In [ ]:
```
Strings write up

Assignment on strings
```

In [ ]:
```
- Read the strings

- type

- max

- min

- len

- in (for loop)

- range

- index

- mutable

- slice

- methods
```

- array of elements

In [1]:
```
string1='python'
```

In [2]:
```
list1=[1,2,3,4]
list1 # You can write numbers
```

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

In [3]:
```
list2=['A','B','C','D']
list2 # you can write strings
```

Out[3]:
```
['A', 'B', 'C', 'D']
```

In [4]:
```
list3=[1,2,3,4,'A','B','C','D']
list3
```

Out[4]:
```
[1, 2, 3, 4, 'A', 'B', 'C', 'D']
```

In [6]:
```
list4=[1,'Apple',10.5,10+20j,True]
list4
```

Out[6]:
```
[1, 'Apple', 10.5, (10+20j), True]
```

In [7]:
```
list5=[100,100,100]
list5
```

Out[7]:
```
[100, 100, 100]
```

In [8]:
```
list6=[[1,2,3,4]]
list6
```

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

- list is an array of elements
- list elements heterogeneous(different) data types are allowed
- duplicates are allowed
- list in list also works

type

In [9]:
```
type(list3)
```

Out[9]:
```
list
```

max

In [10]:
```
list1
```

Out[10]:
```
[1, 2, 3, 4]
```

In [11]:
```
list2
```

Out[11]:
```
['A', 'B', 'C', 'D']
```

In [12]:
```
print(max(list1)) # 4
print(max(list2)) # D ====== > ascii ord
print(max(list3)) # error we cant compare different data types
```

```
4
D

---------------------------------------------------------------------------
-
TypeError                                 Traceback (most recent call las
t)
Cell In[12], line 3
      1 print(max(list1)) # 4
      2 print(max(list2)) # D ====== > ascii ord
----> 3 print(max(list3))

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

min

In [13]:
```
print(min(list1)) # 1
print(min(list2)) # A ====== > ascii ord
print(min(list3)) # error we cant compare different data types
```

```
1
A

---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call las
t)
Cell In[13], line 3
      1 print(min(list1)) # 1
      2 print(min(list2)) # A ====== > ascii ord
----> 3 print(min(list3))

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

len

In [14]:
```
print(len(list1))  # 4
print(len(list2)) # 4
print(len(list3)) #8
```

```
4
4
8
```

sum

In [16]:
```
print(list1)
print(sum(list1))
```

```
[1, 2, 3, 4]
10
```

In [17]:
```
print(list2)
print(sum(list2))
```

```
['A', 'B', 'C', 'D']

---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call las
t)
Cell In[17], line 2
      1 print(list2)
----> 2 print(sum(list2))

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

- sum
- eval
- range

in

In [18]:
```
list1
```

Out[18]:
```
[1, 2, 3, 4]
```

In [21]:
```
1 in list1
2 in list1
3 in list1
4 in list1

i in list1

# can i iterate through for loop
```

Out[21]:
```
False
```

In [22]:
```
for i in list1:
    print(i)
```

```
1
2
3
4
```

Concatenation

In [23]:
```
list1
```

Out[23]:
```
[1, 2, 3, 4]
```

In [24]:
```
list2
```

Out[24]:
```
['A', 'B', 'C', 'D']
```

In [25]:
```
print(list1+list2)
print(list2+list1)
```

```
[1, 2, 3, 4, 'A', 'B', 'C', 'D']
['A', 'B', 'C', 'D', 1, 2, 3, 4]
```

In [26]:
```
new_list=list1+list2
```

In [27]:
```
new_list
```

Out[27]:
```
[1, 2, 3, 4, 'A', 'B', 'C', 'D']
```

In [28]:
```
list1*3  # list1+list1+list1
```

Out[28]:
```
[1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4]
```

In [ ]:
```
list1-list2  # Fail
list1*list2  # Fail
list1/list2  # Fail

#'a'/'b'
```

In [ ]:
```
96 ====== laptop

techshame
```

Index

In [29]:
```
list3
```

Out[29]:
```
[1, 2, 3, 4, 'A', 'B', 'C', 'D']
```

In [33]:
```
# python index start with zero
list3[0],list3[1],list3[2],list3[3]

