Basics of python:

```
In [5]:    1  c=0
           2  c=5/2
           3  type(c)
```

Out[5]: float

```
In [11]:   1  name,rollno,age= 'Python', 234, 23.5
           2  print(type(name),type(rollno),type(age))
           3
```

<class 'str'> <class 'int'> <class 'float'>

```
In [15]:   1  a,b=10,20
           2  print(a,b)
           3  a,b= b,a
           4  print(a,b)
```

10 20
20 10

```
In [16]:   1  2 ** 5
           2
```

Out[16]: 32

```
In [18]:   1  pow(2,5)    #pow(basevalue,power)
```

Out[18]: 32

```
In [ ]:    1  #Memership operators ---->  in, not in.
           2
           3  #Identity operators -----> is, is not. (this cannot be applied to strings)
```

```
In [1]:    1  5 in [1,2,3,4] # Membership operator
```

Out[1]: False

```
In [2]:    1  5 is not 10   # Identity operator
```

Out[2]: True

Conditional Control Statements:

- if 5<6:
    - print("True")
- else:
    - print("False")

```
In [5]:   1  # For comparing more than one condition we use elif :
          2
          3  if 7<6:
          4      print("True")
          5  elif 6<7:
          6      print("True")
          7  else:
          8      print("False")
```

True

Iterative Statements:

- for
- while

```
In [6]:   1  range(10)
```

Out[6]: range(0, 10)

```
In [13]:   1  list(range(2,10,2))   # (starting value, ending value, no of increments)
```

Out[13]: [2, 4, 6, 8]

```
In [16]:   1  for i in range(2,10+1,2):
           2      print(i,end=' ')
```

2 4 6 8 10

```
In [3]:   1  n=1
          2  while n<10:
          3      print(n, end= " ")
          4      n+=1
```

1 2 3 4 5 6 7 8 9

```
In [4]:   1  for i in range(1,6):
          2      print(i, end= " ")
```

1 2 3 4 5

```
In [5]:   1  #without using range method
          2
          3  for i in 1,2,3:
          4      print(i, end=" ")
```

1 2 3

```
In [8]:   1  for i in range(10,4,-1):
          2      print(i, end =" ")
```

10 9 8 7 6 5

In [12]:
```python
for i in "hello":
    print(i, end=" ")
```

h e l l o

In [14]:
```python
print(1,2)
```

1 2

In [21]:
```python
print(1,2,sep=",")    # to get the comma in between values we use sep paramet
```

1,2

In [22]:
```python
print(1,2,sep=",",end=" *")
print(3,sep=";")
```

1,2 *3

In [26]:
```python
n = input("Enter value: ")
print(n,type(n))
```

Enter value: 30
30 <class 'str'>

In [28]:
```python
1+int("2") # for adding two different types we use type conversion
```

Out[28]: 3

In [29]:
```python
num=10
print(num)
```

10

In [33]:
```python
ls=[1,10,20]
ls[2]
```

Out[33]: 20

In [47]:
```python
n=input().split()

count=len(n)
print(count)
```

1 3 4 5
4

In [2]:
```
1  # Another way
2
3  data=input()
4  count=0
5  for c in data.split(" "):
6      count+=1
7  print(count)
```

```
1 2 3 4 5 6
6
```

In [4]:
```
1  for i in 1,2,3:
2      print(i, end=" ")
3  print(i)
```

```
1 2 3 3
```

In [14]:
```
1  # 1,2,3,4,5=15(sum)
2  n=input()
3  m=n.split(",")
4  s=0
5  for i in m:
6      s=s+int(i)
7      count+=1
8  print(s)
```

```
1,2,3,4,5
15
```

In [22]:
```
1  #Another way using map(this is used to convert all the list values into inte
2
3  data=input()
4  values=data.split(",")
5
6  #Syntax of map is:  map(method_name,list_of_values)
7  int_values=list(map(int,values))
8  sum(int_values)
```

```
1,2,3
```

Out[22]: 6

In [21]:
```
1  del sum     #to delete the variable sum
```

In [16]:
```
1  # without using map
2
3  l=['1','2','3']
4  r=[]
5  for value in l:
6      r.append(int(value))
7  r
```

Out[16]: [1, 2, 3]

In [33]:
```python
1  #1+2+5+63+3
2
3  #1
4
5  #63
6
7
8  n=input()
9  values=n.split("+")
10 m=list(map(int,values))
11 for i in m:
12     if i>m:
13         print(max(i))
14     elif i<m:
15         print(min(i))
16 return
17
```

1+2+65

```
-------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-33-21fd0d325667> in <module>
     10 m=list(map(int,values))
     11 for i in m:
---> 12     if i>int(n):
     13         print(max(i))
     14     elif i<int(n):

ValueError: invalid literal for int() with base 10: '1+2+65'
```

In [ ]:
```python
1
```