PYTHON

MahmoudAlipour.a@Gmail.Com
@AminAlipour_IT

filoger

Syllabus

S01
- Basics
- Operators

S02
- Control Flow & Iteration
- Function

S03
- List
- Tuples
- Set
- String

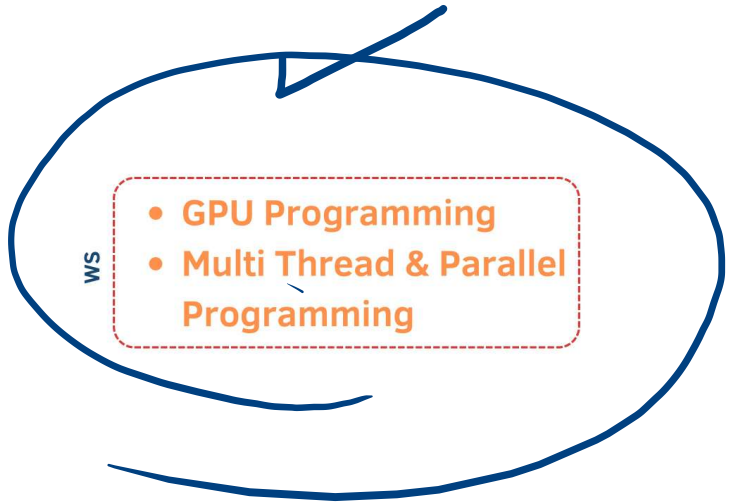
S04
- Exception Handling
- OOP (Object & Class)

