



Python Basics



Session-1





General Information about Python



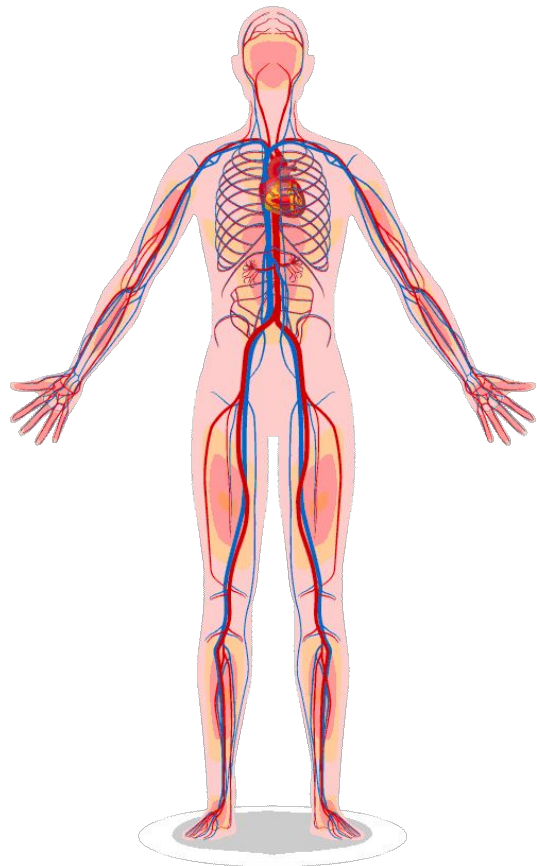
Table of Contents



- ▶ What is Python?
- ▶ Historical Development of Python
- ▶ Review of Tools & Installations
- ▶ First Program 'Hello World!'
- ▶ Matter of Quotes

