

```
In [1]: a="divija ppp"
print(a.split())
a="divija1999"
print(a.split())

['divija', 'ppp']
['divija1999']
```

(5/9/19)

list

tuple

dictionary

```
In [2]: ##list is represented by square braces
##list is mutable
##forward indexing and backward indexin
v=[1,3.3,"satheesh",[1,2,3]]
v[0]
```

Out[2]: 1

```
In [3]: l=[1,2,3,4,5]
s=0
for i in l:
    s=s+1
print(s)
print(sum(l))
```

5
15

```
In [14]: a=-2.8
abs(a)
round(a)
b=[1,2,55,0,3]
print(len(b))
print(sum(b))
print(chr(98))
print(ord('B'))
print(bin(5))
print(hex(15))
print(sorted(b))
print(sorted(b,reverse=True))
print(sorted(b,reverse=False))
print(max(b))
min(b)
#range
#print
#input
print(b[1::2])
```

```
5
61
b
66
0b101
0xf
[0, 1, 2, 3, 55]
[55, 3, 2, 1, 0]
[0, 1, 2, 3, 55]
55
[2, 0]
```

```
In [15]: dir(list)
```

```
Out[15]: ['__add__',
          '__class__',
          '__contains__',
          '__delattr__',
          '__delitem__',
          '__dir__',
          '__doc__',
          '__eq__',
          '__format__',
          '__ge__',
          '__getattribute__',
          '__getitem__',
          '__gt__',
          '__hash__',
          '__iadd__',
          '__imul__',
          '__init__',
          '__init_subclass__',
          '__iter__',
          '__le__',
          '__len__',
          '__lt__',
          '__mul__',
          '__ne__',
          '__new__',
          '__reduce__',
          '__reduce_ex__',
          '__repr__',
          '__reversed__',
          '__rmul__',
          '__setattr__',
          '__setitem__',
          '__sizeof__',
          '__str__',
          '__subclasshook__',
          'append',
          'clear',
          'copy',
          'count',
          'extend',
          'index',
          'insert',
          'pop',
          'remove',
          'reverse',
          'sort']
```

```
In [30]: ##append(we cannot pass more than 1 argument)
l=[]##here we can add any datatype
l.append(4)#if we give any value it will again add newly(it is not saved)
l.append(5)
l.append([1,2])#for list it can be added by taking any elements in list format(being
print(l)
```

```
##index(finds elements in index number)
```

```
[4, 5, [1, 2]]
```

```
In [31]: #in extend list elements will be added as separate elements unlike in append,there
l.extend([22,33,34])
l
```

```
Out[31]: [4, 5, [1, 2], 22, 33, 34]
```

```
In [32]: 1
```

```
Out[32]: [4, 5, [1, 2], 22, 33, 34]
```

```
In [ ]: name=[]
age=[]
for i in range(3):
    i=input("name:")
    name.append(i)
    j=int(input("age:"))
    age.append(j)
print(name)
print(age)
```

```
name:
```

```
In [1]: name=[]
age=[]
for i in range(3):
    i=input("name:")
    name.append(i)
    j=int(input("age:"))
    age.append(j)
print(name)
print(age)
```

```
name:p.sindhu
age:19
name:g.divija
age:17
name:b.geethika
age:18
['p.sindhu', 'g.divija', 'b.geethika']
[19, 17, 18]
```

```
In [12]: l=[1,2,3,4]
l.insert(0,100)
#print(l.sort()) for this we will get output as none
l.sort()
print(l)
l.sort(reverse=True)
print(l)
```

```
[1, 2, 3, 4, 100]
[100, 4, 3, 2, 1]
```

```
In [3]: l.pop()
print(l)
```

```
[100, 4, 3, 2]
```

```
In [4]: l.reverse()
l
```

```
Out[4]: [2, 3, 4, 100]
```

```
In [5]: del l
```

```
In [6]: print(l)
```

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-6-e6f469464218> in <module>()
----> 1 print(l)

NameError: name 'l' is not defined
```

In [7]: `l.reverse(1)`

```
-----  
NameError                                Traceback (most recent call last)  
<ipython-input-7-c65e8e75f6d8> in <module>()  
----> 1 l.reverse(1)  
  
NameError: name 'l' is not defined
```

In [9]: `l=[1,2,3]
a=l.copy()
print(l)
print(a)
a[1]=1999
print(a)
print(l)`

```
[1, 2, 3]  
[1, 2, 3]  
[1, 1999, 3]  
[1, 2, 3]
```

In [20]: `a=("divija","9676047715")
(name,mobile)=a
print(name)
mobile`

```
divija
```

Out[20]: '9676047715'

Dictionaries

```
In [21]: #here we get by key not by index, since dictionary do not have fixed index value
#for key there must be a value,otherwise the default is none
#mutable
#key and value pair is item
dir(dict)
```

```
Out[21]: ['__class__',
          '__contains__',
          '__delattr__',
          '__delitem__',
          '__dir__',
          '__doc__',
          '__eq__',
          '__format__',
          '__ge__',
          '__getattr__',
          '__getitem__',
          '__gt__',
          '__hash__',
          '__init__',
          '__init_subclass__',
          '__iter__',
          '__le__',
          '__len__',
          '__lt__',
          '__ne__',
          '__new__',
          '__reduce__',
          '__reduce_ex__',
          '__repr__',
          '__setattr__',
          '__setitem__',
          '__sizeof__',
          '__str__',
          '__subclasshook__',
          'clear',
          'copy',
          'fromkeys',
          'get',
          'items',
          'keys',
          'pop',
          'popitem',
          'setdefault',
          'update',
          'values']
```

```
In [23]: d={"name":"p.sindhu","roll":"3","age":19}
          d.values()
          d.keys()
          d.items()
```

```
Out[23]: dict_items([('name', 'p.sindhu'), ('roll', '3'), ('age', 19)])
```

In [26]: `d["name"]`

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-26-3dc752b20a59> in <module>()
----> 1 d["name"]
```

TypeError: list indices must be integers or slices, not str

```
In [5]: name=[]
age=[]
for i in range(3):
    i=in
    put("name:")
    name.append(i)
    j=int(input("age:"))
    age.append(j)
print(name)
print(age)
```

```
File "<ipython-input-5-859a3a14b549>", line 4
    i=in
      ^
```

SyntaxError: invalid syntax

```
In [4]: ***
##check whether the given is in range or not
##Lb=Lower bound
##Ub=upper bound
##Lb=20,ub=30
def check_range():
    if(n>=lb and n<=ub):
        print(n,"n is in range")
    else:
        print(n,"n is not in range")
check_range(30,50,2)
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-4-9f5c1501f200> in <module>()
      8     else:
      9         print(n,"n is not in range")
----> 10 check_range(30,50,2)
```

TypeError: check_range() takes 0 positional arguments but 3 were given


```
In [3]: 1
2 def prime():
3     for i in range(2,num):
4         if (num % i) == 0:
5             print(num,"is not a prime number")
6             print(i,"times",num,"is",num)
7             break
8         elif:
9             print(num,"is a prime number")
10        else:
11            print(num,"is not a prime number")
12
```

File "<ipython-input-3-d74180dc169b>", line 8

elif:

^

SyntaxError: invalid syntax

```
In [6]: ##print the prime no.s in given range(2,50)
def prime(lb,ub):
    for i in range(lb,ub+1):
        if isprime(i):
            print(i)
prime(2,50)
```

NameError Traceback (most recent call last)

<ipython-input-6-c1e462a0b820> in <module>()

4 if isprime(i):

5 print(i)

----> 6 prime(2,50)

<ipython-input-6-c1e462a0b820> in prime(lb, ub)

2 def prime(lb,ub):

3 for i in range(lb,ub+1):

----> 4 if isprime(i):

5 print(i)

6 prime(2,50)

NameError: name 'isprime' is not defined

```
In [18]: def per():
          lb=int(input())
          ub=int(input())
          for i in range(lb,ub+1):
              if(per1(i)):
                  print(i)
          per(lb,ub)
```

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-18-efe7d8020874> in <module>()
      5         if(per1(i)):
      6             print(i)
----> 7 per(lb,ub)

NameError: name 'lb' is not defined
```

```
In [15]: ##method2
def elem():
    num=[1,2,3,4,5]
    num.sort()
    print(num)
    print(max(num))
    print(min(num))
    print(num[-2])
elem()
```

```
[1, 2, 3, 4, 5]
5
1
4
```

```
In [19]: #li=[2,3,4,2,3,4]
def li():
    li=[2,3,4,2,3,4]
    l=[]
    for i in li:
        if i not in l:
            l.append(i)
    print(l)
li()
```

```
[2, 3, 4]
```

```
In [25]: def li():
          li=[2,3,4,2,3,4,6,-1]
          l=[]
          for i in li:
              if i not in l:
                  l.append(i)
          print(l)
li()
```

```
[2, 3, 4, 6, -1]
```

In []: