# Assignment 2: Refactoring

This assignment provides you an opportunity to explore the refactoring techniques. It will also help you practice working with Github.

PART1. Creating a repository (10pts)

1. Please login to your Github account and go to <https://github.com/SIU-CS>
2. Use the “+ New repository” link to create a new repository for your project and suffix it with you last name. For example, if you are working on the chess project, name it as *chess\_lastname*.
3. Create an empty directory named {*project\_lastname}* in your computer. If you are on windows, you will also need to download and install Git (<https://git-scm.com/downloads> )
4. In the command prompt or terminal navigate to the {*project\_lastname*} directory. Map the local directory with remote Git

git remote add origin [https://github.com/amiangshu/project\_lastname.git]

1. Stage the file, README.md

git add README.md

1. Commit the file

git commit -m "Adding readme.md file"

1. Push your changes to the remote (i.e., Github)

git push -u origin master

1. Login to Github to verify that your changes are stored.

PART2: Repository for the assignment (20pts)

1. Download the source code for your assigned project (see Projects.xlsx for your project). Extract the source code inside the *{project\_lastname}* folder.
2. Open eclipse. Create a new java project using the “From existing source code” option. Use the *{project\_lastname}* folder for the project.
3. Verify that there is no compilation error and you can run the project.
4. From eclipse use the “Team->Share project” option to let eclipse know that this project will be shared using Git.
5. From eclipse, use “Team->Commit” option to add the project files and commit the changes.
6. Use “Team-> Push to Upstream” option to push your committed changes to Github.
7. Login to github and verify that the project files are added.

PART3: Refactor existing code changes (45pts)

1. Use “Team->Switch To-> New branch” to create a new branch named “development”. In this assignment, you will be working on this branch. Push this branch to Github (Team->Push branch ‘development’)
2. Please refer to Lecture04-Refactoring to identify at least 5 code smells in the project.
3. Apart from the lectures, following web resources can be helpful to identify code smells:
   1. <https://sourcemaking.com/refactoring>
   2. <http://refactoring.com/>
   3. <https://blog.codinghorror.com/code-smells/>
4. Refactor the source code to remove those code smells. Verify that you have not introduced any errors and project runs correctly.
5. Commit and push your changes to Github.

PART4: Assignment report (25pts)

In this report you have to describe following 5 attributes for each of the code smell:

* 1. Code smell type
  2. Source class name and location
  3. Why it was bad?
  4. How did you eliminated the code smell?
  5. How did you verify the correctness of your refactoring?

Save the project report as {project\_lastname\_report}. You can use doc, docx, or pdf. Submit the report in D2L.

Deadline and Grading

* **You have to finish all the four parts by Thursday, September 22, 2016 11:00PM**
* **You have to submit the assignment report on D2L.**
* **The assignment must be your own work. Please do not cheat.**
* **Please always use descriptive and meaningful commit messages.**