S05
- Inheritance

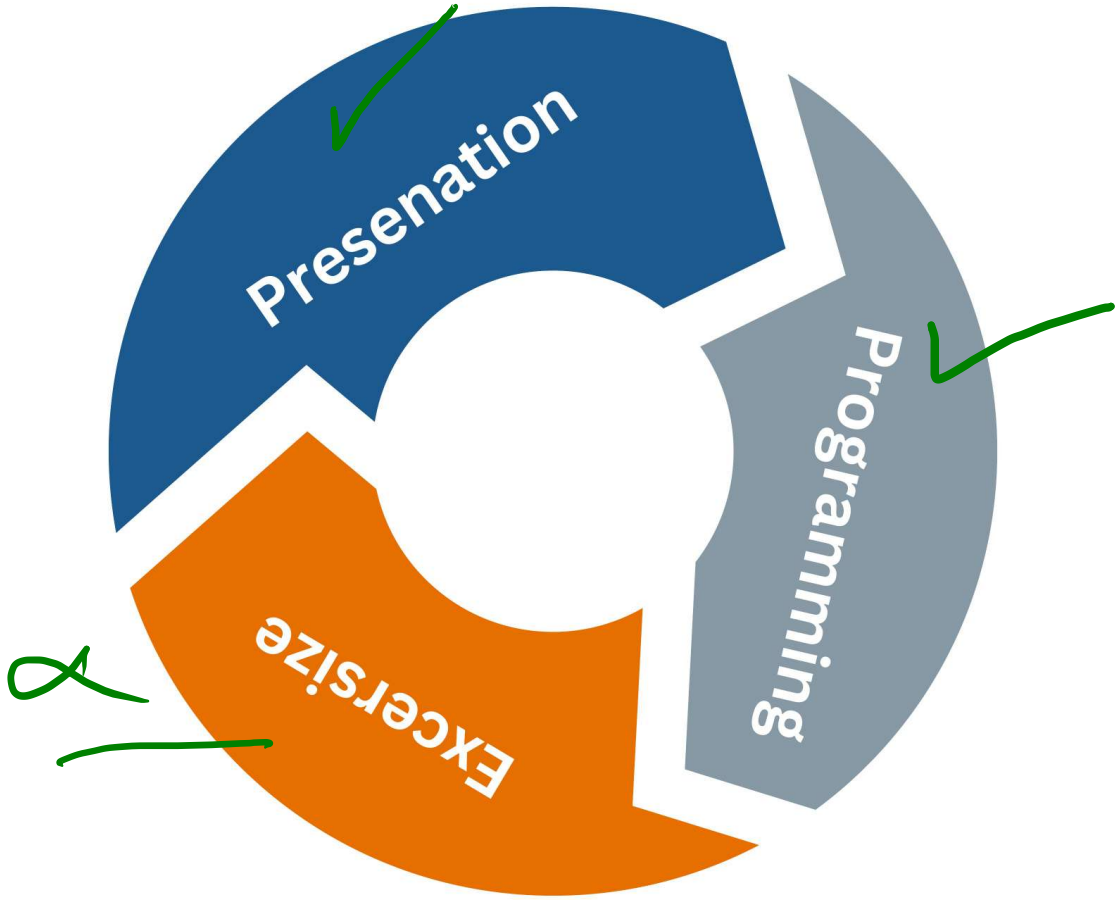
S06
- Abstraction
- Encapsulation

S07
- Numpy

S08
- Pandas

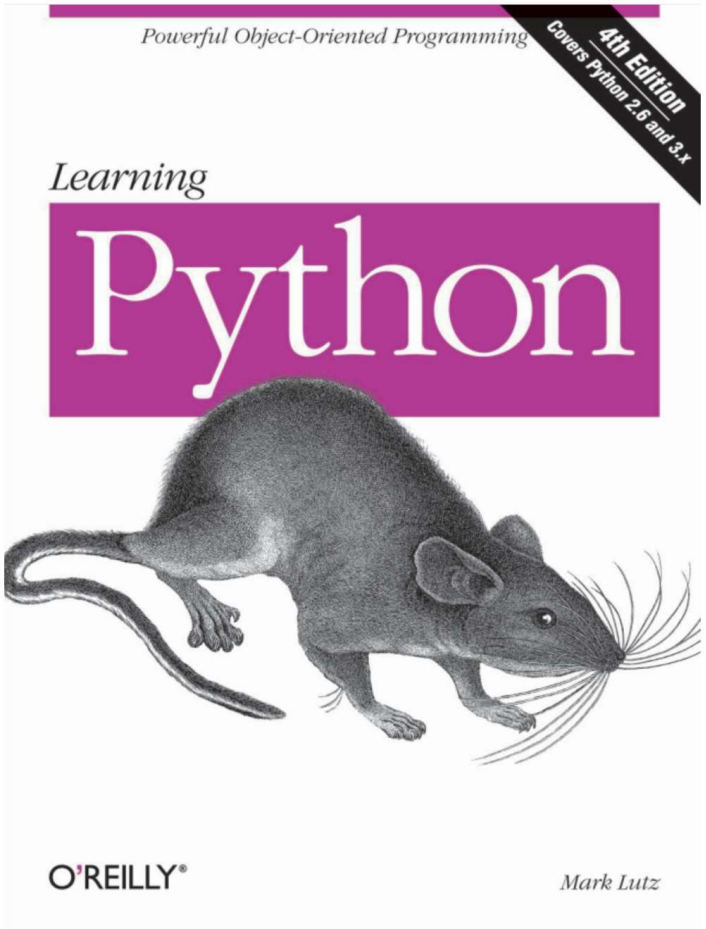


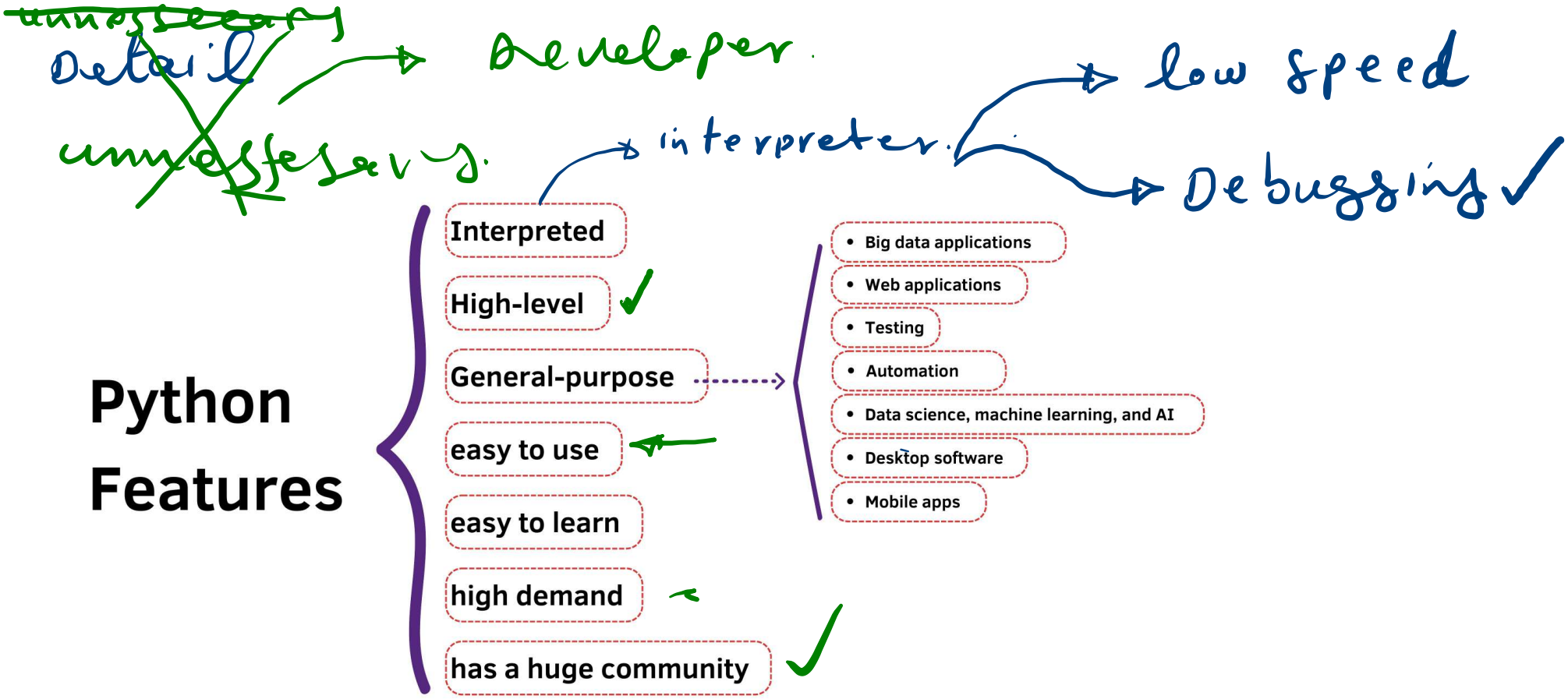
Teaching Methodology

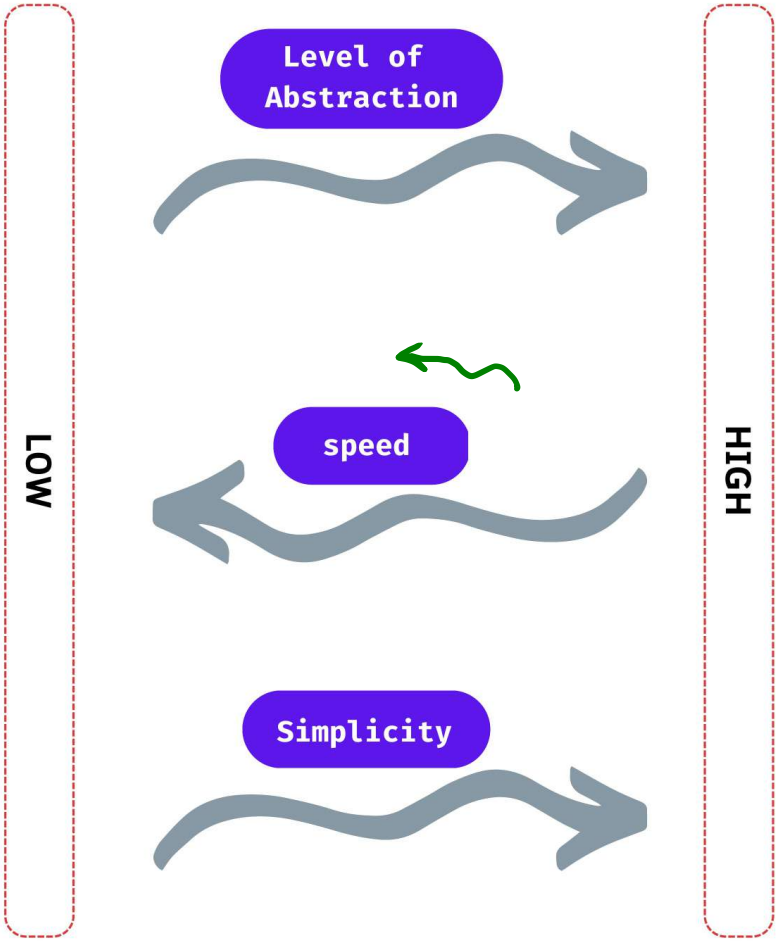


Ref.

- **Learning Python**
  Mark Lutz (4th Edition)
- **ocw.mit.edu**
- **pythontutorial.net**
- **github.com/Asabeneh**

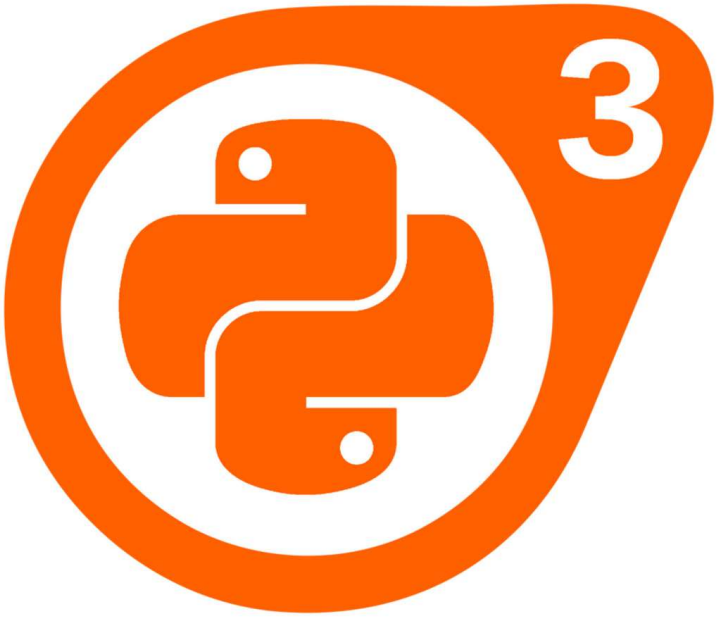



unnecessary
detail
unnecessary.
Developer.
interpreter.
low speed
Debugging ✓
Python Features
Interpreted
High-level ✓
General-purpose
easy to use
easy to learn
high demand
has a huge community ✓
• Big data applications
• Web applications
• Testing
• Automation
• Data science, machine learning, and AI
• Desktop software
• Mobile apps



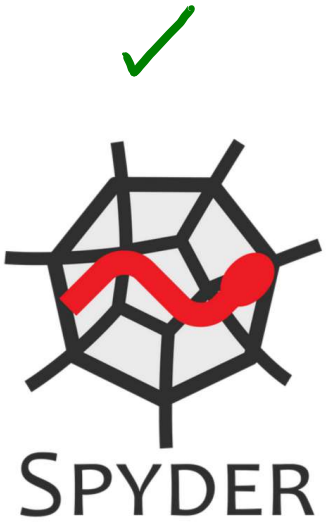
src: c-sharpcorner.com

Py2 | Py3 ?

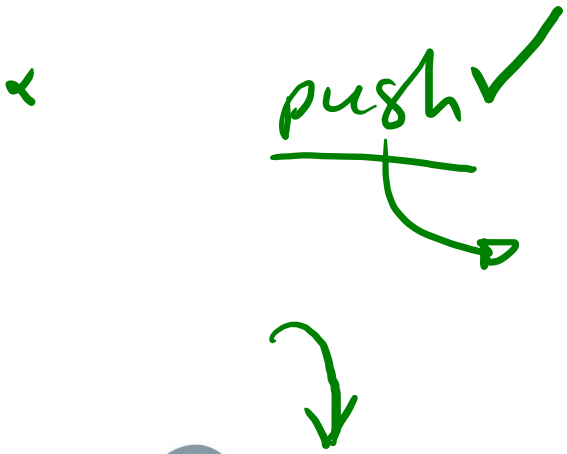


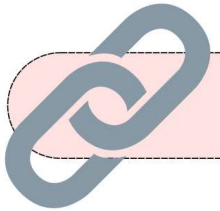
3.10

IDE



File Sharing

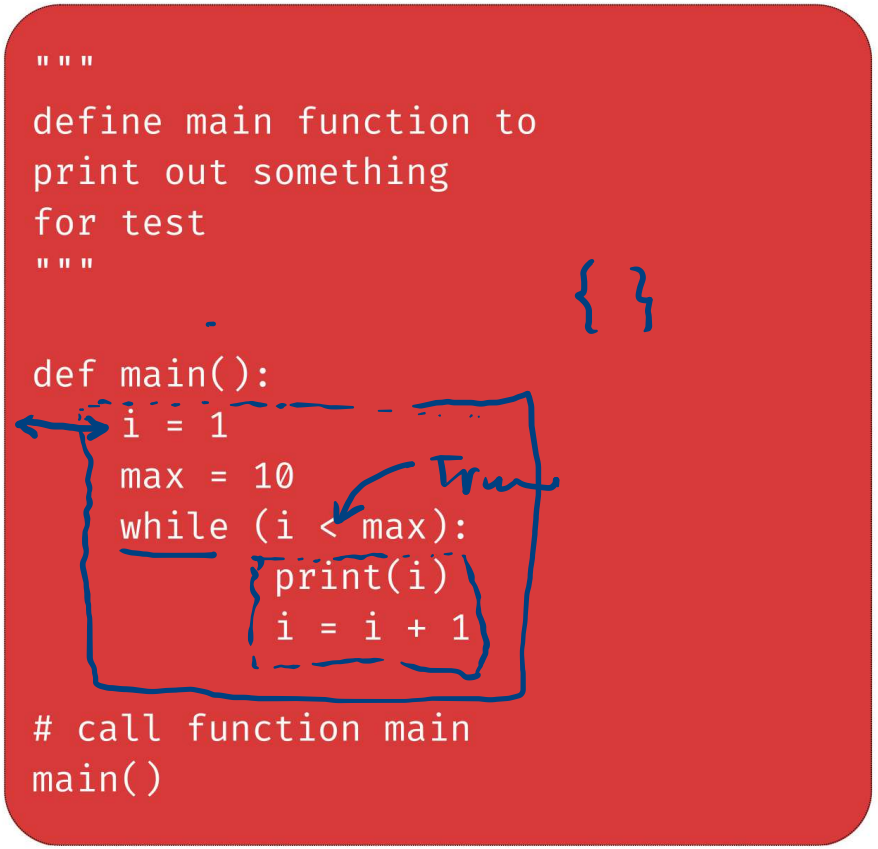




https://github.com/MhmudAlpurd/Vision_Feb2023.git

Basics

White space & Indentation



Comment

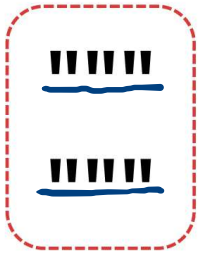
#


```
"""
define main function to
print out something
for test
"""

def main():
    i = 1
    max = 10
    while (i < max):
            print(i)
            i = i + 1

# call function main
main()
```

MultiLine Comment /DocString




```
"""
define main function to
print out something
for test
"""

def main():
    i = 1
    max = 10
    while (i < max):
            print(i)
            i = i + 1

# call function main
main()
```

Continuation of statements

\

```
if (a == True) and \
 (b == False) and (c == True):
     print("Continuation of
statements")
```

var_1 = 1.0

a-z / A-Z / _ / 1. / @ % ...

Identifiers

- Identifiers are names that identify variables, functions, modules, classes, and other objects in Python.
- The name of an identifier needs to begin with a letter or underscore (_). The following characters can be alphanumeric or underscore.
- Python identifiers are case-sensitive.

→ VAR_1 = 1.0

var_1 = 1.0

Keywords

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

Objects

almost

in Python everything is an Object, and Any Object has a Type.

Programs manipulate Data Objects.

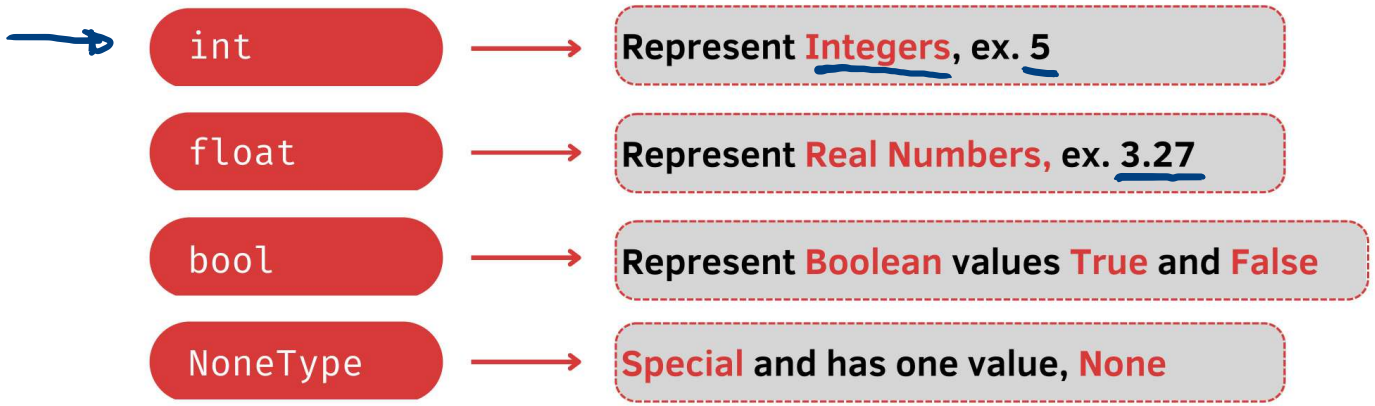
Objects have a Type that defines the kinds of things.

Objects are

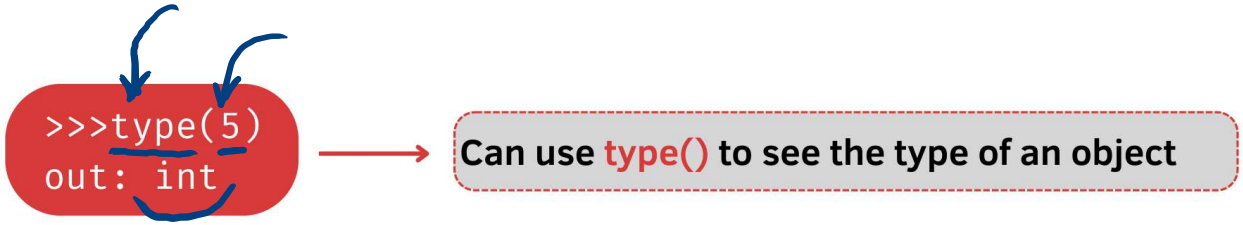
- Scalar (cannot be subdivided)
- Non-scalar (have internal structure that can be accessed)

Type → int
float
list
⋮

Scalar Objects



Type



Double

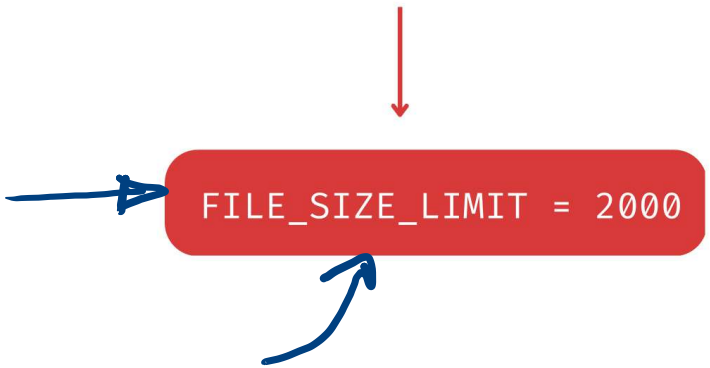
Python does not have an in-built double data type

