Knowledge Test K5

COMP 4081 • Software Engineering • Fall 2019

Name: _____ , _____

                       Last name   First name

Rules:

- No potty breaks.
- Turn off cell phones/devices.
- Closed book, closed note, closed neighbor.

- <u>WEIRD!</u> Do not write on the backs of pages. If you need more pages, ask me for some.

Reminders:

- Verify that you have all pages.
- Don't forget to write your name.
- Read each question <u>carefully</u>.
- Don't forget to answer <u>every</u> question.

Consider the *Echo* web app depicted in the figures. In particular, the app's interface is depicted in Figure 1, and key parts of its implementation are depicted in Figure 2, Figure 3, and Figure 4. Take a moment to familiarize yourself with the app before proceeding.

For each of the next several problems, a scenario is described, and you must provide the various data/events that result from the scenario. Your answer for each of the data/events must be one and only one of the following terms (the one that best describes the data/event):

```
delete          echo            localhost       update
do_echo         get             post            3000
```

Scenario: User enters `http://localhost:3000/echo` into their browser's location bar (as illustrated in Figure 1a).

1. [1] HTTP Request Method: _____

2. [1] Resource Path: _____

3. [1] Host: _____

4. [1] Port: _____

5. [1] Controller Action: _____

Scenario: User enters something into the Utterance field and presses the Submit button (as illustrated in Figure 1b).

6. [1] HTTP Request Method: _____

7. [1] Resource Path: _____

8. [1] Host: _____

9. [1] Port: _____

10. [1] Controller Action: _____

Figure 1a and Figure 1b each have a line that begins with "Here's what you said:" and that is followed by an utterance (e.g., "Hello, world!").

11. [1] Which line of code in Figure 4 renders the utterance that is displayed on the web page? Give the line number.

12. [1] Which line of code in Figure 3 passed the utterance to the view so it could be rendered on the page displayed in Figure 1a? Give the line number.

13. [1] Which line of code in Figure 3 passed the utterance to the view so it could be rendered on the page displayed in Figure 1b? Give the line number.

14. [1] Which line of code in Figure 2 was responsible for the page that was ultimately displayed in Figure 1a? Give the line number.

15. [1] Which line of code in Figure 2 was responsible for the page that was ultimately displayed in Figure 1b? Give the line number.

Rails expects certain naming conventions to be followed, so it's important to understand those conventions.

16. [1] Which of the following might be the name of a model class in Rails?

    a) `truck_driver`

    b) `truck_drivers`

    c) `TruckDriver`

    d) `TruckDrivers`

17. [1] Which of the following might be the basename (i.e., the filename minus the `.rb` file suffix) of the file that contains a model class in Rails?

    a) `truck_driver`

    b) `truck_drivers`

    c) `TruckDriver`

    d) `TruckDrivers`

18. [1] Which of the following might be the name of a model database table in Rails?

    a) `truck_driver`

    b) `truck_drivers`

    c) `TruckDriver`

    d) `TruckDrivers`

19. [1] Which of the following might be the name of a controller class in Rails?

    a) `truck_driver_controller`

    b) `truck_drivers_controller`

    c) `TruckDriverController`

    d) `TruckDriversController`

20. [1] What database model are we using in Rails?

    a) Document

    b) Object-oriented

    c) Relational

21. [5] Consider this UML diagram representing a model in Rails.

How would the Rails ORM represent this in the database? Be thorough in your description.

_____

_____

_____

_____

_____

_____

_____

_____

22. [1] What do database migrations do in Rails?

   a) Migrate data from one database management system to another (e.g., Postgres to MySQL)

   b) Update the database schema and set up the database accordingly

   c) Generate Rails model classes (i.e., in the `app/models` subdirectory)

23. [1] What do model classes do in Rails?

   a) Render HTML to be displayed in a web browser

   b) Translate HTTP requests to invocations of controller actions

   c) Use the database to CRUD records

24. [1] When creating model records in `seeds.rb`, which method should you use?

   a) `create`

   b) `create?`

   c) `create!`

25. [1] Which model class method retrieves all the records for that model from the database?

   a) `all`

   b) `find`

   c) `select`

26. [1] Which model class method retrieves the model record with a particular ID from the database?

   a) `all`

   b) `find`

   c) `select`

Bonus Problems

1. [6] Draw a dataflow diagram depicting the Rails MVC architecture, including at the minimum the browser and the four main Rails MVC subcomponents. Make sure to include all flows (directed edges between subcomponents).

Figures

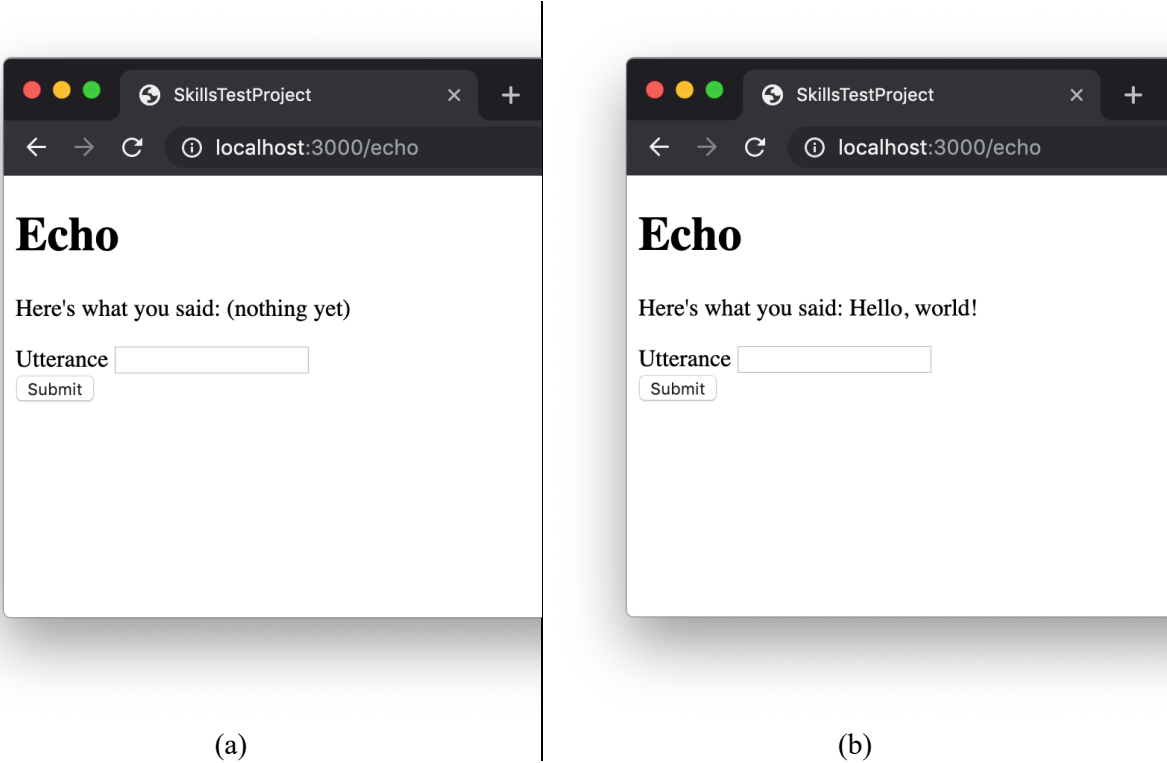

(a)	(b)

Figure 1. Echo page (a) when opened for the first time, and (b) after the user has entered the utterance "Hello, world!" and pressed the Submit button.

```
1  Rails.application.routes.draw do
2    get 'echo', to: 'pages#echo', as: 'echo'
3    post 'echo', to: 'pages#do_echo', as: 'do_echo'
4    root to: redirect('/echo', status: 302)
5  end
```

Figure 2. `config/routes.rb`

```
1  class PagesController < ApplicationController
2    def echo
3      respond_to do |format|
4          format.html { render :echo, locals: { utterance: '(nothing yet)' } }
5      end
6    end
7
8    def do_echo
9      utterance = params[:utterance]
10     respond_to do |format|
11       format.html { render :echo, locals: { utterance: utterance } }
12     end
13   end
14 end
```

Figure 3. `app/controllers/pages_controller.rb`

```
1  <h1>Echo</h1>
2
3  <p>
4    Here's what you said: <%= utterance %>
5  </p>
6
7  <%= form_with url: do_echo_path, local: true, method: :post do %>
8    <div>
9      <%= label_tag "utterance" %>
10     <%= text_field_tag "utterance", nil %>
11   </div>
12   <%= submit_tag "Submit" %>
13 <% end %>
```

Figure 4. `app/views/pages/echo.html.erb`