Introduction to Python: Basic Python, Pandas, Matplotlib

Anas Samara
asamara@bethlehem.edu

2023, AUGUST 14

Outline

- Development environment
- Python basic concepts:
  - Variables and Data types
  - Loops and Conditional statements
  - Functions
  - Arrays, Lists, Tuples and Dictionaries
- First Steps in Python (via HIFIS)

Development Environment

- Cloud:
  - Google Colab https://colab.research.google.com/
  - **Jupyter** Notebook https://hifis.net/aai/

- On-premise:
  - Jupyter Notebook via Anaconda
  - Spyder
  - PyCharm

Variables, types and Casting

- Variables do not need to be declared with any particular **type**

```
x = 4       # x is of type int
x = "Sally" # x is now of type str
```

- You can get the data type of a variable with the **type()** function.

```
x = 5
y = "John"
print(type(x))  #   <class 'int'>
print(type(y))  #   <class 'str'>
```

- **Casting** used to specify the data type of a variable

```
x = str(3)    # x will be '3'
y = int(3)    # y will be 3
z = float(3)  # z will be 3.0
```

Common Data types

Example	Data Type
`x = "Hello World"`	str
`x = 20`	int
`x = 20.5`	float
`x = 1j`	complex
`x = ["apple", "banana", "cherry"]`	list
`x = ("apple", "banana", "cherry")`	tuple
`x = range(6)`	range
`x = {"name" : "John", "age" : 36}`	dict
`x = {"apple", "banana", "cherry"}`	set
`x = True`	bool

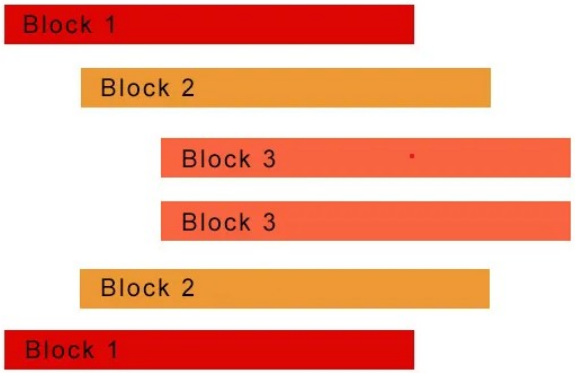
Hands on Exercise 1

- Write a Python script that calculates and prints the ***area*** and ***perimeter*** of a circle with an arbitrary given ***radius***?

Booleans for expressions and variables

```
print(10 > 9)  # True
print(10 == 9) # False
print(10 < 9)  # False


x = "Hello"
y = 15

print(bool(x))  # True
print(bool(y))  # True

print(bool("abc"))                          # True
print(bool(123))                            # True
print(bool(["apple", "cherry", "banana"]))  # True

print(bool(False))  # False
print(bool(None))   # False
print(bool(0))      # False
print(bool(""))     # False
print(bool(()))     # False
print(bool([]))     # False
print(bool({}))     # False
```

Indentation matters in Python



- Python uses indentation to indicate a block of code
- Indentation is mandatory in python to define the blocks of statements

```
def foo():
    print("Hi")

    if True:
        print("true")
    else:
        print("false")

print("Done")
```

```
def foo():
    print("Hi")

    if True:
        print("true")
    else:
        print("false")

print("Done")
```

1st level indentation
foo() method statements

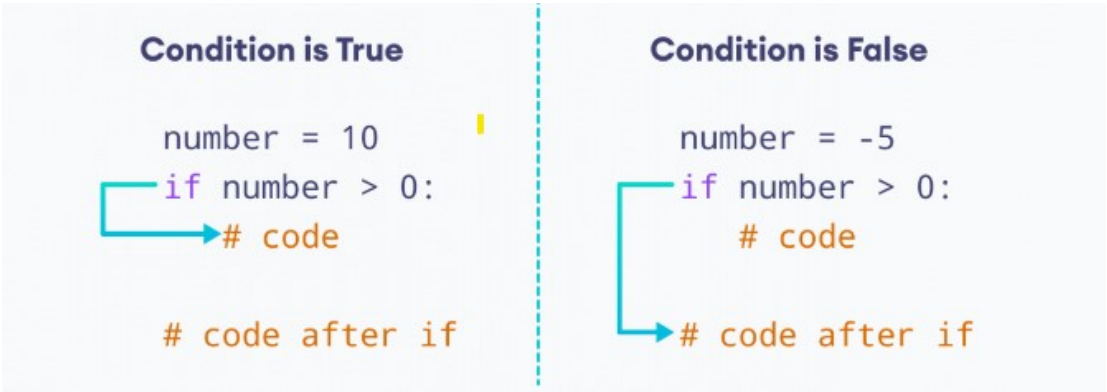
2nd level indentation
if and else block code

