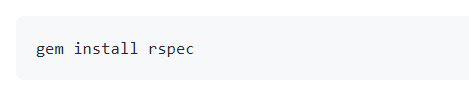
**TASK4 program testing with RSpec**

So for this task I chose to test my rectangle program. I also chose RSpec to test my program, and I chose RSpec because I was kind of familiar with the syntax coming from javascript testing with jest and enzyme.

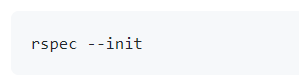
RSpec is a testing tool for for Ruby applications. It is considered a behavior driven development framework. The tests look at a ruby object or class, and describes what that object should be doing. It is intended to be very intuitive, like the rest of ruby. It's used in a lot of ruby and rails enterprise level applications.

[Here’s a link for RSpec documentation.](http://rspec.info/documentation/)

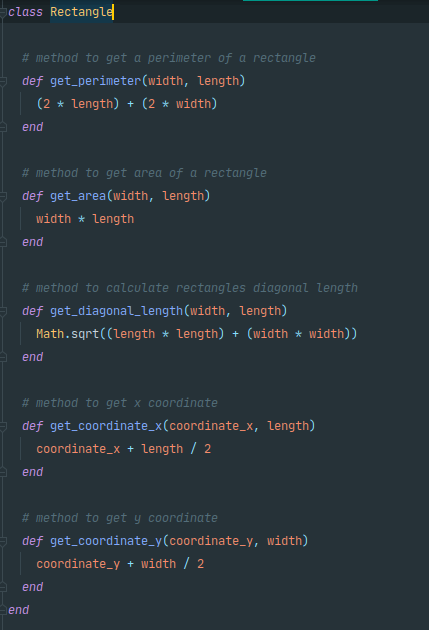
To get started with RSpec first we need to install rspec:



Then to set it up in a project, all we need to do is initialize it with this command:



Doing this task I’ve learnt that I made a few mistakes in TASK3 and corrected them in this task. First for it to work, I removed the constructor in the Rectangle class, so rectangle class looked something like this now:



And I also fixed my object when creating it in calculations file. So I fixed it from this:



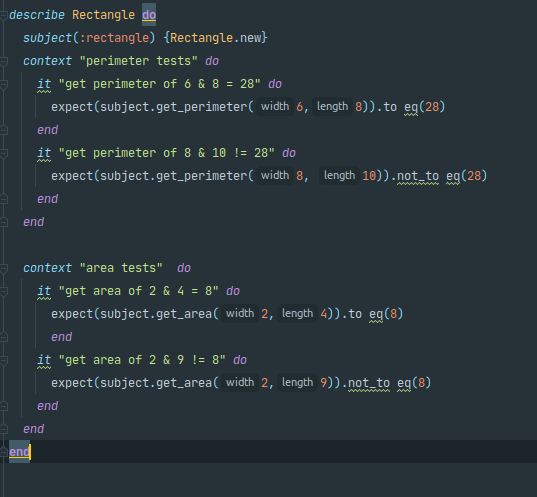
To this:



So in my test file I had to import rspec and the class file I want to test which I’ve done right here:

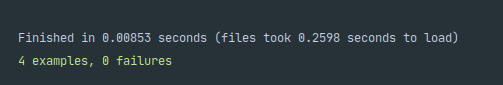


Now I got to write some tests.



I could have written let instead of subject and then it test cases I would have just written the name of it. But I found out that even better method is to write a subject. A subject is basically another version of let. The only difference is that you can only have one subject, and it’s meant to be an instance of the main object you’re testing.

When running these tests I get the output that I pass them:



But what if we want to fail some of them to check it really works?

What I’m going to do for a test to fail, is that I’m going to just change up a few numbers. What I did is change perimeter numbers from 6 to 7 and it still expects for it to be equal to 28, but in fact it’s not equal to 28 anymore, it’s equal to 30.

