

Homework 6-1: Maintain a list of countries

Console

```
Welcome to the Countries Maintenance application

1 - List countries
2 - Add a country
3 - Exit

Enter menu number: 1

India
Japan
Mexico
Spain
United States

1 - List countries
2 - Add a country
3 - Exit

Enter menu number: 2

Enter country: Thailand

This country has been saved.

1 - List countries
2 - Add a country
3 - Exit

Enter menu number: 1

India
Japan
Mexico
Spain
United States
Thailand

1 - List countries
2 - Add a country
3 - Exit

Enter menu number: 3

Goodbye.
```

Operation

- The application begins by displaying a menu with three choices.
- If the user chooses the first menu item, the application displays a list of countries that are saved in a file.
- If the user chooses the second menu item, the application prompts the user to enter a country and then it writes that country to the file of countries.
- If the user chooses the third menu item, the application displays a goodbye message and exits.

Specifications

- Create a class named `CountriesTextFile` that contains one method that allows you to read a list of countries from a file and another method that allows you to write a list of countries to a file. For example:

```
public ArrayList<String> getCountries()  
public boolean saveCountries(ArrayList<String> countries)
```

- Store the list of countries in a text file named `countries.txt` in the same directory as the `CountriesTextFile` class. If the `countries.txt` file doesn't exist, the `CountriesTextFile` class should create it. This class should use buffered I/O streams, and it should close all I/O streams when they're no longer needed.
- Create a class named `CountriesApp` that displays the menu and responds to the user's choices.
- Use the `Validator` class presented in chapter 6 or an enhanced version of it to validate the user's entries. A valid integer is required for the menu choice, and a non-empty string is required for the country.

Enhancements

- Create another class named `CountriesBinaryFile` that can store the list of countries in a binary data file named `countries.dat`. This class should contain a main method that initializes the file for the first time by writing several countries to it. Then, modify the `CountriesApp` class so it uses the `CountriesBinaryFile` class instead of the `CountriesTextFile` class.
- Modify the `CountriesApp` class so it includes a menu choice that allows the user to delete a country from the file.

Homework 6-2: Maintain a conversions table

Console

```
Welcome to the Length Converter

1 - Convert a length
2 - Add a type of conversion
3 - Delete a type of conversion
4 - Exit

Enter menu number: 1

1 - Miles to Kilometers: 1.6093
2 - Kilometers to Miles: 0.6214
3 - Inches to Centimeters: 2.54

Enter conversion number: 2

Enter Kilometers: 10
10.0 Kilometers = 6.214 Miles

1 - Convert a length
2 - Add a type of conversion
3 - Delete a type of conversion
4 - Exit

Enter menu number: 2

Enter 'From' unit: Centimeters
Enter 'To' unit: Inches
Enter the conversion ratio: .3937

This entry has been saved.

1 - Convert a length
2 - Add a type of conversion
3 - Delete a type of conversion
4 - Exit

Enter menu number: 1

1 - Miles to Kilometers: 1.6093
2 - Kilometers to Miles: 0.6214
3 - Inches to Centimeters: 2.54
4 - Centimeters to Inches: 0.3937

Enter conversion number: 4

Enter Centimeters: 2.54
2.54 Centimeters = 1 Inches

1 - Convert a length
2 - Add a type of conversion
3 - Delete a type of conversion
4 - Exit

Enter menu number: 4

Goodbye.
```

Operation

- This application begins by displaying a main menu with four items: (1) Convert a length, (2) Add a type of conversion, (3) Delete a type of conversion, and (4) Exit.
- If the user chooses the first main menu item, the application displays a menu of possible conversions. After the user selects a conversion, the application prompts the user to enter a unit of measurement, calculates the conversion, displays the result, and displays the main menu again.
- If the user chooses the second main menu item, the application prompts the user to enter the values for a new conversion, saves this new conversion to a file, and displays a message to the user.
- If the user chooses the third main menu item, the application displays a menu of possible conversions. After the user selects the conversion, the application deletes that conversion from the file, displays a message to the user, and displays the main menu again.
- If the user chooses the fourth main menu item, the application displays a goodbye message and exits.

Specifications

- Create a class named `Conversion` that can store information about a conversion, including `fromUnit`, `fromValue`, `toUnit`, `toValue`, and `conversionRatio`. This class should also contain the methods that perform the conversion calculations and return the results as a formatted string.
- Create a class named `ConversionsTextFile` that contains one method that reads an array list of `Conversion` objects from a file and another that writes an array list of `Conversion` objects to a file. For example:

```
public ArrayList<Conversion> getConversions()  
  
public boolean saveConversions(ArrayList<Conversion> typesList)
```

- Store the list of conversions in a text file named `conversion_types.txt` in the same directory as the `ConversionsTextFile` class. If the `conversion_types.txt` file doesn't exist, the `ConversionsTextFile` class should create it. This class should use buffered I/O streams, and it should close all I/O streams when they're no longer needed.
- Create a class named `ConversionsApp` that displays the menus shown in the console output and responds to the user's choices.
- Use the `Validator` class or a variation of it to validate the user's entries. A valid integer is required for a menu choice, non-empty strings are required for the "From" and "To" fields, and a valid double is required for the conversion ratio.

Homework 6-3: List artists and albums

Console

```
Artist and Album Listing

Artists
-----
The Beatles
Elvis Presley
John Prine

Albums
-----
Rubber Soul
Revolver
Sgt. Pepper's Lonely Hearts Club Band
The White Album
Elvis at Sun
Elvis 30 #1 Hits
John Prine
Sweet Revenge

Artists and Albums
-----
The Beatles
    Rubber Soul
    Revolver
    Sgt. Pepper's Lonely Hearts Club Band
    The White Album
Elvis Presley
    Elvis at Sun
    Elvis 30 #1 Hits
John Prine
    John Prine
    Sweet Revenge
```

Operation

- This application reads an XML file and displays a list of artists, albums, and albums by artist.

Specifications

- Create a class named MusicArtistsApp that reads an XML file named music_artists.xml and displays a list like that shown in the console output above.
- The music_artists.xml is provided for you and contains the data shown above.