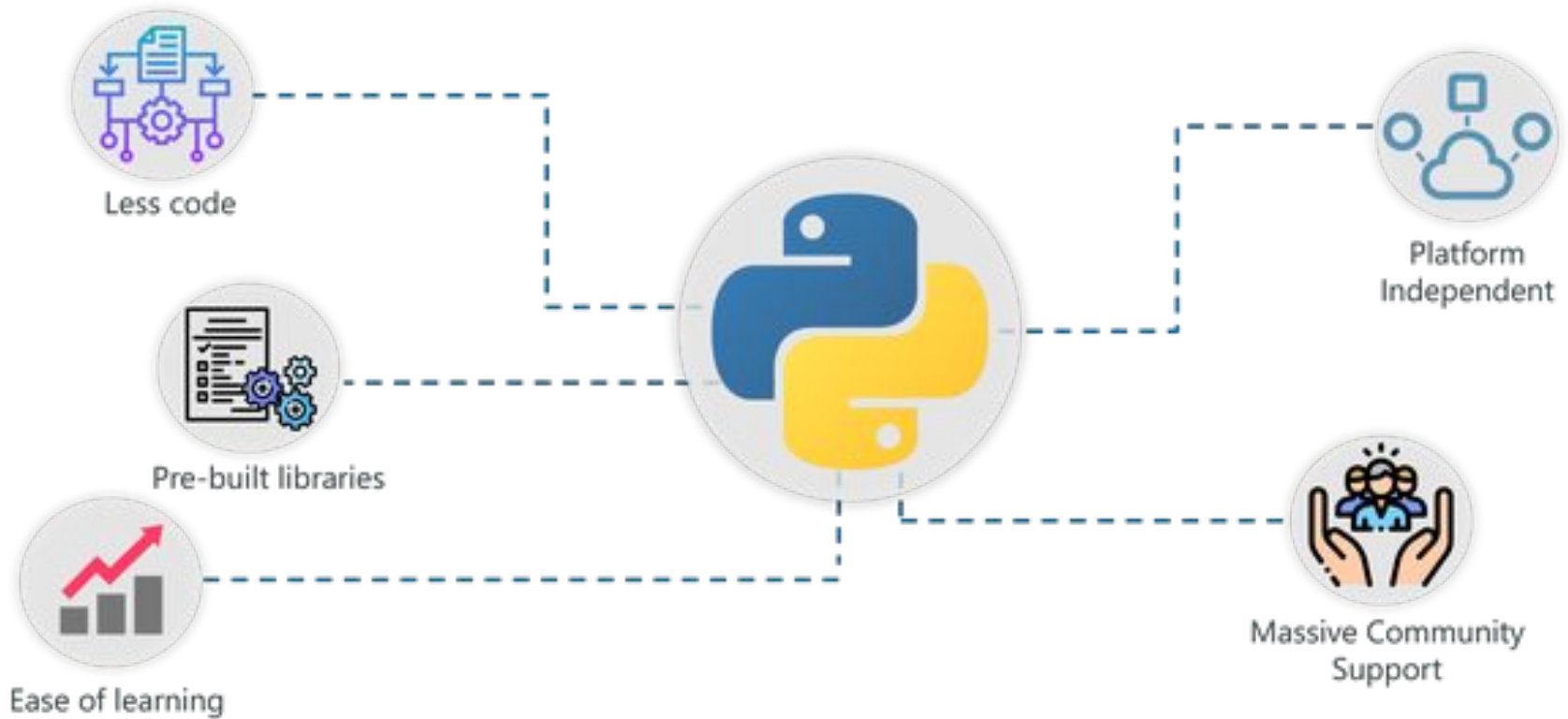


What is Python?



Mr. IT

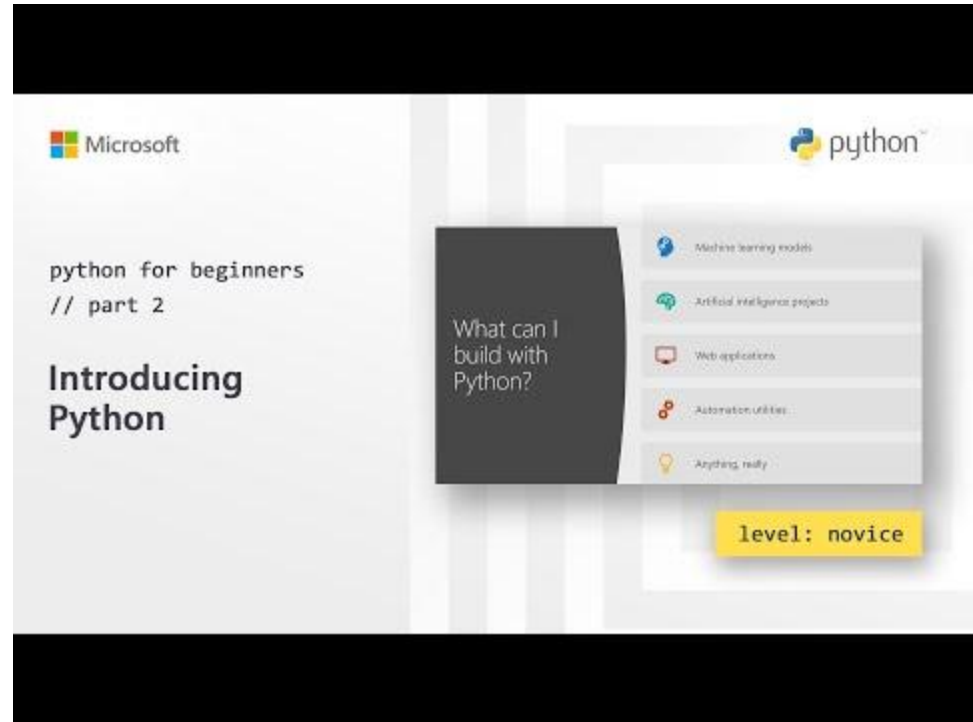
What is Python? (review)





What is Python?

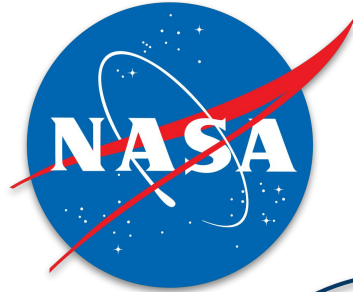
- Easy to **learn**
- Easy to **use**
- Easy to **run**
- Easy to **read**
- Easy to **develop**
- Easy to **teach..**



'Microsoft Developers' on Youtube :

-Introducing Python (*duration: 3 min 9 sec*)

What is Python?

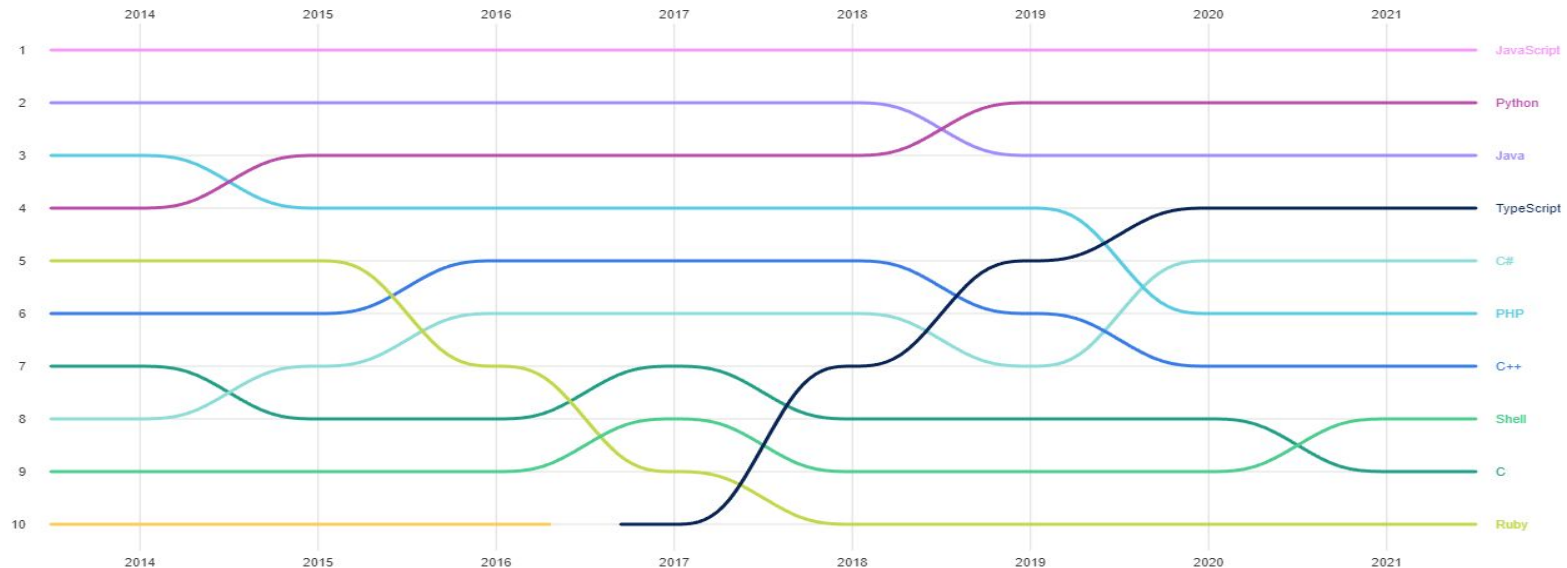


Why Python?



Some indexes showing the position of Python among the other programming languages.

Top languages over the years













2021 Github's State of The Octoverse Report :

-The Second Most Popular Language

Why Python?



Some indexes showing the position of Python among the other programming languages.

| Oct 2022 | Oct 2021 | Change | Programming Language | | Ratings | Change |
|----------|----------|--------|---|-------------------|---------|--------|
| 1 | 1 | |  | Python | 17.08% | +5.81% |
| 2 | 2 | |  | C | 15.21% | +4.05% |
| 3 | 3 | |  | Java | 12.84% | +2.38% |
| 4 | 4 | |  | C++ | 9.92% | +2.42% |
| 5 | 5 | |  | C# | 4.42% | -0.84% |
| 6 | 6 | |  | Visual Basic | 3.95% | -1.29% |
| 7 | 7 | |  | JavaScript | 2.74% | +0.55% |
| 8 | 10 | ▲ |  | Assembly language | 2.39% | +0.33% |
| 9 | 9 | |  | PHP | 2.04% | -0.06% |
| 10 | 8 | ▼ |  | SQL | 1.78% | -0.39% |

2022 Tiobes.com's Index :

- The Most Popular Language



► First Program for 'Hello World!'

- Writing a text is **quite simple** in Python.

```
print('Hello World!')
```



5

Matter of Quotes



Matter of Quotes

What we wrote is a **string datatype**. Strings are always in **quotes**. There are basically two types of quotes we use in Python. **Single** or **double** quotes. Both are the same but we should use them in the correct way.

Single



```
print('Hello World!')
```

Double



```
print("Hello World!")
```



Matter of Quotes



Choose one of them and stick to it.



```
print('Hello World!')  
print("Clarusway School!")  
print("I'm happy to learn!")
```



```
print("Hello World!")  
print('Clarusway School!')  
print('I'm happy to learn!')
```



Matter of Quotes



The alternative valid use of quotes.

```
print(''Lorem ipsum dolor sit amet,  
  fusce ut quisque neque donec,  
  massa metus amet, luctus inceptos.'')  
  
print("""Praesent a, mi mattis velit metus,  
  accumsan adipiscing ipsum sit justo  
  penatibus, amet mauris non tempus justo.""")
```

Triple use
of **'single'**
and
"double"
quotes



Matter of Quotes



The alternative valid use of quotes.

Output1:

```
Lorem ipsum dolor sit amet,  
fusce ut quisque neque donec,  
massa metus amet, luctus inceptos.
```

Output2:

```
Praesent a, mi mattis velit metus,  
accumsan adipiscing ipsum sit justo  
penatibus,  
amet mauris non tempus justo.
```

Triple use of
**'single and
double'**
quotes



Matter of Quotes



What is the difference of these two syntaxes?

```
print('3.14')  
print(3.14)
```

The outputs are the same but
quotes make data string type.

3.14

3.14



Matter of Quotes



What is the output?

```
print('' 'We should have enough time for our family" ' '')
```

```
'We should have enough time for our family'
```



Matter of Quotes



Multiple lines of codes.

```
print('first line')  
print()  
print(''third line'')
```

first line

third line

print() prints
an empty line



PEP 8 Conventions



Table of Contents



- ▶ What is PEP 8?
- ▶ Some Important PEP 8 Rules

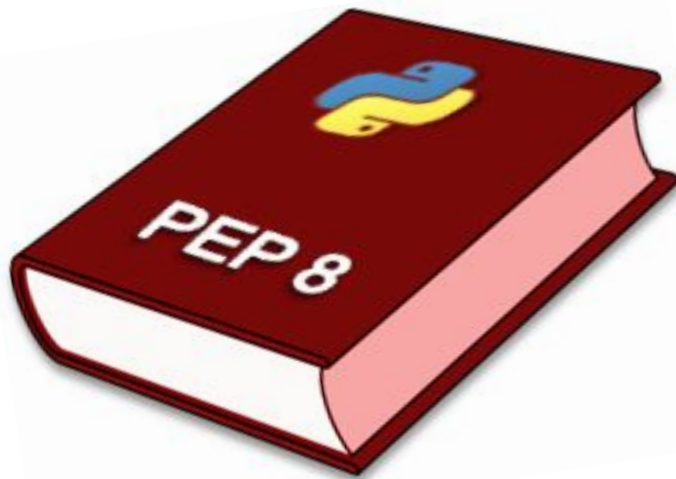


What is PEP 8?

