

```
In [1]: 1 aa = 10
        2 bb = 5
```

```
In [2]: 1 cc = aa
```

```
In [5]: 1 dd = cc + bb
        2 dd
```

```
Out[5]: 15
```

```
In [7]: 1 aa = 0
```

```
In [8]: 1 aa + cc
```

```
Out[8]: 10
```

The different data_types

```
In [2]: 1 #integer
        2 type(1000) #the type function returns the type of an input
```

```
Out[2]: int
```

```
In [4]: 1 my_integer = 3.2
        2 type(my_integer)
```

```
Out[4]: float
```

```
In [5]: 1 #float
        2 my_float = 5.5
        3 type(my_float)
```

```
Out[5]: float
```

```
In [6]: 1 #boolean : bool (True, False)
        2 my_boolean = False
        3 type(my_boolean)
```

```
Out[6]: bool
```

```
In [9]: 1 my_boolean = True
        2 type(my_boolean)
```

```
Out[9]: bool
```

```
In [11]: 1 #strings: str
        2 my_string = hello
        3 type(my_string)
```

```
-----
NameError                                Traceback (most recent call last)
Cell In[11], line 2
      1 #strings: str
----> 2 my_string = hello
      3 type(my_string)
```

```
NameError: name 'hello' is not defined
```

```
In [13]: 1 name = "Segun"
        2 type(name)
```

```
Out[13]: str
```

```
In [14]: 1 type(I am Olanrewaju)
```

```
Cell In[14], line 1
    type(I am Olanrewaju)
      ^
SyntaxError: invalid syntax
```

```
In [15]: 1 type('I am Olanrewaju')
```

```
Out[15]: str
```

Questions

```
In [16]: 1 type('abc')
```

```
Out[16]: str
```

```
In [17]: 1 type(False)
```

```
Out[17]: bool
```

```
In [18]: 1 type(10.1)
```

```
Out[18]: float
```

```
In [ ]: 1 type(10)
```

```
In [19]: 1 a = 3
        2 b = 5.5
```

```
In [20]: 1 type(a)
```

```
Out[20]: int
```

```
In [21]: 1 type(b)
```

```
Out[21]: float
```

```
In [22]: 1 type(a + b)
```

```
Out[22]: float
```

```
In [26]: 1 a = 3.7
        2 b = 2.3
        3 print(a + b)
        4 type(a + b)
```

```
6.0
```

```
Out[26]: float
```

```
In [27]: 1 type(true)
```

```
-----
NameError                                Traceback (most recent call last)
Cell In[27], line 1
----> 1 type(true)

NameError: name 'true' is not defined
```

```
In [28]: 1 bool("False") #This will return True for as long as it is not empty
```

```
Out[28]: True
```

```
In [29]: 1 bool("True")
```

```
Out[29]: True
```

```
In [30]: 1 bool("") #This is an empty string - python will only returns False if and only if the string is empty.
```

```
Out[30]: False
```

```
In [31]: 1 bool(" ") #Please note that this is not empty because it contains a space and a space is a character.
```

```
Out[31]: True
```

```
In [32]: 1 bool(1)
```

```
Out[32]: True
```

```
In [33]: 1 bool(0)
```

```
Out[33]: False
```

Type conversion (casting)

```
In [34]: 1 my_num = 7
        2 print(my_num)
        3 print(type(my_num))
```

```
7
<class 'int'>
```

```
In [35]: 1 # integer to string
        2 str(my_num)
```

```
Out[35]: '7'
```

```
In [40]: 1 my_num + 10
```

```
Out[40]: 17
```

```
In [37]: 1 converted = str(my_num)
```

```
In [41]: 1 type(converted)
```

```
Out[41]: str
```

```
In [38]: 1 converted + 10
```

```
-----
TypeError                                 Traceback (most recent call last)
Cell In[38], line 1
----> 1 converted + 10

TypeError: can only concatenate str (not "int") to str
```

```
In [42]: 1 type(str(my_num))
```

```
Out[42]: str
```

```
In [43]: 1 my_num
```

```
Out[43]: 7
```

```
In [44]: 1 my_num_string = str(my_num)
        2 print(my_num_string)
        3 print(type(my_num_string))
```

```
7
<class 'str'>
```

```
In [ ]: 1 # string to integer
        2 my_num_int = int(my_num_string)
        3 print(my_num_int)
        4 print(type(my_num_int))
```

```
In [46]: 1 list_of_num = "12345"
        2 type(list_of_num)
```

```
Out[46]: str
```

```
In [48]: 1 mynewnum = int(list_of_num)
        2 mynewnum
        3 type(mynewnum)
```

```
Out[48]: int
```

```
In [49]: 1 alphabet = "ABCD"
        2 type(alphabet)
```

```
Out[49]: str
```

```
In [50]: 1 # Moving from Strings to Integer
        2
        3 int(alphabet)
```

```
-----
ValueError                                Traceback (most recent call last)
Cell In[50], line 1
----> 1 int(alphabet)

ValueError: invalid literal for int() with base 10: 'ABCD'
```

```
In [51]: 1 numm = "10.5"
        2 type(numm)
```

