

In [2]:

```
name ="krishna"  
age=19  
course ="B.tech"  
print("my is name is {} and my age is {} and I am pursuing {}".format(name,age,course))
```

my is name is krishna and my age is 19 and I am pursuing B.tech

In [3]:

```
#formatting in python  
name ="krishna"  
age=19  
course ="B.tech"  
print("my is name is {a} and my age is {b} and I am pursuing {c}".format(a=name,b=age,c=course))
```

my is name is krishna and my age is 19 and I am pursuing B.tech

control flow

decision making

In [4]:

```
# if statement
```

In [5]:

```
age = 18  
if age>=18:  
    print("You are eligible to vote ")
```

you are eligible to vote

In [10]:

```
age=int(input("Enter the age :"))  
if age>=18 and age<=45:  
    print("You are young blood ")  
else:  
    print("thank you your application is reject ")
```

Enter the age :50

thank you your application is reject

In [9]:

```
print(age)
```

32

In [25]:

```
product_price=int(input("Enter the product price : "))
if product_price >1000:
    print(f"product price is {product_price*0.8}")
elif product_price >
```

Enter the product price : 100000
product price is 80000.0

In []:

In [6]:

```
age = 18
if age<18:
    print("You are eligible to vote ")
```

In [11]:

```
==,>==,<==
```

File "<ipython-input-11-0535fa98496a>", line 1

```
==,>==,<==
```

^

SyntaxError: invalid syntax

In [12]:

```
100*0.8
```

Out[12]:

80.0

In [13]:

```
5000*0.8
```

Out[13]:

4000.0

In [18]:

```
5000/20
```

Out[18]:

250.0

In [19]:

```
20/100
```

Out[19]:

0.2

In [21]:

```
5000*20/100
```

Out[21]:

1000.0

In [23]:

```
50000*0.8
```

Out[23]:

40000.0

In [26]:

```
join
```

```
-----  
-  
NameError                                Traceback (most recent call last)  
t)  
<ipython-input-26-ca88921fe3b2> in <module>  
----> 1 join
```

NameError: name 'join' is not defined

In [1]:

```
total_amount = 1000  
while total_amount !=0:  
    print(total_amount)  
    total_amount =total_amount-100  
else:  
    print(" put more money bank people")
```

```
1000  
900  
800  
700  
600  
500  
400  
300  
200  
100  
put more money bank people
```

In [2]:

```
## for loop
```

In [10]:

```
a= 7
for i in range(0,a):
    for j in range(0,i+1):
        print('*',end=' ')
    print("\r")
```

```
*
* *
* * *
* * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

logical AND

In [4]:

```
l2=[3,2,4,3,5]
l = [1,2,3,4,5]
```

In [6]:

```
print(id(l))
print(id(l2))
```

```
3082396235136
3082396142976
```

In [8]:

```
a=10
b=5
print(id(a))
print(id(b))
```

```
140713916770384
140713916770224
```

In [9]:

```
a='krishna'
```

In [14]:

```
b=a.replace("krishna","ram")
```

In [15]:

```
b
```

Out[15]:

```
'ram'
```

compersion operation

In [16]:

```
var =10  
print(bin(var))
```

```
0b1010
```

In [17]:

```
~var
```

Out[17]:

```
-11
```

string

In [1]:

```
var =" krishna"  
var1=" jaiswal"
```

In [5]:

```
str1="Weclome to  data scince Master "
```

In [4]:

```
str1[]
```

```
File "<ipython-input-4-1e07dfca845c>", line 1  
    str1[]  
      ^
```

SyntaxError: invalid syntax

string

In [16]:

```
name = "    Data science master    /    "
```

In [2]:

```
name.swapcase()
```

Out[2]:

```
'dATA SCIENCE MASTER '
```

In [3]:

```
name.title()
```

Out[3]:

```
'Data Science Master '
```

In [20]:

```
name.strip(" ")
```

Out[20]:

```
'Data science master  /'
```

In [21]:

```
name.rstrip(" ")
```

Out[21]:

```
'    Data science master    /'
```

In [22]:

```
name.lstrip(" ")
```

Out[22]:

```
'Data science master    /    '
```

In [23]:

```
name.replace("science", " ram")
```

Out[23]:

```
'    Data  ram master    /    '
```

In [30]:

```
' hello \t world '.expandtabs()
```

Out[30]:

```
' hello   world '
```

In [63]:

```
str1 ="wlecome to pwskills . to dat cience Master12345"
```

In [37]:

```
str1=str1.replace("dat","Data").replace("cience","Science")  
str1=str1.replace("cience","Science")
```

In [38]:

```
str1
```

Out[38]:

```
' wlecome to pwskills . to Data SSScience Master '
```

In [53]:

```
str1.endswith("er")
```

Out[53]:

```
True
```

In [57]:

```
str1.startswith("w")
```

Out[57]:

```
True
```

In [66]:

```
## check if all char in string are alphanumeric  
ram= "ram12345"
```

In [68]:

```
ram.isalnum()
```

Out[68]:

```
True
```

In [69]:

```
## to count the number of string
```

In [70]:

```
count = 0  
for i in str1 :  
    count =count+1
```

In [71]:

```
count
```

Out[71]:

47

In [72]:

```
len(str1)
```

Out[72]:

47

In [76]:

```
for i in range(len(str1)):
    print(i,"=",str1[i])
```

```
0 = w
1 = l
2 = e
3 = c
4 = o
5 = m
6 = e
7 = 
8 = t
9 = o
10 = 
11 = p
12 = w
13 = s
14 = k
15 = i
16 = l
17 = l
18 = s
19 = 
20 = .
21 = 
22 = t
23 = o
24 = 
25 = d
26 = a
27 = t
28 = 
29 = c
30 = i
31 = e
32 = n
33 = c
34 = e
35 = 
36 = M
37 = a
38 = s
39 = t
40 = e
41 = r
42 = 1
43 = 2
44 = 3
45 = 4
46 = 5
```

In [15]:

```
# we can use index to iterate string reverse direction
a= " krishna jaiswal is bad guys"
len(a)
```

Out[15]:

28

In [27]:

```
count =0
for i in range(len(a)-1,-1,-1):
    print(i,"=",a[i])
```

```
27 = s
26 = y
25 = u
24 = g
23 = 
22 = d
21 = a
20 = b
19 = 
18 = s
17 = i
16 = 
15 = l
14 = a
13 = w
12 = s
11 = i
10 = a
9 = j
8 = 
7 = a
6 = n
5 = h
4 = s
3 = i
2 = r
1 = k
0 =
```

In [29]:

```
range(len(a)-1,-1,-1)
```

Out[29]:

```
range(27, -1, -1)
```

In [30]:

```
name ="pwwskills"  
vowels =" AaEeIiOoUu"
```

In [37]:

```
for i in name:  
    if i in vowels:  
        print(i,f" this vowels :")  
    else:  
        print(i,"this is not vowels")
```

```
p this is not vowels  
w this is not vowels  
s this is not vowels  
k this is not vowels  
i  this vowels :  
l this is not vowels  
l this is not vowels  
s this is not vowels
```

list

In [40]:

```
type([])
```

Out[40]:

list

In [43]:

```
list1=["krishna ","jaiswal","axis", 19]
```

In [42]:

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

Out[42]:

```
[1, 2, 3, 4, 5]
```

In [47]:

```
list2=list1 +[["ram","shree krishna"]]
```

In [51]:

```
list2[-1][1]
```

Out[51]:

```
'shree krishna'
```

In [52]:

```
list2 *2
```

Out[52]:

```
['krishna ',  
'jaiswal',  
'axis',  
19,  
['ram', 'shree krishna'],  
'krishna ',  
'jaiswal',  
'axis',  
19,  
['ram', 'shree krishna']]
```

In [53]:

```
list2
```

Out[53]:

```
['krishna ', 'jaiswal', 'axis', 19, ['ram', 'shree krishna']]
```

In [59]:

```
#for i in list2:  
    #print(i)  
if "axis" in list2 :  
    print("present")
```

present

In [60]:

```
for i in list2:  
    if i=="axis":  
        print(i)
```

axis

In [61]:

```
## check element inside a list
```

In [63]:

```
list1= [1,2,3,4]  
4 in list1
```

Out[63]:

True

In [64]:

```
list1=["zebra","monkey","donkey","lion"]  
list2 = [5,6,2,4,5,3,2]
```

In [65]:

```
print(max(list1))
```

zebra

In [66]:

```
print(max(list2))
```

6

In [67]:

```
min(list1)
```

Out[67]:

'donkey'

In [68]:

```
min(list2)
```

Out[68]:

2

In [74]:

```
list2 = [5,6,2,4,5,3,2]
```

In [70]:

```
list2.append("krish")
```

In [71]:

```
list2
```

Out[71]:

[5, 6, 2, 4, 5, 3, 2, 'krish']

In [72]:

```
list2.clear()
```

In [76]:

```
list2.pop(4)
```

Out[76]:

5

In [77]:

```
list2
```

Out[77]:

```
[5, 6, 2, 4, 3, 2]
```

In [78]:

```
list2[::-1]
```

Out[78]:

```
[2, 3, 4, 2, 6, 5]
```

In [79]:

```
list2.reverse()
```

In [80]:

```
list2
```

Out[80]:

```
[2, 3, 4, 2, 6, 5]
```

In [81]:

```
list2.sort()
```

In [82]:

```
list2
```

Out[82]:

```
[2, 2, 3, 4, 5, 6]
```

nested list

In [84]:

```
list_1=[1,2,3]
list_2=[4,5,6]
list_3=[7,8,9]
matrix =[list_1,list_2,list_3]
```

In [85]:

```
matrix
```

Out[85]:

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
```

In [86]:

```
matrix[1][2]
```

Out[86]:

6

In [87]:

```
matrix[0][0],[1][1],[2],[2]
```

```
-----  
-  
IndexError                                Traceback (most recent call las  
t)  
<ipython-input-87-c314efc9eb14> in <module>  
----> 1 matrix[0][0],[1][1],[2],[2]
```

IndexError: list index out of range

In [89]:

```
[i for i in matrix]
```

Out[89]:

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
```

list comprehension

even number

In [90]:

```
[i for i in range(20)]
```

Out[90]:

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
```

In []:

In [92]:

```
[ i if i%2==0 else "odd" for i in range(20)]
```

Out[92]:

```
[0,  
'odd',  
2,  
'odd',  
4,  
'odd',  
6,  
'odd',  
8,  
'odd',  
10,  
'odd',  
12,  
'odd',  
14,  
'odd',  
16,  
'odd',  
18,  
'odd']
```

sum of even number and add number

In [1]:

```
list1 =[1,2,3,4,5,6,7,8]
```

In [7]:

```
even_sum=0  
odd_sum2=0  
for i in list1:  
    if i%2==0:  
        even_sum= even_sum+i  
        #print(f"sum of even number is: {even_sum}")  
    elif i%2==1:  
        odd_sum2=odd_sum2+i  
        #print(f"sum of odd number is:{odd_sum2}")
```

In [8]:

```
print(f"sum of even number is: {even_sum}")  
print(f"sum of odd number is:{odd_sum2}")
```

```
sum of even number is: 20  
sum of odd number is:16
```


In [2]:

```
list1 =[1,2,3,4,5,6,7,8]
```

In [11]:

```
sum_even=sum([ i for i in list1 if i%2==0])
```

Out[11]:

20

create a list of only the square of the a given list

In [2]:

```
list2=[1,2,3,4,5,6,7,8,9,10]
```

In [7]:

```
[i**2 for i in list2 ]
```

Out[7]:

[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

create a list of only the postive numbers from a given list

In [8]:

```
numbers =[-2,-1,0,1,2,3,4]
```

In [11]:

```
[ i for i in numbers if i>0]
```

Out[11]:

[1, 2, 3, 4]

create a list of only the first letter of words in a list

In [12]:

```
words = ['apple','banana ','cherry','date']
```

In [16]:

```
[word[0] for word in words]
```

Out[16]:

['a', 'b', 'c', 'd']

convert a list of temperatures from celsius to fahrenheit using list comprehension

In [17]:

```
celsius_temperatures =[0,10,20,30,40,50]
```

In [20]:

```
[(9/5)*temp+32 for temp in celsius_temperatures]
```

Out[20]:

```
[32.0, 50.0, 68.0, 86.0, 104.0, 122.0]
```

flatten a list of lists into a single list

In [21]:

```
lists= [[1,2,3],[4,5,6],[7,8,9]]
```

In [22]:

```
[list1 for list1 in lists]
```

Out[22]:

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
```

In [24]:

```
[num for sublist in lists for num in sublist]
```

Out[24]:

```
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```

In [25]:

```
## assignment  
## using both code and lists comprehension
```

In [31]:

```
numbers =[1,2,3,4,5,6,7,8,9,11,12]
```

In [35]:

```
# [ i for i in numbers if numbers%i==0 and numbers%1==0]
```

Create a list of all the possible combinations of 2 elements from a list

In [36]:

```
numbers =[1,2,3,4,5]
```

In []: