24 oct

Data Types

- python can say data types of the values with out intializing
- which means the data types automatically consider
- python has some inbuilt datatypes structures
- int : integers
- float
- bool : boolean
- complex
- list
- tuple
- dict : dictionary
- set
- frozenset
- bytes
- bytesarray
- memoryview

integer

In [2]:
```
num = 100
type(num)
```

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

In [3]:
```
print(num)
```

```
100
```

binary number system

- digits : 0,1,2,3,4,5,6,7,8,9
- binary : bi means two : 0 and 1
- representation is : 0b001, 0b111, 0b010, 0B010, ...

- wrong represntation is : ob, 0b102

In [4]:
```
0b111
```

Out[4]: 7

In [6]:
```
0b10101
```

Out[6]: 21

In [7]:
```
0B1010
```

Out[7]: 10

In [8]:
```
0b1111
```

Out[8]: 15

In [ ]:
```
  2^3=8     2^2=4   2^1=2    2^0=1
    0         0       0        0          0
    0         0       0        1          1
    0         0       1        0          2
    0         0       1        1          3
    0         1       0        0          4
    0         1       0        1          5
    0         1       1        0          6
    0         1       1        1          7
    1         0       0        0          8
    1         0       0        1          9
    1         0       1        0          10
    1         0       1        1          11
    1         1       0        0          12
    1         1       0        1          13
    1         1       1        0          14
    1         1       1        1          15
```

In [9]:
```
0b1100, 0b1110
```

Out[9]: (12, 14)

octal number system

- digits : 0,1,2,3,4,5,6,7,8,9
- octal : octa means eight : 0,1,2,3,4,5,6,7
- representation is : 0o401, 0o141, 0o016, 0O017, ...
- wrong represntation is : 'o' as start,, 0o789

In [10]:
```
0o765
```

Out[10]: 501

In [11]:
```
0o123
```

Out[11]: 83

In [12]:
```
0o1276
```

Out[12]: 702

In [13]:
```
0O217
```

Out[13]: 143

hexa number system

- digits: 0,1,2,3,4,5,6,7,8,9
- hexa : hexa means 16: 0,1,2,3,4,5,6,7,8,9,A=10,B=11,C=12,D=13,E=14,F=15
- representation is : 0x576, 0xAF, 0X123

In [14]:
```
0x576
```

Out[14]: 1398

In [15]:
```
0xAF
```

Out[15]: 175

In [16]:
```
0x123
```

Out[16]: 291

float numbers

- float numbers means continues points
- float numbers means having some decimals
- integer numbers means discrete points

In [18]:
```
num = 10.23
type(num)
```

Out[18]: float

- float points can represent exponential format **e**

In [35]:
```
print(1e1), # 1*10
print(1e2), # 1*100
print(1e3)  # 1*1000
```

```
10.0
100.0
1000.0
```

In [25]:
```
2e2
```

Out[25]: 200.0

In [29]:
```
print(1e+1) # 1*10
print(1e+2) # 1*100
print(1e+3) #1*1000
```

```
10.0
100.0
1000.0
```

In [34]:
```
print(1e-1)  # 1/10
print(1e-2)  # 1/100
print(1e-3)  # 1/1000
```

```
0.1
0.01
0.001
```

- if + in given example then multiply
- if - in given example then divided

In [44]:
```
1e-5
1/100000
```

Out[44]: 1e-05

String

- String means english characters

In [45]:
```
name = "Python"
type(name)
```

Out[45]: str

In [47]:
```
num = '10'
type(num)
```

Out[47]: str

In [48]:
```
sen1 = "Hello I like 'Python'"
sen1
```

Out[48]: "Hello I like 'Python'"

In [49]:
```
sen2 = 'Hello I like "python"'
```

In [50]:
```
sen2
```

Out[50]: 'Hello I like "python"'

- Entire sentence in single quotes, then use double quotes to highlight the word

- Entire sentence in double qoutes, then use single quotes to highlight the word

In [53]:
```
name = 'Pyhton'
name
```

Out[53]: 'Pyhton'

In [54]:
```
print(name)
```

```
Pyhton
```

- when we want to see two answer in same cell then we can use print to see them

In [ ]:
```
hello good moring
how are you with out /n
```

In [57]:
```
statement = """Hello I like 'Python'
               hii good morning
               class data types"""
print(statement)
```

```
Hello I like 'Python'
               hii good morning
               class data types
```

Doc string

- If we want to convey information to the user
- we willl write a statement before coding part
- is called as Doc String
- In Jupyter notebook to convey the information we have **markdown**
- But in Vscode we dont have a Markdown option
- We convey the information by using **triple quotes**
- If we see anywhere triple quotes means the user trying to convey the information

How we use doc-string

In [60]:
```
"""
    Write something
    """

'''
    write someting
    '''
```

Out[60]: '\n   write someting\n   '

Boolean

In [61]:
```
value = True

type(value)
```

Out[61]:
```
bool
```

In [62]:
```
value = False
type(value)
```

Out[62]:
```
bool
```