**BCPR301 Assignment Two Version Messages**

**Note 1:** The initial version for each version is the last one in the previous version, e.g. version 2.5 is the initial version of 3.1.

**Version One**

1. Initial upload of program as a base for refactoring. 100% test coverage of program

This version no bad smells were fixed. It was the last version from assignment one with 100% unit-test coverage added.

**Version Two**

1. Extract the class “PickleData”, out of the Controller class and place in a new file
2. Extract the class “Save”, out of the Controller class and place in a new file
3. Extract the class “Load”, out of the Controller class and place in a new file
4. Extract the class “Display”, out of the Controller class and place in a new file
5. Extract the class “Validate”, out of the Controller class and place in a new file

The bad smell of a large class has successfully been removed as the number of logical lines of code in my Controller class has significantly been reduced. This did not bring any new bad smells into the program once the refactoring was done. My program now has a much higher software quality as the Controller class and the extracted classes have their own purpose rather than one big class having multiple purposes.

**Version Three**

1. Create the display\_data() method to choose which sub class to use for displaying data
2. Create the builder method and setup the base class and its sub classes
3. Fill the sub classes display\_data() methods so they each have their own individual purpose

The switch case bad smell was successfully removed at the end, as I was able to remove the multiple conditional statements that were being used to decide how the data was going to be displayed. Doing this did not bring any new bad smells into the program that I have spotted. My software is now at a higher quality as I have been able to remove my duplicate code and are now able to add different display types by adding new sub classes.

**Version Four**

1. Extract the check\_valid\_data() method from the check\_data() method
2. Extract check\_item\_validity() method from the check\_valid\_data() method
3. Extract check\_date() method from check\_item\_validity() method
4. Extract valid\_date() method from check\_date() method

**Version Five**

1. Create the load\_data() method to choose which sub class to use for displaying data
2. Create the builder method and setup the base class and its sub classes
3. Fill the sub classes load\_data() methods so they each have their own individual purpose