Week 02
Dealing with User Input

Lab 02 - Square Root

In class we learned about using **`cout`** and **`cin`** to communicate with the user. For today's lab I want you to write a program that asks the user to input a number and then you return the square root. I expect your program to have the following:

- A descriptive header
- Include the **`<iostream>`** and the **`<cmath>`** libraries.
- Make use of **`cout`** and **`cin`**
- Make sure your main function ends with **`return 0;`**
- Make sure your code compiles correctly and without issue
- Make sure you name your file **`lastname_lab02.cpp`**

Notes:
You'll be using the **`sqrt()`** function from the **`<cmath>`** library to figure out the square root. Store the result of the **`sqrt()`** function in a **`double`** type variable.

Example Input:

```
4
```

Expected Example Output:

```
Please enter in a number: 4
The square root of 4 is: 2
```

Homework 02 - Pythagorean Theorem

You now know how to deal with square roots and arithmetic between variables. Let's put it to the test and make a program that can calculate the length of the third side of a triangle given the first two sides. I expect your program to have the following:

- A descriptive header
- Include the **<iostream>** and the **<cmath>** libraries.
- Make use of **cout** and **cin**
- Make sure your main function ends with **return 0;**
- Make sure your code compiles correctly and without issue
- Make sure you name your file **lastname_hw02.cpp**
- Only submit one file, if you did the bonus, submit that.

Notes:
If you don't remember what Pythagorean theorem is, it's:

$$a^2 + b^2 = c^2$$

where a is the first side, b is the second side, and the answer you return to the user is c.

Example Input:

```
a = 3
b = 4
```

Expected Example Output:

```
Enter in a: 3
Enter in b: 4
c is: 5
```

Example Input 2:

```
a = 0.5
b = 1.2
```

Expected Example Output 2:

```
Enter in a: 0.5
Enter in b: 1.2
c is: 1.3
```

Bonus Objectives:

1) To get a squared and b squared, you probably just multiplied them by themselves. You can however use the **pow()** function, also apart of the **<cmath>** library. You can use it like so:

   ```
   pow(a,2);
   ```

   Try to use the **pow()** function instead of **a * a** and **b * b**.