In [1]:
```
print("Hello World")
```
```
Hello World
```

In [2]:
```
type("Hello World")
```
Out[2]:
```
str
```

In [1]:
```
print('Heloo furqaan,r u excited to learn python again from scratch')
```
```
Heloo furqaan,r u excited to learn python again from scratch
```

In [4]:
```
print(Hello world)  # error string must be enclosed in '  ' or "   " or '''  '''
```
```
  File "C:\Users\furqa\AppData\Local\Temp\ipykernel_18584\2715667532.py", line 1
    print(Hello world)  # error
                ^
SyntaxError: invalid syntax
```

In [5]:
```
 type('a')
```
Out[5]:
```
str
```

In [6]:
```
type("a")
```
Out[6]:
```
str
```

In [7]:
```
type('''a''')
```
Out[7]:
```
str
```

In [8]:
```
a=5 ##Assignment
```

In [9]:
```
a
```
Out[9]:
```
5
```

In [10]:
```
type(a)
```
Out[10]:
```
int
```

In [11]:
```
a==5 ##comparison
```
Out[11]:
```
True
```

In [12]:
```
a==6
```
Out[12]:
```
False
```

In [13]:
```
type(a==5) ## comparison operator returns boolean(True or False)
```
Out[13]:
```
bool
```

In [14]:
```
range(10)
```
Out[14]:
```
range(0, 10)
```

In [15]:
```
range(1,10)
```
Out[15]:
```
range(1, 10)
```

In [16]:
```
list(range(10)) # is equal to range(0,10),start=0,end=10, end is exclusive,start is inclusive
```
Out[16]:
```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

In [17]:
```
list(range(3,10))
```
Out[17]:
```
[3, 4, 5, 6, 7, 8, 9]
```

In [18]:
```
list(range(0,10,2))  # start=0,end=10 and 2 refers to step size
```
Out[18]: [0, 2, 4, 6, 8]

In [19]:
```
list(range(0,10,1))
```
Out[19]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

In [20]:
```
list(range(10,3)) # need to b rectified
```
Out[20]: []

In [23]:
```
list(range(10,3,-1)) # step size -1(negative)
```
Out[23]: [10, 9, 8, 7, 6, 5, 4]

In [21]:
```
print(list(range(2)))
```
```
[0, 1]
```

In [22]:
```
list(range(3,10,-1))
```
Out[22]: []

In [24]:
```
list(range(3,10.5)) #error
```
```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18584\1628764533.py in <module>
----> 1 list(range(3,10.5))

TypeError: 'float' object cannot be interpreted as an integer
```

In [25]:
```
list(range(3.5,10.5)) #error
```
```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18584\1585570156.py in <module>
----> 1 list(range(3.5,10.5))

TypeError: 'float' object cannot be interpreted as an integer
```

In [26]:
```
list(range(-10,-3))
```
Out[26]: [-10, -9, -8, -7, -6, -5, -4]

In [27]:
```
list(range(10,3,-1))
```
Out[27]: [10, 9, 8, 7, 6, 5, 4]

In [28]:
```
'anand'+'jha'
```
Out[28]: 'anandjha'

In [29]:
```
'anand'+' jha'
```
Out[29]: 'anand jha'

In [30]:
```
'anand'+' '+'jha'
```
Out[30]: 'anand jha'

In [31]:
```
'Furqaan'+' '+'Shafi'
```
Out[31]: 'Furqaan Shafi'

In [32]:
```
a=input("enter a number ")  # input ftn takes values as string
```
```
enter a number 10
```

In [33]:
```
a
```
Out[33]: '10'

In [34]:
```
type(a)
```

Out[34]:
```
str
```

In [35]:
```
num1=input("enter 1st number ") ##used to take dynamic values from user
num2=input("enter 2nd number ")
summ=num1+num2
print(summ)    # here string concatination happens
```

```
enter 1st number 55
enter 2nd number 34
5534
```

In [36]:
```
num1=int(input("enter 1st number ")) ##typecasted i.e converted string to int
num2=int(input("enter 2nd number "))
summ=num1+num2
print(summ)
```

```
enter 1st number 55
enter 2nd number 34
89
```

In [37]:
```
num1=int(input("enter 1st number ")) ##typecaste
num2=int(input("enter 2nd number "))
summ=num1+num2
print(summ)          # error as num2 will accept only in values
```

