David Andalcio

Software Testing and Quality Assurance

Assignment 9 Coverage

5/30/17

Additional Classes and changes

1. For Orders, changed the state of some of the tests to have for the 6% sales tax to include each form for the three states. (Line 14, 18, 22)
2. Added tests that used the 6% sales tax to include the rest of the forms of the states. (Line 45, 47)
3. Created and used an Orders object so it didn’t go right to the methods in Orders. (Line 50, 51)
4. In calculateTotal, the second parameter is Orders.ShippingMethod.valueOf(“NextDay”) so all of ShippingMethod enum is used. (Line 51)
5. For StringUtil, created and used StringUtil object so it didn’t bypass the constructor and go right to the methods. (Line 79, 81, 83, 89, 90, 94)
6. Tested null value to make sure it returned null. (Line 81)
7. Put an “!” in the word so it would throw an IllegalArgumentException. (Line 83)
8. The word is “equipment” so it would throw the same since it doesn’t change since it is uncountable. (Line 89)
9. The word is “person” so it would return “people” since it is irregular. (Line 90)
10. The word is “bay” to test a word that ended with a vowel and a “y” to return -s or -ies. (Line 91)
11. Called the main method on the object to include the main method in the coverage. (Line 94)

Defects

1. In Orders, Illinois is spelled wrong in the program. It has “Illinios”.
2. Also, it the state abbreviation isn’t a real state, it doesn’t say so. It will still accept it.
3. In StringUtil, when the word ends with an “x”, it returns with an “en”. I used “box” and return “boxen” rather than “boxes”.
4. If the word isn’t a noun, it still treats it as a noun. I used “very” and it still returned “very”.
5. If the word has numbers, it will still accept the word like the previous defect.
6. When a word ends with a vowel followed by a “y”, it always returns with a -ies. Entered bay, and got baies rather than bay.

Discussion

1. StringUtil can’t have complete branch coverage because when it tests for a vowel with a “y”, it always returns the word with -ies. It never reaches the return where the word would end with “s”. This is because if c2 comes to false on the first condition, it will then return true on the second. Since it is all Or, only one true is needed. For example, if c2 is “a”, it returns false on the first condition but is true for the second on. If c2 isn’t “a”, then it returns true on the first condition. Therefore, testing will never reach the other conditions and no more branches will be made. All other coverages and for the two classes can be achieved. I believe my test suite is adequate because it considers different outcomes from the orders or the words that can be placed. Also, it shows where there are defects in the classes like the non-existent states or common mistakes like entering a number into word by accident.