```
In [13]:  pwd

Out[13]: 'C:\\Users\\Rahul Sapireddy\\AppData\\Roaming\\SPB_16.6\\Python Programming'
```

Python Variables

```
In [3]:   a = 10
          b = "John"
          c = 10.5
```

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

Out[6]:  float
```

```
In [7]:   a, b, c = 20, "Srinivas", 20.5
```

```
In [8]:   type(b)

Out[8]:  str
```

```
In [9]:   @value= 20

  File "<ipython-input-9-54522a37d203>", line 1
    @value= 20
         ^
SyntaxError: invalid syntax
```

```
In [10]:  _value = 20
```

```
In [11]:  bool a= 20

  File "<ipython-input-11-0bf298907f3e>", line 1
    bool a= 20
         ^
SyntaxError: invalid syntax
```

```
In [12]:  var = 10
          Var = 20
```

Python Data Types

Integers

In [14]:

```
a = 10
b = 20
c = 30
```

In [15]:

```
print(a)
print(b)
print(c)
```

```
10
20
30
```

In [16]:

```
print(a,b,c)
```

```
10 20 30
```

In [17]:

```
print("Value of a is", a, "Value of b is", b, "Value of c is", c)
```

```
Value of a is 10 Value of b is 20 Value of c is 30
```

In [22]:

```
print("Value of a is %d, Value of b is %d, Value of c is %d" %(a,b,c))
```

```
Value of a is 10, Value of b is 20, Value of c is 30
```

In [24]:

```
print("The value of a, b, c is {0} {1} {2}".format(a,b,c))
```

```
The value of a, b, c is 10 20 30
```

In [25]:

```
a = input("Enter the value of a")
b = input("Enter the value of b")
c = a + b
print(c)
```

```
Enter the value of a10
Enter the value of b20
1020
```

In [26]:

```
type(a)
```

Out[26]:

```
str
```

In [27]:

```
type(b)
```

Out[27]:

```
str
```

In [28]:

```
a = int(input("Enter the value of a"))
b = int(input("Enter the value of b"))
c = a + b
print(c)
```

```
Enter the value of a10
Enter the value of b20
30
```

In [29]:

```
a = 10
b = str(a)
type(b)
```

Out[29]:

```
str
```

In [30]:

```
print(b)
```

```
10
```

In [31]:

```
print(a)
```

```
10
```

In [32]:

```
str(a)
```

Out[32]:

```
'10'
```

Python Boolean and Strings

In [33]:

```
a = 0
type(a)
```

Out[33]:

```
int
```

In [34]:

```
a = True
type(a)
```

Out[34]:

```
bool
```

In [35]:

```
a = False
type(a)
```

Out[35]:

```
bool
```

In [36]:

```
string1 = 'welcome to python'
```

In [37]:

```
string2 = "welcome to python"
```

In [38]:

```
string1 = """hi
welcome to
python"""
```

In [39]:

```
string = "apple"10"banana"10.5  # Strings are homogenous -> Only made of characters
```

```
  File "<ipython-input-39-0733b8d5e470>", line 1
    string = "apple"10"banana"10.5
                     ^
SyntaxError: invalid syntax
```

In [40]:
```
string = "apple and banana"
```

In [42]:
```
string[10]
```

Out[42]:
```
'b'
```

In [43]:
```
string[-6]
```

Out[43]:
```
'b'
```

In [44]:
```
string[1]
```

Out[44]:
```
'p'
```

In [45]:
```
string[1] = "y"
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-45-ba6cfca4b478> in <module>
----> 1 string[1] = "y"

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

In [47]:
```
string[6:]   # Staring Slicing
```

Out[47]:
```
'and banana'
```

In [48]:
```
string[:6] # Ending Slicing
```

Out[48]:
```
'apple '
```

In [49]:
```
string[3:5] # Stating and Ending Slicing
```

Out[49]:
```
'le'
```

In [50]:
```
# Functions of the Strings
string.capitalize()  # Capitalize the first letter of the string
```

Out[50]:
```
'Apple and banana'
```

In [51]:
```
string.find('a')
```

Out[51]:
```
0
```

In [52]:
```
string.split()
```

Out[52]:
```
['apple', 'and', 'banana']
```

In [53]:
```
string.split('a')
```

Out[53]:
```
['', 'pple ', 'nd b', 'n', 'n', '']
```

In [54]:

```
string.count('a')
```

Out[54]:

```
5
```

In [56]:

```
string.isalpha()  # To check the presence of spaces
```

Out[56]:

```
False
```

In [57]:

```
dir(string)
```

Out[57]:

```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__getnewargs__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mod__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__rmod__',
 '__rmul__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'capitalize',
 'casefold',
 'center',
 'count',
 'encode',
 'endswith',
 'expandtabs',
 'find',
 'format',
 'format_map',
 'index',
 'isalnum',
 'isalpha',
 'isascii',
 'isdecimal',
 'isdigit',
 'isidentifier',
 'islower',
 'isnumeric',
 'isprintable',
 'isspace',
 'istitle',
 'isupper',
 'join',
 'ljust',
 'lower',
 'lstrip',
 'maketrans',
 'partition',
 'replace',
 'rfind',
 'rindex',
 'rjust',
 'rpartition',
 'rsplit',
 'rstrip',
 'split',
 'splitlines',
 'startswith',
 'strip',
 'swapcase',
 'title',
 'translate',
 'upper',
 'zfill']
```

Python List

In [71]:

```
list1 = [1,"John", "Srinivas Rahul", 20.5, 30]
```

In [60]:

```
list1[-1]
```

Out[60]:

```
30
```

In [63]:

```
list1[1:4]
```

Out[63]:

```
['John', 'Srinivas Rahul', 20.5]
```

In [64]:

```
list1[1] = "Sapireddy"
```

In [65]:

```
list1
```

Out[65]:

```
[1, 'Sapireddy', 'Srinivas Rahul', 20.5, 30]
```

In [66]:

```
# Functions of List
dir(list1)
```

Out[66]:

```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__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']
```

In [67]:

```
list1.append(20)
```

In [68]:

```
list1
```

Out[68]:

```
[1, 'Sapireddy', 'Srinivas Rahul', 20.5, 30, 20]
```

```
In [69]:  list1.clear()

In [70]:  list1

Out[70]:  []

In [72]:  list1

Out[72]:  [1, 'John', 'Srinivas Rahul', 20.5, 30]

In [74]:  list2 = list1.copy()

In [75]:  list1

Out[75]:  [1, 'John', 'Srinivas Rahul', 20.5, 30]

In [76]:  list2

Out[76]:  [1, 'John', 'Srinivas Rahul', 20.5, 30]

In [78]:  list1.count("Srinivas")

Out[78]:  0

In [79]:  list1.extend([1,2]) # Extending a list with other list

In [80]:  list1

Out[80]:  [1, 'John', 'Srinivas Rahul', 20.5, 30, 1, 2]

In [81]:  list1.index(2)

Out[81]:  6

In [83]:  list1.insert(4,4)

In [84]:  list1

Out[84]:  [1, 'John', 'Srinivas Rahul', 20.5, 4, 30, 1, 2]

In [85]:  list1.insert(6,"Sri")

In [86]:  list1

Out[86]:  [1, 'John', 'Srinivas Rahul', 20.5, 4, 30, 'Sri', 1, 2]

In [87]:  list1.pop() # Last in first out

Out[87]:  2
```

```
In [88]:   list1

Out[88]:   [1, 'John', 'Srinivas Rahul', 20.5, 4, 30, 'Sri', 1]

In [89]:   list1.remove("Sri")

In [90]:   list1

Out[90]:   [1, 'John', 'Srinivas Rahul', 20.5, 4, 30, 1]

In [91]:   list1.reverse()

In [92]:   list1

Out[92]:   [1, 30, 4, 20.5, 'Srinivas Rahul', 'John', 1]

In [94]:   list1.sort

Out[94]:   <function list.sort(*, key=None, reverse=False)>

In [95]:   list3 = [2,5,3,2,6,8,3]

In [96]:   list3.sort

Out[96]:   <function list.sort(*, key=None, reverse=False)>

In [97]:   list3

Out[97]:   [2, 5, 3, 2, 6, 8, 3]

In [98]:   list3.sort()

In [99]:   list3

Out[99]:   [2, 2, 3, 3, 5, 6, 8]

In [100]:  # Comprehensive Techniques
           list4 = [i for i in list1]

In [101]:  list4

Out[101]:  [1, 4, 20.5, 30, 'Srinivas Rahul', 'John', 1]

In [102]:  list5 = [i for i in range(10)]

In [103]:  list5

Out[103]:  [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

In [104]:  list5 = [i for i in range(10) if(i%2==0)]
In [105]:  list5

Out[105]:  [0, 2, 4, 6, 8]
```

```
In [107]:  st = i for i in "Python"

             File "<ipython-input-107-d76ba470d20a>", line 1
               st = i for i in "Python"
                        ^
           SyntaxError: invalid syntax
```

```
In [108]:  st = [i for i in "Python"]
```

```
In [109]:  st

Out[109]:  ['P', 'y', 't', 'h', 'o', 'n']
```

