

Function

```
In [2]: def baru():  
        print("venky")  
baru()
```

venky

```
In [4]: s=input("enter string")  
def reverse(s):  
    print(s[::-1])  
reverse(s)
```

enter stringbaru
urab

```
In [8]: n=int(input("enter number"))  
def square(n):  
    print(n*n)  
square(n)
```

enter number2
4

```
In [9]: a=int(input("enter number"))  
b=int(input("enter number"))  
def sum(a,b):  
    print(a+b)  
sum(a,b)
```

enter number3
enter number2
5

```
In [11]: n=input("enter number")  
len(n)
```

enter number435

Out[11]: 3

```
In [20]: s=input("enter string")  
def palindrome(s):  
    d=s[::-1]  
    if(d==s):  
        print("palondrome")  
    else:  
        print("no")  
palindrome(s)
```

enter stringlevel
palondrome

```
In [11]: # leap year
def leap(n):
    if n%400==0 or n%4==0 and n%100!=0:
        print("leap")
    else:
        print("not a leap")
leap(2014)
```

not a leap

```
In [14]: # Counting the nuber of digits
def count(n):
    return len(str(n))
count(44)
```

Out[14]: 2

```
In [15]: # Biggest of 4
def biggest(a,b,c,d):
    if a>b and a>c and a>d:
        print(a)
    elif b>a and b>c and b>d:
        print(b)
    elif c>a and c>b and c>d:
        print(c)
    else:
        print(d)
biggest(2,45,73,4)
```

73

```
In [19]: # Factorial with out recursion
def factorial(n):
    fact=1
    for i in range(2,n+1):
        fact*=i
    return fact
factorial(4)
```

Out[19]: 24

Types of Functions

```
In [26]: def adding():
    n1=10
    n2=12
    add=n1+n2
    print("with out arguments adn without return value",add)
adding()
```

with out arguments adn without return value 22

2) Without arguments and with return value

```
In [33]: def mul():
          m1=4
          m2=4
          mul=m1*m2
          return mul
          mul()
```

Out[33]: 16

3)With arguments without return value

```
In [37]: def mul(m1,m2):
          m1=4
          m2=4
          mul=m1*m2
          print("With arguments without return value",mul)
          mul(4,4)
```

With arguments without return value 16

4)with arguments and with return

```
In [16]: def add(a,b):
          sum=a+b
          return sum
          add(4,8)
```

Out[16]: 12

Recursive function

- A function call itself

```
In [16]: # function to calculate factorial of a number
def factorial(n):
    if n>1:
        return n*factorial(n-1)
    elif n==0:
        return 1
    else:
        return n
fact=factorial(-4)
print(fact)
```

-4

Tasks

Function to print all numbers divisible by 6 and not a factor of 100 in a given range(lb, ub) inclusive

```
In [1]: def divisible(lb,ub):  
        for i in range(lb,ub+1):  
            if i%6==0 and 100%i!=0:  
                print(i)  
        divisible(6,10)
```

6

Function to find the average of cubes of all the even numbers in a given range(lb, ub) inclusive

```
In [6]: def even(lb,ub):  
        s=0  
        j=0  
        for i in range(lb,ub+1):  
            if i%2==0:  
                c=i*i*i  
                s=s+c  
                j=j+1  
        print(s/j)  
        even(2,4)
```

36.0

Function to generate the list of factors for a given number

```
In [3]: def factors(n):  
        i=1  
        for i in range(1,n+1):  
            if n%i==0:  
                print(i)  
        factors(6)
```

1
2
3
6

Function to check if a given number is Prime by using recursive function

```
In [ ]: def prime(n):  
        i=1  
        c=0  
        while i<=n:  
            if n%i==0:  
                c=c+1  
        if c==2:  
            print("prime")  
        prime(3)
```

Function to calculate the average first N Prime numbers by using recursive function

In []:

Function to generate all Perfect numbers in a given range by using recursive function

In []:

Function to calculate the average of all factorials in a given range

```
In [21]: def factorial(n):
s=0
for i in range(1,n+1):
    fact=1
    for j in range(1,i+1):
        fact=fact*j
    s=s+fact
print(s/i)
factorial(4)
```

8.25

Function to generate N odd armstrong numbers

```
In [44]: def odd(n):
for j in range(1,n+1):
    q=j
    s=0
    while(j!=0):
        r=j%10
        m=r**3
        s=s+m
        j=j//10
    if(s==q):
        if(s%2!=0):
            print(q)
```

odd(160)

1
153

Function to generate Multiplication table for a number in a given range

10 in the range(100, 102) inclusive

```
In [4]: def mul(n):  
        for i in range(100,103):  
            print(n*i)  
mul(10)  
  
1000  
1010  
1020
```

```
In [ ]:
```