CITP 190 – Introduction to Java

Project 6

This project requires you to create one project. The project will store e-mail addresses and phone numbers. Please be sure to download and unzip the project start files in the `Project6Start.zip` file. See the instructions for adding these files to your project. The instructions are located in the course Web site in the Project6 folder under the Projects folder.

To receive full credit for this project you must submit the following:

- A detailed design diagram for the `main()` method and for each method in the AddressBookEntry and Validator classes. Note: Each method will be a separate diagram.
- Class diagrams for the AddressBookEntry and Validator classes.
- The source code for each each class (the .java file) following the coding standards.
- The bytecode for each class (the .class file)
- Proof of the correctness of your output using the test data provided. Please note that you must provide proof for all test data. You may provide additional test data.
- A capture of the output of the program. The output must show all data from your proof.
- Your final `address_book.txt` file.

Submit all files as one (1) ZIP file to the Project 6 Drop Box in the course site.

Grading:

Program design	5 points
Design diagrams and class diagrams	5 points
Following course standards	5 points
Proof	3 points
Screen captures	2 points

Important notes:

1. Incorrect calculations will result in a 0 grade for this project.
2. Output that is not presented as shown (including spaces and spelling) will result in a 0 grade for this project.
3. Code that does not follow the standards (with the exception of the AddressBookIO code) will result in a 0 grade for this project. Since you did not write the code for the AddressBookIO class and you must not change the code for this class, it is okay if this code is not formatted to standards.
4. Changing the AddressBookIO source file will result in a 0 grade for this project.
5. Uploading more than one zip file will result in a 0 grade for this project.
6. Not using all the test data provided will result in a 0 grade for this project.
7. Not using the `address_book.txt` file that was provided will result in a 0 grade for this project. If you are using the file correctly, when the user selects menu option 1 four address book entries will be displayed even though the user has not entered any data.
8. Using a `continue` statement or misusing a `break` statement will result in a 0 grade for this project.
9. Not including the `address_book.txt` file will result in a 0 grade for this project.
10. Using material from beyond Chapter 6 will result in a 0 grade for this project. Do not attempt to validate the format of the email address or phone number.

Project 6: Store email addresses and phone numbers

Console

```
Welcome to the Address Book application

1 - List entries
2 - Add entry
3 - Exit

Enter menu number: 1

Name                Email               Phone
------------------  ------------------  ------------------
Bill Gates          bill@microsoft.com  (111) 222-3333
Larry Ellison       larry@sun.com       (444) 555-6666
Steve Jobs          steve@apple.com     777-888-9999

1 - List entries
2 - Add entry
3 - Exit

Enter menu number: 2

Enter name: Mike Murach
Enter email address: mike@murach.com
Enter phone number: 800-221-5528

This entry has been saved.

1 - List entries
2 - Add entry
3 - Exit

Enter menu number: 1

Name                Email               Phone
------------------  ------------------  ------------------
Bill Gates          bill@microsoft.com  (111) 222-3333
Larry Ellison       larry@sun.com       (444) 555-6666
Steve Jobs          steve@apple.com     777-888-9999
Mike Murach         mike@murach.com     800-221-5528

1 - List entries
2 - Add entry
3 - Exit

Enter menu number: 3

Goodbye.
```

Operation

- If the user selects the first menu option, the application displays the email addresses and phone numbers that have been saved. Then, it displays the menu again.
- If the user selects the second menu option, the application prompts the user to enter a name, email address, and phone number. Then, it displays the menu again.
- If the user selects the third menu option, the application exits.

Specifications

- Use the AddressBookIO class that's provided to get and save entries. This class contains two static methods that you can use to read and write data to the `address_book.txt` file. They are:

  ```
  // get a String that displays all entries in columns
  public static String getEntriesString()
  ```

  ```
  // save an AddressBookEntry object to the file
  public static boolean saveEntry(AddressBookEntry entry)
  ```

  If necessary, you can open the `address_book.txt` file in a text editor to debug this application.

- The AddressBookIO class uses the AddressBookEntry class. Create a class named AddressBookEntry to store the data for each entry. This class must contain these instance variables (and no other instance variables):

  ```
  private String name;
  private String emailAddress;
  private String phoneNumber;
  ```

  And it must contain a constructor, get methods for each of the instance variables, and set methods for each of the instance variables. The AddressBookIO class expects the AddressBookEntry class to contain the following get methods:

  ```
      getName()
      getEmailAddress()
      getPhoneNumber()
  ```

  Your AddressBookEntry class must define these as the get methods.

- Create a class named AddressBookEntryApp which contains your `main()` method. This class should display the menu and respond to the user's menu choices using the AddressBookEntry and AddressBookIO classes as necessary.
- Create a class named Validator that contains static methods that can be used to validate the data in this application. The user should only be able to select one of the menu choices, and the user must enter some text for name, email address, and phone number. Do not attempt to validate the format of the email address or phone number. We have not yet covered that material.

Test Data

Menu Option	Name	e-mail	Phone
1			
2	Brent Knight	President@lcc.edu	517-555-1851
2	Walt Disney	disneyw@disney.com	(567) 555-4896
1			

Note: Include the following in your proof:

- For option 1, include what would be displayed on the screen in the appropriate format (spaces and spelling). You will need to look at the contents of the `address_book.txt` file to determine what will be displayed the first time the user chooses option 1.
- For option 2, include what the `addess_book.txt` file will contain. You must include the entire file for each set of test data. Fields in the file are separated by a tab.