Ruby on Rails Sessions - Week 1 Supplementary Homework

In order to make sure we don't lose too much time, I'm asking each of you to go through the following exercises. These would have been done in the week 1 session as individual exercises, and we will go through them as a group in our next in-person session.
If you have any trouble or questions please feel free to contact me.

Setting up Ruby and Rails

- Go to the Rails Installer website and download the latest package.
  - The setup is one step for Windows, but a little more complex for Mac.
  - Essentially, this will get Ruby installed, set up the Rails libraries, and install a Git command line tool.
    - Note, the Git tool that gets installed is a little out of date. It has a TLS problem and won't actually connect to Github. We will have to install a newer version together.
  - In our next class we will cover all of the various things this actually did for you.
    - Note that there is now a RailsInstaller group on your start menu
- Verify your setup using the "Command Prompt with Ruby and Rails" which is on your start menu.
  - When you start the command prompt you should see some software versions specified
    - Ruby 2.3.2
    - Rails 5.1.3
    - Git 2.20 (I updated mine for the TLS problem, yours might be a little older)
  - Run 'ruby -v'
  - Run 'rails -v'
  - Run 'gem list'
  - If all of that worked, you probably have a working setup

Make sure you have a Github account

- In the future we will fork some repositories for our projects, but for now just ensure you have on
- Mine is williadd

Make sure you have a development environment set up

- I will use a combination of Brackets, RubyMine, a command prompt, and the Git Bash terminal.
  - Please use whatever you find most effective, there is no specific reason I use the tools listed above.
  - When I started using Git I stopped using the version controls built into the IDE. This is mostly because IDEs did not always have good support for Git when it came out, and they tried to map older version control ideas to Git.

- I keep all of my source code in C:\workspace. All of the examples will be in project folders under that main directory.

Hello World in Ruby

- Create a project called HelloWorldRuby
    - Remember, I'm starting in c:\workspace
- Create a Ruby source file named 'hello.rb'
    - RB is the ruby source code file extension
- Add 1 line to the file
    - puts 'hello'
- Open the Command Prompt with Ruby and Rails
    - Cd to the HelloWorldRuby folder
    - Run the app using 'ruby hello.rb'
    - You should see the app print 'hello'

Hello World in Rails

- Go to your workspace folder
- Create a basic rails app with 'rails new HelloWorldRails'
    - This may take a few minutes, it needs to install many gems (ruby libraries, we'll cover that a little later)
    - If there is any errors you probably won't be able to run the hello world app
- Run the new Rails app
    - Cd to Hello World Rails
    - Run 'ruby bin\rails about' to get some info about the app you just created
        - Notice the version of Rails updated to 5.1.6
    - Run the app using 'ruby bin\rails server'
        - The app is running in a Puma web server
        - Go to http://localhost:3000 to view the app, you should get to the default Rails welcome page
- Shut down Rails using Ctrl-C

Install some other gems we'll need

- Quick note about installing gems. If there is an error read it carefully. Sometimes it's just an error installing the documentation for a gem.
- Unit testing
    - gem install rspec
- Code quality
    - gem install excellent
    - gem install reek
    - gem install flay