data structures

list-> elements are seperated by comma and enclosed with square bracket

list can be hetrogeneous

list can be retrived using index

mutable -> can make the changes after creating the list

In [5]:
```
sample_list=[1,13.6,"hi",12,1]
sample_list
```

Out[5]:
```
[1, 13.6, 'hi', 12, 1]
```

In [6]:
```
type(sample_list)
```

Out[6]:
```
list
```

In [8]:
```
sample_list[1]
```

Out[8]:
```
13.6
```

In [9]:
```
sample_list[-1]
```

Out[9]:
```
1
```

In [10]:
```
sample_list[2]
```

Out[10]:
```
'hi'
```

In [11]:
```
sample_list[5]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Cell In[11], line 1
----> 1 sample_list[5]

IndexError: list index out of range
```

In [12]:
```
sample_list[0]=10 #list is mutable,we can make changes
```

In [13]:
```
sample_list
```

Out[13]:
```
[10, 13.6, 'hi', 12, 1]
```

In [14]:
```
sample_list[-2]
```

Out[14]:
```
12
```

Tuple

hetrogeneous

paranthesis

can retrived data using index

immutable

In [19]:
```
sample_tuple=(1,13.6,"hi",12,1) #we re using here small baracket
sample_tuple
```

Out[19]:
```
(1, 13.6, 'hi', 12, 1)
```

In [20]:
```
type(sample_tuple)
```

Out[20]:
```
tuple
```

In [22]:
```
sample_tuple[1] #we re using here square baracket while retriving
```

Out[22]:
```
13.6
```

In [23]:
```
sample_tuple[-2]
```

Out[23]:
```
12
```

In [24]:
```
sample_tuple[0]=10 #we can not make changes means tuples are immutable
sample_tuple
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[24], line 1
----> 1 sample_tuple[0]=10
      2 sample_tuple

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

set

set is mutable but we can not make cahnges using index

can not retrive using index

numbers are sorted according to first number

set define using curl bracket

duplicate numbers will be removed

In [27]:
```
sample_set={1,13.6,"hi",12,1,127,25,25,"apple","king","ball"}
sample_set
```

Out[27]:
```
{1, 12, 127, 13.6, 25, 'apple', 'ball', 'hi', 'king'}
```

In [28]:
```
sample_set[0]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[28], line 1
----> 1 sample_set[0]

TypeError: 'set' object is not subscriptable
```

In [29]:
```
sample_set[1]=10
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[29], line 1
----> 1 sample_set[1]=10

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

In [30]:
```
type(sample_set)
```

Out[30]:
```
set
```

In [31]:
```
sample_set.add(1000)
sample_set
```

Out[31]:
```
{1, 1000, 12, 127, 13.6, 25, 'apple', 'ball', 'hi', 'king'}
```

In [32]:
```
sample_set.remove(1)
sample_set
```

Out[32]:
```
{1000, 12, 127, 13.6, 25, 'apple', 'ball', 'hi', 'king'}
```

In [33]:
```
sample_set.remove("hi")
sample_set
```

Out[33]:
```
{1000, 12, 127, 13.6, 25, 'apple', 'ball', 'king'}
```

Dictionary

Has key-value pair data structure

key is unique

values can be duplicated

values can be retrived using key

key can not be retrived using value

In [34]:
```
sample_dict={1:"apple",2:35.4,3:"data",2:"every",6:35.4,7:35.4,2:"hi"}
sample_dict #key is unique and cosider last value
```

Out[34]:
```
{1: 'apple', 2: 'hi', 3: 'data', 6: 35.4, 7: 35.4}
```

In [35]:
```
sample_dict[3]
```

Out[35]:
```
'data'
```

In [36]:
```
 sample_dict["data"]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[36], line 1
----> 1 sample_dict["data"]

KeyError: 'data'
```

In [37]:
```
sample_dict[7]=100
sample_dict
```

Out[37]:
```
{1: 'apple', 2: 'hi', 3: 'data', 6: 35.4, 7: 100}
```

In [38]:
```
sample_dict[10]="business"
sample_dict
```

Out[38]:
```
{1: 'apple', 2: 'hi', 3: 'data', 6: 35.4, 7: 100, 10: 'business'}
```

In [ ]:
```

```