Python is the General purpose Language bcz we can use it at every where like machine learning, GUI, software development, web development.

python is a interpreted object-oriented high-level language.

Python Interpreter

IDE → Integrated Development Environment

Integer divison :- 5 // 2 ⇒ 2

Float :- 5 / 2 ⇒ 2.5

power off :- 2 ** 3 ⇒ 8

String :- Combination of characters → 'PYTHON'

Function :- set of tasks we are performing

print :- Inbuild function

E.g. print ('Python') → Brackets used for pass some parameteres

print ('navin's laptop') → Invalid syntax

Instead of single quote use

print (" navin's laptop")

print ('navin "laptop"') → navin "laptop"

print ('navin's "laptop"') → Error

To tell python that skip the meaning of that special ' write

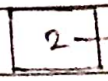
print ('navin\'s "aptop".) → navin's "laptop"

10 * 'Me' → prin the 'Me' word 10 times

Raw string :- print ('c:\docs\navin') → c:\docs avin

print (r 'c:\docs\navin') → c:\docs\navin

Variables :- Container where you can put your values



x ⟶ Variable Name

If we want to use output of previous operations then use '_' there

e.g. x+10 → 19

_ +4 → 22

Fetch a one character from string

e.g. name = 'youtube'

name[0] → 'y'

name[6] → 'e'

name[-1] → 'e'

name[0:2] → 'yo'

name[1:4] → 'out'

name[1:] → 'outube'

name[:4] → 'yout'

-7	-6	-5	-4	-3	-2	-1
Y	O	U	T	U	B	E
0	1	2	3	4	5	6

** Strings in python are immutable

↓

You cannot change the value once assigned

myname = 'navin reddy'

len (myname) → 11

* Lists :- Lists are mutable

Lists can contain a different types of values like values = [9.5, 'navin', 25]

num.append (45) → Add this no. at the end of the list

num.insert (2, 77) → Add this no. at the position of inde

no. 2

Add multiple values in the list -

```
nums.extend([29, 12, 14, 36])
nums
```

* Tuple → Immutable

E.g. tup = (21, 36, 14, 25)

* set → Collection of unique elements, not support index & duplicate values

e.g. s = {22, 25, 14, 21, 5}

* Dictionary → key - value

e.g.

```
data = {1: 'navin', 2: 'kiran', 4: 'harsh'}
data[4] → harsh
```

Don't have a key then print 'not found'

```
data.get(3, 'Not Found') → Not Found
```

With the help of 2 lists make a dictionary

```
>>> keys = ['navin', 'kiran', 'Harsh']
>>> values = ['python', 'Java', 'JS']
>>> data = dict(zip(keys, values))
>>> data → o/p → {'navin': 'python', 'kiran': 'java', 'Harsh': 'JS'}
```

* find the address of variable

E.g. num = 5

id(num) → 1554841536

* Data Types:-

None → variable not assigned with any value

Numeric → int, float, complex, bool

List

Tuple

Set

String

Range → for iterate betⁿ values

Dictionary → key have to be unique

complex → No. + imaginary no.

* operators:-

Arithmetic

Assignment

Relational

Logical → AND, OR, NOT

Unary

* Number system conversion in python

Decimal	→	Binary	BIT → Binary Digit
base 10 (0-9)		base 2 (0-1)	

Octal	Hexa Decimal
base 8 (0-7)	base 16 (0-9 a-F)