1. A description (2-3 paragraphs) of what you learned from this project and how it relates to some of the topics covered in lecture
   1. We concentrate on refactoring in this project. Refactoring is only altering the structure of the code, not the meaning. We need an automation to check all the structure of the project before and after refactoring. It’s the reason why we use Junit in this project
   2. Refactoring is used to improve the code structure, to get rid of bad design smells. We have to change many line of code when we and to change a small part of the design to preserve its observable behavior.
2. A description (2-3 paragraphs) of what you liked about Eclipse’s support for refactoring including its strengths and limitations as well as your impression of JDeodorant
   1. Eclipse strengths
      1. We can update many lines of code that relate to our new changes automatically
      2. Eclipse saves us from a lot of mutual labor when comparing to the simple find all and replace method
      3. There is no chance for human error
   2. Eclipse limitations
      1. Eclipse doesn’t support “Change Method Signature refactoring” operation very well. We have to fix the error on our own
   3. My impression of JDeodorant
      1. It identifies design problems and attempts to suggest possible refactorings to the source code.
      2. JDeodorant UI is quite easy to use and understandable
3. A description (1-2 paragraphs) of how/why unit tests are important when doing refactoring
   1. We change code structure while doing refactoring, we need a quick method to be sure that the project observable behavior is preserved