



Courses > 310 Build Robust and Production Quality Applications > Week 7 > Lesson: Transactions and test database setup

# Lesson: Transactions and test database setup

If you look at the instructions for capybara here: <https://github.com/jnicklas/capybara>

In the "Transactions and database setup" section, it talks about the Selenium driver needs to run against a real HTTP server and that capybara will start one for you but it runs on a different thread. In this case using "transactions" as the strategy to reset the database after each spec can be a problem, and this is where you want to use the "database\_cleaner" gem so that you can use the "truncation" strategy, which effectively just goes to your database and delete all the records. It's a bit slower than transaction rollbacks, but in this case guarantees data will be reset.

You can add the "database\_cleaner" gem in your test group (for the setup below, make sure to use version 2.7.0), and use the following configuration in your `spec_helper` file inside of the `RSpec.configure do |config| ... end` block.

```
1  config.before(:suite) do
2    DatabaseCleaner.clean_with(:truncation)
3  end
4
5  config.before(:each) do
6    DatabaseCleaner.strategy = :truncation
7  end
8
9  config.before(:each, :js => true) do
10   DatabaseCleaner.strategy = :truncation
11 end
12
13 config.before(:each) do
14   DatabaseCleaner.start
15 end
16
17 config.after(:each) do
18   DatabaseCleaner.clean
19 end
```

You marked this topic or exercise as completed.