Python Tuple

```
In [110]:  tuple = (1,"SB", "remore", 10.5)
```

```
In [111]:  tuple

Out[111]:  (1, 'SB', 'remore', 10.5)
```

```
In [112]:  tuple[0] = 22

           ---------------------------------------------------------------------------
           TypeError                                 Traceback (most recent call last)
           <ipython-input-112-c650779f1ca8> in <module>
           ----> 1 tuple[0] = 22

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

```
In [114]:  tuple[0]

Out[114]:  1
```

```
In [116]:  tuple[0:4]

Out[116]:  (1, 'SB', 'remore', 10.5)
```

In [117]:
```
dir(tuple)
```

Out[117]:
```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__getnewargs__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__rmul__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'count',
 'index']
```

In [118]:
```
tuple.count(1)
```

Out[118]:
```
1
```

In [119]:
```
tuple.index(1)
```

Out[119]:
```
0
```

In [120]:
```
# Comprehensive Technique
tuple2 = i for i in range(10)
```

```
  File "<ipython-input-120-a29954179d0c>", line 2
    tuple2 = i for i in range(10)
                 ^
SyntaxError: invalid syntax
```

Python Dictionaties

In [121]:

```
dict = {"name": "Srinivas", "Qualification": "Master of Science", "Roll No": 35}
```

In [122]:

```
dict
```

Out[122]:

```
{'name': 'Srinivas', 'Qualification': 'Master of Science', 'Roll No': 35}
```

In [123]:

```
dict["name"]
```

Out[123]:

```
'Srinivas'
```

In [124]:

```
dict["name"] = "Name"
```

In [125]:

```
dict
```

Out[125]:

```
{'name': 'Name', 'Qualification': 'Master of Science', 'Roll No': 35}
```

In [126]:

```
# Dictionary Functions
dir(dict)
```

Out[126]:

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

In [127]:

```
dict2 = dict.copy()
```

In [129]:

```
dict2
```

Out[129]:

```
{'name': 'Name', 'Qualification': 'Master of Science', 'Roll No': 35}
```

In [130]:

```
dict.items()
```

Out[130]:

```
dict_items([('name', 'Name'), ('Qualification', 'Master of Science'), ('Roll No', 35)])
```

In [132]:

```
dict.keys()
```

Out[132]:

```
dict_keys(['name', 'Qualification', 'Roll No'])
```

In [133]:

```
dict.values()
```

Out[133]:

```
dict_values(['Name', 'Master of Science', 35])
```

Python Sets

In [134]:

```
set = {1,2,3,4,5,6}
```

In [135]:

```
set[0]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-135-5a7e82b41e48> in <module>
----> 1 set[0]

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

In [138]:

```
dir(set)
```

Out[138]:

```
['__and__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__gt__',
 '__hash__',
 '__iand__',
 '__init__',
 '__init_subclass__',
 '__ior__',
 '__isub__',
 '__iter__',
 '__ixor__',
 '__le__',
 '__len__',
 '__lt__',
 '__ne__',
 '__new__',
 '__or__',
 '__rand__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__ror__',
 '__rsub__',
 '__rxor__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__sub__',
 '__subclasshook__',
 '__xor__',
 'add',
 'clear',
 'copy',
 'difference',
 'difference_update',
 'discard',
 'intersection',
 'intersection_update',
 'isdisjoint',
 'issubset',
 'issuperset',
 'pop',
 'remove',
 'symmetric_difference',
 'symmetric_difference_update',
 'union',
 'update']
```

Python Conditional Statements

if ifelse if elif nested ifelse

In [141]:

```
# if condition
a = int(input("Enter a value: "))
b = int(input("Enter b value: "))
c = a + b
print("Value of c: ", c)
if(c>70):
    print("He is passed.")
else:
    print("He is failed.")
```

```
Enter a value: 20
Enter b value: 30
Value of c:  50
He is failed.
```

