CIS 251 • C++ Programming
Project 3: Classes and Objects

A commercial property management company maintains a file containing information about each property it manages. Each line of the file, named `properties.txt`, includes the property's identification number (six characters), tenant name (with underscores between words if necessary), the area of the property in square feet, and the monthly rent for the property. The following is an example of how a typical property's information appears in the file:

```
SV0017 Valleydale_Drugs 10152 125407.89
```

The company wants to process this data in an organized manner. Define a class named `RetailProp` that contains the following as `private` members:

- The identification number (a `string` member variable)
- The tenant name (a `string` member variable)
- The area (an `int` member variable)
- The monthly rent (a `double` member variable)

The class should also contain the following as `public` members:

- A constructor that accepts no arguments; it should assign `"??????"` to the identification number member variable, `"(vacant)"` to the tenant name member variable, and zero to each of the numeric member variables
- An accessor member function for each member variable that returns the value of the member variable (four member functions)
- A mutator member function for each member variable that assigns the argument to the member variable (four member functions); assume no validation is required for regular credit
- A member function named `getRentPerSqFoot` that returns the monthly rent per square foot, which is the monthly rent divided by the area (do not store this value in a member variable; it should be calculated and returned by the member function itself)
- A member function named `input` that accepts an input stream and does not return a value; it should read the first value from the input stream into the identification number member variable, the second into the tenant member variable, the third into the area member variable, and the fourth into the monthly rent member variable
- A member function named `output` that accepts an output stream and does not return a value; it should send the property's identification number, tenant name, area, monthly rent, and rent per square foot with labels to the output stream in a manner similar to the following:

  ```
  Property #SV0017
  Occupied by Valleydale_Drugs
  Area:  10152 sq. ft.
  Rent:  $125407.89 ($12.35 per sq. ft.)
  ```

  The rent amounts should appear in fixed notation with two digits after the decimal point.

Only the prototypes for the member functions should appear in the class definition itself; the function definitions should appear separately.

In the `main` function, the program should open the input file, displaying an error message and terminating if the input file does not exist. The program should create five `RetailProp` objects. It should call the `input` member function using each of the `RetailProp` objects in series (assume that the file contains information for at least five properties; the program is not required to read more than the first five sets of values). Finally, it should call the `output` member function using each of the `RetailProp` objects in series to display the information for each property on the screen (there is no file output in this program).

POINTS: 75

BONUS (5 points): In the `output` member function, if the identification number member variable contains a value of `"??????"`, the program should send an error message to the output stream indicating that the information for that property has yet to be provided; it should not send the values of the member variables to the output stream.

BONUS (5 points): When assigning a new value to the tenant name member variable (in the mutator or the `input` member function), the program should change any underscores in the argument to regular spaces.

BONUS (5 points): Each mutator member function should validate the argument to ensure that it is a legal value for the member variable being changed; if the argument is invalid, the program should display an error message and leave the member variable as-is. Define the `input` member function so that it uses the mutator member functions to validate the input before it is assigned to each member variable. Use the following criteria for the validation:

- The identification number should be exactly six characters
- No validation is required for the tenant name
- The area should be no less than 50 square feet (the size of a kiosk)
- The rent should be no less than $500

No bonus credit will be given unless the standard requirements of the assignment are met! Make sure that you have the basic functionality working regardless of whether or not you are able to complete the bonus portion successfully!

Your code should include comments at the top listing your name, the course, the project number, and a brief description of the program. It should also include **at least five inline comments** in your code explaining what your code is doing at that given point.

No late work will be accepted! The deadline will be strictly enforced by Blackboard!

If you need assistance with your project code, e-mail your instructor **at least one day in advance** of the deadline. E-mail messages sent too close to the deadline may not receive a response in adequate time to make corrections to the project.

You may discuss general details of the project with your classmates, but **do not look at or borrow another student's project code!** In addition, **do not post the details of the project or ask for assistance with specifics of this project on any public discussion board, forum, or newsgroup!**