Homework 4 Grade Sheet

Name: Kyle James

|  |  |  |  |
| --- | --- | --- | --- |
| **Part** | **Criteria** | **Points** | **Earned** |
| **Code** | ItemCategory Enum appropriately defined  Item Class   * Is appropriately abstract * Contains 2 non-enum fields * Also contains an ItemCategory field * Contains appropriate properties * Has an abstract Use() method * Has an overridden ToString()   Item Sub Classes   * At least 3 Sub Classes * Each contains an appropriate extra field * Implements the Use() method appropriately   Inventory Class   * Contains correct fields * Count property * Add method * RemoveRandomItem method * RemoveRandomItem(ItemCategory) method * Exceptions are thrown appropriately   Main Method   * Program compiles and runs correctly * Exceptions are handled appropriately | 4  3  3  2  4  3  3  9  9  9  2  1  2  4  7  5  10  10 | 4  3  3  2  4  3  3  9  9  9  2  1  2  4  7  5  10  9 |
| **General** | * Follows C# coding standards for commenting, indentation and naming | **10** | 10 |
|  | Total | **100** | 99 |

***Comments:***

Solid work!

One piece of major feedback:

Using the catch-all 'Exception e' does work, but isn't very nuanced - using a more specific exception type provides more flexibility to the developer and makes the program more readable by knowing what to expect at a glance. In this case, catching exception types such as "InvalidOperationException" or "ArgumentNullException" would be ideal.