In [1]:
```
''' Ques 1 Calculate the sum of first N natural numbers and odd numbers. Use of a
formulae not allowed.'''''
for z in range(1,6): # creating a loop to generate 5 outputs

  N = float(input("Enter the Value of N "))
  def sum_function1(n):  # Sum of Natural numbers

      i =1
      sum_natural =0
      if N>=0 and N == int(N)  :
          while i <N+1:
                  sum_natural = sum_natural +i
                  i = i+1
          return sum_natural
      else:
         return("not valid , wrong value entered ")







  def sum_function2(n):      # Sum of Odd numbers

      p=1
      k =1
      sum_odd = 0
      if N>=0 and N == int(N) :

                while p <N+1:
                  sum_odd = sum_odd + k
                  k = k+2
                  p=p+1
                return sum_odd
      else:
               return("not valid , wrong value entered ")



  print(f"********************Output{z}******************")
  print(f"sum of first "+ str(N) +f" natural numbers is {sum_function1(N)}")
  print(f"sum of first "+ str(N) +f" odd numbers is {sum_function2(N)}")
```

```
********************Output1******************
sum of first 3.0 natural numbers is 6
sum of first 3.0 odd numbers is 9
********************Output2******************
sum of first 7.0 natural numbers is 28
sum of first 7.0 odd numbers is 49
********************Output3******************
sum of first 4.3 natural numbers is not valid , wrong value entered
sum of first 4.3 odd numbers is not valid , wrong value entered
********************Output4******************
sum of first -6.0 natural numbers is not valid , wrong value entered
sum of first -6.0 odd numbers is not valid , wrong value entered
********************Output5******************
```

```
sum of first 9.0 natural numbers is 45
sum of first 9.0 odd numbers is 81
```

In [73]:
```
''' Ques 2 Calculate the sum of N terms of an AP, GP and HP series for common dif
1.5 and common ratio 0.5. Use of analytical formulae not allowed.'''

for x in range(1,6):  # creating a loop to generate 5 outputs
    N , a = float(input("Enter the Value of N ")) , float(input("Enter the Value
    def sumAP(n,z):
          # first term of all series i
          i =1


          # i ,k , p, t  are all counters
          sum_AP= 0

          v=a


          # Sum of AP
          if N>=0 and N == int(N) :
                    for _ in range(int(N)):
                      sum_AP = sum_AP +v
                      v = v+ 1.5

                    return(print(f"********Output{x} ********** \n Sum of first '
          else:

                    print(f"********Output{x} ********** \n Entered value of N is
          # sum of GP
    def sumGP(n,z):
          p=1
          sum_GP = 0
          if N>=0 and N == int(N) :
                    while p <N+1:
                      sum_GP=  sum_GP+ z
                      z = z*0.5
                      p=p+1

                    return(print(f"Sum of first "+ str(int(N))+ f" GP terms start

          else:
                    print("Entered value of N is not  valid ")

              # Sum of HP
    def sumHP(n,z):
          t=0
          g= 0
          sum_harmonic = 0
          if N>=0 and N == int(N) and a!=0 :
                    while t<N:
                      g= 1/(1/(a)+1.5*t)
                      sum_harmonic = sum_harmonic + g
                      t = t+1

                    return(print(f"Sum of first "+ str(int(N))+ f" HP terms start
          else:
                    print("Either the entered value of N is not valid or the firs
```

```
    sumAP(x,x)
    sumGP(x,x)
    sumHP(x,x)
```

```
********Output1 **********
 Sum of first 5 AP terms starting with -1.0 is 10.0
Sum of first 5 GP terms starting with -1.0 is 1.9375
Sum of first 5 HP terms starting with -1.0 is 1.9857142857142855
********Output2 **********
 Sum of first 5 AP terms starting with 0.0 is 15.0
Sum of first 5 GP terms starting with 0.0 is 3.875
Either the entered value of N is not valid or the first term was entered 0 whic
h is also not valid for HP because 1/0 case is encountered
********Output3 **********
 Sum of first 4 AP terms starting with 3.2 is 21.8
Sum of first 4 GP terms starting with 3.2 is 5.625
Sum of first 4 HP terms starting with 3.2 is 4.261403138176073
********Output4 **********
 Entered value of N is not  valid
Entered value of N is not  valid
Either the entered value of N is not valid or the first term was entered 0 whic
h is also not valid for HP because 1/0 case is encountered
********Output5 **********
 Sum of first 4 AP terms starting with 9.0 is 45.0
Sum of first 4 GP terms starting with 9.0 is 9.375
Sum of first 4 HP terms starting with 9.0 is 10.158985696480503
```

In [7]:
```
''' Ques 3 Calculate  factorial'''
for x in range(1,6): # Loop to create 5 outputs
    N = float(input("Enter the Value of N "))
    def factorial(n):

        # first term of all series i
        i =1
        fact = 1
        # Sum of GP
        if N>0 and N == int(N) :
                  while i <N+1:
                    fact=  fact*i
                    i=i+1
                  return(print(f"**********output{x}******** \nFactorial of  "+ s
        elif N == 0 :
                  return(print(f"**********output{x}******** \nFactorial of  "+ s

        else:
                  return(print(f"**********output{x}******** \n Entered value is

    factorial(x)
```

```
**********output1********
Factorial of  4  is 24
**********output2********
Factorial of  7  is 5040
**********output3********
 Entered value is not a valid
**********output4********
Factorial of  0  is 1
**********output5********
 Entered value is not a valid
```

In [34]:
```
# Question 4

import matplotlib.pyplot as plt
import math
```

Type *Markdown* and LaTeX: α^2

In [66]:
```
def factorial12(N):
    if N==0:
        return(1)
    elif N<0:
        print("Not applicable")
    else:
        return(N*factorial12(N-1))
```

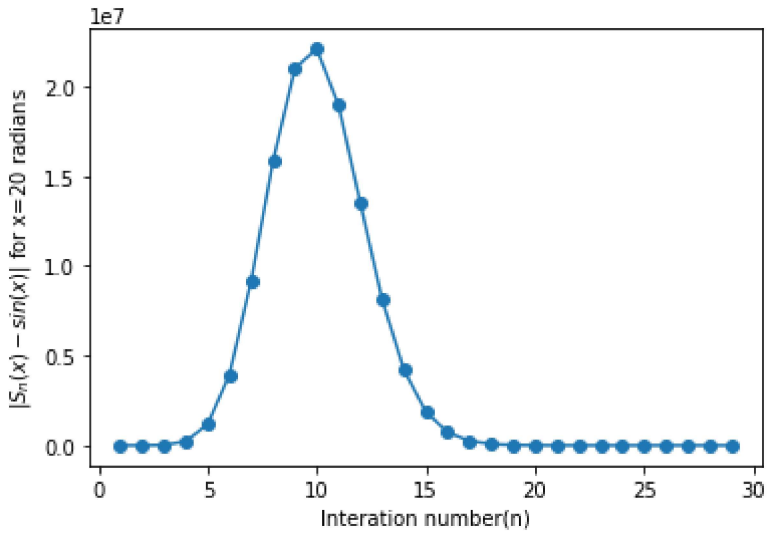
In [60]:
```
#sine of x series taking n terms of the taylor series
def S_n(x,n):
    val=0
    for i in range(n):
        val=val+(-1)**i*x**(2*i+1)/(factorial12(2*i+1))
    return(val)
```

In [61]:
```
#sine of x rounded off upto n decimal places using the taylor series expansion

def sine(x,n):
    #m is the number of terms to be taken to minimise the error, we increase m ti
    m=0
    while abs(math.sin(x)-S_n(x,m))>10**(-n-1):
        m+=1
    return(S_n(x,m))
```

In [62]:
```
x=[i for i in range(1,30)]
y=[abs(S_n(20,j)-math.sin(20)) for j in range(1,30)]
plt.plot(x,y,'-o')
plt.xlabel("Interation number(n)")
plt.ylabel("$|S_n(x)-sin(x)|$ for x=20 radians")
```

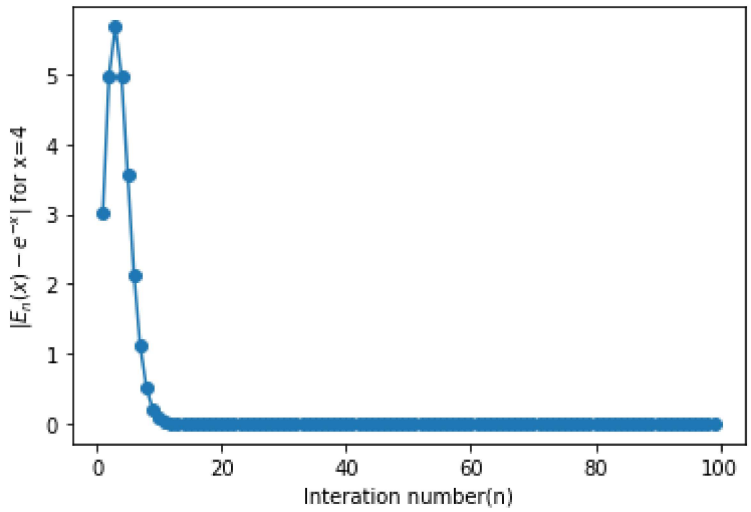
Out[62]: Text(0, 0.5, '$|S_n(x)-sin(x)|$ for x=20 radians')



In [65]:
```
#Taylor series sum of exp(-x) function taking n terms
def E_n(x,n):
    val=0
    for i in range(n+1):
        val=val+(-x)**(i)/factorial12(i)
    return(val)

#Exp(-x) function correct upto to n decimal places using taylor series
#e.g.,if x=1.7 exp(-1.7) will be evaluated
def expo(x,n):
    m=0
    while abs(math.exp(-x)-E_n(x,m))>10**(-n-1):
        m+=1
    return(E_n(x,m))
x=[i for i in range(1,100)]
y=[abs(E_n(4,j)-math.exp(-4)) for j in range(1,100)]
plt.xlabel("Interation number(n)")
plt.ylabel("$|E_n(x)-e^{-x}|$ for x=4")
plt.plot(x,y,'-o')
```

Out[65]: [<matplotlib.lines.Line2D at 0x268f556c3a0>]



In [ ]: