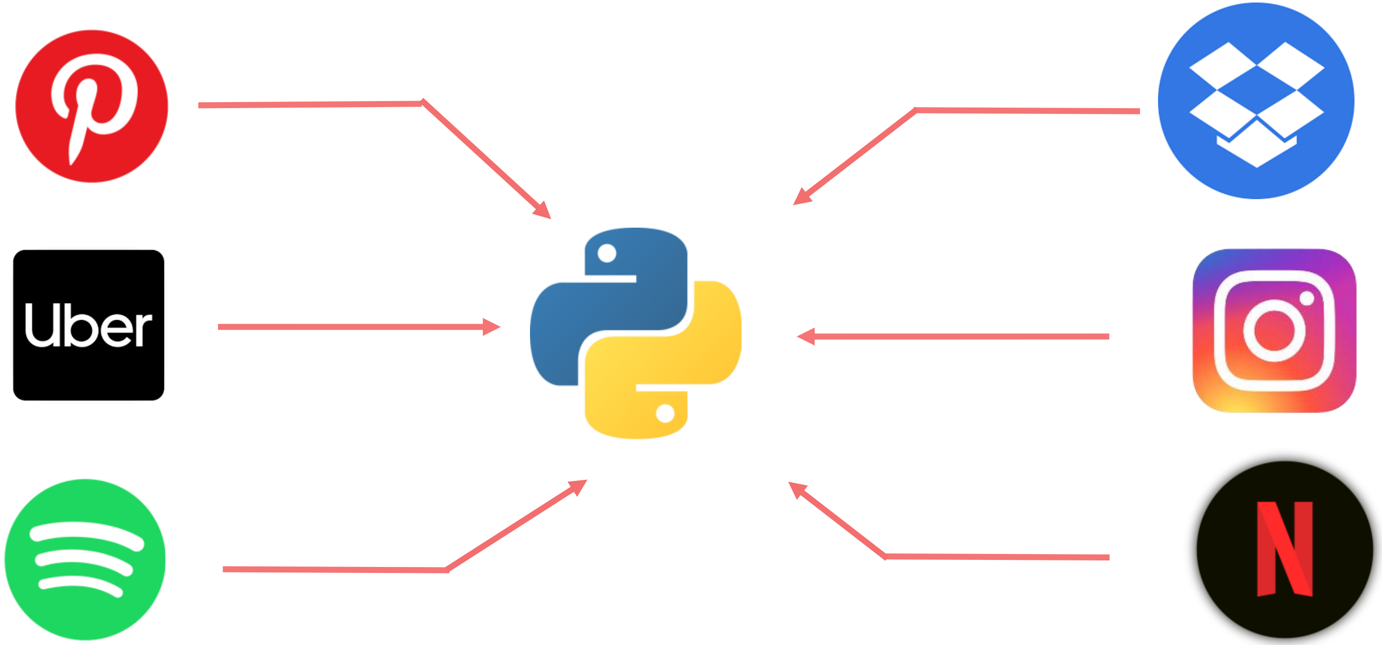
ELYSIAN EDU
AI LEARNING CENTER

1.2. What can Python do ?

- Python can be used **on a server to create web applications**.
- Python can **connect to database systems**. It can also read and modify files.
- Python can be used to **handle big data and perform complex mathematics.**
- Python can be used for **rapid prototyping, or for production-ready software development.**

Python

1.3 Application Developed by Python



Python

1.4. Why Python for Data Analysis

ELYSIAN EDU
AI LEARNING CENTER

- Python **works on different platforms**.
- Python has a **simple syntax** similar to the English language.
- Python has **abundant open-source libraries**.
- Python has syntax that **allows developers to write programs with fewer lines** than some other programming languages.
- Python **relies on indentation**, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

