print is used for answer

In [1]:
```
a = 10
b = 20
a
```

Out[1]: 10

In [3]:
```
b
```

Out[3]: 20

In [5]:
```
print (a + b)
```

```
30
```

In [6]:
```
print (a)
print (b)
```

```
10
20
```

In [7]:
```
print (10)
print (10, 20)
print (10, 20, 'python')
```

```
10
10 20
10 20 python
```

In [8]:
```
num1=20
num2=30
add=num1+num2
print (add)
```

```
50
```

Print result with string

In [12]:
```
num1 = 20
num2 = 30
add=num1+num2
print ('The addition of', num1, 'and', num2, 'is=', add)
```

```
The addition of 20 and 30 is= 50
```

In [13]:
```
name='Python'
age=20
city='New York'
print ('Hello my name is', name, 'I am', age, 'years old and I am from', city)
```

```
Hello my name is Python I am 20 years old and I am from New York
```

In [14]:
```
num1=20
num2=30
add=num1+num2
print ('The addition of {} and {} is = {}'.format (num1, num2, add))
```

```
The addition of 20 and 30 is = 50
```

In [16]:
```
name='Python'
age=20
city='New York'
print ('Hello, my name is, {}, I am {} years old and I am from {}' .format (name, a
```

```
Hello, my name is, Python, I am 20 years old and I am from New York
```

In [ ]:
```
```

In [17]:
```
num1=100
num2=25
num3=333
avg=(num1+num2+num3)/3
```

In [18]:
```
print (avg)
```

```
152.66666666666666
```

In [22]:
```
print (round(avg, 0))
```

```
153.0
```

In [23]:
```
print (round(avg, 2))
```

```
152.67
```

In [24]:
```
print ('The average of {}, {} and {} is={}'.format (num1, num2, num3, avg))
```

```
The average of 100, 25 and 333 is=152.66666666666666
```

In [25]:
```
avg1=round ((num1+num2+num3)/3,2)
```

In [26]:
```
print ('The average of {}, {} and {} is={} or {}'.format (num1, num2, num3, avg, av
```

```
The average of 100, 25 and 333 is=152.66666666666666 or 152.67
```

More short format meythod(f string method)

In [29]:
```
num1 = 20
num2 = 30
add = num1+num2
print(f'The addition of {num1} and {num2} is = {add}')
```

```
The addition of 20 and 30 is = 50
```

In [33]:
```
name='Python'
age = 20
```

```
city = 'New York'
print (f'Hello, my name is {name}, I am {age} years old and I live in {city} city.'
```

```
Hello, my name is Python, I am 20 years old and I live in New York city.
```

In [34]:
```
num1=100
num2=230
num3=444
avg=round((num1+num2+num3)/2,2)
print (f'The average of {num1}, {num2} and {num3} is ={avg}')
```

```
The average of 100, 230 and 444 is =387.0
```

In [41]:
```
num1 = 10
num2 = 20
add = num1+num2
print ('The addition of', num1, 'and', num2, 'is=',add)
print ('The additioin of {} and {} is={}'.format (num1, num2,add))
print (f'The addition of {num1} and {num2} is={add}')
```

```
The addition of 10 and 20 is= 30
The additioin of 10 and 20 is=30
The addition of 10 and 20 is=30
```

end statement

Here we will use end statement that joint line from end of one string to starting of other string

In [43]:
```
print ('hello')
print ('good morning')
print ('hello ', end = '')
print ('world good day')
```

```
hello
good morning
hello world good day
```

seprator

In [44]:
```
print ('hello', 'hi', 'how are you?', sep='--->')
```

```
hello--->hi--->how are you?
```

In [45]:
```
print ('hello', 'hi', 'how are you?', sep='@')
```

```
hello@hi@how are you?
```

In [46]:
```
print (3, '.')
```

```
3 .
```

In [47]:
```
print (3, '.', sep='')
```

```
3.
```

In [51]:
```
num1 = 10
num2 = 20
add = num1+num2
print (f'The addition of {num1} and {num2} is={add}', sep='')
```

```
The addition of 10 and 20 is=30
```

In [49]:
```
print (f'The addition of {num1} and {num2} is={add}')
```

```
The addition of 10 and 20 is=30
```

In [50]:
```
print ('The additioin of {} and {} is={}'.format (num1, num2,add))
```

```
The additioin of 10 and 20 is=30
```

In [54]:
```
num1 = 10
num2 = 20
add = num1+num2
print ('The addition of', num1, 'and', num2, 'is=',add, sep=' ')
print ('The additioin of {} and {} is={}'.format (num1, num2,add, sep=''))
print (f'The addition of {num1} and {num2} is={add}', sep='')
```

```
The addition of 10 and 20 is= 30
The additioin of 10 and 20 is=30
The addition of 10 and 20 is=30
```

COMPLEX DATA TYPE

In Python, the complex type is used to represent complex numbers, which consist of a real and an imaginary part. # You can create complex numbers and perform mathematical operations on them using the built-in support for complex numbers.

In [55]:
```
z = 3 + 4j
print (z.real)
print (z.imag)
```

```
3.0
4.0
```

In [56]:
```
a = 3 + 4j
b = 1 + 2j
print (a + b)
```

```
(4+6j)
```

In [57]:
```
print (a - b)
```

```
(2+2j)
```

In [58]:
```
print (a * b)
```

```
(-5+10j)
```

In [59]:
```
print (a/b)
```

```
(2.2-0.4j)
```

In [60]:
```
z = 3+4j
print (abs(z))
```

```
5.0
```

In [61]:
```
print (z.conjugate())
```

```
(3-4j)
```

End of today's practice :)

In [ ]:
```

```

In [ ]:
```

```