In [144]:

```
# if condition
a = int(input("Enter a value: "))
b = int(input("Enter b value: "))
c = a + b
print("Value of c: ", c)
if(c>70):
    print("He is passed.")
elif (c>65 and c<70):
    d = 70 - c
    print("Failed with a difference of : ", d, "marks")
else:
    print("He is failed.")
```

```
Enter a value: 32
Enter b value: 35
Value of c:  67
Failed with a difference of :  3 marks
```

In [150]:

```
user_input = input("Enter the symbol: ")
a = int(input("Enter a value: "))
b = int(input("Enter b value"))
if(user_input == "+"):
    d = a+b
    print("Addition is: ", d)
    if(d>60):
        print("Just a Statement")
elif(user_input == "*"):
    d = a*b
    print("Multiplication is: ", d)
elif(user_input == "-"):
    d = a-b
    print("Subtraction is: ", d)
elif(user_input == "/"):
    d = a/b
    print("Division is: ", d)
else:
    print("No correct operation is given")
```

```
Enter the symbol: +
Enter a value: 50
Enter b value58
Addition is:  108
Just a Statement
```

Python Loops

In [151]:

```
"""for
while
continue
break
"""
```

Out[151]:

```
'for\nwhile\ncontinue\nbreak\n'
```

In [153]:

```
for i in range(10):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
```

In [155]:

```
for i in range(1,10,2):
    print(i)
```

```
1
3
5
7
9
```

In [156]:

```
string = "hi i am srinivas"
for letter in string:
    print(letter)
```

```
h
i

i

a
m

s
r
i
n
i
v
a
s
```

In [158]:

```
string = "hi i am srinivas"
for letter in range(len(string)):
    print(letter)
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
```

In [161]:

```
string = "hi i am srinivas"
for letter in range(len(string)):
    print(string[letter])
```

```
h
i

i

a
m

s
r
i
n
i
v
a
s
```

In [162]:

```
list = [1,2,3,4,5]
for i in list:
    print(i)
```

```
1
2
3
4
5
```

In [1]:

```
# while loop
i = 1
while(i<10):
    print(i)
    i = i+1
```

```
1
2
3
4
5
6
7
8
9
```

In [4]:

```
for i in range(41):
    if(i>30):
        break
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
```

In [6]:

```python
for i in range(41):
    if(i%2==0):
        continue; # To skip some conditionality
    print(i)
```

```
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
```

Python Functions

In [13]:

```python
def add(a,b):
    c = a+b
    print(c)
```

In [14]:

```python
add(1,3)
```

```
4
```

In [15]:

```python
p = add(3,5)
```

```
8
```

In [17]:

```python
print(p) # As the function is not returning anything
```

```
None
```

In [19]:

```python
def add(a,b):
    c = a+b
    print(c)
    return c
```

In [20]:

```
add(1,2)
```

```
3
```

Out[20]:

```
3
```

In [21]:

```
def function(a,b):
    c = a+b
    d = a-b
    e = a/b
    f = a*b
    return (c,d,e,f)
```

In [24]:

```
h,i,j,k = function(2,3)
```

In [25]:

```
print(h)
print(i)
print(j)
print(k)
```

```
5
-1
0.6666666666666666
6
```

In [26]:

```
def fun(a,b):
    for i in range(a,b):
        if(i==a+10):
            continue
        if(i==(a*2)+10):
            break
        print(i)
fun(5,50)
```

```
5
6
7
8
9
10
11
12
13
14
16
17
18
19
```

In [27]:

```
def functionname(*args):
    for i in args:
        print(i)
```

In [28]:

```
functionname(1,2,3,4,5,6,7,8,9,0)  # args are used to give n number of arguments
```

```
1
2
3
4
5
6
7
8
9
0
```

In [30]:

```
def fun(**args):
    for key,value in args.items():
        print(key,value)
```

In [31]:

```
fun(name = "Srinivas", compamy = "no job")    # **args -> To give key value pairs
```

```
name Srinivas
compamy no job
```

Python File Handling

In [35]:

```
f = open("newtext1.txt", "x")
```

```
---------------------------------------------------------------------------
FileExistsError                           Traceback (most recent call last)
<ipython-input-35-fab3a725be0b> in <module>
----> 1 f = open("newtext1.txt",  "x")

FileExistsError: [Errno 17] File exists: 'newtext1.txt'
```

In [38]:

```
f = open("newtext1.txt",  "w")
f.write("\nwelcome to python")
f.close()
```

In [45]:

```
f = open("newtext1.txt",  "a")
a = int(input("Enter a: "))
b = int(input("Enter b: "))
c = a+b
f.write("\nThe value of a is " +  str(a) +"Value of b is: " +  str(b) + "And the addition of these two num
f.close()
```

```
Enter a: 55
Enter b: 44
```

In [46]:

```
f = open("newtext1.txt", "r")
p = f.read()
print(p)
f.close()
```

```

120
91
The value of a is40Value of b is: 55And the addition of these two numners is: 95
The value of a is 55Value of b is: 44And the addition of these two numners is: 99
```

In [47]:

```
f = open("newtext1.txt", "r")
p = f.readlines()
print(p)
f.close()
```

```
['\n', '120\n', '91\n', 'The value of a is40Value of b is: 55And the addition of these two numners is:
95\n', 'The value of a is 55Value of b is: 44And the addition of these two numners is: 99']
```

In [49]:

```
f = open("newtext1.txt", "r")
p = f.readlines()
for i in p:
    print(i)
# print(p)
f.close()
```

```


120

91

The value of a is40Value of b is: 55And the addition of these two numners is: 95

The value of a is 55Value of b is: 44And the addition of these two numners is: 99
```

In [50]:

```
f = open("newtext1.txt", "r")
f1 = open("newtext2.txt", "a")
p = f.read()
for i in p:
    f1.write(i)
f1.close()
f.close()
```

In [53]:

```
f = open("Capture.PNG", "rb")
p = f.read()
print(p)
f.close()
```

```
ZZZ;a)\x04jiiiiiii\xed\x84\xa5\x10\xa8\xa5\xa5\xa5\xa5\xa5\xa5\xa5\xb5\x13\x96B\xa0
\x96\x96\x96\x96\x96\x96\x96\xd6NX\n\x81ZZZZZZZZ;a)\x04jiiiiiii\xed\x84\xa5\x10\xa8
\xa5\xa5\xa5\xa5\xa5\xa5\xa5\xb5\x13\x96B\xa0\x96\x96\x96\x96\x96\x96\x96\xd6NX
\n\x81ZZZZZZZZ;a)\x04jiiiiiii\xed\x84\xf5\xa9\xbf\xfa_\'\xd3_\xfdO\x96\xb8\xd1I\r<\xa3
\xff1\x92>m\xd6#\xadL\x9f\xf3og\xfc\xdb\xb1\xb3\xfe\xea\x7f\x9c\xd8\xc0O\xac\xfb?\x
8cp\xbds\xd3\xb3\x9c\xfb\xfe\x04\xd7[\xb7=+t\xa3o5\xf0\x8c\xfe\xfex\xb7v.\xbdq\x16
\xfb.\xcem\x7f\x9c\xeb\x8f\xe3\x9eez\xe7\xd8\xf6\xf7\xc3\x1b\xf8\xf0\xba\xff\xdd\xb1
\xaew.\xbd\xf1c3\xfeM\xe3\xbb\x18\xff\xa6\xebYp\x96\xe9\x8d\x86e<\xa3/\x1d\xe3\xd
6\xce\xa57\xce\xaa\xf9\xd1\x19g\xc1\x9dv\r\xd6\xbb|\xe9(\xd7\x1fUi\xd7\xd0\xbf\xf8
\r\xd7;\x97\xde\xf872~d\xd9Yu\xff7\xe3\xbb\x1a\xff7\xd7{\xed\xf2\x05Y\x7f\xdd\xfa\x
17Bg\xd5\xfc\x88\xb2gux\xc6\x03}\xfep\xda-\xea\x0f\xb3\xbd\xf1\xc32~h\xd9Yu?\xa4\x
81\x1fB\xbb\xee\xe5\xbc\xea\x0f6\xbe\x9b\xf1\x83]\xcf\xaa\xf5\x07\xd5=\xab\x03\x1b
8\xebs\xce\xab\xfe\x80\x9a\xef\xbe\x17\xafMo\xdd\xf6C]?\xb4\xeaw\xc7\xb6\x9a\xef\x
df\xc0\xf7\xa7\xdd>\xeb\xbc\xea\xbf\x96\xf1\xaf\xd5|\xf7\xaa\xffW\xd7\xb3\xb3L\x1f
\xe9\xab\r\x9c\xf5\xb7\xce\xab\xfe+\r\x9cU\xf3\x7f\xf1\xee\xb4G\xb0\xf6\xfd\x97]\xff
\xe5\xaa\xdf\x03\xdb\xf6t^\xf5\xfb5\xf0\xfd2>\xc4;o\xb3>\xa4\xe6\xa2\xdd\xf7\xdc\xd
7\xad\xe1\xfb\xd2\xee\x9f\xb1^\xe93\xfb4pV\xcd\xf7\xf6\x8em\x91\xabT\x83[\x9f\x02
\xdc\xfd\x7f\x06\xe0<\xec\xc5=\xeb\x7f8\xafz\x06\xb8\xa2\x03\xeeRg\x80\x13/H \xcf\x
f6\x1e\xfa\x8c\xfe\xa1\x00}Y\xf7\xd0\xe7\x9d\x01N\x9c\xb7Y?\xa1\x81\x03\xe8\xb0\x
86\x7f\x8b\xfe\xea\xef\xad\xfb\x1e0\xc7\xeb\xac\x03\xeeRg\x80\x13\xc7\xb6\xc8s\x02
```