Python

1.4. Essential Python Libraries for ML

pandas

SciPy

matplotlib

NumPy

PyTorch

scikit
learn

ELYSIAN EDU
AI LEARNING CENTER

02

Environment Setup

Python

2.1 Conda Installation

Installation

- conda install <package_name>
- pip install <package_name>

Update

- conda update <package_name>
- pip install —upgrade <package_name>

Python

ELYSIAN EDU
AI LEARNING CENTER

03

Python 2
vs
Python 3

Python

Python

3.1 Python 2 vs Python 3

ELYSIAN EDU
AI LEARNING CENTER

Comparison Parameter	Python 2	Python 3
Year of Release	released in the year 2000.	released in the year 2008.
Storage of Strings	strings are stored as ASCII by default.	strings are stored as UNICODE by default.
Division of Integers	On the division of two integers, we get an integral value in Python 2. For instance, 7/2 yields 3 in Python 2.	On the division of two integers, we get a floating-point value in Python 3. For instance, 7/2 yields 3.5 in Python 3.
Ease of Syntax	has more complicated syntax than Python 3.	as an easier syntax compared to Python 2.
Libraries	A lot of libraries of Python 2 are not forward compatible.	A lot of libraries are created in Python 3 to be strictly used with Python 3.
Application	Python 2 was mostly used to become a DevOps Engineer. It is no longer in use after 2020.	Python 3 is used in a lot of fields like Software Engineering, Data Science, etc.

Python

3.2 Python 2 vs Python 3

ELYSIAN EDU
AI LEARNING CENTER

Python 2

```
def main():
  print "Hi! This is Python 2"

if __name__== "__main__":
    main()
```

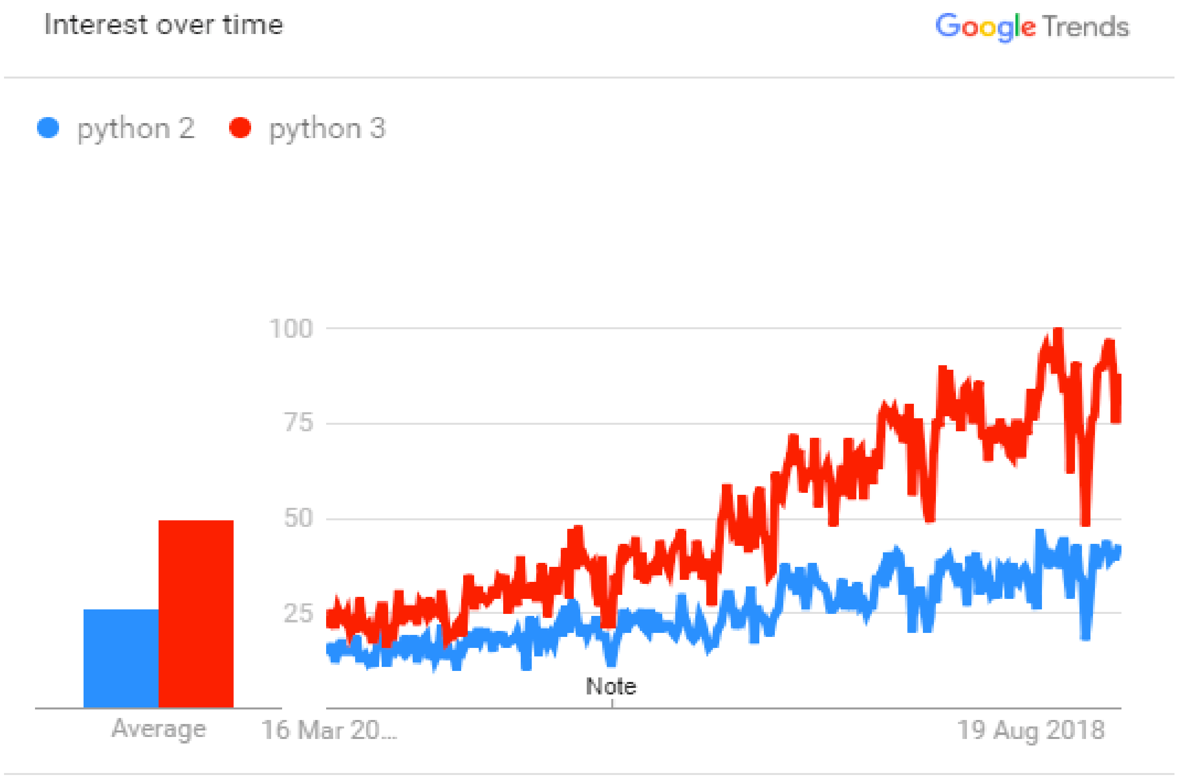
Python 3

```
def main():
  print ("Hi! This is Python 2“)

if __name__== "__main__":
    main()
```

