

"Codebase Cleanup" Assignment

Prerequisites:

- [Software Maintenance and Quality Control](#), including all linked notes
- ["Code Simplification" Exercise](#)
- ["Testing, 1,2,3" Exercise](#)
- ["CI, 1,2,3" Exercise](#)
- [Code Climate](#)

Learning Objectives

- Practice simplifying and refactoring code to improve readability and quality and decrease maintenance costs.
- Practice writing and executing automated tests to verify your program behaves as expected, and incorporating automated testing into your development workflow.
- Practice incorporating GitHub Pull Requests into your development workflow.
- Practice running automated tests on a Continuous Integration server, and incorporating CI checks into your development workflow.
- Practice incorporating peer reviews and automated quality checks into your development workflow.

Instructions

Revisit your previous projects and tackle their respective "testing challenges":

- [Shopping Cart Project - Testing Challenges](#)
- [Robo Advisor Project - Testing Challenges](#)

Start developing on a new branch called something like "cleanup" or "qc", and incorporate [Git Branch operations](#) and GitHub Pull Requests and code self-reviews into your development workflow. Optionally ask a friend to review your PR and make a comment, or add them as a collaborator on your repository and they'll be able to conduct an official "review".

Also configure your application's tests to run on a platform like Travis CI, and incorporate Continuous Integration (CI) into your workflow.

Also configure your repository to integrate with a platform like Code Climate, and incorporate automated code quality checks into your workflow (BONUS).

Examples

See the following examples of fully-tested projects:

- [Shopping Cart Tests](#)
- [Robo Advisor Tests](#)

Evaluation

Each of the two projects will comprise half of the weight for each of the categories below:

Category	Description	Weight
Pull Request Workflow	At least one or two Pull Requests used during development	20%
Simplification	Current state of the code is reasonably well organized and simplified	20%
Documentation	Most if not all functions are documented via "docstring" comments	20%
Automated Testing	Presence of at least a few tests which directly invoke functions from the source code to verify the program behaves as expected	30%
Continuous Integration	Tests have been run and passed on a CI server at least once, ideally as part of the Pull Request workflow	10%
Code Quality	The Code Climate service has been integrated into the Pull Request workflow to provide automated code quality checks	10% (BONUS)
Peer Reviews	A peer has provided an unofficial comment or official review as part of the Pull Request workflow	0% (OPTIONAL)

NOTE: you'll be evaluated according to the rubric above, not any previous functionality requirements. Instructors will be reading your code and running your tests and inspecting your GitHub Pull Request history, but not necessarily running your application code directly.