Code Review Guidelines

Upon completing a course project, you are welcome to request a code review in the lesson's discussion forum. Code reviews are a great way to receive valuable feedback and improve your code.

Guidelines

* Some projects will also specify optional 'bonus features' that you are encouraged to implement. You may choose to request a code review either before or after completing the 'bonus features' portion of a project, but we are only able to support one code review per project.
* Sometimes students will request that we look at code again after corrections or updates after the initial code review. While this is ok, we will not do another code review and only take a quick look.
* We cannot perform code reviews on material external to Launch School. We also cannot do code reviews on materials that's within Launch School but not assignments, such as exercises or optional content. The exercises and optional content is for you to gain more practice, and we can't perform code reviews on those.
* "Language Check" requests: we cannot do reviews of language checks. This typically happens more in 109 and 129 when students are starting to study for the assessment and want to verify if the way they're describing code is appropriate. Unfortunately, we cannot review these, as it would be akin to grading an essay before submission of the essay.

Tips for Success

* If you are struggling to debug a particular piece of code or understand a course concept, don't wait until you are submitting a code review of your whole project to ask for help-- go ahead and create a post for that issue in the lesson forum. This allows us to focus in on your question and address it specifically. It also makes it easier for other students with the same question to find your post.
* Asking good questions is essential to mastery based learning; good questions should be specific. If you are having trouble debugging a piece of code, include a link to the GitHub repository or gist where we can view your code. Identify the issue with as much precision as possible and provide details about how you have tested the problem already. If you are using Ruby, for example, Pry, IRB, and the [Ruby documentation](http://ruby-doc.org/) are helpful tools for debugging and testing programming concepts.
* Reading code is an important skill. Feel free to spend time reading through other code reviews in the lesson forums.
* Before submitting a code review request for a project, be sure that your program runs without error and is as close to a 'final product' as possible. This helps our staff give you focused and useful feedback. If you are using Ruby, run your code through Rubocop and do your best to address any complaints.