Formative Assignment for Module #2, Lesson #1: Case Statement

The overall goal of this assignment is to implement a program control mechanism using Ruby that

- properly tests for object equality
- properly tests for a `nil` value
- properly tests for a boolean `true`/`false` value
- implements these tests using a `case` statement

The functional goal of the assignment is to rewrite a provided if/else statement using a case statement.

Functional Requirements

1. Re-write the following if/else statement using a `case` statement in Ruby.

```
some_var = "false"
another_var = "nil"

if some_var == "pink elephant"
  puts "Don't think about the pink elephant!"

elsif another_var.nil?
  puts "Question mark in the method name?"

elsif some_var == false
  puts "Looks like this one should execute"

else
  puts "I guess nothing matched... But why?"
end
```

2. Analyze the outcome of the original script and state the reasons why each of the first three (3) tests fail and how they could be made to succeed.

```
$ ruby module2_lesson1_formative.rb

I guess nothing matched... But why?
```

Change the source code of the solution to verify your conclusions.

Getting Started

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

   ```
   --- student-start
      |-- .rspec(important hidden file)
      |-- module2_lesson1_formative.rb
      `-- spec
          |-- lesson1_spec.rb
          `-- spec_helper.rb
   ```

   - .rspec - configuration file for unit tests. If you move your files you must take care to also copy this file.
   - module2_lesson1_formative.rb - contains the starting if/else statements. 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

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 provided Ruby script containing the if/else

   ```
   $ ruby module2_lesson1_formative.rb

   I guess nothing matched... But why?
   ```

4. 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

   .FFFF

   Failures:
   ...
   Finished in 0.02247 seconds (files took 0.1567 seconds to load)
   5 examples, 4 failures

   Failed examples:
   ```

```
rspec ./spec/case_spec.rb:17 # lesson1 check implementation remove if clause
rspec ./spec/case_spec.rb:21 # lesson1 check implementation remove elsif clause
rspec ./spec/case_spec.rb:25 # lesson1 check implementation remove elsif clause
rspec ./spec/case_spec.rb:29 # lesson1 check implementation missing case
```

5. Implement the solution and re-test.

Technical Requirements

1. Implement all parts of this assignment within the module2_lesson1_formative.rb file in the root directory of your solution. The grader will load this specific file from this location.
2. Remove all traces of `if`, `elsif`.
3. Re-implement the solution in terms of `case` and associated constructs.
4. Return the same result as the initial solution after applying the case statement.

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 0.00304 seconds (files took 0.16353 seconds to load)
5 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.