

#DAY

1

DAY 1 OF 200 DAY'S PYTHON CHALLENGE



BASIC MATH OPERATIONS



Day1.py

-
- $10 + 20$
 - $30 - 20$
 - $5 * 5$
 - $56 / 8$
 - $4 ** 3$
 - $56 \% 7$



VARIABLE DECLARATIONS

- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive (age, Age and AGE are three different variables)
- A variable name cannot be any of the Python keywords.

EXAMPLE



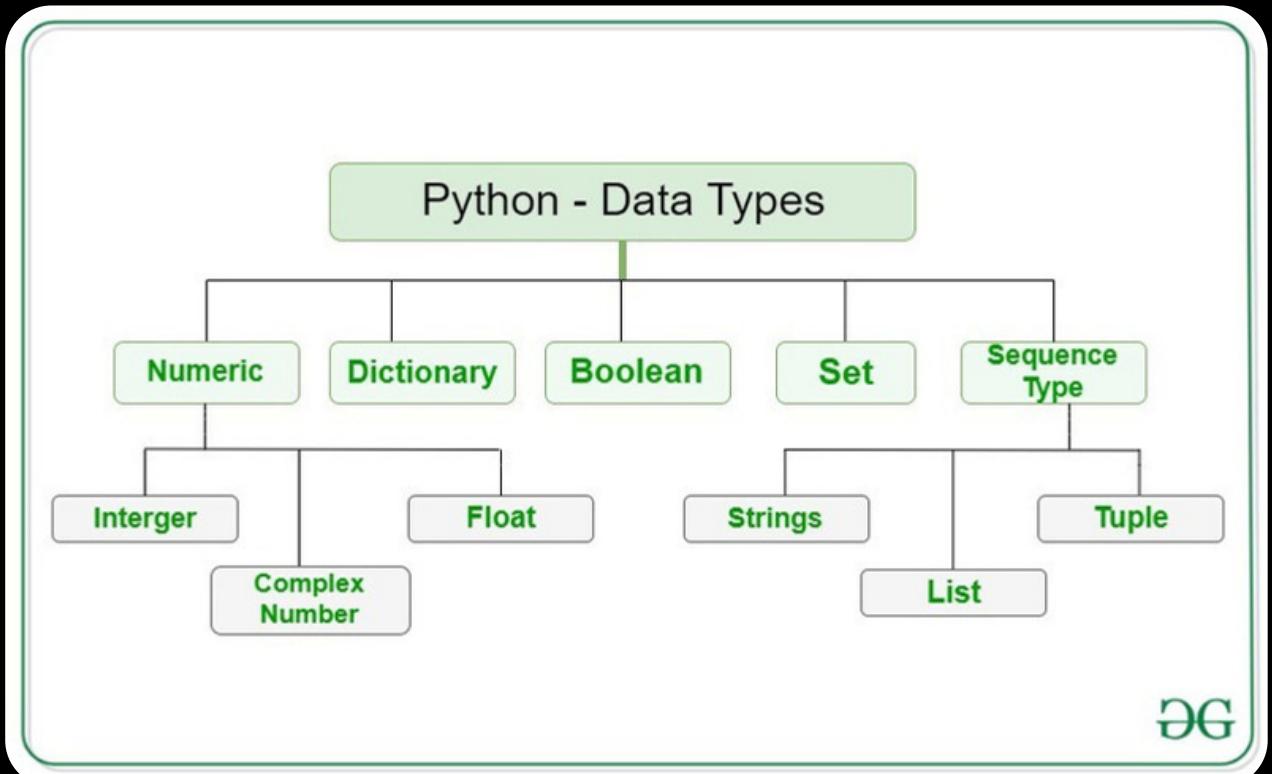
Day1.py

```
# Can
• Var = 10
• _var = 20
• var1 = 30

# Cannot
• 4var = 30
• va r = 34
• v@r = 46
```

DATA TYPES

- Integer
- Float
- Boolean
- String
- Complex Number
- List
- Tuple
- Set
- Dictionary



EXAMPLE



Day1.py

```
a = 20
type(a)
b = 90.38
type(b)
c = "Tech1.Abhi"
type(c)
d = "Day 1 of 200 day's Python
challenge"
type(d)
e = True or False
type(e)
f = 5 + 7i
type(f)
```

int

float

str

bool

complex

COMPLEX DATA TYPE

It is a combination of real number



Day1.py

```
>>> c = 5 + 4i
>>> c.real
5.0
>>> c.imag
4.0
```

BOOLEAN DATA TYPE

In Python programming value of True = 1
and False = 0.



Day1.py

```
>>> False + False
0
>>> True + False # And wiseversa
1
>>> True + True
2
>>> False - True
-1
>>> True/False # Here like 1/0 so..
Error: Division by zero
>>> False/True # It is like 0/1
0
```

PRINT FUNCTION

It is use to print any variable or print all things those in quets it may be Single quet or Double quet.



