Lists

In [ ]:

```
> A List is A collection of characters variables ,and
 number variables and boolean values datatypes
> a list is a to store multiple data with in a single
variable
> a list is a ordered type of data
> a list as denoted as []
> a list item as denoted with double quotes.

syntax:
    items=["item1","item2","item3"]
    print(items)
```

In [1]:

```
# Example for the list

li=["apple","bananna","orange","grapes","milk"]
li
```

Out[1]:

```
['apple', 'bananna', 'orange', 'grapes', 'milk']
```

In [2]:

```
# type of the list

print(type(li))
```

```
<class 'list'>
```

In [3]:

```
# length if the list
print(len(li))
```

```
5
```

In [4]:

```
# accessing first element in a list

print(li[0])
```

```
apple
```

In [5]:

```
# accessing a last elemtn in a list
print(li[-1])
```

```
milk
```

In [7]:

```
# accessing the item in a list or not

if "apple" in li:
    print("yes")
else:
    print("no")
```

```
yes
```

In [8]:

```
li
```

Out[8]:

```
['apple', 'bananna', 'orange', 'grapes', 'milk']
```

In [9]:

```
li[0]="pinaple"
li
```

Out[9]:

```
['pinaple', 'bananna', 'orange', 'grapes', 'milk']
```

In [11]:

```
li.insert(1,"gopal")
li
```

Out[11]:

```
['pinaple', 'gopal', 'bananna', 'orange', 'grapes', 'milk']
```

In [12]:

```
li1=["gopal","123","li"]
li1
```

Out[12]:

```
['gopal', '123', 'True']
```

In [13]:

```
ligrapes
```

Out[13]:

```
['pinaple', 'gopal', 'bananna', 'orange', 'grapes', 'milk']
```

In [14]:

```
li[2:5]
```

Out[14]:

```
['bananna', 'orange', 'grapes']
```

In [15]:

```
li[2:]
```

Out[15]:

```
['bananna', 'orange', 'grapes', 'milk']
```

In [16]:

```
li[:4]
```

Out[16]:

```
['pinaple', 'gopal', 'bananna', 'orange']
```

In [17]:

```
li
```

Out[17]:

```
['pinaple', 'gopal', 'bananna', 'orange', 'grapes', 'milk']
```

In [18]:

```
li[:4]
```

Out[18]:

```
['pinaple', 'gopal', 'bananna', 'orange']
```

In [19]:

```
li
```

Out[19]:

```
['pinaple', 'gopal', 'bananna', 'orange', 'grapes', 'milk']
```

In [20]:

```
li.remove("gopal")
li
```

Out[20]:

```
['pinaple', 'bananna', 'orange', 'grapes', 'milk']
```

In [23]:

```
li1=["sbi","axes","panjab"]
li+li1
```

Out[23]:

```
['pinaple', 'bananna', 'orange', 'grapes', 'milk', 'sbi', 'axes', 'panjab']
```

In [24]:

```
li1
```

Out[24]:

```
['sbi', 'axes', 'panjab']
```

In [26]:

```
li1.clear()
```

In [27]:

```
li1
```

Out[27]:

```
[]
```

In [28]:

```
li
```

Out[28]:

```
['pinaple', 'bananna', 'orange', 'grapes', 'milk']
```

In [31]:

```
li.sort()
li
```

Out[31]:

```
['bananna', 'grapes', 'milk', 'orange', 'pinaple']
```

In [32]:

```
li.remove("milk")
```

In [33]:

```
li
```

Out[33]:

```
['bananna', 'grapes', 'orange', 'pinaple']
```

In [34]:

```
del li[1]
```

In [35]:

```
li
```

Out[35]:

```
['bananna', 'orange', 'pinaple']
```

In [37]:

```
# list using loop

for i in li:
    print(i,end=" ")
```

```
bananna orange pinaple
```

Tuple

it is a collection of different types of data.

it is immutable(can't change)

we can using round brackets() to write a tuple.

to create the empty tuple

tuple_name=()

to create single values

tuple_name=(values)

to create Multiple values

tuple_name=(values1,value2...)

In [38]:

```
#create tuple
t1=(10,20,30)
t1
print(type(t1))
```

```
<class 'tuple'>
```

In [39]:

```
#single value tuple
t2=(10)
print(type(t2))
t3=(20,)
print(type(t3))
```

```
<class 'int'>
<class 'tuple'>
```

In [40]:

```
t3
```

Out[40]:

```
(20,)
```

In [41]:

```
t2
```

Out[41]:

```
10
```

In [45]:

```
# how to access the values from the tuple

t1
print(t1[2])
```

```
30
```

In [47]:

```
print(t1[0:1])
```

```
(10,)
```

In [48]:

```
t2=(10,20,10,20,30,20,20,30,10)
# to count the number of ocurences
t2.count(10)
```

Out[48]:

```
3
```

In [49]:

```
t2=(10,20,10,20,30,20,20,30,10)
# to count the number of ocurences
t2.count(30)
```

Out[49]:

```
2
```

In [50]:

```
t2=(10,20,10,20,30,20,20,30,10)
# to count the number of ocurences
t2.count(20)
```

Out[50]:

```
4
```

In [51]:

```
#index
t2.index(20)
```

Out[51]:

```
1
```

In [52]:

```
t2.index(10)
```

Out[52]:

```
0
```

In [53]:

```
t2.index(30)
```

Out[53]:

```
4
```

In [54]:

```
tuple1 = ("abc", 34, True, 40, "male")

print(tuple1)
```

```
('abc', 34, True, 40, 'male')
```

In [55]:

```
# length if the list
print(len(tuple1))
```

```
5
```

In [ ]:

```
#Dictionary :
-it is collection of different data types.
-it is group of key and values(key:value)->item
-in dictionary keys are unique
- written in({})
-each and every item seperated with commas(,)
-acessing dictionaries values by using key names
-it is a mutable(changable)
```

In [ ]:

```
to create a empty dictionary:
    -dictionary_name={}
```

In [ ]:

```
to create the dictionaries values:
    dictionaries_name={key:value,key:value2...}
```

In [56]:

```
#  to create a dictionaries with values
d1={'a':10,'b':34,'c':45}
print(d1)
print(type(d1))
```

```
{'a': 10, 'b': 34, 'c': 45}
<class 'dict'>
```

In [57]:

```
# to create a dictionaries with different data types..
d2={'a':100,'name':'anusha','branch':'cse','b':45.8}
print(d2)
```

```
{'a': 100, 'name': 'anusha', 'branch': 'cse', 'b': 45.8}
```

In [58]:

```
#acessing the dictionaries values using the key names
print(d2['name'])
print(d2['b'])
print(d2['a'])
```

```
anusha
45.8
100
```

In [59]:

```
#update the dictionairs values
print(d2)
d2['branch']='EEE'
print(d2)
```

```
{'a': 100, 'name': 'anusha', 'branch': 'cse', 'b': 45.8}
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8}
```

In [60]:

```
print(dir(dict))
```

```
['__class__', '__contains__', '__delattr__', '__delitem__', '__dir__', '__do
c__', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__',
'__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__le__',
'__len__', '__lt__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__
repr__', '__reversed__', '__setattr__', '__setitem__', '__sizeof__', '__str_
_', '__subclasshook__', 'clear', 'copy', 'fromkeys', 'get', 'items', 'keys',
'pop', 'popitem', 'setdefault', 'update', 'values']
```

In [61]:

```
#keys
print(d2)
print(d2.keys())
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8}
dict_keys(['a', 'name', 'branch', 'b'])
```

In [62]:

```
#values()
print(d2)
print(d2.values())
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8}
dict_values([100, 'anusha', 'EEE', 45.8])
```

In [63]:

```
#items
print(d2)
print(d2.items())
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8}
dict_items([('a', 100), ('name', 'anusha'), ('branch', 'EEE'), ('b', 45.8)])
```

In [64]:

```
#copy()
print(d2)
d3=d2.copy()
print(d3)
print(type(d3))
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8}
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8}
<class 'dict'>
```

In [65]:

```
#get
print(d2)
print(d2.get('a'))
print(d2.get('name'))
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8}
100
anusha
```

In [66]:

```
#setdefault
print(d2)
print(d2.setdefault('rollno',310))
print(d2)
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8}
310
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8, 'rollno': 310}
```

In [67]:

```
#pop
print(d2)
print(d2.pop('b'))
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'b': 45.8, 'rollno': 310}
45.8
```

In [68]:

```
#popitem
print(d2)
print(d2.popitem())
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE', 'rollno': 310}
('rollno', 310)
```

In [69]:

```
#popitem
print(d2)
print(d2.popitem())
```

```
{'a': 100, 'name': 'anusha', 'branch': 'EEE'}
('branch', 'EEE')
```

In [70]:

```
#popitem
print(d2)
print(d2.popitem())
```

```
{'a': 100, 'name': 'anusha'}
('name', 'anusha')
```

In [71]:

```
#clear
print(d2)
print(d2.clear())
```

```
{'a': 100}
None
```

In [72]:

```
#clear
print(d2)
print(d2.clear())
```

```
{}
None
```

In [ ]:

```

```