Rectangle Calculator 5.0

Create a program that calculates the perimeter and area of a rectangle only when both width and height provided by the user are positive.

Note: **Bold** words are output while non-bold words are input in the following console samples.

Console Sample 1 (valid input)

```
Rectangle Calculator 5.0

Enter the height: .1
Enter the width: .2

Perimeter: .6
Area: .02
```

Console Sample 2 (invalid height)

```
Rectangle Calculator 5.0

Enter the height: -.1
Enter the width: .2

Height and width must be both positive!
```

Console Sample 3 (invalid width)

```
Rectangle Calculator 5.0

Enter the height: .1
Enter the width: .0

Height and width must be both positive!
```

Console Sample 4 (invalid height and width)

```
Rectangle Calculator 5.0

Enter the height: .0
Enter the width: -.2

Height and width must be both positive!
```

Specifications

- You have to define a class *Rectangle* with two private data members corresponding to height and width.
- Declare and define a non-default constructor which requires two parameters corresponding to the private data members for the class *Rectangle*.
- Declare and define a getter and a setter for each private member in the class *Rectangle*.

- Declare and define a public member function *get_perimeter()* in the class *Rectangle* to calculate the perimeter and return the result. The formula for calculating the perimeter of a rectangle is:

  ```
  perimeter = 2 * (height + width)
  ```

- Declare and define a public member function *get_area()* in the class *Rectangle* to calculate the area and return the result. The formula for calculating the area of a rectangle is:

  ```
  area = height * width
  ```

- Declare and define a public member function *display_results()* in the class *Rectangle* to display all the calculation results in the console as shown in the Console Sample 1 above by calling the other member functions *get_perimeter()* and *get_area()*.
- In the *main()*, after displaying the title, you must first check the validity of the input of both height and width provided by the user. If they are both positive, then create a *Rectangle* object based on them and then call the member function *display_results()* on the object. Otherwise, display an error message as shown in the Console Sample 2, 3 or 4.
- The program should accept decimal entries like 35.5 and 14.25.
- There is no requirement of precision for the output.