1. Given an integer n, write a program to return the sum of squares of first n natural numbers in integer format.
   Note1:
   The formula for the sum of squares of first n natural numbers is: n(n+1)(2n+1)/6
   Note2:
   You need not take input in this problem, only implement the function provided.
   Problem Constraints:
   `1 <= N <= 100`

2. Considering the Interest() function:
   ```
   def Interest(p,c,t=2,r=0.09):
    return p*t*r
   ```
   Which of the following function calls are legal?

   a. Interest(p=1000,c=5)
   b. Interest(r=0.05,5000,3)
   c. Interest(500,t=2,r=0.05)
   d. Interest(c=4,r=0.12,p=5000)

   3. What will be the o/p of the code:

   ```
        x = "Hello"
   def greet():
       global x
       print(x, end=",")
       x = "Bye"
       print(x, end=",")

   greet()
   print(x, end="")
   ```

   4. Write a program to print all Natural numbers from 1 to N and from N to 1 inclusive, where N is given as an argument.
   Note1: First print from 1 to N then in new line print again from N to 1
   Note2: *You need not take input in this problem, you need to only implement the function provided.*
   Note3: There is 1 line gap between two output lines and a space between numbers in same line.

5. What does the following code print?

```
x = 55
def hi():
 return x
x +=10
print(hi())
```

6. Write a function to check whether a given number n as an input to the function is a perfect number or not. If the given integer is a perfect number return 1 else return 0.
Note: In number theory, a perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself (also known as its aliquot sum).

7. You are given a positive integer r denoting the radius of a sphere as a parameter. Write a program to calculate the volume of the sphere. The volume of a sphere having radius R is given by $(4 * \pi * R_3) / 3$.
NOTE: Return the volume of the sphere up to two decimal places. You can use round().
NOTE2: Use pi as 22/7 (not *math.pi*).