

Course Day 12, June 8, SAT

Takeway challenge

- ☒ cloned repo
- ☒ installing open ssl

```
make sure to report any problems to https://github.com/rvm/rvm/issues

Requirements installation successful.
Installing Ruby from source to: /Users/Aleksandra/.rvm/rubies/ruby-2.5.0, this may take a while depending on your cpu(s)...
ruby-2.5.0 - #downloading ruby-2.5.0, this may take a while depending on your connection...
ruby-2.5.0 - #extracting ruby-2.5.0 to /Users/Aleksandra/.rvm/src/ruby-2.5.0.....
ruby-2.5.0 - #applying patch /Users/Aleksandra/.rvm/patches/ruby/2.5.0/prelude_gcc_diagnostic.patch.
ruby-2.5.0 - #applying patch /Users/Aleksandra/.rvm/patches/ruby/2.5.0/libressl_2_7.patch.
ruby-2.5.0 - #configuring.....
ruby-2.5.0 - #post-configuration.
ruby-2.5.0 - #compiling.....
ruby-2.5.0 - #installing.....
ruby-2.5.0 - #making binaries executable..
ruby-2.5.0 - #downloading rubygems-2.7.9
ruby-2.5.0 - #extracting rubygems-2.7.9.....
ruby-2.5.0 - #removing old rubygems.....
ruby-2.5.0 - #installing rubygems-2.7.9.....
ruby-2.5.0 - #gemset created /Users/Aleksandra/.rvm/gems/ruby-2.5.0@global
ruby-2.5.0 - #importing gemset /Users/Aleksandra/.rvm/gemsets/global.gems.....
ruby-2.5.0 - #generating global wrappers.....
ruby-2.5.0 - #gemset created /Users/Aleksandra/.rvm/gems/ruby-2.5.0
ruby-2.5.0 - #importing gemsetfile /Users/Aleksandra/.rvm/gemsets/default.gems evaluated to empty gem list
ruby-2.5.0 - #generating default wrappers.....
ruby-2.5.0 - #adjusting #shebangs for (gem irb erb ri rdoc testrb rake).
Install of ruby-2.5.0 - #complete
Please be aware that you just installed a ruby that requires      2 patches just to be compiled on an up to date linux system.
This may have known and unaccounted for security vulnerabilities.
Please consider upgrading to ruby-2.6.0 which will have all of the latest security patches.
Ruby was built without documentation, to build it run: rvm docs generate-ri
Making gemset ruby-2.5.0 pristine.....
Making gemset ruby-2.5.0@global pristine.....
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ bundle
Fetching gem metadata from https://rubygems.org/.....
Fetching rake 12.3.2
Installing rake 12.3.2
Fetching public_suffix 3.0.3
Installing public_suffix 3.0.3
Fetching addressable 2.5.2
Installing addressable 2.5.2
Fetching ansi 1.5.0
Installing ansi 1.5.0
Fetching ast 2.4.0
Installing ast 2.4.0
Using bundler 1.16.6
Fetching mini_mime 1.0.1
Installing mini_mime 1.0.1
Fetching mini_portile2 2.3.0
Installing mini_portile2 2.3.0
Fetching nokogiri 1.8.5
Installing nokogiri 1.8.5 with native extensions
```

- ☒ And done:

```
Installing rspec 3.8.0
Fetching ruby-progressbar 1.10.0
Installing ruby-progressbar 1.10.0
Fetching unicode-display_width 1.4.0
Installing unicode-display_width 1.4.0
Fetching rubocop 0.56.0
Installing rubocop 0.56.0
Fetching simplecov-html 0.10.2
Installing simplecov-html 0.10.2
Fetching simplecov 0.16.1
Installing simplecov 0.16.1
Fetching simplecov-console 0.4.2
Installing simplecov-console 0.4.2
Bundle complete! 6 Gemfile dependencies, 33 gems now installed.
Use `bundle info [gemname]` to see where a bundled gem is installed.
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$
```

☒ Ok, read the instruction:

Instructions

- Challenge time: rest of the day and weekend, until Monday 9am
- Feel free to use google, your notes, books, etc. but work on your own
- If you refer to the solution of another coach or student, please put a link to that in your README
- If you have a partial solution, **still check in a partial solution**
- You must submit a pull request to this repo with your code by 9am Monday morning

Task

- Fork this repo
- Run the command 'bundle' in the project directory to ensure you have all the gems
- Write a Takeaway program with the following user stories:

```
As a customer
So that I can check if I want to order something
I would like to see a list of dishes with prices

As a customer
So that I can order the meal I want
I would like to be able to select some number of several available dishes

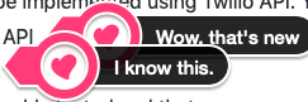
As a customer
So that I can verify that my order is correct
I would like to check that the total I have been given matches the sum of the various dishes in my order

As a customer
So that I am reassured that my order will be delivered on time
I would like to receive a text such as "Thank you! Your order was placed and will be delivered before 1
```

Read through the hints:

- Hints on functionality to implement:

- Ensure you have a list of dishes with prices
- Place the order by giving the list of dishes, their quantities and a number that should be the exact total. If the sum is not correct the method should raise an error, otherwise the customer is sent a text saying that the order was placed successfully and that it will be delivered 1 hour from now, e.g. "Thank you! Your order was placed and will be delivered before 18:52".
- The text sending functionality should be implemented using Twilio API. You'll need to register for it. It's free.
- Use the twilio-ruby gem to access the API
- Use the Gemfile to manage your gems
- Make sure that your Takeaway is thoroughly tested and that you use mocks and/or stubs, as necessary to not to send texts when your tests are run
- However, if your Takeaway is loaded into IRB and the order is placed, the text should actually be sent
- Note that you can only send texts in the same country as you have your account. I.e. if you have a UK account you can only send to UK numbers.



- Advanced! (have a go if you're feeling adventurous):

- Implement the ability to place orders via text message.
- A free account on Twilio will only allow you to send texts to "verified" numbers. Use your mobile phone number, don't worry about the customer's mobile phone.
- **WARNING** think twice before you push your mobile number or any private details to a public space like Github. Now is a great time to think about security and how you can keep your private information secret. You might want to explore environment variables.
- Finally submit a pull request before Monday at 9am with your solution or partial solution. However much or little amount of code you wrote please please please submit a pull request before Monday at 9am

☒ Look at the the code review rubric - but not to close. General scan so I know what is there but not the details of the solutions. This is learning from the mistakes of the the previous weekend challenge - I didn't look in to it until submission. It would be SO much easier if I did - I could get further and learn more. <https://github.com/ajmpawlik/takeaway-challenge/blob/master/docs/review.md>

☒ in the rubric review I noticed some things I don't know that were spoken about like `gem install rubocop ; rubocop` I know that rubocop aims to analyze your code for violations: intonations, spacing, ends.

☒ read tutorial on the use of rubocop https://rubocop.readthedocs.io/en/latest/basic_usage/

I run rubocop by typing

```
rubocop #and it basically checks the whole directory
```

I created a file with example code from the tutorial and it produced nice errors emphasised by robocop. Nice work rubocop

```

More information: https://github.com/rubocop-hq/rubocop-rails
Successfully installed rubocop-0.71.0
Parsing documentation for jaro_winkler-1.5.2
Installing ri documentation for jaro_winkler-1.5.2
Parsing documentation for parser-2.6.3.0
Installing ri documentation for parser-2.6.3.0
Parsing documentation for rubocop-0.71.0
Installing ri documentation for rubocop-0.71.0
Done installing documentation for jaro_winkler, parser, rubocop after 36 seconds
3 gems installed
Inspecting 3 files
...

3 files inspected, no offenses detected
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ ls
CONTRIBUTING.md Gemfile      Gemfile.lock  LICENSE      README.md    Rakefile     docs         spec
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ touch rubocop_try.rb
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ atom .
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ rubocop
Inspecting 4 files
..W.

Offenses:

rubocop_try.rb:1:5: C: Naming/MethodName: Use snake_case for method names.
def badName
  ^^^^^^^
rubocop_try.rb:2:3: C: Style/GuardClause: Use a guard clause instead of wrapping the code inside a conditional expression.
if something
  ^^
rubocop_try.rb:4:5: W: Layout/EndAlignment: end at 4, 4 is not aligned with if at 2, 2.
  end
  ^^^

4 files inspected, 3 offenses detected
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$

```

☒ Yep, I understand rubocop. Nice tool. That took me max 10 mins and gave me a feeling of catching up on the material I wasn't full into. One step closer to the peace of mind.

☒ Matt suggested using this. How's it going people? Getting super hungry over the weekend challenge! I've installed the Rubocop linter for atom (gives you 'live' syntax recommendations etc. as you type). Finding it really helpful with making the code as readable as possible! <https://atom.io/packages/linter-rubocop>

☐ That was rubocop, now time for test-coverage. I know it's about simple cove.

☒ read test-coverage

pill https://github.com/makersacademy/course/blob/master/pills/test_coverage.md

☒ follow the link to simple-cove <https://github.com/colszowka/simplecov>

☒ checked - simple cove is already in the Gemfile forked from the Makers repo with the challenge

☒ youtube on simple cove <https://www.youtube.com/watch?v=WMgDD2IU5nY>

☒ put the code in the spec_helper.rb on the very top of the helper file

```
require 'simplecov'
SimpleCov.start
```

☒ run test by **rspec** on the example file with three tests failing

```
Aleksandras-MacBook-Air:my_oyster_card Aleksandra$ rspec
FFF

Failures:

1) Oystercard
  Failure/Error: it { is_expected.to respond_to(balance) }

  NameError:
    undefined local variable or method `balance' for #<RSpec::ExampleGroups::Oystercard:0x00007f854c30f178>
    # ./spec/oystercard_spec.rb:4:in `block (2 levels) in <top (required)>'

2) Oystercard second option
  Failure/Error: expect(subject).to respond_to(balance)

  NameError:
    undefined local variable or method `balance' for #<RSpec::ExampleGroups::Oystercard:0x00007f854c30cf68>
    # ./spec/oystercard_spec.rb:7:in `block (2 levels) in <top (required)>'

3) Oystercard third option
  Failure/Error: expect(oystercard).to respond_to(balance)

  NameError:
    undefined local variable or method `balance' for #<RSpec::ExampleGroups::Oystercard:0x00007f854c317eb8>
    # ./spec/oystercard_spec.rb:12:in `block (2 levels) in <top (required)>'

Finished in 0.06815 seconds (files took 1.79 seconds to load)
3 examples, 3 failures

Failed examples:

rspec ./spec/oystercard_spec.rb:4 # Oystercard
rspec ./spec/oystercard_spec.rb:6 # Oystercard second option
rspec ./spec/oystercard_spec.rb:10 # Oystercard third option

Coverage report generated for RSpec to /Users/Aleksandra/Desktop/Coding_All/Course/week_2/my_oyster_card/coverage. 10 / 11 LOC (90.91%) covered.
Aleksandras-MacBook-Air:my_oyster_card Aleksandra$
```

It creates directory coverage with file index.html to be opened in the browser

Generated about a minute ago

All Files (90.91% covered at 0.99 hits/line)

2 files in total. 11 relevant lines. 10 lines covered and 1 lines missed

File	% covered	Lines	Relevant Lines	Lines covered	Lines missed	Avg. Hits / Line
lib/oystercard.rb	66.67 %	5	3	2	1	0.7
spec/oystercard_spec.rb	100.0 %	15	8	8	0	1.1

Showing 1 to 2 of 2 entries

Generated by simplecov v0.16.1 and simplecov-html v0.10.2 using RSpec

☒ From reading about simplecov I see the reading documentation is a key. I wasn't getting how to actually "activate" the simple cov functionality, and there it is "run your test". Of course. I rspec-ed the project and it worked well.

I got the simplecov and rubycop. Let's decide what next.

☐ Start with the first user story.

As a customer

So that I can check if I want to order something

I would like to see a list of dishes with prices

☒ run in IRB intro feature test

☒ first test in rspec failed and passed

The screenshot shows a code editor with a project named 'takeaway-challenge'. The file explorer on the left shows the project structure, including a 'spec' directory. The main editor window displays the content of 'list_spec.rb', which contains a RSpec test for a 'List' class. The test is currently incomplete, with a 'describe' block and an 'end' statement. The terminal window on the right shows the output of running 'rubocop' on the project. It reports that 6 files were inspected, 3 offenses were detected, and an error occurred while loading './spec/list_spec.rb'. The error is a 'NameError: uninitialized constant List' in 'spec/list_spec.rb:3:in <top (required)>'. The terminal also shows the output of running 'rubocop' again, which reports that 5 files were inspected, no offenses were detected, and the test passed. The terminal output includes a message: 'Have you considered running rubocop? It will help you improve your code! Try it now! Just run: rubocop'.

```
1 require 'list'
2
3 describe List do
4 end
5
```

```
rubocop_pty.rb:4:5: W: Layout/EndAlignment: end at 4, 4 is not aligned with if at 2, 2.
end
AAA

6 files inspected, 3 offenses detected
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ rspec

An error occurred while loading ./spec/list_spec.rb.
Failure/Error:
  describe List do
  end

NameError:
  uninitialized constant List
# ./spec/list_spec.rb:3:in <top (required)>
No examples found.

Have you considered running rubocop? It will help you improve your code!
Try it now! Just run: rubocop

Finished in 0.00053 seconds (files took 2.24 seconds to load)
0 examples, 0 failures, 1 error occurred outside of examples

COVERAGE: 100.00% -- 2/2 lines in 2 files

Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ rubocop
-rbash: rubocop: command not found
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ rubocop

Inspecting 5 files
-----

5 files inspected, no offenses detected
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ rspec
No examples found.

Have you considered running rubocop? It will help you improve your code!
Try it now! Just run: rubocop

Finished in 0.00084 seconds (files took 2.76 seconds to load)
0 examples, 0 failures

COVERAGE: 100.00% -- 3/3 lines in 2 files
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$
```

