1.1 Write a Python Program(with class concepts) to find the area of the triangle using the below formula.

area = (s*(s-a)*(s-b)*(s-c))* 0.5

Function to take the length of the sides of triangle from user should be defined in the parent class and function to calculate the area should be defined in subclass.

In [139]:
```
class Object:

    def __init__(self, a, b, c):
        self._a = a
        self._b = b
        self._c = c

class Triangle(Object):

    def __init__(self, *ar):
        super().__init__(*ar)

    def compute_area(self):
        s = (self._a + self._b + self._c)/2
        return (s*(s-self._a)*(s-self._b)*(s-self._c))**0.5

area = Triangle(24, 30, 18)
print("Area of the triangle is : ", area.compute_area())
```

```
Area of the triangle is :  216.0
```

1.2 Write a function filter_long_words() that takes a list of words and an integer n and returns the list of words that are longer than n

In [140]:
```
def find_long_words(e, n):
    if(len(e) > n):
        return e

def filter_long_words(func, list, n):
    l = []
    for e in list:
        if(func(e, n)):
            l.append(e)
    return l

a = ["Apple","Elephant", "Aeroplane"]
target_length = 5

print("Using user defined function : ", filter_long_words(find_long_words, a, targe

print("Using pre-defined function : ", list(filter(lambda x: len(x) > target_length
```

```
Using user defined function :  ['Elephant', 'Aeroplane']
Using pre-defined function :  ['Elephant', 'Aeroplane']
```

2.1 Write a Python program using function concept that maps list of words into a list of integers representing the lengths of the corresponding words.

Hint: If a list [ab,cde,erty] is passed on to the python function output should come as [2,3,4]

Here 2,3 and 4 are the lengths of the words in the list.

In [141]:
```
def mymapfunc(func, list):
    l = []
    for e in list:
        l.append(func(e))
    return l

print("Using user defined map : ", list(mymapfunc(len, ["ab", "cde", "erty", "hello
print("Using pre-defined map : ", list(map(len, ["ab", "cde", "erty"])))
```

```
Using user defined map :  [2, 3, 4, 5]
Using pre-defined map :  [2, 3, 4]
```

2.2 Write a Python function which takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise.

In [144]:
```
volwes = ['a', 'e', 'i', 'o', 'u']

def is_ovwel(char):
    if(len(char) > 1):
        return "Invalid operation!! More than one character supplied"
    return char.lower() in volwes

print(is_ovwel('I'))
print(is_ovwel('X'))
print(is_ovwel('ABCD'))
```

```
True
False
Invalid operation!! More than one character supplied
```