Input()

In [1]:
```
x=input()
y=input()
z=x+y
print(z)
```

```
53
```

In [3]:
```
x1=input("enter the first number")
y1=input("enter the second number")
z1=x1+y1
print(z1)
```

```
65
```

In [8]:
```
print('x1 is',type(x1))
print('y1 is',type(y1))
```

```
x1 is <class 'str'>
y1 is <class 'str'>
```

In [9]:
```
x1 = input('Enter the 1st number')
a1 = int(x1)
y1 = input('Enter the 2nd number')
b1 = int(y1)
z1 = a1 + b1
print(z1)
```

```
12
```

In [10]:
```
x2=int(input("enter the first number"))
y2=int(input("enter the second number"))
z2=x2+y2
print(z2)
```

```
9
```

In [13]:
```
x2 = input('user name :')
y2 = input('password :')
z2 = x2 + y2
print(z2)
```

```
joseph@gmail.com
```

In [12]:
```
st=input('enter string')
print(st)
```

```
hello
```

In [15]:
```
print(st[0])
```

```
h
```

In [16]:
```
print(st[0:2])
```

```
he
```

In [17]:
```
st=input('enter string')[1]
print(st)
```

```
e
```

In [18]:
```
st=input('enter string')[-1]
print(st)
```

```
o
```

In [20]:
```
st=input('enter string')[5:8]
print(st)
```

```
nin
```

In [21]:
```
result=input("enter expression")
print(result)
```

```
5+8-3
```

Eval

In [22]:
```
result=eval(input("enter expression"))
print(result)
```

```
10
```

In [23]:
```
result=eval(input("enter expression"))
print(result)
```

```
113
```

In [24]:
```
result=eval(input("enter expression"))
print(result)
```

```
6666
```

NUMPY

In [25]:
```
import numpy as np
```

In [26]:
```
np.__version__
```

Out[26]:
```
'2.1.3'
```

In [28]:
```
np.arange(10)
```

Out[28]:
```
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

In [29]:
```
np.arange(2,12,2)
```

Out[29]:
```
array([ 2,  4,  6,  8, 10])
```

arrays creation in numpy

In [31]:
```
np.array([1,2,3])
```

Out[31]:
```
array([1, 2, 3])
```

In [32]:
```
np.array([
    [1,2,3],
    [3,4,5]
])
```

Out[32]:
```
array([[1, 2, 3],
       [3, 4, 5]])
```

In [35]:
```
np.zeros((2,3))
```

Out[35]:
```
array([[0., 0., 0.],
       [0., 0., 0.]])
```

In [36]:
```
np.ones((2,3))
```

Out[36]:
```
array([[1., 1., 1.],
       [1., 1., 1.]])
```

In [40]:
```
np.ones((3,3))
```

Out[40]:
```
array([[1., 1., 1.],
       [1., 1., 1.],
       [1., 1., 1.]])
```

In [ ]: