2.2. Write an appropriate docstring for the following function

```python
def mystery(n):
    
    total = 0
    for i in range(n):
        # Roll a 6-sided die (with sides 1-6)
        total = total + randint(1, 6)
    return total
```

2.3. What value should you assign to the variable \( n \) to so that \( \text{total} \) has the specified value after this code executes. The value for \( n \) must be an integer that results in valid Python code (i.e., not cause an error).

```python
total = 0
for i in range(2):
    for j in range(n):
        total = total + 2
```

- **a)** total is 4
- **b)** total is 12

2.1. Write a function named `area` that has four parameters \( x_1, y_1, x_2, y_2 \), (representing the \( x,y \) coordinates of two points) and returns the area between the line connecting those points and the \( x \)-axis (shaded area below). You can assume \( x_2 > x_1, y_1>0 \) and \( y_2>0 \). Recall that the area of a trapezoid is \( \frac{1}{2}(base_1 + base_2) \cdot height \). You do not need to include comments or docstrings.
Numeric Operators
+ , - , / , * : Addition, subtraction, division, multiplication
//: Floor division: Round division result down to nearest whole number
%: Modulus: Evaluate to remainder of division
- Range
range(stop): Equivalent to range(0, stop, 1)
range(start, stop[, step]): Create sequence of integers from inclusive start to exclusive stop by step

Strings
- The following functions are built-in
  len(string): Returns the number of characters in the string
  int(string), float(string): Converts numeric string to int or float
  str(object): Converts object, e.g. int or float, to a string
- String operators
  string1 + string2: Returns a new string that is the concatenation of string1 and string2
  string * int: Returns a new string that is string repeated int times

Modules
- turtle module
  forward(dist), backward(dist): Move the turtle forward/backward by the length dist. Doesn’t change heading.
  right(angle) left(angle): Turn the turtle right/left by angle (in degrees)
  goto(x, y): Move turtle to position x, y
  setheading(angle): Set the turtles heading to angle
  circle(radius): Draw a circle with specified radius; the center is radius units left of the turtle
  dot(size): Draw a filled circle with diameter size centered on current position of the turtle
  penup(): Pull the pen up – no drawing when moving
  pendown(): Put the pen down – drawing when moving
  fillcolor(color): Change the fill color to color, where color is a string
  begin_fill(), end_fill(): Start and end filling shapes with fill color
- random module
  randint(a, b): Return a random integer N such that a ≤ N ≤ b
  uniform(a, b): Return a random floating point number N such that a ≤ N ≤ b
- math module
  sqrt(num): Return the square root of num