SHEET-1

In [1]:
```
#2. Perform Following on Python Shell Window
```

In [2]:
```
5**9
```

Out[2]:
```
1953125
```

In [3]:
```
3//2
```

Out[3]:
```
1
```

In [4]:
```
7//3
```

Out[4]:
```
2
```

In [5]:
```
7/3
```

Out[5]:
```
2.3333333333333335
```

In [6]:
```
6==6
```

Out[6]:
```
True
```

In [7]:
```
a = 20; a+= 30; a%=3; print(a)
```

```
2
```

In [8]:
```
True is False
```

Out[8]:
```
False
```

In [9]:
```
False in 'False'
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-9-2d8d452715a3> in <module>
----> 1 False in 'False'

TypeError: 'in <string>' requires string as left operand, not bool
```

In [10]:
```
((True == False) or (False > True)) and (False <= True)
```

Out[10]:
```
False
```

In [11]:
```
#3. Try to get following output from two python strings
```

In [12]:
```
s1="Nice to have it"
```

In [13]:
```
s2="here"
```

In [14]:
```
s1+" "+s2
```

Out[14]:
```
'Nice to have it here'
```

In [15]:
```
#4. Given this nested list, use indexing to grab the word "hello"
```

In [16]:
```
a = [1,2,[3,4],[5,[100,200,['hello']],23,11],1,7]
```

In [17]:
```
a[3][1][2]
```

Out[17]:
```
['hello']
```

In [18]:
```
#5. Try to insert above strings s1 and s2 in the list ‘a’ mentioned in que 4, in the beginning and end of it respectively
```

In [19]:
```
[s1]+a[0:]+[s2]
```

Out[19]:
```
['Nice to have it',
 1,
 2,
 [3, 4],
 [5, [100, 200, ['hello']], 23, 11],
 1,
 7,
 'here']
```

In [20]:
```
#6. Write a Python program to print all even numbers from a given numbers list in the same order and stop the printing if any numbers that come after 237 in t
```

In [21]:
```
numbers = [386, 462, 47, 418, 907, 344, 236, 375, 823, 566, 597, 978, 328, 615, 953, 345, 399, 162, 758, 219, 918, 237, 412, 566, 826, 248, 866, 950, 626, 949
742, 717, 958,743, 527]
```

In [22]:
```
for i in numbers:
    if i%2==0 and i<=237:
        print(i,end=" ")
```

```
236 162 104 58 24
```

In [23]:
```
#7. Write a Python program to print out a set containing all the colours from color_list_1 which are not present in color_list_2.
```

In [24]:
```
color_list_1 = set(["White", "Black", "Red"])
color_list_2 = set(["Red", "Green"])
```

In [25]:
```
l=set([])
for i in color_list_1:
    if i not in color_list_2:
        l.add(i)
```

In [26]:
```
l
```

Out[26]:
```
{'Black', 'White'}
```

In [27]:
```
#8. WAP to find if the given input string is Pangram or not
```

In [29]:
```
s=input("enter the sentence: ")
letter="abcdefghijklmnopqrstuvwxyz"
flag=1
for ch in letter:
    if ch not in s:
        flag=0
        break
if flag==0:
    print("not pangram")
else:
    print("pangram")
```

```
enter the sentence: hones is the bes policy
not pangram
```

In [30]:
```
#9. Write a Python program that accepts an integer (n) and computes the value of
```

In [31]:
```
n=int(input("enter the number: "))
a=(n+((n*10)+n)+((n*100)+(n*10)+n))
print(a)
```

```
enter the number: 5
615
```

In [32]:
```
#10. Write a python program to take input from console in following fashion 23 54 12#98 3 17 and generate the corresponding two list having integers inside (n
```

In [33]:
```
s=input("enter a fashion: ")
a,b=s.split("#")
x=a.split(" ")
y=b.split(" ")
l1=[]
l2=[]
for i in x:
    l1.append(int(i))
for j in y:
    l2.append(int(j))
print(l1)
print(l2)
```

```
enter a fashion: 23 54 12#98 3 17
[23, 54, 12]
[98, 3, 17]
```

In [34]:
```
#11. Write a program that accepts a comma separated sequence of words as input and prints the words in a comma-separated sequence after sorting them alphabeti
```

In [35]:
```
words=input("enter a comma seperated words: ").split(',')
words.sort()
s=','.join(words)
print(s)
```

```
enter a comma seperated words: without,hello,bag,world
bag,hello,without,world
```

In [36]:
```
#12. Write a Python function to find the name of person obtained highest marks in exam from given dictionary
```

In [37]:
```
d = {'Student':['Rahul', 'Kishore', 'Vidhya', 'Raakhi'], 'Marks':[57,87,67,79]}
l1=tuple(d['Marks'])
d['Marks'].sort()
print(d['Student'][l1.index(d['Marks'][-1])])
```

```
Kishore
```

In [38]:
```
#13. Write a program that accepts a sentence and calculate the number of letters and digits.Suppose the following input is supplied to the program:
```

In [39]:
```
s="hello world! 123"
LETTERS=DIGIT=0
for i in s:
    if i.isalpha():
        LETTERS+=1
    elif i.isdigit():
        DIGIT+=1
print('LETTERS {} \n DIGITS {}'.format(LETTERS,DIGIT))
```

```
LETTERS 10
 DIGITS 3
```

In [40]:
```
#14. Write a python function which creates a new dictionary of students from a given Dataset of various subject to a specific subject or topic only.Example Da
```

In [41]:
```
d = {'Name':['Akash','Soniya','Vishakha','Akshay','Rahul','Vikas'],'Subject':['Python','Java','Python','C','Python','Java'],'Ratings':[8.4, 7.8, 8, 9, 8.2, 5.
```

In [42]:
```
x=list(zip(d['Name'],d['Subject'],d['Ratings']))
y=[]
for i in range(len(x)):
    if x[i][1]=='Python':
        y.append(x[i])

l1=[]
l2=[]
l3=[]
for i in range(len(y)):
    l1.append(y[i][0])
    l2.append(y[i][1])
    l3.append(y[i][2])
d1={}
d1['Name']=l1
d1['Subject']=l2
d1['Ratings']=l3
print(d1)
```

```
{'Name': ['Akash', 'Vishakha', 'Rahul'], 'Subject': ['Python', 'Python', 'Python'], 'Ratings': [8.4, 8, 8.2]}
```

In [43]:
```
#15. Define a class with a generator which can iterate the numbers, which are divisible by 7, between a given range 0 and n.
```

In [44]:
```
def divby7(n):
    for i in range(1,n):
        if i%7==0:
            yield i
n=int(input())
k=divby7(n)
for i in k:
    print(i)
```

```
69
7
14
21
28
35
42
49
56
63
```

In [45]:
```
##16. A robot moves in a plane starting from the original point (0,0). The robot can move toward UP, DOWN, LEFT and RIGHT with a given steps. The trace of rob
#UP 5
#DOWN 3
#LEFT 3
#RIGHT 2
#The numbers after the direction are steps. Please write a program to compute the distance from current position after a sequence of movement and original poi
```

In [46]:
```
l=[['up',5],['down',3],['left',3],['right',2]]
x=l[0][1]-l[1][1]
y=l[2][1]-l[3][1]
d=int(((x**2)+(y**2))**(0.5)))
round(d)
print('the steps are {}'.format(d))
```

```
the steps are 2
```

In [ ]:
```

```

In [ ]:
```

```