Rubocop

CHEF

Lesson Objectives

- After completing this lesson, you should be able to
  - Evaluate your Ruby coding style against community standards
  - Improve your code by iterating through style warnings
  - Prevent specific rules from generating warnings

The Problem and the Success Criteria

- **The Problem:** We want to make sure our Apache cookbook follows best practice for Ruby code

- **Success Criteria:** We use Rubocop to check our cookbook before committing it

Rubocop

- New Ruby developers often ask for guidance on writing idiomatic Ruby

- Rubocop gives the same kind of feedback for your Ruby style that Foodcritic gives for your Chef Code

- Evaluates your Chef code for Ruby best practices, not for Chef style

- Documentation located here

    https://github.com/bbatsov/rubocop

Navigate to Apache Cookbook

```
$ cd cookbooks/apache
```

Exercise: Use Additional Rules

```
$ rubocop
```

```
recipes/default.rb:53:7: C: Use the new Ruby 1.9 hash syntax.
      :site_name => site_name,
      ^^^^^^^^^^^^^
recipes/default.rb:54:7: C: Use the new Ruby 1.9 hash syntax.
      :port => site_data["port"]
      ^^^^^^^^
recipes/default.rb:54:26: C: Prefer single-quoted strings when you don't need
string interpolation or special symbols.
      :port => site_data["port"]
                          ^^^^^^


4 files inspected, 50 offences detected
```

Fixing Rubocop Offenses

- We can mask offenses using the rubocop configuration file `.rubocop.yml`

- Rubocop can automatically generate a configuration called file named `.rubocop_todo.yml` that contains entries to mask all of your offenses

- We can use `.rubocop_todo.yml` to iterate through the offenses

- `.rubocop_todo.yml` settings are documented at

  https://github.com/bbatsov/rubocop/blob/master/README.md

Exercise: Generate .rubocop_todo.yml

```
$ rubocop --auto-gen-config
```

```
recipes/default.rb:54:7: C: Use the new Ruby 1.9 hash syntax.
      :port => site_data["port"]
      ^^^^^^^^
recipes/default.rb:54:26: C: Prefer single-quoted strings when you don't need
string interpolation or special symbols.
      :port => site_data["port"]
                         ^^^^^^


7 files inspected, 38 offences detected
Created .rubocop_todo.yml.
Run rubocop with --config .rubocop_todo.yml, or
add inherit from: .rubocop_todo.yml in a .rubocop.yml file.
```

Exercise: Review .rubocop_todo.yml

OPEN IN EDITOR: .rubocop_todo.yml

```
 This configuration was generated by `rubocop --auto-gen-config`
# on 2014-10-09 10:42:37 -0400 using RuboCop version 0.18.1.
# The point is for the user to remove these configuration records
# one by one as the offences are removed from the code base.
# Note that changes in the inspected code, or installation of new
# versions of RuboCop, may require this file to be generated again.

# Offence count: 4
# Cop supports --auto-correct.
# Configuration parameters: SupportedStyles.
HashSyntax:
  EnforcedStyle: hash_rockets

# Offence count: 2
LineLength:
  Max: 127
```

Exercise: Review .rubocop_todo.yml

OPEN IN EDITOR: .rubocop_todo.yml

```
# Offence count: 2
# Cop supports --auto-correct.
SpaceInsideBrackets:
  Enabled: false


# Offence count: 28
# Cop supports --auto-correct.
# Configuration parameters: EnforcedStyle, SupportedStyles.
StringLiterals:
  Enabled: false


# Offence count: 1
# Cop supports --auto-correct.
TrailingWhitespace:
  Enabled: false
```

Exercise: Update .rubocop.yml

📄 **OPEN IN EDITOR:** .rubocop.yml

```
inherit from: .rubocop_todo.yml
```

SAVE FILE!

Exercise: Re-run rubocop

```
$ rubocop
```

```
Inspecting 7 files
.......

4 files inspected, no offences detected
```

Work Through the Issues

- We now have a configuration file that masks all of our offenses

- We want to fix some of the offenses and mask the others

- Move the configuration rules for the offenses we want to continue masking from
  `.rubocop_todo.yml` **to** `.rubocop.yml`

Exercise: Update Rubocop Configuration

OPEN IN EDITOR: cookbooks/apache/.rubocop.yml

```
#inherit from: .rubocop_todo.yml  <<- Comment out this line

Encoding:
  Enabled: false


LineLength:
  Max: 200


HashSyntax:
  EnforcedStyle: hash_rockets


StringLiterals:
  Enabled: false
```

SAVE FILE!

Exercise: Re-generate .rubocop_todo.yml

```
$ rubocop --auto-gen-config
```

```
recipes/default.rb:15:11: C: Space inside square brackets detected.
  action [ :enable, :start ]
         ^
recipes/default.rb:15:27: C: Space inside square brackets detected.
  action [ :enable, :start ]
                         ^
recipes/default.rb:19:87: C: Trailing whitespace detected.
execute "mv /etc/httpd/conf.d/welcome.conf /etc/httpd/conf.d/welcome.conf.disabled" do
                                                                                    ^


7 files inspected, 4 offences detected
Created .rubocop_todo.yml.
Run rubocop with --config .rubocop_todo.yml, or
add inherit from: .rubocop_todo.yml in a .rubocop.yml file.
```

Fixing Rubocop Offenses

We decide we want to fix these offenses

1. Add `inherit from:` to `.rubocop.yml`

2. Remove an entry from `.rubocop_todo.yml`

3. Fix offenses

4. Verify fix by running Rubocop

5. Lather-Rinse-Repeat (steps 2-4)

6. Remove `.rubocop_todo.yml`

Exercise: Update .rubocop.yml

📄 **OPEN IN EDITOR:** .rubocop.yml

```
inherit from: .rubocop_todo.yml     <<- Un-comment this line

Encoding:
  Enabled: false


LineLength:
  Max: 200


HashSyntax:
  EnforcedStyle: hash_rockets


StringLiterals:
  Enabled: false
```

SAVE FILE!

Exercise: Delete Entry From .rubocop_todo.yml

OPEN IN EDITOR: `.rubocop_todo.yml`

```
# Offence count: 1
# Cop supports --auto-correct.
Style/SpaceAfterComma:
  Enabled: false


# Offence count: 2
# Cop supports --auto-correct.
Style/SpaceInsideBrackets:
  Enabled: false


# Offence count: 1
# Cop supports --auto-correct.
Style/TrailingWhitespace:
  Enabled: false
```

SAVE FILE!

Exercise: Re-run rubocop

```
$ rubocop
```

```
Offences:

recipes/default.rb:19:87: C: Trailing whitespace detected.
execute "mv /etc/httpd/conf.d/welcome.conf /etc/httpd/conf.d/welcome.conf.disabled" do
                                                                                      ^

4 files inspected, 1 offence detected
```

Exercise: Fix Default Recipe

OPEN IN EDITOR: recipes/default.rb

```
package "httpd" do
  action :install
end

service "httpd" do
  action [ :enable, :start ]
end

# Disable the default virtual host
execute "mv /etc/httpd/conf.d/welcome.conf /etc/httpd/conf.d/
welcome.conf.disabled" do
```

Remove trailing whitespace from line 19

SAVE FILE!

Exercise: Re-run rubocop

```
$ rubocop
```

Inspecting files

....

7 files inspected, o ence etected

WIN!

Review Questions

- What guidelines does Rubocop help you enforce?
- Why might you want to exclude certain cops?
- What is the name of Rubocop configuration file?