# list3[i]
# i=0 1 2 3
```

Out[33]:
```
(1, 2, 3, 4)
```

In [37]:
```
list3[-1]

-8  -7  -6  -5  -4  -3  -2  -1
1   2   3   4    A   B   C  D
0   1   2   3    4   5   6  7
```

Out[37]:
```
'D'
```

In [39]:
```
for i in range(len(list3)):
    print('postive index is: {} and negative index is {}: for an element {}
```

```
postive index is: 0 and negative index is -8: for an element 1
postive index is: 1 and negative index is -7: for an element 2
postive index is: 2 and negative index is -6: for an element 3
postive index is: 3 and negative index is -5: for an element 4
postive index is: 4 and negative index is -4: for an element A
postive index is: 5 and negative index is -3: for an element B
postive index is: 6 and negative index is -2: for an element C
postive index is: 7 and negative index is -1: for an element D
```

In [42]:
```
#WAP find the elements which are having len<3
# list=['Apple','Ball','Cat','Ab','Cd','Ef']

# step-1:  iterate the list using for loop
# step-2:  apply the if condition len(<element>)<3:
# step-3:            print(element)

list1=['Apple','Ball','Cat','Ab','Cd','Ef']
for i in range(len(list1)):
    if len(list1[i])<3:
        print(list1[i])


for i in list1:
    if len(i)<3:
        print(i)
```

```
Ab
Cd
Ef
Ab
Cd
Ef
```

In [43]:
```
#WAP find the elements which are having #
# list=['App#e','B#ll','C#t','Ab','Cd','Ef']

# step-1:  iterate the list using for loop
# step-2:  apply the if condition:
# step-3:            print(element)

list1=['App#e','B#ll','C#t','Ab','Cd','Ef']
for i in list1:
    if '#' in i:
        print(i)
```

```
App#e
B#ll
C#t
```

In [44]:
```
list1=['App#e','B#ll','C#t','Ab','Cd','Ef']
count=0
for i in list1:
    if '#' in i:
        count=count+1

print(count)
```

```
3
```

In [46]:
```
list1=[1,2,3,['Apple','Ball']]

# retrive the ball using index
# In the given list how many elements are there: 4
list1[3][1]
```

Out[46]:
```
'Ball'
```

In [53]:
```
list2=[[[[[[['Cherry']]]]]]]
list2[0][0][0][0][0][0][0]
```

Out[53]:
```
'Cherry'
```

In [ ]:
```
list3=[[[['A','B',[[[1,2,3,['Car']]]]]]]]
# retrive the car
```

In [54]:
```
list3=[[[['A','B',[[[1,2,3,['car']]]]]]]]
list3[0][0][0][2][0][0][3][0]
```

Out[54]:
```
'car'
```

Mutabel

In [56]:
```
string1='welcome'
# 'l'  to 'L'
string1[2]='L'

# strings are immutable
```

```
---------------------------------------------------------------------------
-
TypeError                                 Traceback (most recent call las
t)
Cell In[56], line 3
      1 string1='welcome'
      2 # 'l'  to 'L'
----> 3 string1[2]='L'

TypeError: 'str' object does not support item assignment
```

In [58]:
```
list1=['A','B','C']
list1[0]=100
list1
```

Out[58]:
```
[100, 'B', 'C']
```

slice

In [59]:
```
list1=[10,20,30,40,50,'P','Y','T','H','O','N','a','b','c','D']

print(list1[2:14:3])   # p
print(list1[2:14:-3])  # np
print(list1[2:-14:3])  # np
print(list1[2:-14:-3]) # p
print(list1[-2:14:3])  # p
print(list1[-2:-14:3])  # np
print(list1[-2:-14:-3])  # p
print(list1[-2:14:-3])  # np
```

```
[30, 'P', 'H', 'a']
[]
[]
[30]
['c']
[]
['c', 'N', 'T', 50]
[]
```

- Reading a list
- Different ways to provide elements
- type/min/max/sum/len
- in
- concatenation
- index
- mutable
- slice

In [60]:
```
dir([])
```

Out[60]:
```
['__add__',
 '__class__',
 '__class_getitem__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__getstate__',
 '__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']
```

clear-copy

In [62]:
```
list1=[1,2,3,4]

# I wanat copy these elements in a list2

list2=list1.copy()

list1.clear()

print('list1:',list1)  # []
print('list2:',list2)  # [1,2,3,4]
```

```
list1: []
list2: [1, 2, 3, 4]
```

In [ ]:
```
# Side heading

# write the code

# insight/observation from above code
```

In [ ]:
```

```

In [ ]:
```

```

In [ ]:
```

```