LISTS

In [ ]:
```
str1=''
```

In [1]:
```
l1=[1,2,3,4,5]
l1
```

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

In [2]:
```
type(l1)
```

Out[2]:
```
list
```

- List will be in the sqaure bracket

In [4]:
```
l2=['Apple','Banana','Cherr']
l2
```

Out[4]:
```
['Apple', 'Banana', 'Cherr']
```

In [7]:
```
l3=[1,2,3,'A','B','C']
l3
```

Out[7]:
```
[1, 2, 3, 'A', 'B', 'C']
```

In [8]:
```
l4=[1,2,3,'A','B','C',10.5,True,20+3j]
l4
```

Out[8]:
```
[1, 2, 3, 'A', 'B', 'C', 10.5, True, (20+3j)]
```

In [10]:
```
l5=[100,100,100]
l5
```

Out[10]:
```
[100, 100, 100]
```

In [11]:
```
l6=[1,2,3,['A','B','C']]
l6
```

Out[11]:
```
[1, 2, 3, ['A', 'B', 'C']]
```

In [14]:
```
l7=['$','%','<','^']
l7
```

Out[14]:
```
['$', '%', '<', '^']
```

In [17]:
```
l8=[_]
l8
```

Out[17]:
```
[[[['$', '%', '<', '^']]]]
```

- List is array of elements
- That elemnts can be any data type
- Heterogeneous
- list allows duplicates
- list can extend with any values
- list in list is posiible
- list of underscore, is nothing but list in list

In [19]:
```
l1
#len
#max
#min
#sum
m1=len(l1)
m2=max(l1)
m3=min(l1)
m4=sum(l1)
m1,m2,m3,m4
```

Out[19]:
```
(5, 5, 1, 15)
```

In [20]:
```
len(l1),max(l1),min(l1),sum(l1)
```

Out[20]:
```
(5, 5, 1, 15)
```

In [21]:
```
len(l2),max(l2),min(l2),sum(l2)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[21], line 1
----> 1 len(l2),max(l2),min(l2),sum(l2)

TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

In [22]:
```
len(l3),max(l3),min(l3),sum(l3)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[22], line 1
----> 1 len(l3),max(l3),min(l3),sum(l3)

TypeError: '>' not supported between instances of 'str' and 'int'
```

In [23]:
```
len(l4),max(l4),min(l4),sum(l4)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[23], line 1
----> 1 len(l4),max(l4),min(l4),sum(l4)

TypeError: '>' not supported between instances of 'str' and 'int'
```

In [24]:
```
len(l5),max(l5),min(l5),sum(l5)
```

Out[24]:
```
(3, 100, 100, 300)
```

In [25]:
```
len(l6),max(l6),min(l6),sum(l6)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[25], line 1
----> 1 len(l6),max(l6),min(l6),sum(l6)

TypeError: '>' not supported between instances of 'list' and 'int'
```

In [26]:
```
sorted(l1,reverse=True)
```

Out[26]:
```
[5, 4, 3, 2, 1]
```

In [27]:
```
sorted(l1)
#by default reverse =False
#whic means ascending order
```

Out[27]:
```
[1, 2, 3, 4, 5]
```

In [28]:
```
sorted(l2,reverse=True)
```

Out[28]:
```
['Cherr', 'Banana', 'Apple']
```

In [29]:
```
sorted(l3,reverse=True)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[29], line 1
----> 1 sorted(l3,reverse=True)

TypeError: '<' not supported between instances of 'int' and 'str'
```

In [30]:
```
sorted(l4,reverse=True)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[30], line 1
----> 1 sorted(l4,reverse=True)

TypeError: '<' not supported between instances of 'bool' and 'complex'
```

In [31]:
```
sorted(l5,reverse=True)
```

Out[31]:
```
[100, 100, 100]
```

In [32]:
```
val=reversed(l1)
for i in val:
    print(i)
```

```
5
4
3
2
1
```

In [33]:
```
val=reversed(l2)
for i in val:
    print(i)
```

```
Cherr
Banana
Apple
```

In [34]:
```
val=reversed(l3)
for i in val:
    print(i)
```

```
C
B
A
3
2
1
```

In [35]:
```
val=reversed(l4)
for i in val:
    print(i)
```

```
(20+3j)
True
10.5
C
B
A
3
2
1
```

In [36]:
```
val=reversed(l5)
for i in val:
    print(i)
```

```
100
100
100
```

In [38]:
```
for i in l1:
    print(i)
```

```
1
2
3
4
5
```

In [39]:
```
l2[0],l2[1],l2[2]
```

Out[39]:
```
('Apple', 'Banana', 'Cherr')
```

In [40]:
```
l2[-3],l2[-2],l2[-1]
```

Out[40]:
```
('Apple', 'Banana', 'Cherr')
```

In [41]:
```
for i in range(len(l2)):
    print(i)
```

```
0
1
2
```

In [49]:
```
for i in range(len(l2)):
    print(i)
```

```
0
1
2
```

In [42]:
```
for i in range(len(l1)):
    print(i)
```

```
0
1
2
3
4
```

In [43]:
```
for i in range(len(l3)):
    print(i)
```

```
0
1
2
3
4
5
```

In [45]:
```
for i in range(len(l4)):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
```

In [46]:
```
for i in range(len(l5)):
    print(i)
```

```
0
1
2
```

In [47]:
```
for i in range(len(l6)):
    print(i)
```

```
0
1
2
3
```

In [51]:
```
for i in range(len(l1)):
    print(i,l1[i])
```

```
0 1
1 2
2 3
3 4
4 5
```

In [55]:
```
#positive index
#negative index
#psoitive and negative both

for i in range(len(l2)):
    print(f" Positive index of {l2[i]} is {i}")
    print(f" Negative index of {l2[i]} is {i-3}")
    print(f" Positive index {i} and Negative index of {l2[i]} is {i-3}")
```

```
 Positive index of Apple is 0
 Negative index of Apple is -3
 Positive index 0 and Negative index of Apple is -3
 Positive index of Banana is 1
 Negative index of Banana is -2
 Positive index 1 and Negative index of Banana is -2
 Positive index of Cherr is 2
 Negative index of Cherr is -1
 Positive index 2 and Negative index of Cherr is -1
```

mutable-immutable

In [56]:
```
l7=[1,2,3,4,5]
l7[0]=100
l7
#if we are abe to change the index value then its called as mutable
#lists are mutable
#strings are immutable
```

Out[56]:
```
[100, 2, 3, 4, 5]
```

In [ ]:
```
lists1=[1,2,3,4,5,'A','B','C','D','E',100,200,300,400,"APPLE"]
list1[3:14:2] #p
list1[3:14:-2] #np
list1[3:-14:2] #np
list1[3:-14:-2]
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: