In [1]:
```
## List
```

In [2]:
```
L = [1,2,3,4,6,8]
```

In [3]:
```
L
```

Out[3]:
```
[1, 2, 3, 4, 6, 8]
```

In [4]:
```
## Nesting of the list
```

In [5]:
```
A = [[1,2,3],[11,22,33],[45,56,77]]
```

In [7]:
```
A
```

Out[7]:
```
[[1, 2, 3], [11, 22, 33], [45, 56, 77]]
```

In [8]:
```
A[0]
```

Out[8]:
```
[1, 2, 3]
```

In [9]:
```
A[1]
```

Out[9]:
```
[11, 22, 33]
```

In [10]:
```
A[1][1]
```

Out[10]:
```
22
```

In [11]:
```
A = [[1,2,3],[11,22,33,[500,502]],[45,56,77]]

A
```

Out[11]:
```
[[1, 2, 3], [11, 22, 33, [500, 502]], [45, 56, 77]]
```

In [13]:
```
A[1][3]
```

Out[13]:
```
[500, 502]
```

In [12]:
```
A[1][3][0]
```

Out[12]:
```
500
```

In [18]:
```
A = [[1,2,3],[11,22,33,[500,502]],[45,[239,"Hii"],56,77]]
```

In [19]:
```
A
```

Out[19]:
```
[[1, 2, 3], [11, 22, 33, [500, 502]], [45, [239, 'Hii'], 56, 77]]
```

In [21]:
```
A[2]
```

Out[21]:
```
[45, [239, 'Hii'], 56, 77]
```

In [22]:
```
A[2][1]
```

Out[22]:
```
[239, 'Hii']
```

In [20]:
```
A[2][1][1]
```

Out[20]:
```
'Hii'
```

In [23]:
```
## by using negative indices
```

In [24]:
```
A[-1][1][1]
```

Out[24]:
```
'Hii'
```

In [25]:
```
amazon_cart= [["watch",5000],["phone",10000],["laptop",50000]]

total_cost = 0
```

In [27]:
```
amazon_cart[0][1]
```

Out[27]:
```
5000
```

In [28]:
```
amazon_cart[1][1]
```

Out[28]:
```
10000
```

In [29]:
```
amazon_cart[2][1]
```

Out[29]:
```
50000
```

In [30]:
```
amazon_cart[0][1]+amazon_cart[1][1]+amazon_cart[2][1]
```

Out[30]:
```
65000
```

In [31]:
```
##with for loop
```

In [33]:
```
total_cost = 0
for i in range(len(amazon_cart)):
    print(amazon_cart[i][1])
    total_cost = total_cost + amazon_cart[i][1]

print (total_cost)
```

```
5000
10000
50000
65000
```

In [36]:
```
total_cost = 0
for item in amazon_cart:
    print(item[1])
    total_cost = total_cost + item[1]
    print(total_cost)
```

```
5000
5000
10000
15000
50000
65000
```

In [38]:
```
total_cost = 0
for item in amazon_cart:
    print(item[1])
    total_cost = total_cost + item[1]
print(total_cost)
```

```
5000
10000
50000
65000
```

In [39]:
```
total_cost = 0
print(f"empty cart: {total_cost}")
for item in amazon_cart:
    print(item[1])
    total_cost = total_cost + item[1]
    print(f"cart after adding {item[0]}: {total_cost}")
print(f"total payable amount:{total_cost}")
```

```
empty cart: 0
5000
cart after adding watch: 5000
10000
cart after adding phone: 15000
50000
cart after adding laptop: 65000
total payable amount:65000
```

In [40]:
```
## List comprehension
```

In [42]:
```
A =[1,2,3,4,5,6,7,8,9,10]
A
```

Out[42]:
```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

In [44]:
```
ans = list()

for element in A:
    print(element**2)
```

```
1
4
9
16
25
36
49
64
81
100
```

In [45]:
```
ans.append(element**2)
```

In [46]:
```
ans
```

Out[46]:
```
[100]
```

In [47]:
```
ans = list()

for element in A:
    print(element**2)
```

```
    ans.append(element**2)
print(ans)
```

```
1
4
9
16
25
36
49
64
81
100
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

In [48]:
```
ans = [ele**2 for ele in A]

print(ans)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

In [49]:
```
## only take square root of odd numbers
```

In [51]:
```
ans= list()

for element in A:
    if element%2 != 0:

        print(element**2)
        ans.append(element**2)
ans
```

```
1
9
25
49
81
```
Out[51]:
```
[1, 9, 25, 49, 81]
```

In [52]:
```
ans = [element**2 for element in A if element%2 != 0]

print(ans)
```

```
[1, 9, 25, 49, 81]
```

In [55]:
```
## count
```

In [56]:
```
A =[1,2,1,4,5,6,7,7,7]
```

In [57]:
```
A.count(1)
```

Out[57]:
```
2
```

In [58]:
```
A.count(7)
```

Out[58]:
```
3
```

In [59]:
```
A.count(99)
```

Out[59]:
```
0
```

In [60]:
```
A =[1,2,1,4,5,6,[7,7],[7,7]]
```

In [61]:
```
A.count([7,7])
```

Out[61]:
```
2
```

In [62]:
```
A.count(7)
```

Out[62]:
```
0
```

In [63]:
```
A[-1].count(7)
```

Out[63]:
```
2
```

In [64]:
```
A =[1,2,1,4,5,6,7,7,7]
for i in A:
    print(i,A.count(i))
```

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

In [65]:
```
A = "sunny"
for i in A:
    print(i, A.count(i))
```

```
s 1
u 1
n 2
n 2
y 1
```

In [66]:
```
A = ['sunny','sunny','chandra']
for i in A:
    print (i, A.count(i))
```

```
sunny 2
sunny 2
chandra 1
```

In [67]:
```
A = ['sunny','SUNNY','chandra']
for i in A:
    print (i, A.count(i))
```

```
sunny 1
SUNNY 1
chandra 1
```

In [68]:
```
## Extend
```

In [69]:
```
A = [1,2,3]

B = [11,22,33]

A+B
```

Out[69]:
```
[1, 2, 3, 11, 22, 33]
```

In [78]:
```
A = [1,2,3]

B = [11,22,33]

A.extend(B)  ## extend is a permanent operation inplace operation
```

In [79]:
```
A
```

Out[79]:
```
[1, 2, 3, 11, 22, 33]
```

In [80]:
```
A.append(B)  ## append is a permanent operation inplace operation
```

In [81]:
```
A
```

Out[81]:
```
[1, 2, 3, 11, 22, 33, [11, 22, 33]]
```

In [82]:
```
## Index Method
```

In [83]:
```
A =[11,3,4,56,"Sunny", 1+2j]
```

In [85]:
```
A
```

Out[85]:
```
[11, 3, 4, 56, 'Sunny', (1+2j)]
```

In [86]:
```
A.index(11)
```

Out[86]:
```
0
```

In [87]:
```
A.index("Sunny")
```

Out[87]:
```
4
```

In [89]:
```
## print the index of all the complex number
# do a for loop
# if condition for type check
#if type is complex then print the index
```

In [91]:
```
for element in A:
    print (f"element : {element}| {type(element)}| Index: {A.index(element)}")
```

```
element : 11| <class 'int'>| Index: 0
element : 3| <class 'int'>| Index: 1
element : 4| <class 'int'>| Index: 2
element : 56| <class 'int'>| Index: 3
element : Sunny| <class 'str'>| Index: 4
element : (1+2j)| <class 'complex'>| Index: 5
```

In [92]:
```
for element in A:
    if type(element) == complex:

        print (f"element : {element}| {type(element)}| Index: {A.index(element)}")
```

```
element : (1+2j)| <class 'complex'>| Index: 5
```

In [93]:
```
for element in A:
    if type(element) == str:
```

```
        print (f"element : {element}| {type(element)}| Index: {A.index(element)}")
```

```
element : Sunny| <class 'str'>| Index: 4
```

In [94]:
```
for element in A:
    if type(element) == int:

        print (f"element : {element}| {type(element)}| Index: {A.index(element)}")
```

```
element : 11| <class 'int'>| Index: 0
element : 3| <class 'int'>| Index: 1
element : 4| <class 'int'>| Index: 2
element : 56| <class 'int'>| Index: 3
```

In [95]:
```
#isinstance
```

In [98]:
```
for element in A:
    ##if type(element) == complex
    if isinstance(element,complex):
         print (f"element : {element}| {type(element)}| Index: {A.index(element)}"
```

```
element : (1+2j)| <class 'complex'>| Index: 5
```

In [99]:
```
## print the index all the complex and int also
```

In [100…
```
for element in A:
    ##if type(element) == complex
    if isinstance(element,complex) or isinstance(element,int)  :
         print (f"element : {element}| {type(element)}| Index: {A.index(element)}"
```

```
element : 11| <class 'int'>| Index: 0
element : 3| <class 'int'>| Index: 1
element : 4| <class 'int'>| Index: 2
element : 56| <class 'int'>| Index: 3
element : (1+2j)| <class 'complex'>| Index: 5
```

In [102…
```
for element in A:
    ##if type(element) == complex
    if isinstance(element,(complex,int)):
         print (f"element : {element}| {type(element)}| Index: {A.index(element)}"
```

```
element : 11| <class 'int'>| Index: 0
element : 3| <class 'int'>| Index: 1
element : 4| <class 'int'>| Index: 2
element : 56| <class 'int'>| Index: 3
element : (1+2j)| <class 'complex'>| Index: 5
```

In [103…
```
for element in A:
    ##if type(element) == complex
    if isinstance(element,complex) and isinstance(element,int)  :
         print (f"element : {element}| {type(element)}| Index: {A.index(element)}"
```

In [104…
```
## there is no and condition because there will be no two variable together
```

In [107…
```
ans = [element for element in A if isinstance(element,(complex,int))]

ans
```

Out[107]:
```
[11, 3, 4, 56, (1+2j)]
```

In [110…
```
A = [11, 3, 4, 56+2j, "Sunny", (1+2j),3,3,3,]
A
```

Out[110]:
```
[11, 3, 4, (56+2j), 'Sunny', (1+2j), 3, 3, 3]
```

In [111…
```
A.index(3)
```

Out[111]:
```
1
```

In [112…
```
## indexing will give you the first value index in the list
```

In [113…
```
## Insert
```

In [115…
```
A = [1,3,4,56,7,8,9]
A
```

Out[115]:
```
[1, 3, 4, 56, 7, 8, 9]
```

In [117…
```
A.insert(2,[2,5,8,90,88])

A
```

Out[117]:
```
[1, 3, [2, 5, 8, 90, 88], 4, 56, 7, 8, 9]
```

In [118…
```
## Remove
```

In [119…
```
A = [1,3,4,56,7,8,9]
A
```

Out[119]:
```
[1, 3, 4, 56, 7, 8, 9]
```

In [120…
```
A.remove(56)
A
```

Out[120]:
```
[1, 3, 4, 7, 8, 9]
```

In [121…
```
A = [1,1,1,3,4,56,7,8,9]
A
```

Out[121]:
```
[1, 1, 1, 3, 4, 56, 7, 8, 9]
```

In [123…
```
A.remove(1)
A
```

Out[123]:
```
[1, 3, 4, 56, 7, 8, 9]
```

In [124…
```
## in remove only first occurance of the element will be removed
```

In [125…
```
## POP
```

In [126…
```
A = [1,1,1,3,4,56,7,8,9]
A
```

Out[126]:
```
[1, 1, 1, 3, 4, 56, 7, 8, 9]
```

In [128…
```
A.pop()
```

Out[128]:
```
9
```

In [131…
```
new_list=[]

[new_list.append (item) for item in A if item not in new_list]

print(new_list)
```

```
[1, 3, 4, 56, 7, 8]
```

In [ ]: