Exercises for Chapter 14 (Week 8)

Introductory Programming 2020

Exercises

14.11

Save a copy of the **address-book-v1t** project to work on under another name. Make changes to the removeDetails method to avoid a NullPointerException arising if the key value does not have a corresponding entry in the address book. Use an if-statement. If the key is not valid, then the method should do nothing.

14.13

Are there any other methods in the AddressBook class that are vulnerable to similar errors? If so, try to correct them in your copy of the project. Is it acceptable in all cases for the method simply to do nothing if its parameter values are inappropriate? Do the errors need reporting in some way? If so, how would you do it, and would it be the same way for each error?

14.26

Review all of the methods of the AddressBook class and decide whether there are any additional situations in which they should throw an IllegalArgumentException. Add the necessary checks and throw statements.

14.29

The **address-book-v3t** project includes some throwing of unchecked exceptions if parameter values are null. The project source code also includes the checked exception class NoMatchingDetailsException, which is currently unused. Modify the removeDetails method of AddressBook so that it throws this exception if its key parameter is not a key that is in use. Add an exception handler to the remove method of AddressBookTextInterface to catch and report occurrences of this exception.

14.30

Make use of NoMatchingDetailsException in the changeDetails method of AddressBook. Enhance the user interface so that the details of an existing entry may be changed. Catch and report exceptions in AddressBookTextInterface that arise from use of a key that does not match any existing entry.

14.32

Enhance the try statements you wrote as solutions to Exercises 14.29 and 14.30 so that they handle checked and unchecked exceptions in different catch blocks.

14.34

In the **address-book-v3t** project, define a new checked exception class: DuplicateKeyException. This should be thrown by the addDetails method if either of the non-blank key fields of its actual parameter is already currently in use. The exception class should store details of the offending key(s). Make any further changes to the user interface class that are necessary to catch and report the exception.