

Andrew Sychtysz

Professor Nagpal

Software Process Management CS-348-01

6 November 2020

## Assignment #4

Issue #	Location		Issue Description	Clean Code Chapter # and Practice
	File Name	Line #		
1	Conversion.java	2,5,11,12	First thing I realized was that my class comments, method comments, were still a little bit confusing. Almost as if they were useless comments. I tried to reword them to make them more informative to someone reviewing the code or someone who isn't that familiar with the language.	Ch. 4 <b>Informative Comments</b>
2	Conversion.java	14 it begins, throughout the entire method it is replaced.	The reason why I believed this was a good fix is because the previous variable being named "roman", not only leaves a question in the air to it potentially being the converted string but also the	Ch. 3: <b>Function arguments</b>  When the variable was named "roman" it made it seem like that was the return result in the method rather than the parameter which is the unconverted version of the user input.

			fact that is now easier to follow.	
3	Conversion.java	15,16, and all throughout the method	<p>Changed String result variable to String convertedResult, and Integer currentNum variable to originalUserInput.</p> <p>Going back to this code on my own, I had to investigate to find out what the currentNum variable was doing. Once I realized, I figured that changing it's variable name will make this a hump when reviewing this code no more. Now, I feel I will understand how the code works instantly since the variable's changed name. As well as convertedResult, because when it was just result, I had to investigate differences in my code.</p>	<p><b>Ch. 2. Use Intention-Revealing Names &amp; Ch. 3. Verbs and Keywords</b></p> <p>These definitely have more intention revealing names than before. Now you understand that currentNum was just a place holder to retain the originalUserInput for the method's <code>System.out.println("");</code> statement.</p> <p>As well as changing "roman" to userInput, changing roman a noun to a verb "userInput" which requires an action to create that variable.</p>
4	Conversion.java, Project4.java	Conversion: Line 14. Project4: Line 22.	Now, if I wanted to further develop this code, the Conversion class doesn't seem like it's stuck to one method. Now it would make sense to have multiple types of	<b>Ch. 2. Method Names</b>

			<p>conversions. I could make methods that convert integer variables to binary or hexadecimal. It seems like the conversion class has much more purpose with a simple method name change. Which has expanded the possibilities of the code much more.</p>	
5	Conversion.java	Line 120.	<p>A little more explanation towards the method's return statement. Show's that it is not simply returning a variable but a statement containing that variable.</p>	Ch. 4 <b>Informative comments.</b>
6	Project4.java, Conversion.java	<p>Project4.java:Line 16, 20, 23, 26, 30, 33</p> <p>Conversion.java: Line 120.</p>	<p>I added these lines to the output that space out the user output a little more. It makes the program feel a lot more interactive for the user. They know what section they are working on as they work their way through the program. It reads more like a newspaper now. From top to</p>	Ch. 5. <b>Newspaper metaphor &amp; Ch. 5 Vertical Openness Between Concepts.</b>

			bottom left to right.	
--	--	--	--------------------------	--