

## Final Assessment – Ruby on Rails

Name: \_\_\_\_\_

Date: \_\_\_\_\_

Refer to code below for questions 1-10:

```
class Team                                     # Line 1
  attr_accessor :name                         #|
  attr_reader :city                          #|
  attr_writer :state                         #|
                                           # Line 5
  def initialize(players)                    #|
    @player_count = players                 #|
  end                                       #|
                                           #|
  def self.number_of_players                 # Line 10
    @player_count                          #|
  end                                       #|
                                           #|
  def cheer_for_team                         #|
    puts "Go #{@name}!"                    # Line 15
                                           #|
    puts "Go #{self.name}!"                #|
                                           #|
    set_the_city                            #|
    return nil                             # Line 20
  end                                       #|
                                           #|
  private                                   #|
                                           #|
  def set_the_city                           # Line 25
    if @state == 'Texas'                   #|
      @city = "Austin"                     #|
    elsif @state == 'Florida'              #|
      @city = 'Tampa'                      #|
    else                                   # Line 30
      @city = "No clue"                    #|
    end                                    #|
  end                                       #|
end                                         # Line 35
```

*Note the line numbers start after the first line of code.*

1. On line 1, what is being defined? \_\_\_\_\_

2. What type of scope is the variable on line 7? \_\_\_\_\_

3. Explain the difference between the three `attr_` methods on lines 2, 3, and 4. \_\_\_\_\_

4. On line 10, what type of method is defined? \_\_\_\_\_  
\_\_\_\_\_
5. What is the purpose of line 11? \_\_\_\_\_  
\_\_\_\_\_
6. What type of method is defined on line 14? \_\_\_\_\_
7. On line 15, could single quotes be used instead of double quotes? Why? \_\_\_\_\_  
\_\_\_\_\_
8. What is line 17 doing? Is `self` needed? \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_
9. What is the purpose of line 20? What would the method return if that line was gone? \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_
10. What code can call the method defined on line 25? Could you replace the `if/elsif/else` with some other construct? What is it called? \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

In the following code, what are the types of each line? What is their class?

ex.

```
n = 2 #Type of n is Fixnum
```

11.

```
file = File.new('content.txt') _____  
content = file.read _____
```

12.

```
words = "these are my words".split _____
```

13.

```
user = User.new _____  
user.name = 'Me' _____  
user.city = 'Austin' _____
```

14. In the following code, what type of variable (that is, what scope) is `count` ?

```
def repeat
  limit = 10
  count = 0
  while count < limit
    puts count
    count += 1
  end
end

repeat
puts count
```

15. In the code from question 14, what will happen on the last line and why? \_\_\_\_\_

16. What commands do you run to install devise? \_\_\_\_\_

17. Devise is an example of Authorization or Authentication? \_\_\_\_\_

18. What is a gem you can use to do pagination? What is pagination? \_\_\_\_\_

19. What is paperclip used for? \_\_\_\_\_

20. Where do your paperclip files go by default? \_\_\_\_\_

21. What is the value in using a gem like Figaro, dotenv, or the built in rails secrets.yml file? \_\_\_\_\_

22. You are building your own authorization code for your app. Using pseudo-code, explain below the logic needed to make sure the current user of the app only sees their records.

23. The app you are building needs to have users who have many applications. Using nested resources, add those below (watch your plurality):

```
Rails.application.routes.draw do
```

---

---

---

```
end
```

24. Where can you access the params hash?

- (a) The Controller
- (b) The View
- (c) Both
- (d) Neither

25. Is it required to have a completed controller to create new records via rails c ? \_\_\_\_\_

26. What command do you run to create a migration? \_\_\_\_\_

27. What command do you run to migrate your database? \_\_\_\_\_

28. What command do you run to deploy your code to heroku? \_\_\_\_\_

\_\_\_\_\_

29. How do you migrate your database that exists on heroku? \_\_\_\_\_

30. What is bootstrap? \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

31 - 33. Below, create a syntactically valid model named `User` (or do the best you can from memory). Using the class you created above, add a validation to make sure the `:first_name` is always present. Add a validation that makes sure the `:ssn` attribute of the user is always unique.

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

34. Fill in the route file below so that the root\_path of the application goes to the `welcome` controller's `index` action  
`Rails.application.routes.draw do`

`root` \_\_\_\_\_

`end`

35. Where do you put methods that you want to use in your views? \_\_\_\_\_

36.

```
1  #!/usr/bin/ruby
2  # Note for this exercise, follow these simplified pig latin rules
3  # If the first letter is a vowel, add "way" to the end
4  # If the first letter is a consonant, move it to the end and add "ay"
5  # Hint: Use regex
6
7  def pigatize(text)
8    if starts_with_vowel?(text)
9      else
10     end
11  end
12
13  def starts_with_vowel?(text)
14    true
15  end
16
17  loop do
18    puts "Please enter a word and I will translate to pig latin"
19    text = gets.chomp
20    break if text.length == 0 # Break out of the loop if I say nothing
21    puts pigatize(text)
22  end
```