(factorial) 3.python programs using fuctions

```
def fact1(a) :
  if(a!=1) :
    a=a*fact1(a-1)
  return(a)

fact1(5)
```

```
120
```

```
def fact2(a) :
  prod=1
  for a in range(a,1,-1) :
    prod=prod*a
  return(prod)
fact2(5)
```

```
120
```

(sum of all the even number)3.python programs using fuctions

```
def evensum(a) :
  sum = 0
  for i in range(0,a+1) :
    if (i%2==0) :
      sum+=i
  return sum
evensum(10)
```

```
30
```

(sum of squre of given number)3.python programs using fuctions

```
def sumsq(num) :
  x=[int(a) for a in str(num)]
  import functools
  z=functools.reduce(lambda m,n : (m**2)+(n**2) , x)
  return z

sumsq(17) #holds only for 2 digits
```

```
50
```

```
def sumsq2(num) :
  x=[int(a) for a in str(num)]
  res=0
  for i in x :
    res+=i**2
  return res

sumsq2(123)
```

```
14
```

"WAP for the following conditions

- Given the price of chocolates on 7days as 8,10,15,3,12,14,1
- You can buy and sell only once eg: buy on day1 and sell on day3 i.e buy=8, sell=15, profit=7
- Goal is to maximize profit"

```
prchoc = [8,10,15,3,12,14,1]
print("Min Price to Buy on Day #",prchoc.index(min(prchoc))+1,"at price ₹",min(prchoc))
print("Max Price to Sel on Day #",prchoc.index(max(prchoc))+1,"at price ₹",max(prchoc))
print("profit will be ₹",max(prchoc)-min(prchoc))
```

```
Min Price to Buy on Day # 7 at price ₹ 1
Max Price to Sel on Day # 3 at price ₹ 15
profit will be ₹ 14
```

python program for fabonacci of n numbers

```
def·fib(a):
··n_2=0
··n_1=1
··ls=[n_2,n_1]
··for·i·in·range(0,a-2):
····n=n_1+n_2
····n_2=n_1
····n_1=n
····ls.append(n)
··return·ls
fib(10)
```

```
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

python program to find substring

```
str="SWAROOP"
str.find('Z') #not found
```

```
-1
```

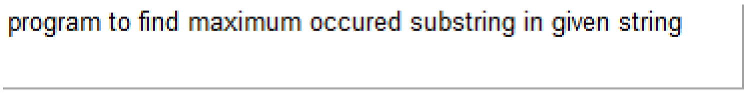
```python
str.find('A')
```

```
2
```

program to find pair of number which sum is 4 in a given list

```python
l=[0,1,2,3,4,5]
la=[]
for i in l :
  for j in l :
    if(i+j)==4 :
      la.append([i,j])
la
```

```
[[0, 4], [1, 3], [2, 2], [3, 1], [4, 0]]
```



program to find maximum occured substring in given string

```python
str1='DUSHYANTH'
from collections import Counter
countstr=Counter(str1)
countstr
```

```
Counter({'A': 1, 'D': 1, 'H': 2, 'N': 1, 'S': 1, 'T': 1, 'U': 1, 'Y': 1})
```

```python
for k,v in countstr.items() :
  if v==max(countstr.values()) :
    print(k , "repeating ",countstr[k],"times")
```

```
H repeating  2 times
```

1.WAP to find prime number

```python
n=9
prime=True
for i in range(2,n) :
  if (n%i==0) :
    prime=False
    break
prime
```

```
False
```

You have a list that contains a random number with a fibonacci number and your goal is to find a fibonacci triple sequence. For example: List = [55, 8, 34, 12, 17, 28, 5, 21] The out will be: 21, 34, 55

```
def fib(a):
  n_2=0
  n_1=1
  ls=[n_2,n_1]
  for i in range(0,a-2):
    n=n_1+n_2
    n_2=n_1
    n_1=n
    ls.append(n)
  return ls
fib(12)
```

```
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
```

```
List = [55, 8, 34, 12, 17, 28, 5, 21]
print(List)
List =sorted(List)
print(List)
fibtrip=[]
for i in List :
  for j in List :
    for k in List :
      if (i+j==k) :
        fibtrip.append([i,j,k])
fibtrip
```

```
[55, 8, 34, 12, 17, 28, 5, 21]
[5, 8, 12, 17, 21, 28, 34, 55]
[[5, 12, 17], [12, 5, 17], [17, 17, 34], [21, 34, 55], [34, 21, 55]]
```

```
fibtrip2=[]
for i in fibtrip :
  for j in range(2,len(i)) :
    if i[j-2] in fib(12) and i[j-1] in fib(12) and i[j] in fib(12) :
      fibtrip2=[i[j-2],i[j-1],i[j]]
sorted(fibtrip2)
```

```
[21, 34, 55]
```

+ Code + Text

✓ 0s completed at 8:13 AM ● ✕