Code without indentation
Belongs to the source file

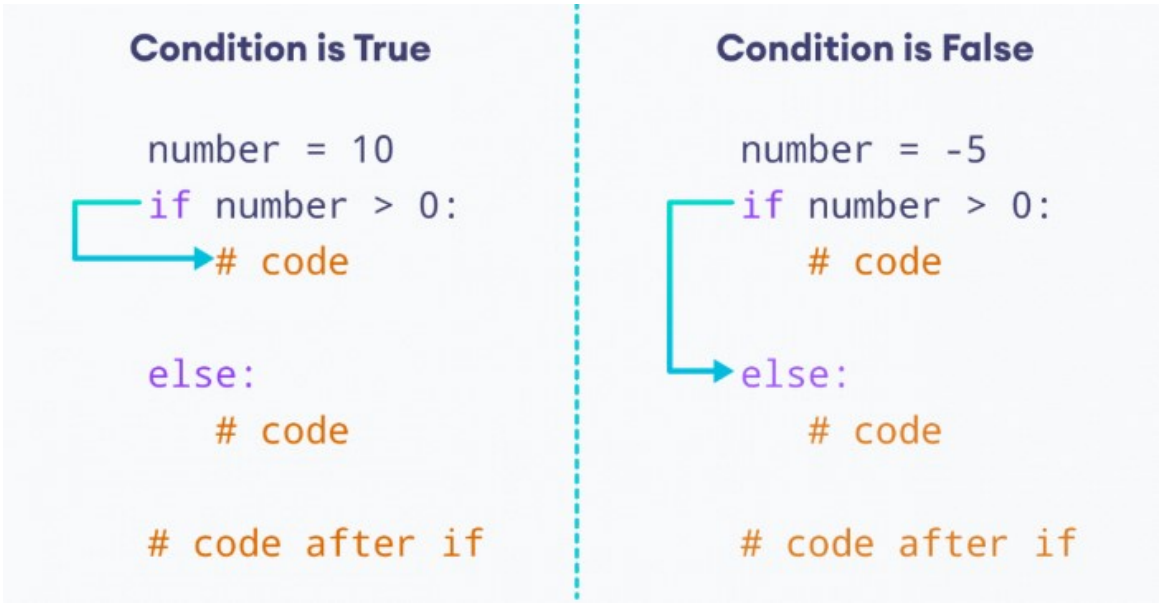
Conditions and If statements

Python Operator	*Mathematics Symbol*	*Name*	*Example* (radius is 5)	*Result*
`<`	<	less than	`radius < 0`	`False`
`<=`	≤	less than or equal to	`radius <= 0`	`False`
`>`	>	greater than	`radius > 0`	`True`
`>=`	≥	greater than or equal to	`radius >= 0`	`True`
`==`	=	equal to	`radius == 0`	`False`
`!=`	≠	not equal to	`radius != 0`	`True`

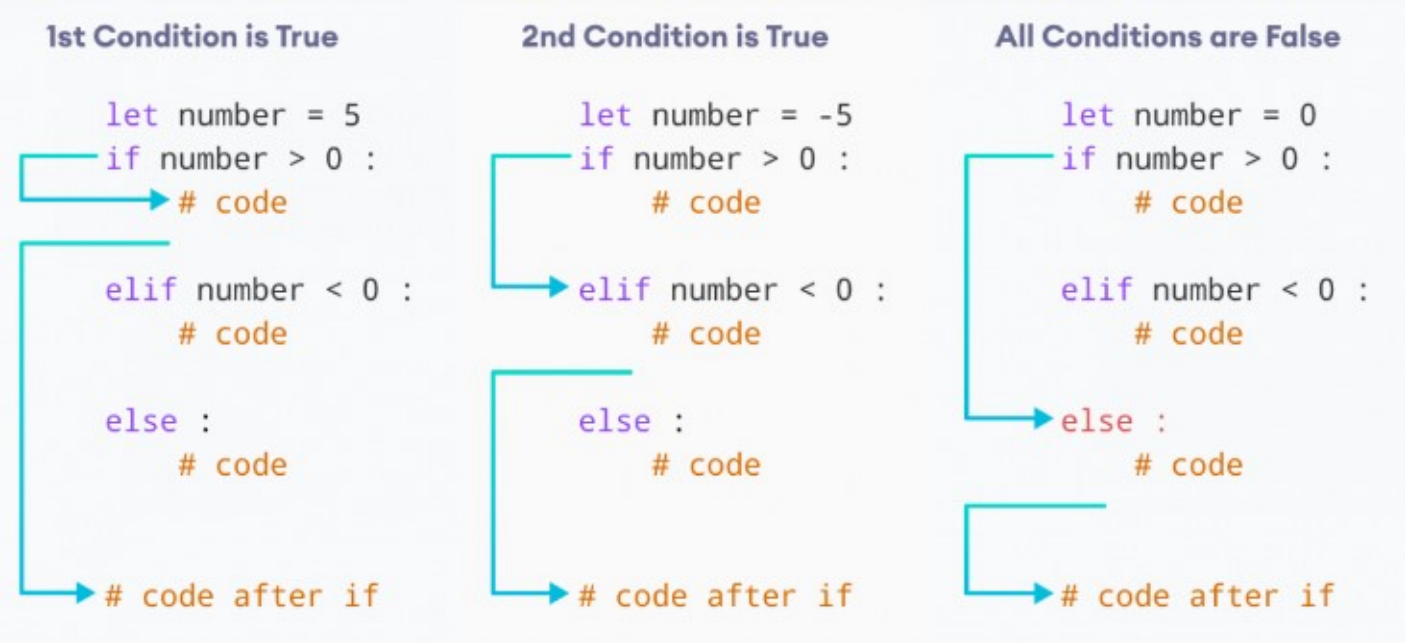
if statement



if else statement



if...elif...else


1st Condition is True
let number = 5
if number > 0 :
code
elif number < 0 :
code
else :
code
code after if
2nd Condition is True
let number = -5
if number > 0 :
code
elif number < 0 :
code
else :
code
code after if
All Conditions are False
let number = 0
if number > 0 :
code
elif number < 0 :
code
else :
code
code after if

Getting input from user

- The **input()** function takes input from the user.

```
inputValue = input([prompt])
```

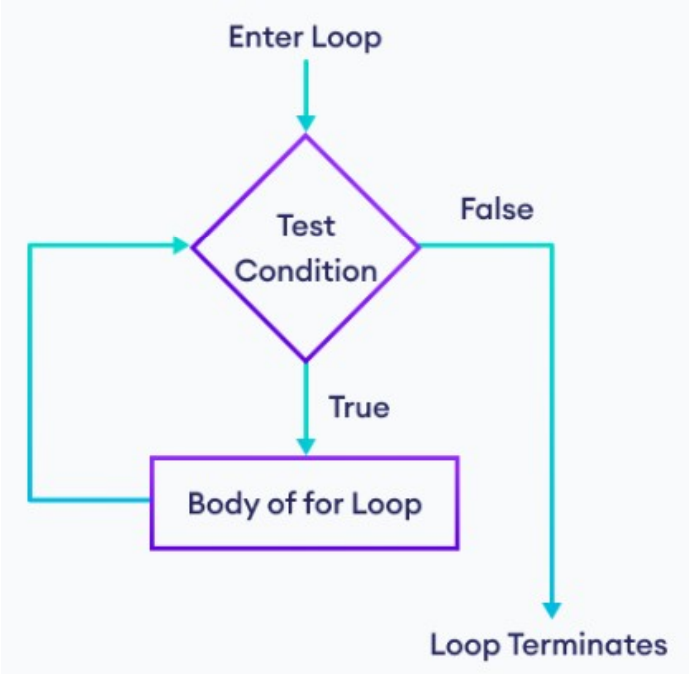
- ***prompt***: represents a message before the input.
- ***inputValue***: is the user input in a **String** type variable.

Hands on Exercise 2

- Refactor the Python script you did in exercise 1 :
  - Where the radius of the circle is given from the user input?

Loops

```
while condition:
    # body of while loop
```




```
for x in range(6):
  print(x)
```

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
  print(x)
```

Hands on Exercise 3

- Refactor the Python script you did in exercise 2 :
  - If the user input for the radius is negative value; then show an error message and ask the user to enter another valid radius (i.e. positive value)?

Functions

- A function is a block of code which only runs when it is called.

```
def function_name(arguments):
    # function body

    return
```

- To call a function, use the function name followed by parenthesis:

```
function_name("args")
```

Hands on Exercise 4

- Refactor the Python script you did in exercise 1 and 2 so it makes the calculations for ***area*** and ***perimeter*** in separate functions?

Python Collections (Arrays)

- **List:** is a collection which is
  - Ordered.
  - Changeable
  - Allows duplicate.
- **Tuple:** is a collection which is
  - Ordered.
  - UnChangeable
  - Allows duplicate.
- **Set**:
  - Unordered.
  - No duplicate.
- **Dictionary:**
  - Ordered.
  - No duplicate.

```
list1 = ["apple", "banana", "cherry"]
list2 = [1, 5, 7, 9, 3]
list3 = [True, False, False]
list4 = ["abc", 34, True, 40, "male"]
```

```
tuple1 = ("apple", "banana", "cherry")
tuple2 = (1, 5, 7, 9, 3)
tuple3 = (True, False, False)
tuple4 = ("abc", 34, True, 40, "male")
```

```
set1 = {"apple", "banana", "cherry"}
set2 = {1, 5, 7, 9, 3}
set3 = {True, False, False}  # {False, True}
set4 = {"abc", 34, True, 40, "male"}
```

```
thisdict = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
```

Built in Methods (Lists/Arrays)

Method	Description
append()	Adds an element at the end of the list
clear()	Removes all the elements from the list
copy()	Returns a copy of the list
count()	Returns the number of elements with the specified value
extend()	Add the elements of a list (or any iterable), to the end of the current list
index()	Returns the index of the first element with the specified value
insert()	Adds an element at the specified position
pop()	Removes the element at the specified position
remove()	Removes the first item with the specified value
reverse()	Reverses the order of the list
sort()	Sorts the list

Hands on Exercise 5

1. Write a Python script that calculates the **sum** and **average** of the given list of values [471, 500, 254, 845, 430]
2. Sort the values of the list

Workshop - First Steps in Python

- https://hifis.net/workshop-materials/python-first-steps/