```
a=10
print(a)

     10
```

```
b=20
print(b)

     20
```

hi everyone!

**Python Feature:**

1. Simple and easy to learn.
2. Free and open source.
3. Portable: Mac/windows/linux
4. Rapid Deveplopment : Interpreted High level Language.
5. High Level Langauge: Automatic garbage collector & memory managemnt system.
6. Extensible: We can use C++ & JAVA libraries.

**Python Applications/Uses:**

1. Desktop GUI : Tikenter
2. Web Application development: Flask , Django, Streamlit
3. Mobile Development : Beewire, Kiwi
4. Gaming Application & 3D Graphics : Pygame, Pykra
5. Testing : PySelenium
6. Data Analysis : Pandas, Matplotlib, Seaborn, Numpy
7. Machine Learning: SckitLearn

**Components Of Python**

**Literals:** Data or any Value

10, 100.2, 'Hello', " Welcome to this course", binary no., decimal no., hexadecimal, Octa decimal

Boolean Literals: True/False

Special Literal: None

```
s1 = "Welcome to this course"
print(s1)

     Welcome to this course
```

```
type(s1)

     str
```

```
id(s1)

     134634595677280
```

```
a=None
print(a)

     None
```

```
type(a)

     NoneType
```

```
id(a)
```

```
     95927217095648
```

```
b = 0b1100
print(b)
type(b)

     12
     int
```

```
c = 0o10
print(c)
type(c)

     8
     int
```

```
h = 0xa
print(h)

     10
```

```
hex(c)

     '0x8'
```

```
f = 10.7
print(type(f))
print(f)

     <class 'float'>
     10.7
```

```
int(f)

     10
```

```
my_val = True
print(my_val)
#case sensitive

     True
```

**String Literals:**

```
a =''''Python'''''
print(a)


     'Python
```

```
type(a)

     str
```

```
#for multi-line quotation we have provide 3 consecutive single/double quotation.

my_var = """welcome to
this course"""

print(my_var)

     welcome to
     this course
```

**COMMENTS**

# #single-line comment

'''Multi-line comments'''

```
'''addition
of 2 variables'''
A= 10  #assigning B with 20 value
B= 20  #assigning B with 20 value
c = A+B
print(c)
```

```
    30
```

**Identifiers:** Names of Variables of any class, function(def) or any object.

eg. num =1000 #num is my identifier.

1. A-Z, a-z
2. Cannot begin with no.s
3. we can begin '_'
4. case sensitive variables i.e. A & a ... both are different.
5. We cannot begin with keywords/reserved words eg. int, float, break etc.
6. No Special character except '_'
7. we cannot provide whitespaces in our identifier.

```
a_b = 10
print(a_b)
```

```
    10
```

### Keywords/Reserved Words

```
import keyword
print(keyword.kwlist)
```

```
    ['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except',
```

```
len(keyword.kwlist)
```

```
    35
```

### TypeCasting

```
f=7.88
print(f)
```

```
    7.88
```

```
10
```

```
    10
```

```
a=10
```

```
int(f)
```

```
    7
```

```
#round-off function
round(f)
```

```
    8
```

```
f= 7.4
print(round(f))
```

```
    7
```

```
f= 7.5
print(round(f))
```

```
     8
```

```
a=11/3
print(a) #Division & getting qouteint
```

```
     3.6666666666666665
```

```
11//3 #Floor Division
```

```
     3
```

```
11+3
```

```
     14
```

### Complex No.

```
c = 10 + 3j
print(c)
type(c)
```

```
     (10+3j)
     complex
```

```
print(c.real)
print(c.imag)
```

```
     10.0
     3.0
```

```
math.sqrt(16)
```

```
     4.0
```

### Inputs from Users:

```
a = int(input("enter your no.:"))
b = int(input("enter your 2nd no.:"))
c = a+b
print(c)
```

```
     enter your no.:45
     enter your 2nd no.:89
     134
```

```
my_var = input("enter your name:")
my_roll = int(input("enter your roll no.:"))
print(my_var)
print(my_roll)
```

```
     enter your name:nandita
     enter your roll no.:56
     nandita
     56
```

```
a= 1999
b= 20.67
c = a+b
print(c)
print(a,b)
```

```
2019.67
1999 20.67
```