<!---

This file was generated on 2019-12-05T21:32:23+00:00 from the rspec-dev repo.

DO NOT modify it by hand as your changes will get lost the next time it is generated.

-->

# Contributing

RSpec is a community-driven project that has benefited from improvements from over \*500\* contributors.

We welcome contributions from \*everyone\*. While contributing, please follow the project [code of conduct](CODE\_OF\_CONDUCT.md), so that everyone can be included.

If you'd like to help make RSpec better, here are some ways you can contribute:

- by running RSpec HEAD to help us catch bugs before new releases

- by [reporting bugs you encounter](https://github.com/rspec/rspec-rails/issues/new?template=bug\_report.md)

- by [suggesting new features](https://github.com/rspec/rspec-rails/issues/new?template=feature\_request.md)

- by improving RSpec's [Relish](https://relishapp.com/rspec) or [API](https://rspec.info/documentation/) documentation

- by improving [RSpec's website](https://rspec.info/) ([source](https://github.com/rspec/rspec.github.io))

- by taking part in [feature and issue discussions](https://github.com/rspec/rspec-rails/issues)

- by adding a failing test for reproducible [reported bugs](https://github.com/rspec/rspec-rails/issues)

- by reviewing [pull requests](https://github.com/rspec/rspec-rails/pulls) and suggesting improvements

- by [writing code](DEVELOPMENT.md) (no patch is too small! fix typos or bad whitespace)

If you need help getting started, check out the [DEVELOPMENT](DEVELOPMENT.md) file for steps that will get you up and running.

Thanks for helping us make RSpec better!

## Rspec issues labels definition

### `Your first PR` issues

These issues are the ones that we be believe are best suited for new

contributors to get started on. They represent a potential meaningful

contribution to the project that should not be too hard to pull off.

### `Needs reproduction case` issues

These issues are ones that have been labelled by the maintainers that we

believe do not currently have enough information to be reproduced the RSpec

team. While not directly counted by the GitHub contribution graph, we consider

helping us to reproduce the issue with a repro case as an extremely meaningful

contribution.

### `Has reproduction case` issues

These issues are the ones that have reproduction cases, able to start working on

immediately. These are good ones to tackle to help us actively fix bugs.

## Maintenance branches

Maintenance branches are how we manage the different supported point releases

of RSpec. As such, while they might look like good candidates to merge into

main, please do not open pull requests to merge them.

## How do the cukes work?

The cucumber features for RSpec rails document how it works, but are also quasi

executable tests for the framework. They execute in the context of a pre-setup

Rails app.

1. Before the cucumber specs run, the directory `tmp/aruba` is cleared

2. If the example app hasn't already been created,

`bundle exec rake generate:app generate:stuff` is executed.

3. The example app is copied in to `tmp/aruba`

4. Everything in `tmp/aruba/spec/\*` is deleted apart from `spec/spec\_helper.rb` and

`spec/rails\_helper.rb`

5. the cucumber suite executes, creating files in that app and executing them

The best way to debug the app is to run a failing cucumber feature, which will

leave the test files intact in `tmp/aruba`, then you can cd in to that director

and run it in the bundle context of the aruba app.