Code Review Defect List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reviewer: \_Dustin Howarth\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | GH Repo: \_\_\_dghowart\_review\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
|  | |  | | |
|  | |  | | |
| **ID #** | **Location** | | **Problem Description** | | Problem | | |
| **File and Line Number** | | **Category** | **Severity** | |
| 01 | PowerSet.java Line 80 | | The method powerSetImplementation2 does not correctly throw a NullPointerException when a null array is passed as an argument. | | FD | MJ | |
| 02 | PowerSetData.java Line 26 | | The initializeArray method incorrectly attempts to split (using the split method) the passed string with the regex “\s” which is not a valid regex. | | FD | BR | |
| 03 | Item.java Line 2 | | The Item class does not have a class banner comment present. | | CG | LOW | |
| 04 | PowerSet.java Line 127 | | The { bracket does not appear on the correct line (not at the end of the previous line) | | CG | LOW | |
| 05 | RandomNumber.java (Whole Class Issue) | | This class generates a random number, though it does not differ in functionality from Math.random or java.util.Random (copied code/unnecessary class) | | CS | LOW | |
| 06 | PowerSetData.java Line 26 | | The array String RESULT is in all caps (name) and its values stay constant, but it is never declared as a constant. | | CG | LOW | |
| 07 | PowerSet.java Line 29 | | The powerSetImplementation1 is supposed to return the power set of the passed array but is set to null. | | FD | BR | |
| 08 | PowerSet.java Line 35 | | Int N is too short of an identifier and does not give any meaningful description of the variable (there are comments however N is used throughout the program so its still bad practice) | | CS | LOW | |
|  |  | |  | |  |  | |
|  |  | |  | |  |  | |
|  |  | |  | |  |  | |
|  |  | |  | |  |  | |
|  |  | |  | |  |  | |
|  |  | |  | |  |  | |

Category: **CS –** Code Smell defect**. CG –** Violation of a coding guideline. Provide the guideline number. **FD** – Functional defect. Code will not produce the expected result. **MD –** Miscellaneous defect, for all other defects.

Severity: **BR** - Blocker, must be fixed asap. **MJ** – Major, of high importance but not a Blocker **LOW** – Low.