Out[51]: str

```
In [53]: 1 print(float(numm))

        10.5
```

```
In [ ]: 1 # integer to float
        2 my_num_float = float(my_num_int)
        3 print(my_num_float)
        4 print(type(my_num_float))
```

```
In [55]: 1 integra = 56
        2 float(integra)
```

Out[55]: 56.0

```
In [57]: 1 # Float to integer
        2
        3
        4 floaaat = 78.5
        5 int(floaaat)
```

Out[57]: 78

```
In [ ]: 1 # float to boolean
        2 my_num_bool = bool(my_num_float)
        3 print(my_num_bool)
        4 print(type(my_num_bool))
```

```
In [58]: 1 # float to boolean
        2
        3 bool(floaaat)
```

Out[58]: True

```
In [ ]: 1 #int = 7
```

```
In [ ]: 1 bool(0)
```

```
In [ ]: 1 bool(1)
```

```
In [ ]: 1 bool('')
```

```
In [ ]: 1 bool('hello')
```

```
In [ ]: 1 # impossible to convert a string with non numeric characters to an integer
        2 alphabet = 'abcde'
        3 #int(alphabet)
```

```
In [59]: 1 bool(5.5)
```

Out[59]: True

```
In [60]: 1 bool(0.1)
```

Out[60]: True

Questions

```
In [ ]: 1 new_variable = 'True'
        2 type(new_variable)
```

```
In [ ]: 1 bool(new_variable)
```

```
In [ ]: 1 new_variable = 'False'
        2 type(new_variable)
```

```
In [ ]: 1 bool(new_variable)
```

```
In [ ]: 1 my_letter = 'g'
        2 #int(my_letter)
```

```
In [ ]: 1 name = 48.9
        2
        3 print(name)
        4 print(type(name))
        5
        6
        7 name_a = str(name)
        8
        9 print(name_a)
        10
        11 print(type(name_a))
        12
```

```
In [ ]: 1 name_var = 4.5
        2 print(name_var)
        3 print(type(name_var))
```

```
In [ ]: 1
```

String operations

```
In [61]: 1 my_word = 'tables'  #[0,1,2,3,4,5] - position of letters in the word #[-6,-5,-4,-3,-2,-1]
```

```
In [62]: 1 my_word[0]
```

```
Out[62]: 't'
```

```
In [63]: 1 my_word[1]
```

```
Out[63]: 'a'
```

```
In [64]: 1 my_word[8]
```

```
-----
IndexError                                Traceback (most recent call last)
Cell In[64], line 1
----> 1 my_word[8]

IndexError: string index out of range
```

```
In [65]: 1 #We want to access the 3rd letter of our word
        2 my_word[2] #strings are indexed from zero
```

```
Out[65]: 'b'
```

```
In [66]: 1 my_word[5]
```

```
Out[66]: 's'
```

```
In [68]: 1 aword = "JEALOUSY"  #[0,1,2,3,4,5,6,7]
```

```
In [69]: 1 aword[6]
```

```
Out[69]: 'S'
```

```
In [ ]: 1
```

End of Class for 9th of June 2023

You expected to reproduce the entire operations we have seen in class on your on jupyter notebook and you are allowed to be creative in practicing with more examples.

Share the notebook as an attachment to: larrysman2004@gmail.com before 14th June 2023 by 11:59pm.

Links to the recordings – Live Session

https://drive.google.com/file/d/1_f7gNhYn24I3PGPj4og27aRP_Tye7RrI/view?usp=drive_link