Formative Assignment for Module #2, Lesson #2: Collections

The overall goal of this assignment is to manipulate collections to derive a result.

The functional goal of the assignment is to chain a series of functions together to locate selected values within a collection.

Functional Requirements

1. Given the following collection examples . . .

```ruby
# Grab 23 random elements between 0 and 10000
arr = (1..10000).to_a.sample(23)
p arr

#=> [61, 6969, 9885, 4722, 158, 9979, 4568, 3079, 2590, 2345, 9086, 9611, 1384, 8444, 7815,

# This selects only elements that when divided by 3 have a remainder of 0
# using the % (modulus) operator
p arr.select { |element| element % 3 == 0 }

#=> [6969, 9885, 4722, 7815, 1317, 8466]
```

Write a single chain of commands to find all numbers that

- are from an array of numbers 1..10000 inclusive
- are divisible by 3 (i.e., element % 3 == 0)
- are not less than 5000
- sorted in reverse order

Getting Started

1. Download and extract the starter set of files. The root directory of this starter set will be referred to as the root directory of your solution.

```
|-- module2_lesson2_formative.rb
|-- .rspec (important hidden file)
'-- spec
    |-- lesson2_spec.rb
    '-- spec_helper.rb
```

- module2_lesson2_formative.rb - contains the starting examples. Your solution must be placed within this file.

- spec - this directory contains tests to verify your solution. You should not modify anything in this directory
- .rspec - configuration file for unit tests. If you move your files you must take care to also copy this file.

2. Install the following gems used by the rspec unit tests. You may have some of these already installed.

```
$ gem install rspec
$ gem install rspec-its
```

3. Run the rspec command to execute the unit tests within the spec directory. This command should be run from the root directory of the project. This should result in several failures until you complete your solution.

```
$ rspec

FF.F....

Failures:

  1) lesson2 check results unexpected number of output lines
  2) lesson2 check results unexpected minimum number
  3) lesson2 check results unexpected sort order

Finished in 0.02205 seconds (files took 0.15601 seconds to load)
8 examples, 3 failures

Failed examples:

rspec ./spec/lesson2_spec.rb:16 # lesson2 check results unexpected number of output lines
rspec ./spec/lesson2_spec.rb:20 # lesson2 check results unexpected minimum number
rspec ./spec/lesson2_spec.rb:32 # lesson2 check results unexpected sort order
```

4. Implement the solution and re-test. You will likely find the `reject`, `sort`, and `reverse` methods of value when completing the solution.

Technical Requirements

1. Implement all parts of this assignment within the module2_lesson2_formative.rb file in the root directory of your solution. The grader will load this specific file from this location.

2. You script must locate a single chain of commands that locate numbers that

- are from an array of numbers 1..10000 inclusive
- are divisible by 3 (i.e., element % 3 == 0)
- are not less than 5000
- sorted in reverse order

3. Your answer must be printed as an array of numbers using the `p` print command as the last line of the script output.

```
...
[(number), (number), (number), ...]
```

Self Grading/Feedback

Unit tests have been provided in the bootstrap files that can be used to evaluate your solution. They must be run from the same directory as your solution.

```
$ rspec
........

Finished in 0.00838 seconds (files took 0.17347 seconds to load)
8 examples, 0 failures
```

Submission

There is no submission required for this assignment but the skills learned will be part of a follow-on assignment so please complete this to the requirements of the unit test.