In [55]:

```
f = open("Capture.PNG", "rb")
f1 = open("Capture2.PNG", "wb")
for i in f:
   f1.write(i)
f1.close()
f.close()
```

Python Exception Handling

In [57]:

```
a = 10
b = 0
c = a/b
print(c)
print("bye")
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-57-e002922df2aa> in <module>
      1 a = 10
      2 b = 0
----> 3 c = a/b
      4 print(c)
      5 print("bye")

ZeroDivisionError: division by zero
```

In [59]:

```
a = int(input("Enter a value: "))
b = int(input("Enter b value: "))
c = a/b      # Critical Statement
print(c)
print("bye")
```

```
Enter a value: 10
Enter b value: 0


---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-59-0b75443641ab> in <module>
      1 a = int(input("Enter a value: "))
      2 b = int(input("Enter b value: "))
----> 3 c = a/b
      4 print(c)
      5 print("bye")

ZeroDivisionError: division by zero
```

In [60]:

```
a = int(input("Enter a value: "))
b = int(input("Enter b value: "))
try:
    c = a/b      # Critical Statement
    print(c)
except Exception as e:
    print(e)
print("bye")
```

```
Enter a value: 10
Enter b value: 0
division by zero
bye
```

In [61]:

```
a = int(input("Enter a value: "))
b = int(input("Enter b value: "))
try:
    c = a/b      # Critical Statement
    print(c)
except :
    print("You cannot divide any number with 0")
print("bye")
```

```
Enter a value: 10
Enter b value: 0
You cannot divide any number with 0
bye
```

In [66]:

```
a = int(input("Enter a value: "))
b = int(input("Enter b value: "))
try:
    print("File is open")
    c = a/b      # Critical Statement
    print(c)

except :
    print("You cannot divide any number with 0")
finally: # Executed in case of any error and in case of no errors
    print("File is close")
print("bye")
```

```
Enter a value: 10
Enter b value: 10
File is open
1.0
File is close
bye
```

In [71]:

```


try:
    a = int(input("Enter a value: "))
    b = int(input("Enter b value: "))
except ValueError:
    print("You cannot give float or integer as a or b")

try:
    c = a/b      # Critical Statement
    print(c)
except ZeroDivisionError:
    print("You cannot divide any number with 0")

finally: # Executed in case of any error and in case of no errors
    print("File is close")
print("bye")
```

```
Enter a value: 10.5
You cannot give float or integer as a or b
You cannot divide any number with 0
File is close
bye
```

In [73]:

```
a = int(input("Enter a:"))
b = input("Enter b: ")
try:
    c = a+b # Critical Statement
except TypeError:
    print("Do not add string to an integer")
print(c)
```

```
Enter a:10
Enter b: d
Do not add string to an integer
1.0
```

In [74]:

```
list = [1,2,3,4,5]
print(list[5])
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-74-00270da38a18> in <module>
      1 list = [1,2,3,4,5]
----> 2 print(list[5])

IndexError: list index out of range
```

In [75]:

```
list = [1,2,3,4,5]
try:
   print(list[5]) # Critical Statement
except IndexError:
   print("Index out of range")
```

```
Index out of range
```

In [ ]: