



> RubyThursday

# Ruby Snack #7

Shoulda Matchers

Hosted By: Melissa Wahnish

# Want To Code Along?



1. Rails app created
2. RSpec installed and set up
3. Parent and Child model created

Check out these videos...



Host: Melissa Wahnish



Hosted By: Melissa Wahnish

# Gemfile Recap



```
group :development, :test do
  gem 'better_errors'
  gem 'capybara-webkit'
  gem 'factory_girl_rails'
  gem 'ffaker'
  gem 'database_cleaner'
  gem 'letter_opener'
  gem 'rspec-rails'
  gem 'pry'
  gem 'pry-nav'
  gem 'pry-rails', '~> 0.3.2'
  gem 'simple_bdd'
gem 'shoulda-matchers'
gem 'spring'
end
```

# Shoulda Matchers



“Shoulda Matchers provides RSpec- and Minitest-compatible one-liners that test common Rails functionality. These tests would otherwise be much longer, more complex, and error-prone.”

<https://github.com/thoughtbot/shoulda-matchers>



# Example:

```
# spec/models/hacker_spec.rb
RSpec.describe Hacker, type: :model do
  it "is invalid without an email" do
    Factory.build(:hacker, email: nil).should_not be_valid
  end
end
```

## Versus

```
RSpec.describe Hacker, type: :model do
  it { should validate_presence_of (:email) }
end
```

# Starship Model:



```
/spec/models/starship_spec.rb
```

```
require 'rails_helper'

RSpec.describe Starship, type: :model do
  it { should validate_presence_of(:name) }
end
```

```
/app/models/starship.rb
```

```
validates_presence_of :name
```

Run the Spec:



```
$ rspec spec/models/starship_spec.rb
```

# Starship Model Part 2:

/spec/models/starship\_spec.rb



```
require 'rails_helper'

RSpec.describe Starship, type: :model do
  it { should validate_presence_of (:name) }
  it { should accept_nested_attributes_for (:crew_members) }
end
```

# Starship Controller Spec: /spec/controllers/starships\_controller\_spec.rb



```
let(:valid_attributes) {
  FactoryGirl.build(:starship).attributes
}

let(:invalid_attributes) {
  FactoryGirl.build(:starship, name: nil).attributes
}

#describe "PUT #update" do
let(:new_attributes) {
  { name: "Renegade" }
}

  new_attributes.each_pair do |key, value|
    expect(starship[key]).to eq(value)
  end
}
```

Run the Spec:



```
$ rspec spec/controllers/starships_controller_spec.rb
```

## Controller Spec Part 2:

/spec/controllers/starships\_controller\_  
spec.rb



```
describe 'GET #create_your_own_starship' do
  before { get :create_your_own_starship }

  it { should redirect_to (new_starship_path) }
  it { should redirect_to (action: :new) }

end
```



## Controller and Routes:

```
/app/controllers/starships_controller.rb
```

```
def create_your_own_starship
  redirect_to new_starship_path
end
```

```
/app/config/routes.rb
```

```
get 'starships/create_your_own_starship'
```

# Spring



“Spring is a Rails application preloader. It speeds up development by keeping your application running in the background so you don't need to boot it every time you run a test, rake task or migration.”

“in Rails 4.1 and up Spring is included by default”

Want More Free Ruby Snacks?



Subscribe



> RubyThursday