- ☒ During a **break** I found an interesting article about chronic stress and stress management. The lesson about importance of breathing as self-regulation tool we get at Makers is confirmed from all the angles.
https://hbr.org/2016/04/steps-to-take-if-youre-suffering-from-chronic-stress?utm_source=facebook&utm_campaign=hbr&utm_medium=social&fbclid=IwAR0eQn3DVTJnsyyXEGVg81YiCRbZHcw7lqMTfq7o6oeFcOPagATAwuF05g
- ☒ Pass the first more complex test. I wrote a test to check if the list contains a menu of dishes with assigned prices, and then created instance variable that carries this information for every instance of List. Test passed.

The screenshot shows a code editor with three panes. The left pane shows a project structure for 'takeaway-challenge' with files like .git, coverage, docs, review.md, lib, .DS_Store, list.rb, spec, list_spec.rb, spec_helper.rb, .gitignore, .rspec, .rubocop.yml, CONTRIBUTING.md, Gemfile, Gemfile.lock, LICENSE, Rakefile, and README.md. The middle pane shows the code for 'list_spec.rb' and 'list.rb'. The right pane shows the output of running the specs.

```
list_spec.rb
1 require 'list'
2
3 describe List do
4
5   it "contains a menu" do
6     list = List.new
7     expect(list.menu).to eq({ "chinese" => 10, "pierogi" => 8, "curry"
8   end
9 end
10
```

```
list.rb
1 class List
2
3   attr_reader :menu
4
5   def initialize
6     @menu = { "chinese" => 10, "pierogi" => 8, "curry" => 9, "carbonara"
7   end
8 end
9
10
```

Output of running the specs:

```
coverage | file | lines | missed | missing |
+-----+-----+-----+-----+-----+
| 20.00% | spec/list_spec.rb | 5 | 4 | 3, 5-7 |
+-----+-----+-----+-----+
1 file(s) with 100% coverage not shown
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ rspec
List
contains a menu (FAILED - 1)

Have you considered running rubocop? It will help you improve
your code!
Try it now! Just run: rubocop

Failures:

  1) List contains a menu
     Failure/Error: expect(list.menu).to eq({ "chinese" => 10,
"pierogi" => 8, "curry" => 9, "carbonara" => 11, "salad" => 6
    })
     NoMethodError:
       undefined method `menu' for #<List:0x00007fb44288a10>
       # ./spec/list_spec.rb:7:in `block (2 levels) in <top (req
       uired>'
Finished in 0.0054 seconds (files took 1.06 seconds to load)
1 example, 1 failure

Failed examples:

rspec ./spec/list_spec.rb:5 # List contains a menu

COVERAGE: 100.00% -- 8/8 lines in 2 files
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$ rspec
List
contains a menu

Have you considered running rubocop? It will help you improve
your code!
Try it now! Just run: rubocop

Finished in 0.01278 seconds (files took 2.33 seconds to load)
1 example, 0 failures

COVERAGE: 100.00% -- 9/9 lines in 2 files
Aleksandras-MacBook-Air:takeaway-challenge Aleksandra$
```

Test in IRB confirms that it worked.

The screenshot shows an IRB session with the following commands and output:

```
2.5.0 :002 > require './lib/list.rb'
=> true
2.5.0 :003 > list = List.new
=> #<List:0x00007fc6d686b5d8 @menu={"chinese"=>10, "pierogi"=
>8, "curry"=>9, "carbonara"=>11, "salad"=>6}>
2.5.0 :004 >
```

☒ Now, back to the user story. I want to see the menu.