Day1.py

```
>>> print("My name is 'Abhi Kumar' or Your 'Ratan'")  
My name is 'Abhi Kumar' or Your 'Ratan'  
>>> a = 10  
>>> print(a)  
10
```

STRING

**In string each character store in specific block that is called index and it is start from 0(By default).
And it can be Positive or Negative**



Day1.py

```
>>> s = "Tech1.Abhi"  
>>> s[0]  
'T'  
>>> s[90000]  
Error: Out of range error  
>>> s[-1]  
'h'
```

STRING SLICING

**It return a substring of the string where we define a
2 way of executing slicing in python. Using start and
stop**



Day1.py

```
>>> S = "Tech1.Abhi"  
>>> S[0:10] # Here 0 starting index and 10 stop + 1  
'Tech1.Abhi'  
>>> S[:]  
'Tech1.Abhi'  
>>> S[:5]  
'Tech1'  
>>> S[6:10]  
'Abhi'
```

STRING SLICING

It return a substring of the string where we define a 2 way of executing slicing in python. Using start, stop and steps



Day1.py

```
>>> S = "Tech1.Abhi"  
>>> S[0:10:1] # Here 0 starting index and 10 stop + 1 and then 1 is step  
'Tech1.Abhi'  
>>> S[::-1]  
'ihbA.1hceT'  
>>> S[:10:2]  
'Tech1'  
>>> S[:-9000:1]  
''
```

LEN FUNCTION

It is used to find a length of any string.



Day1.py

```
>>> S = "Tech1.Abhi"  
>>> len(S)  
10  
>>> w = "Day 1 of 200 day's Python challenge"  
>>> len(w)  
35
```

FIND FUNCTION

It is used to find a any character or substring index



Day1.py

```
>>> S = "Tech1.Abhi"  
>>> S.find('s')  
-1  
>>> w = "Day 1 of 200 day's Python challenge"  
>>> w.find('Python')  
19  
>>> w.find('a')  
1
```

COUNT FUNCTION

It is used to find a any character or substring how many times occurs



Day1.py

```
>>> w = "Day 1 of 200 day's Python challenge"
>>> w.count('Python')
1
>>> w.count('a')
3
>>> w.count('Z')
0
```

UPPER FUNCTION

This function is used to change all the character in upper case



Day1.py

```
>>> w = "Day 1 of 200 day's Python challenge"
>>> w.upper()
"DAY 1 OF 200 DAY'S PYTHON CHALLENGE"
>>> s = 'this is my first python class'
>>> s.upper()
'THIS IS MY FIRST PYTHON CLASS'
```

LOWER FUNCTION

This function is used to change all the character in lower case



Day1.py

```
>>> w = "Day 1 of 200 Day's Python Challenge"  
>>> w.upper()  
"Day 1 of 200 day's python challenge"  
>>> s = 'THIS IS MY FIRST PYTHON CLASS'  
>>> s.upper()  
'this is my first python class'
```

TITLE FUNCTION

This is used to change the initial character in each word to Uppercase and the subsequent characters to Lowercase and then returns a new string.



Day1.py

```
>>> w = "Day 1 of 200 day's Python challenge"
>>> w.upper()
"Day 1 Of 200 Day'S Python Challenge"
>>> s = 'THIS IS MY FIRST PYTHON CLASS'
>>> s.upper()
'This Is My First Python Class'
```

CAPITALIZE FUNCTION

**changes a string's first character to uppercase while
converting the rest to lowercase**



Day1.py

```
>>> w = "Day 1 of 200 day's Python challenge"
>>> w.capitalize()
"Day 1 of 200 day's python challenge"
>>> s = 'this is my first python class'
>>> s.capitalize()
'This is my first python class'
```

ARITHMATIC OPERATION WITH STRING

‘-’(minus) and ‘/’ (Division) Operator not work in string any how. It will Always gives a Error.



Day1.py

```
>>> s = "Tech1.Abhi"  
>>> s + " Kumar"  
Tech1.Abhi Kumar  
>>> s + 1  
Error: Type Error  
>>> s * 3  
'Tech1.Abhi Tech1.Abhi Tech1.Abhi'  
>>> s * "kumar"  
Error : Type Error
```

DIFFERENCE BETWEEN SINGLE QOUTES AND DOUBLE QOUTES

In python no major difference between single qoutes and double quotes but when we use a both in single String then we always remember that which qoute where we use If we use outer side Double qoute then we need to use single qoutes in inner side.

Or

It is usable for regular Expression in python



Day1.py

```
>>> s = "don't do copy and paste in my class"  
>>> s  
"don't do copy and paste in my class"
```

COMMENT

Single Line Comment :

is use to make any code line to comment line but it is a work only for single line.

```
# Comment line
```

Multi-Line Comment :

For multiline code to comment we use a three time's qoutes it can be single qoute or double qoute.

```
''' Multiline  
comment '''
```

FOLLOW FOR MORE



FOLLOW ME