- const

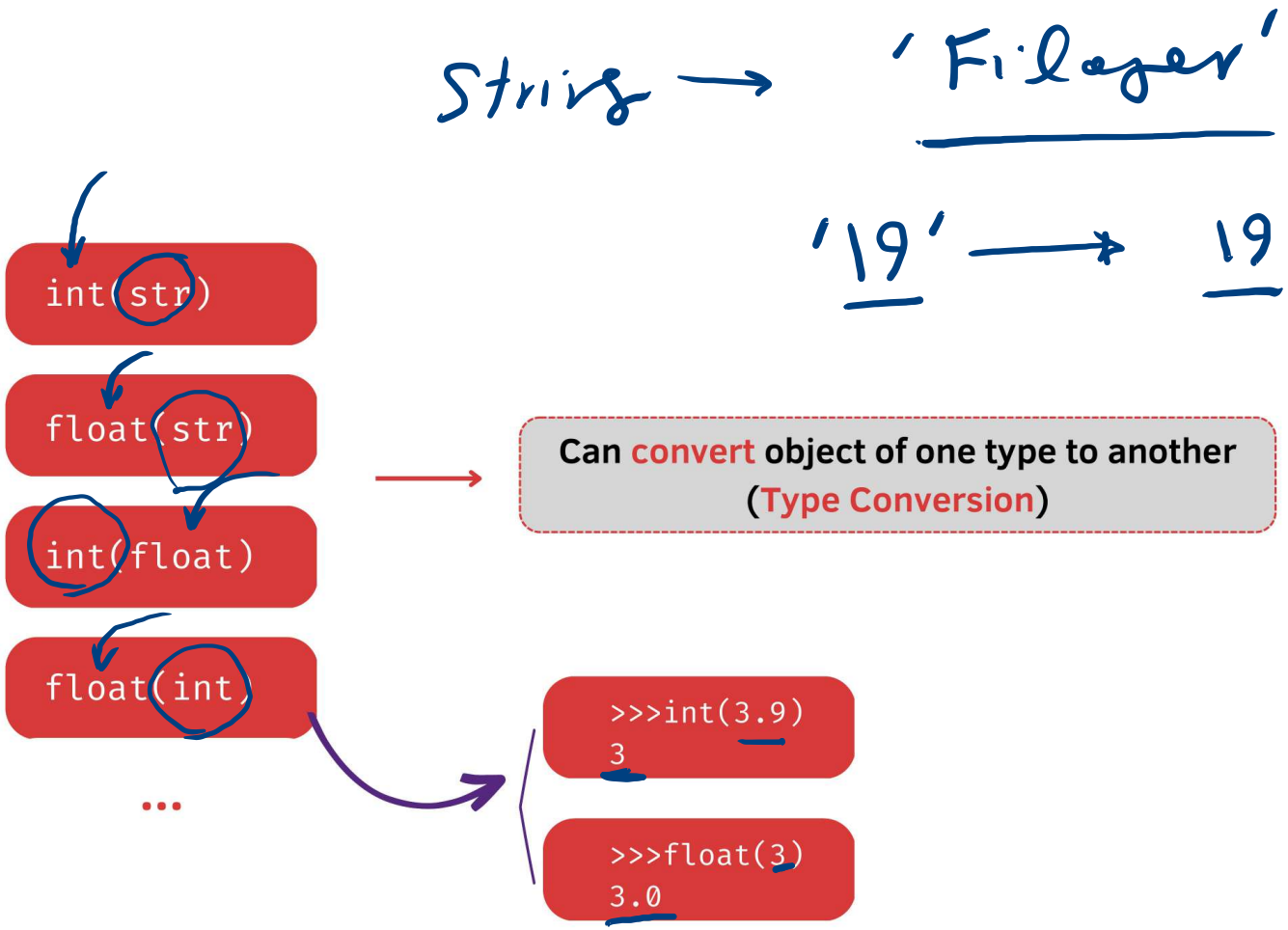
num_Tree_const = 19

Constant

The bad news is that Python doesn't support constants.



Cast


String → 'Filoger'
'19' → 19
int(str)
float(str)
int(float)
float(int)
...
Can convert object of one type to another
(Type Conversion)
>>>int(3.9)
3
>>>float(3)
3.0

Exercises 01

Basics

600

Basics

Exercise 01

- What is the type of the following data?

1. "Filoger"
2. "19.0"
3. 18
4. 13.0
5. True

ipynb

EXs 01

EX 01

EX .2

Basics

Exercise 02

int

- Convert the following data to integer?

1. "Filoger"
2. "19.0"
3. 18
4. 13.0
5. True

Basics

Exercise 03 (Search!)

- Print size of the following data in memory (bytes)?

1. "Filoger"
2. "19.0"
3. 18
4. 13.0
5. True

Basics

Exercise 04 (Search!)

- Print running time of program to find type of the following data ?

1. "Filoger"
2. "19.0"
3. 18
4. 13.0
5. True