```
enter 1st number 33
enter 2nd number 22.3

---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18584\3461254877.py in <module>
      1 num1=int(input("enter 1st number ")) ##typecaste
----> 2 num2=int(input("enter 2nd number "))
      3 summ=num1+num2
      4 print(summ)

ValueError: invalid literal for int() with base 10: '22.3'
```

In [38]:
```
num1=int(input("enter 1st number ")) ##typecaste
num2=float(input("enter 2nd number "))   ## typecaste to float value
summ=num1+num2
print(summ)
```

```
enter 1st number 33
enter 2nd number 22.5
55.5
```

In [39]:
```
type(num1)
```

Out[39]:
```
int
```

In [40]:
```
type(num2)
```

Out[40]:
```
float
```

In [41]:
```
type(summ)
```

Out[41]:
```
float
```

In [43]:
```
##designing calculator

num1=float(input("enter 1st number "))
operation=input("enter an operation ")
num2=float(input("enter 2nd number "))
if operation=='+':
    print(num1+num2)
elif operation=='-':
    print(num1-num2)
elif operation=='/':
    print(num1/num2)
elif operation=='*':
    print(num1*num2)
elif operation=='%':
    print(num1%num2)
else:
    print("wrong operation,enter a correct operation")
```

```
enter 1st number 33
enter an operation /
enter 2nd number 4
8.25
```

In [46]:
```
33%3 #returns the reminder of 33/3
```

Out[46]: 0

In [47]:
```
33%2
```

Out[47]: 1

In [49]:
```
33//3 # returns the quotient
```

Out[49]: 11

In [51]:
```
3**2  # 3*2,,,,3 raised to pwr 2
```

Out[51]: 9

In [52]:
```
3**3 ## 3 raised to pwr 3
```

Out[52]: 27

list

In [53]:
```
LIST['FSB']   # python is case sensitive,so error
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18584\1226588391.py in <module>
----> 1 LIST['FSB']

NameError: name 'LIST' is not defined
```

In [54]:
```
LIST('FSB') #error
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18584\3812464464.py in <module>
----> 1 LIST('FSB')

NameError: name 'LIST' is not defined
```

In [55]:
```
list('fsb ') # takes space as character as well,gotta b carefull
```

Out[55]: ['f', 's', 'b', ' ']

In [56]:
```
list('fsb')
```

Out[56]: ['f', 's', 'b']

In [57]:
```
a=list('fsb')
```

In [58]:
```
a
```

Out[58]:
```
['f', 's', 'b']
```

Slicing

You can return a range of characters by using the slice syntax.

Specify the start index and the end index, separated by a colon, to return a part of the string.

In [59]:
```
a[0]
```

Out[59]:
```
'f'
```

In [60]:
```
len(a)
```

Out[60]:
```
3
```

In [61]:
```
a[3]  #error
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18584\3565805452.py in <module>
----> 1 a[3]

IndexError: list index out of range
```

In [62]:
```
a[2]
```

Out[62]:
```
'b'
```

In [64]:
```
len(a)  # returns the length of string
```

Out[64]:
```
3
```

In [63]:
```
a[len(a)-1]
```

Out[63]:
```
'b'
```

In [65]:
```
a[-2]
```

Out[65]:
```
's'
```

In [84]:
```
a=list('independence')
```

In [85]:
```
a
```

Out[85]:
```
['i', 'n', 'd', 'e', 'p', 'e', 'n', 'd', 'e', 'n', 'c', 'e']
```

In [86]:
```
a[4]
```

Out[86]:
```
'p'
```

In [87]:
```
a[-1]
```

Out[87]:
```
'e'
```

In [88]:
```
a[0:4]  # end exclusive
```

Out[88]:
```
['i', 'n', 'd', 'e']
```

In [89]:
```
a[::1]
```

Out[89]:
```
['i', 'n', 'd', 'e', 'p', 'e', 'n', 'd', 'e', 'n', 'c', 'e']
```

In [90]:
```
a[::2] ##every alternate character
```

Out[90]:
```
['i', 'd', 'p', 'n', 'e', 'c']
```

In [91]:
```
len(a)
```

Out[91]:
```
12
```

In [92]:
```
a[12] # error as counting starts from 0 so end would be 11
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18584\433484699.py in <module>
----> 1 a[12] # error as counting starts from 0 so end would be 11

IndexError: list index out of range
```

In [93]:
```
a[11]
```

Out[93]:
```
'e'
```

In [94]:
```
a[::-1] # reverse a string,its nt saved internally
```

Out[94]:
```
['e', 'c', 'n', 'e', 'd', 'n', 'e', 'p', 'e', 'd', 'n', 'i']
```

In [95]:
```
a
```

Out[95]:
```
['i', 'n', 'd', 'e', 'p', 'e', 'n', 'd', 'e', 'n', 'c', 'e']
```

In [96]:
```
b=a[::-1]  # reverse a string
```

In [97]:
```
b
```

Out[97]:
```
['e', 'c', 'n', 'e', 'd', 'n', 'e', 'p', 'e', 'd', 'n', 'i']
```

In [98]:
```
a
```

Out[98]:
```
['i', 'n', 'd', 'e', 'p', 'e', 'n', 'd', 'e', 'n', 'c', 'e']
```

In [99]:
```
a="hey how are you"
b=a[::-1]
```

In [100]:
```
a
```

Out[100]:
```
'hey how are you'
```

In [101]:
```
b
```

Out[101]:
```
'uoy era woh yeh'
```

In [102]:
```
type(a)
```

Out[102]:
```
str
```

In [103]:
```
type(b)
```

Out[103]:
```
str
```

In [104]:
```
a='furqaan'
```

In [105]:
```
a[2]
```

Out[105]:
```
'r'
```

In [107]:
```
a[2]='e'  # error----- string is immutable
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18584\3904754911.py in <module>
----> 1 a[2]='e'  # error----- string is immutable

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

In [110]:
```
cont_str=list(a)
cont_str
```

Out[110]: ['f', 'u', 'r', 'q', 'a', 'a', 'n']

In [111]:
```
cont_str[2]
```

Out[111]: 'r'

In [112]:
```
cont_str[2]='s'  # list----is muttable
```

In [113]:
```
cont_str
```

Out[113]: ['f', 'u', 's', 'q', 'a', 'a', 'n']

In [114]:
```
empty_list=[] ##empty list creation
```

In [116]:
```
type(empty_list)
```

Out[116]: list

In [117]:
```
empty_list
```

Out[117]: []

In [118]:
```
list_obj=['furqaan',22,'d','jazlan',333,'furqaan' ]
```

In [119]:
```
list_obj
```

Out[119]: ['furqaan', 22, 'd', 'jazlan', 333, 'furqaan']

In [121]:
```
list_obj=['furqaan',22,'d',['jazlan',333,'furqaan'] ]   # list within list
```

In [122]:
```
list_obj
```

Out[122]: ['furqaan', 22, 'd', ['jazlan', 333, 'furqaan']]

In [123]:
```
len(list_obj)
```

Out[123]: 4

In [124]:
```
list_obj[3]
```

Out[124]: ['jazlan', 333, 'furqaan']

In [125]:
```
list_obj[3][1]
```

Out[125]: 333

In [126]:
```
list_obj[-1][0]
```

Out[126]: 'jazlan'

In [127]:
```
list_obj[-1][-1] ##btr for computation
```

Out[127]: 'furqaan'

In [128]:
```
l=['f','a','b']
```

In [129]:
```
l.append("furqaan")
```

In [130]:
```
l
```

Out[130]: ['f', 'a', 'b', 'furqaan']

In [131]:
```
l.insert(3,'ssss')
```

In [132]:
```
l
```

Out[132]: ['f', 'a', 'b', 'ssss', 'furqaan']

task add even and odd in seperate lists

In [133]:
```
Listofnumbers=[2,3,4,5,6,7,8,9,10,11,12,13,14,15]
```

In [134]:
```
type(Listofnumbers)
```

Out[134]:
```
list
```

In [135]:
```
Listofnumbers[::2]
```

Out[135]:
```
[2, 4, 6, 8, 10, 12, 14]
```

In [136]:
```
Listofnumbers[1::2]
```

Out[136]:
```
[3, 5, 7, 9, 11, 13, 15]
```

In [137]:
```
a=[]
b=[]
```

In [138]:
```
len(Listofnumbers)
```

Out[138]:
```
14
```

In [139]:
```
a
```

Out[139]:
```
[]
```

In [ ]:
```

```

In [140]:
```
range(len(Listofnumbers))
```

Out[140]:
```
range(0, 14)
```

In [143]:
```
for i in range(len(Listofnumbers)):
    if Listofnumbers[i]%2==0:
        a.append(Listofnumbers[i])
    else:
        b.append(Listofnumbers[i])
```

In [144]:
```
a
```

Out[144]:
```
[2, 4, 6, 8, 10, 12, 14]
```

In [145]:
```
b
```

Out[145]:
```
[3, 5, 7, 9, 11, 13, 15]
```

In [146]:
```
s='furqaan'
```

In [147]:
```
s.find('qaan')
```

Out[147]:
```
3
```

In [148]:
```
s.find('l')
```

Out[148]:
```
-1
```

In [149]:
```
s='abcjdefjghijkl'
```

In [150]:
```
s.split('j')
```

Out[150]:
```
['abc', 'def', 'ghi', 'kl']
```

In [151]:
```
s.partition("j")
```

Out[151]:
```
('abc', 'j', 'defjghijkl')
```

In [152]:
```
f='furqaan'
```

In [154]:
```
f.center(11,'*')
```
Out[154]: '**furqaan**'

In [155]:
```
reversed(f)
```
Out[155]: <reversed at 0x262f18e6220>

In [156]:
```
s=reversed(f)
```

In [157]:
```
s
```
Out[157]: <reversed at 0x262f18e6c10>

In [158]:
```
list(s)
```
Out[158]: ['n', 'a', 'a', 'q', 'r', 'u', 'f']

In [159]:
```
s=[1,2,3,4,5]
```

In [160]:
```
s
```
Out[160]: [1, 2, 3, 4, 5]

In [161]:
```
s.append([3,4,5,6,7])
```

In [162]:
```
s
```
Out[162]: [1, 2, 3, 4, 5, [3, 4, 5, 6, 7]]

In [163]:
```
s.extend([3,34,45,5])
```

In [164]:
```
s
```
Out[164]: [1, 2, 3, 4, 5, [3, 4, 5, 6, 7], 3, 34, 45, 5]

In [165]:
```
d=[]
```

In [166]:
```
type(d)
```
Out[166]: list

In [167]:
```
s[5]
```
Out[167]: [3, 4, 5, 6, 7]

In [168]:
```
s[5][-2]
```
Out[168]: 6

In [169]:
```
'f' in 'furqaan'
```
Out[169]: True

In [170]:
```
str_obj = "aesbhfwkqfhwjflKDFQWLjrpkil2wnnhlkiwfkqhrpi23ho"
```

In [171]:
```
str_obj
```
Out[171]: 'aesbhfwkqfhwjflKDFQWLjrpkil2wnnhlkiwfkqhrpi23ho'

In [172]:
```
str_obj.find('w') ## index of the first occurrence
```
Out[172]: 6

In [173]:
```
str_obj.find('45') ## if it doesnt exist it returns -1
```
Out[173]: -1

In [175]:
```
e = str_obj.split('j')  # ommits the specified character
```

In [176]:
```
e
```

Out[176]:
```
['aesbhfwkqfhw', 'flKDFQWL', 'rpkil2wnnhlkiwfkqhrpi23ho']
```

In [177]:
```
e[-1]
```

Out[177]:
```
'rpkil2wnnhlkiwfkqhrpi23ho'
```

In [178]:
```
str_obj.partition('j') # includes the specified character as well
```

Out[178]:
```
('aesbhfwkqfhw', 'j', 'flKDFQWLjrpkil2wnnhlkiwfkqhrpi23ho')
```

In [179]:
```
text = "INDIA IS MY COUNTRY"
```

In [180]:
```
text.center(50,'*')
```

Out[180]:
```
'***************INDIA IS MY COUNTRY***************'
```

In [181]:
```
text[::-1]
```

Out[181]:
```
'YRTNUOC YM SI AIDNI'
```

In [182]:
```
var = input('Please input your name')
```

```
Please input your namefurqaan shafi
```

In [183]:
```
'My name is {}'.format(var)
```

Out[183]:
```
'My name is furqaan shafi'
```

In [184]:
```
print("My name is ", var)
```

```
My name is  furqaan shafi
```

In [185]:
```
list_num = [24,45,67,32,67,90]
```

In [187]:
```
[i * 2 for i in list_num]   # list comprehension
```

Out[187]:
```
[48, 90, 134, 64, 134, 180]
```

In [ ]:
```

```