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 [11]:

```
class Triangle:
    def __init__(self, side1, side2, side3):
      self.side1 = side1
      self.side2 = side2
      self.side3 = side3
      print ("Initialize the Triangle super class is [" +  str(side1) + "," + str
```

In [12]:

```
class Triangle_Utilities(Triangle):
    def __init__(self, side1, side2, side3):
       print ("Initialize Triangle Utilities Child class" )
       super(Triangle_Utilities, self).__init__(side1, side2, side3)

    def get_area(self):
      s = (self.side1 + self.side2 + self.side3)/2
      print (str(s))
      return (s*(s-self.side1)*(s-self.side2)*(s-self.side3))**0.5

instance = Triangle_Utilities(3,4,5)

print ("Area of the triangle is " + str(instance.get_area()) )
```

```
Initialize Triangle Utilities Child class
Initialize the Triangle super class is [3,4,5]
6.0
Area of the triangle is 6.0
```

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 [13]:

```
class list_Utilities:
    def __init__(self, wordlist):
      self.wordlist = wordlist
      print ("Initialize list_Utilities")

    def filter_long_words(self, n):
      return list(filter(lambda x:len(x) > n, self.wordlist))

instance = list_Utilities(['Investment', 'is', 'a', 'science,', 'not', 'an', 'art

print ("List of Words of Length greater than 2: " + str(instance.filter_long_word
print ("List of Words of length greater than 3: " + str(instance.filter_long_word
```

```
Initialize list_Utilities
List of Words of Length greater than 2: ['Investment', 'science,', 'not', 'ar
t']
List of Words of length greater than 3: ['Investment', 'science,']
```

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

In [14]:

```
wordlist = ['Investment', 'is', 'a', 'science,', 'not', 'an', 'art']
def wordlength(wordlist):
  return list(map(lambda x: len(x), wordlist))

print ("words list : " + str(wordlist))
print ("length of words :" + str(wordlength(wordlist)))
```

```
words list : ['Investment', 'is', 'a', 'science,', 'not', 'an', 'art']
length of words :[10, 2, 1, 8, 3, 2, 3]
```

4 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 [15]:

```
def vowel_check(char):
  if(char == 'a' or char == 'e' or char == 'i' or char == 'o' or char == 'u'):
    return True
  else:
    return False

# Take user input
char = input("Enter the character: ");

# If Invalid input, exit
if (char.isalpha() == False):
 exit();

# Invoke function
if (vowel_check(char)):
  print(char, "is a vowel.");
else:
 print(char, "is not a vowel.");
```

```
Enter the character: x
x is not a vowel.
```

In [ ]: