Python in One Shot

This video has been made with a lot of love & I hope you guys have an amazing programming journey :)

Why to Use Python?

Python can be used for :

- 1. Programming (for Placements/online contests/DSA)
- 2. Development (using a backend framework called Django)
- 3. Machine Learning / Data Science / Artificial Intelligence

Websites built using Python include Google, Youtube, Instagram, Netflix, Uber & much more.

What to Install?

- 1. Python (https://www.python.org/)
- 2. PyScripter (https://rb.gy/bvnn69)
- 3. PyCharm (https://www.jetbrains.com/pycharm/)

Our First Python Program

print("Hello World")

A Key Point to know about Python

- It is a case sensitive language

Variables

Basic Types in Python - numbers(integers, floating), boolean, strings

Example 1 : name = "shradha" age = 22 print(name) print(age)

Example 2 : name = "shradha" age = 22

name = "aman"
age = 24
<pre>print(name)</pre>
<pre>print(age)</pre>

Example 3 :

first_name = "shradha"
last_name = "khapra"
age = 19
is_adult = True

<pre>print(first_name</pre>	+	 	+	last_	_name)
<pre>print(age)</pre>					
<pre>print(is adult)</pre>					

> Exercise Solution

first_name = "Tony" last_name = "Stark" age = 52 is_genius = True

Taking Input

name = input("What is your name? ")
print("Hello " + name)
print("Welcome to our cool Python class")

> Exercise Solution

superhero = input("What is your superhero name? ")
print(superhero)

Type Conversion

old_age = input("Enter your age : ")
#new_age = old_age + 2
#print(new_age)

new_age = int(old_age) + 2 print(new_age)

#Useful	converion	functions
# 1.	<pre>float()</pre>	
# 2.	bool()	
# 3.	str()	
# 4.	int()	

> Code for Sum of 2 Numbers

first_number = input("Enter 1st number : ")
second_number = input("Enter 2nd number : ")
sum = float(first_number) + float(second_number)

print("the sum is : " + str(sum))

Strings

name	=	"Tony	Stark"
print	t(r	name.uj	<pre>oper())</pre>
print	t(r	name)	

print(name.lower())
print(name)

<pre>print(name.find('y'))</pre>
<pre>print(name.find('Y'))</pre>
<pre>print(name.find("Stark"))</pre>
<pre>print(name.find("stark"))</pre>

print(name.replace("Tony Stark", "Ironman"))
print(name)

#to check if a character/string is part of the main string
print("Stark" in name)
print("S" in name)
print("s" in name)

Arithmetic Operators

print	(5	+	2)	
print	(5	-	2)	
print	(5	*	2)	
print	(5	/	2)	
print	(5	/	7	2)
print	(5	90	2)	
print	(5	**	2)



Operator Precedence

result = 3 + 5 * 2 # 16 or 13 ?
print(result)

Operators	Meaning
()	Parentheses
**	Exponent
+x, -x, ~x	Unary plus, Unary minus, Bitwise NOT
*, /, //, %	Multiplication, Division, Floor division, Modulus
+, -	Addition, Subtraction
<< , >>	Bitwise shift operators
&	Bitwise AND
	Bitwise XOR
	Bitwise OR
<pre>==, !=, >, >=, <, <=, is, is not, in, not in</pre>	Comparisons, Identity, Membership operators
not	Logical NOT
and	Logical AND
or	Logical OR

Comments

This is a comment & useful for people reading your code # This is another line

Comparison Operators

is_greater = 1 > 5
$is_{lesser} = 1 < 5$
1 <= 5
1 >= 5
<pre>is_not_equal = 1 != 5</pre>
is_equal = 1 == 5

Logical Operators

# or -> (atleast one is tru	e)
<pre># and -> (both are true)</pre>	
<pre># not -> (reverses any valu</pre>	e)
number = 2	

Humber – Z
<pre>print(number > 3)</pre>
<pre>print(number < 3)</pre>
<pre>print(not number > 3)</pre>
<pre>print(not number < 3)</pre>
<pre>print(number > 3 and number > 1)</pre>
print(number > 3 or number > 1)