Python

3.3 Python 2 vs Python 3



Python

ELYSIAN EDU
AI LEARNING CENTER

04

Python Basis Syntax

Python

Python

4.1 Python Indentation

- Indentation refers to the spaces at the beginning of a code line.
- Where in other programming languages the indentation in code is for readability only, the indentation in Python is very important.
- Python uses indentation to indicate a block of code.
- The number of spaces is up to you as a programmer, the most common use is four, but it has to be at least one.

```
if 5 > 2:
 print("Five is greater than two!")
if 5 > 2:
        print("Five is greater than
two!")
```

```
if 5 > 2:
 print("Five is greater than two!")
        print("Five is greater than
two!")
```

Python

ELYSIAN EDU
AI LEARNING CENTER

4.2 Variables

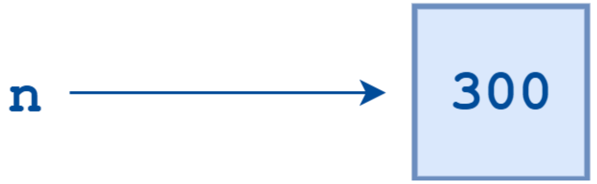
- **Variable Assignment**
  - n=300
  - a=b=c=300
- **Variable Types in Python**
  - a = 20
  - print(a)

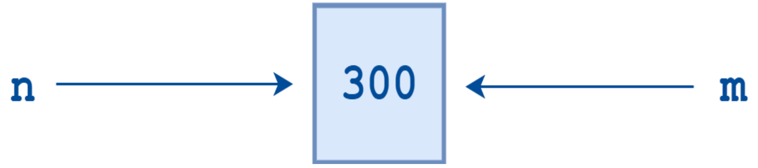
Python

4.4 Object References

- A Python variable is a symbolic name that is a reference or pointer to an object. Once an object is assigned to a variable, you can refer to the object by that name. But the data itself is still contained within the object.

```
m=300
n=300
```





- M = 200
- N = M.copy()
- N

Python

4.5 Object References (cont'd)

ELYSIAN EDU
AI LEARNING CENTER

- Python is a highly object-oriented language. In fact, virtually every item of data in a Python program is an object of a specific type or class.

```
print(300)
```

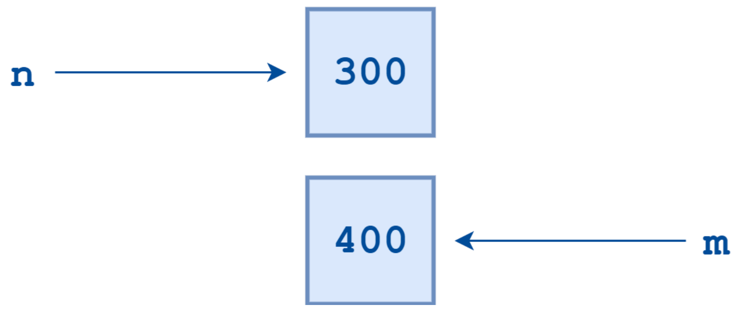
- When presented with the statement print(300), the interpreter does the following:
    - Creates an integer object
    - Gives it the value 300
    - Displays it to the console

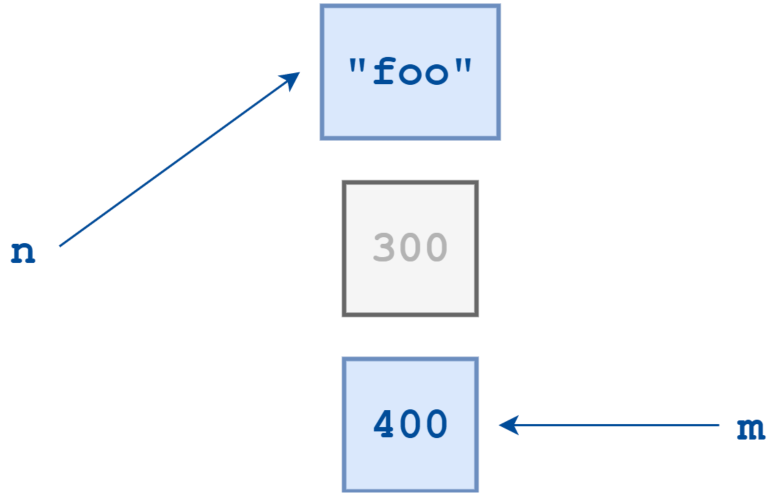
Python

4.5 Object References

```
m=400
```

```
n = "foo"
```





Python

4.6 Variable Names

Officially, variable names in Python can be any length and can consist of uppercase and lowercase letters (`A-Z`, `a-z`), digits (`0-9`), and the underscore character (_). An additional restriction is that, although a variable name can contain digits, the first character of a variable name cannot be a digit.