PEP 8



Python **E**nhancement **P**roposal

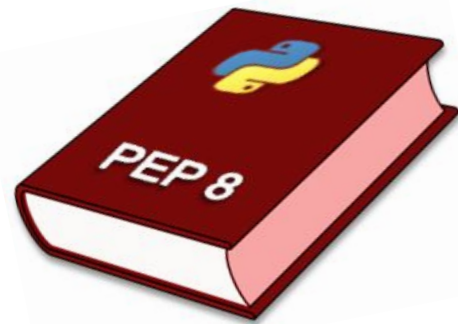




▶ What is PEP 8?

PEP 8 is a style guide about consistency.

- ▶ Consistency with this style guide is **important**.
- ▶ Consistency within a project is **more important**.
- ▶ Consistency within one module or function is the **most important coding aspect**.





Some Important PEP8 Rules

- ▶ Writing a text is **quite simple** in Python.

```
print('being a good person')
```

Limit the code lines to a maximum **79** characters



Some Important PEP8 Rules

- ▶ Avoid extraneous **spaces** in situations such as:

Immediately inside parentheses, brackets or braces :

YES : `spam(meat[1], {milk: 2})` , **NO** : `spam(meat[1], { milk: 2 })`

Between a trailing comma and a following close parenthesis :

YES : `df[0,]` or `foo = (2,)` , **NO** : `df[0,]` or `foo = (2,)`

Immediately before a comma, semicolon, or colon :

YES : `if y == 3: print x, y; x, y = y, x` , **NO** : `if y == 3 : print x , y ; x , y = y , x`

Immediately before the open parenthesis that starts the argument list of a function call:

YES : `print('peace')` , **NO** : `print ('peace')`



Some Important PEP8 Rules

- More than one space around an assignment (or other) operator to align it with another:

| YES | NO |
|----------------------------|----------------------------|
| <code>x = 3</code> | <code>x = 3</code> |
| <code>y = 4</code> | <code>y = 4</code> |
| <code>long_vars = 5</code> | <code>long_vars = 5</code> |



Some Important PEP8 Rules

- ▶ Always surround these operators with a single space on either side. Such as:

=, +=, ==, <, >, >=, **in**, **not in**, **is**, **and**, **or**, **not**

- ▶ Failure to follow the basic rules of PEP 8 **does not make** your program **wrong** or **unable to work**.



Don't ask '**Why**' regarding the conventional rules.





First Steps into Coding





First Program for 'Being a Good Person'



- ▶ Writing a text is **quite simple** in Python.

```
print('Hello World!')
```

```
Hello World!
```



Comments



Introduction



I can't
remember
why I typed
these codes...

```
x = 10  
y = 5  
a = (x / y) * 100  
  
print(a)
```

```
x = 10  
y = 5  
a = (x / y) * 25  
  
print(a ** 2)
```





► Comments

► **Single-line Comments :**

```
# This is a single line comment
```



Comments

► Single-line Comments :

```
# This is a single line comment
```

► Inline Comments :

```
print('hello') # This is an inline comment
```

two spaces

one space



These spacing principles are just PEP8 conventional rules.

```
hello
```




Comments

► Multi-line Comments :

```
print('hello')  
# First multi-line comment  
# Second multi-line comment  
# Third multi-line comment
```

```
hello
```



Comments

Keep these in your mind ! :

► **Comments should be :**

- ▷ **Sufficient**
- ▷ **Necessary**
- ▷ **Updated**

