Additional Resources

FactoryGirl

FactoryGirl is a thoughtbot gem that makes it easy to generate data for your tests.

```
# spec/factories.rb
FactoryGirl.define do
  sequence(:email) {|n| "person#{n}@example.com" }

  factory :post do
    user
  end

  factory :user do
    name "John Doe"
    email
  end
end
```

With factories defined, you can easily instantiate Active Record objects or save them to the database.

```
describe Post do
  let(:user) { FactoryGirl.create(:user, name: "Alexander Graham") }
  subject    { FactoryGirl.create(:post, author: user) }

  it "has an author" do
    subject.author.should == user
  end
end
```

Timecop

Timecop allows for easy time manipulation. It's perfect for interacting with scopes of data based on date or time.

```
describe Post, ".recent" do
  before { Timecop.freeze(5.days.ago) }
  after  { Timecop.return }

  it "finds recent posts" do
    old_post = Post.create(:title => "Old-ish")   # created 5 days ago

    Timecop.travel(2.days.from_now)
    new_post = Post.create(:title => "New")       # created 3 days ago

    Timecop.travel(3.days.from_now)
    newer_post = Post.create(:title => "New-ish") # created today

    Post.recent.should == [newer_post, new_post]
  end
end
```

You can also use it within Cucumber; imagine these steps:

```
# features/step_definitions/date_time_steps.rb
Given /^today is "(.*)"$/ do |date|
  Timecop.freeze Date.parse(date)
end

Given /^(\d+) days? go(?:es)? by$/ do |day_count|
  Timecop.travel day_count.to_i.days.from_now
end

After { Timecop.return }
```

You can then use these steps to modify the current date and travel to different dates easily.

```
# features/sign_in.feature
Scenario: A signed in user gets signed out at the end of every month
  Given today is "10/30/2012"
  When I sign in
  And 1 day goes by
  Then I should be signed in
  When 1 day goes by
  Then I should not be signed in
```

Copyright © 2012 - thoughtbot, inc.