- **Camel Case**: Second and subsequent words are capitalized
  - Example: `numberOfCollegeGraduates`
- **Pascal Case**: Identical to Camel Case, except the first word is also capitalized.
  - Example: `NumberOfCollegeGraduates`
- **Snake Case**: Words are separated by underscores.
  - Example: `number_of_college_graduate`

**Remember that variable names are case-sensitive

Python

ELYSIAN EDU
AI LEARNING CENTER

4.7 Naming Styles

Type	Naming Convention	Examples
Function	Use a lowercase word or words. Separate words by underscores to improve readability.	function, my_function
Variable	Use a lowercase single letter, word, or words. Separate words with underscores to improve readability.	x, var, my_variable
Class	Start each word with a capital letter. Do not separate words with underscores. This style is called camel case.	Model, MyClass
Method	Use a lowercase word or words. Separate words with underscores to improve readability.	class_method, method

Python

4.7 Naming Styles

Constant	Use an uppercase single letter, word, or words. Separate words with underscores to improve readability.	CONSTANT, MY_CONSTANT, MY_LONG_CONSTANT
Module	Use a short, lowercase word or words. Separate words with underscores to improve readability.	module.py, my_module.py
Package	Use a short, lowercase word or words. Do not separate words with underscores.	package, mypackage
Constant	Use an uppercase single letter, word, or words. Separate words with underscores to improve readability.	CONSTANT, MY_CONSTANT, MY_LONG_CONSTANT
Module	Use a short, lowercase word or words. Separate words with underscores to improve readability.	module.py, my_module.py

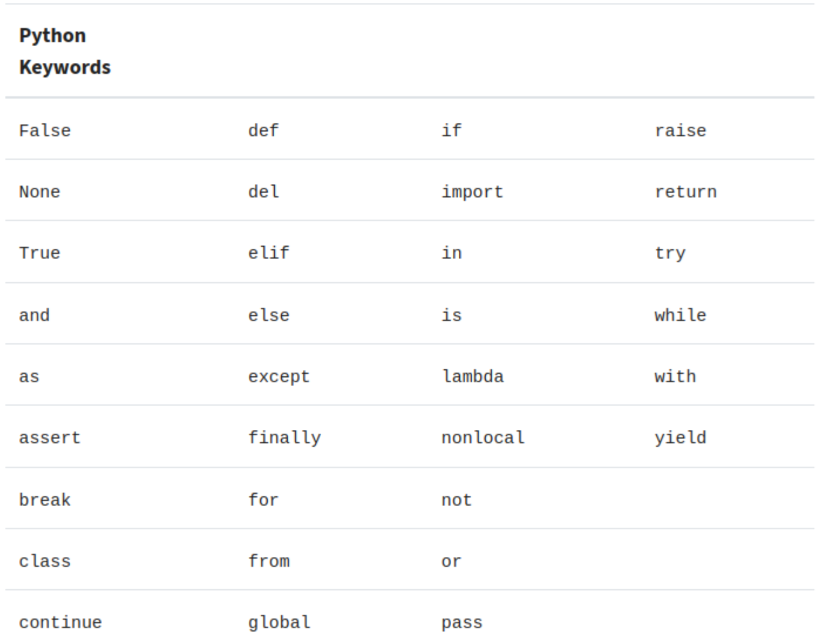
Python

4.8 Reserved Keywords

There is one more restriction on identifier names. The Python language reserves a small set of keywords that designate special language functionality. No object can have the same name as a reserved word.

In Python 3.6, there are

33 reserved keywords:



Python Keywords			
`False`	`def`	`if`	`raise`
`None`	`del`	`import`	`return`
`True`	`elif`	`in`	`try`
`and`	`else`	`is`	`while`
`as`	`except`	`lambda`	`with`
`assert`	`finally`	`nonlocal`	`yield`
`break`	`for`	`not`	
`class`	`from`	`or`	
`continue`	`global`	`pass`	

ELYSIAN EDU
AI LEARNING CENTER

Thanks!

Python

Do you have any questions?