If statements age = 13

if age >= 18:
<pre>print("you are an adult")</pre>
<pre>print("you can vote")</pre>
elif age < 3:
<pre>print("you are a child")</pre>
else:
<pre>print("you are in school")</pre>
<pre>print("thank you")</pre>

Let's build a Calculator

#Our Calculator

first = input("Enter first number : ")
second = input("Enter second number : ")
first = int(first)

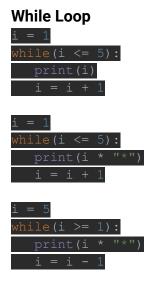
<pre>second = int(second)</pre>
<pre>print("press keys for operator (+,-,*,/,%)")</pre>
<pre>operator = input("Enter operator : ")</pre>
<pre>if operator == "+":</pre>
<pre>print(first + second)</pre>
<pre>elif operator == "-":</pre>
print(first - second)
<pre>elif operator == "*":</pre>
<pre>print(first * second)</pre>
<pre>elif operator == "/":</pre>
<pre>print(first / second)</pre>
elif operator == "%":
print(first % second)
else:
<pre>print("Invalid Operation")</pre>

Range in Python

range() function returns a range object that is a sequence of numbers.

numbers = range(5)
print(numbers)

For iteration (see For Loop section)



For Loop (to iterate over a list)
for i in range(5):
 print(i)

i = i + 1

for i in range(5):
 print(i * "*")
 i = i + 1

Lists

List is a complex type in Python.

friends = ["amar", "akbar", "anthony"]
print(friends[0])
print(friends[1])
print(friends[-1])
print(friends[-2])

friends[0] = "aman"
print(friends)

print(friends[0:2]) #returns a new list

for friend in friends:
 print(friend)

List Methods :

marks = ["english", 95,	"chemistry",	98]
<pre>marks.append("physics")</pre>		
marks.append(97)		
print(marks)		
<pre>marks.insert(0, "math")</pre>		
<pre>marks.insert(1, 99)</pre>		
print(marks)		
<pre>print("math" in marks)</pre>		
print(len(marks)/2)		
<pre>marks.clear()</pre>		
print(marks)		
i = 0		

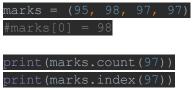
while i < len(marks):
 print(marks[i])
 print(marks[i+1])
 i = i + 2</pre>

Break & Continue
students = ["ram", "shyam", "kishan", "radha", "radhika"]

Tuples

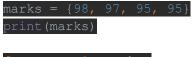
They are like lists (sequence of objects) but they are immutable i.e. once they have been defined we cannot change them.

Parenthesis in tuples are optional.



Sets

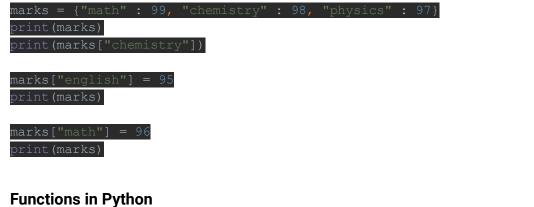
Sets are a collection of all unique elements. Indexing is not supported in sets.



for score in marks:
 print(score)

Dictionary

Dictionary is an unordered collection of Items. Dictionary stores a (key, value) pair.



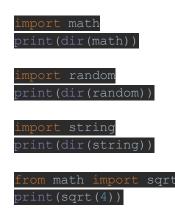
Function is a piece of code that performs some task. (In a tv remote, each button performs a functions, so a function is like that button in code)

There are 3 types of functions in Java :

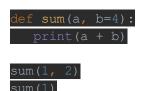
- a. In-built functions
 # int() str() float() min() range() max()
- b. Module functions

Module is a file that contains some functions & variables which can be imported for use in other files.

Each module should contain some related tasks Example : math, random, string



c. User-defined functions



For Machine Learning, refer : <u>https://www.youtube.com/watch?v=1vsmaEfbnoE</u>

Some additional Links :

• <u>https://rb.gy/gjpmwg</u> (A Python GUI)

Some useful Modules

- <u>https://github.com/Embarcadero/DelphiFMX4Python</u>
- <u>https://github.com/Embarcadero/DelphiVCL4Python</u>