In [1]:
```
#Neha Nemade
#Roll_no:22150
```

In [3]:
```
'''1. Numeric Data type
      a) Int
      b)Float
      c)Complex '''
# Int data type

x = 10
y = 20
z = x + y
print("Data type of the variable x is :-",type(x))
print("Data type of the variable y is :-",type(y))
print("Data type of the variable y is :-",type(y))
print("The value of z is:-",z)
```

```
Data type of the variable x is :- <class 'int'>
Data type of the variable y is :- <class 'int'>
Data type of the variable y is :- <class 'int'>
The value of z is:- 30
```

In [4]:
```
# float data type
k = 52.66
print("\nThe data type of variable k is:- ",type(k))
```

```
The data type of variable k is:-  <class 'float'>
```

In [5]:
```
# complex data type
m = complex(x,y)
print("\nThe complex value of m is :-",m)
print("The type of the variable m is:-",type(m))
```

```
The complex value of m is :- (10+20j)
The type of the variable m is:- <class 'complex'>
```

In [6]:
```
# int to float and vice versa
print("\nThe integer part of k is :-",int(k))
print("The float value of x is :-",float(y))
```

```
The integer part of k is :- 52
The float value of x is :- 20.0
```

In [7]:
```
'''2.Boolean datatype'''
# Examples of boolean data type
p = True
print("\nThe type of variable p is:-",type(p))
k =  1>8
print("The result of k is:-",k)
f = 45<96
print("The result of f is:-",f)
```

```
The type of variable p is:- <class 'bool'>
The result of k is:- False
The result of f is:- True
```

In [8]:
```
# We can get the integer notation of the boolean as follows :-
print(int(k))
print(int(f))
print()
'''3.Sequence data types
    a)String
    b)List
    c)Tuple'''
```

```
0
1
```

Out[8]:
```
'3.Sequence data types \n    a)String \n    b)List\n    c)Tuple'
```

In [6]:
```
# String data type
a = "Neha"
print(a)
s = "My name is Neha Nemade"
print(s)
print("The data type of the variable a is:-",type(a))
print("The data type of the variable s is:-",type(s))
print()
# Various methods in strings
k = "Neha"
```

```
Neha
My name is Neha Nemade
The data type of the variable a is:- <class 'str'>
The data type of the variable s is:- <class 'str'>
```

In [7]:
```
#1. Length of string
print("The length of the string is:-",len(k))
#2. Upper case
print("The upper case of k is",k.upper())
#2. Lower case
print("The lower case of k is",k.lower())
#3. Capitalize
k.capitalize()
s = "My name is Neha Nemade"
```

```
The length of the string is:- 4
The upper case of k is NEHA
The lower case of k is neha
```

In [8]:
```
#4. title() converts first letter of sentence to upper case
print(s.title())
print()

a = "Python"
b = "Strings"
c = a+b
print("The string after concatenation is :-",c)
#If we want space we can do as
d = a+" "+b
print("The string after concatenation is :-",d)
```

```
My Name Is Neha Nemade

The string after concatenation is :- PythonStrings
The string after concatenation is :- Python Strings
```

In [9]:
```
# string indexing
print("The first five letters of the string s are :-",s[0:5])
print()

#List data type
list = [5,9,55,'Neha',2,'python']
print(list)
print("The data type of list is ",type(list))
# slicing
print(list[0:4])
# list is mutable ie we can modify the elements
list[2]='k'
print(list)
print(type(list))
```

```
The first five letters of the string s are :- My na

[5, 9, 55, 'Neha', 2, 'python']
The data type of list is  <class 'list'>
[5, 9, 55, 'Neha']
[5, 9, 'k', 'Neha', 2, 'python']
<class 'list'>
```

In [14]:
```
#tuple data type
k = ('python',10,85,66)
print(k)
print(type(k))
```

```
('python', 10, 85, 66)
<class 'tuple'>
```

In [ ]:
```

```