

# Routes, Controllers, & Views

Week 4 / Lesson 2

(Do you know what an Oxford comma is?)

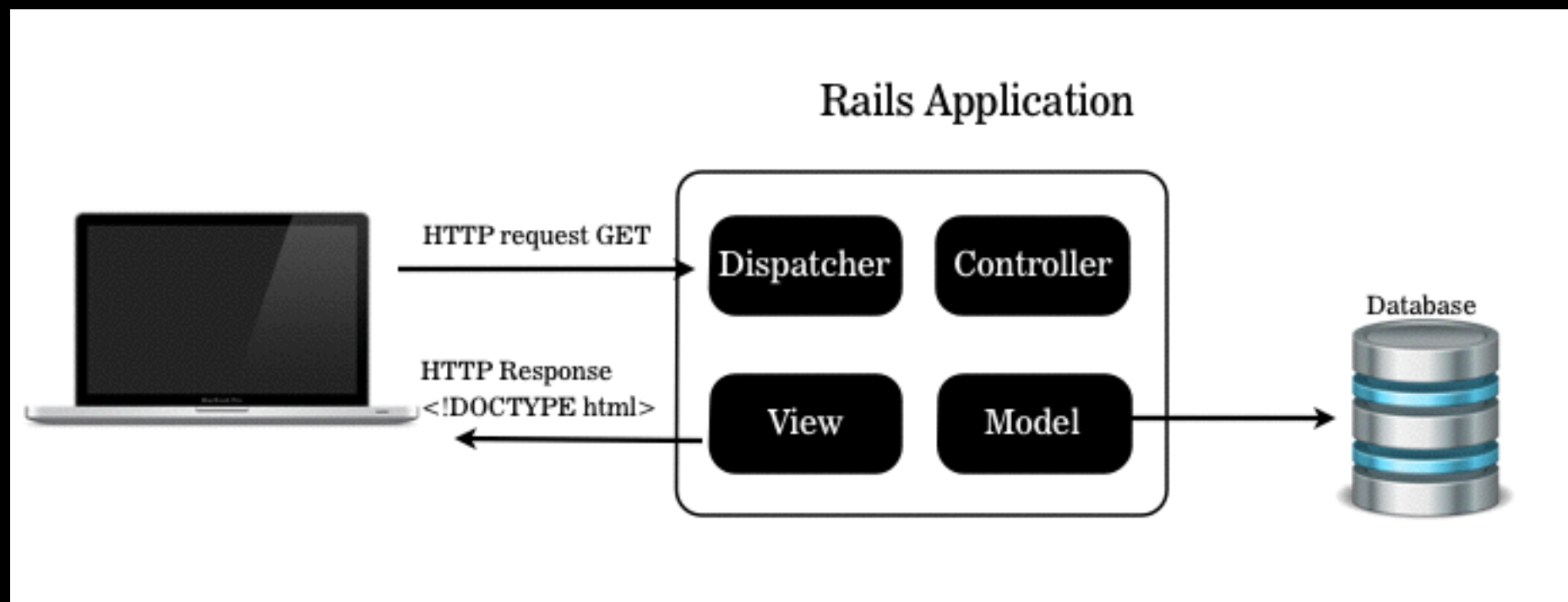
# Agenda

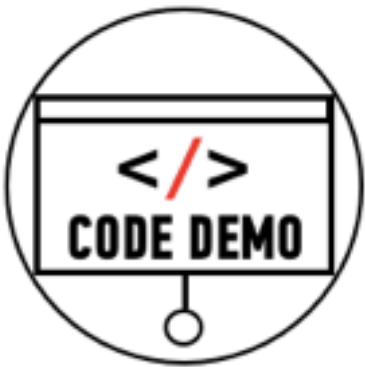
Learning Objective: Understand flow of control in a Rails app.

- Routes
- Controllers
- Views
- Lab Time

# Request Response

## Review





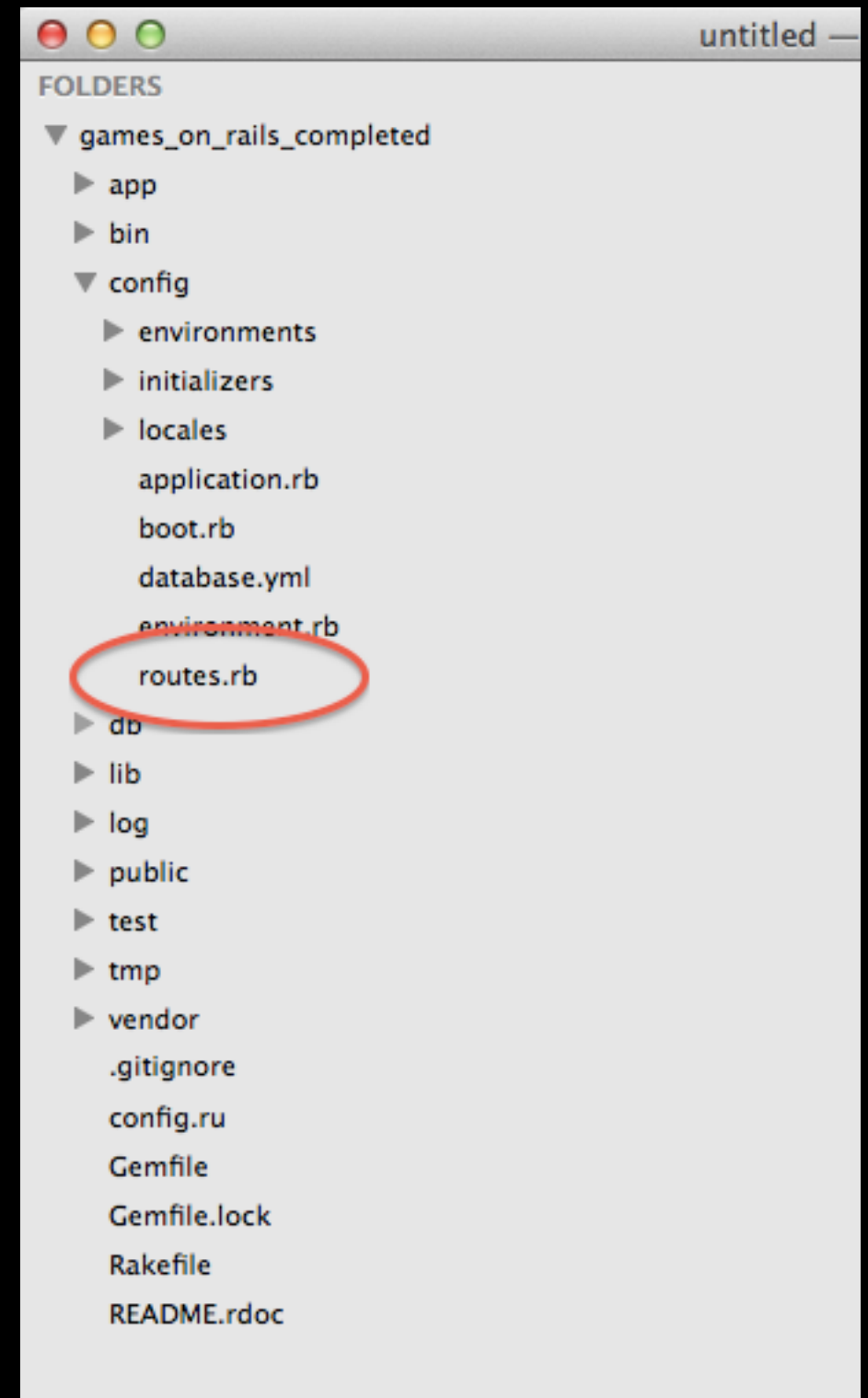
# Games on Rails

- Logic in the controller can render a specific view
- How can views link to a path and pass param to the controller?
- How can the app be styled using CSS?
- How would you add images to the application.

# The Dispatcher

## routes.rb

- Located at config/routes.rb
- Connects URLs to code
  - Used for mapping urls to your code
- Current Routes: `$ rake routes`
- Rails 4: Can be seen at <http://localhost:3000/rails/info/routes>



# Routes

## http verbs

GET, POST, PUT/PATCH, DELETE

HTTP Verb	Path	Action	Used For
GET	/books	index	display all books
GET	/books/new	new	return an HTML form for creating a new book
POST	/books	create	create a new book
GET	/books/:id	show	display a specific book
GET	/books/:id/edit	edit	return an HTML form for editing a book
PUT	/books/:id	update	update a specific book
DELETE	/books/:id	destroy	delete a specific book

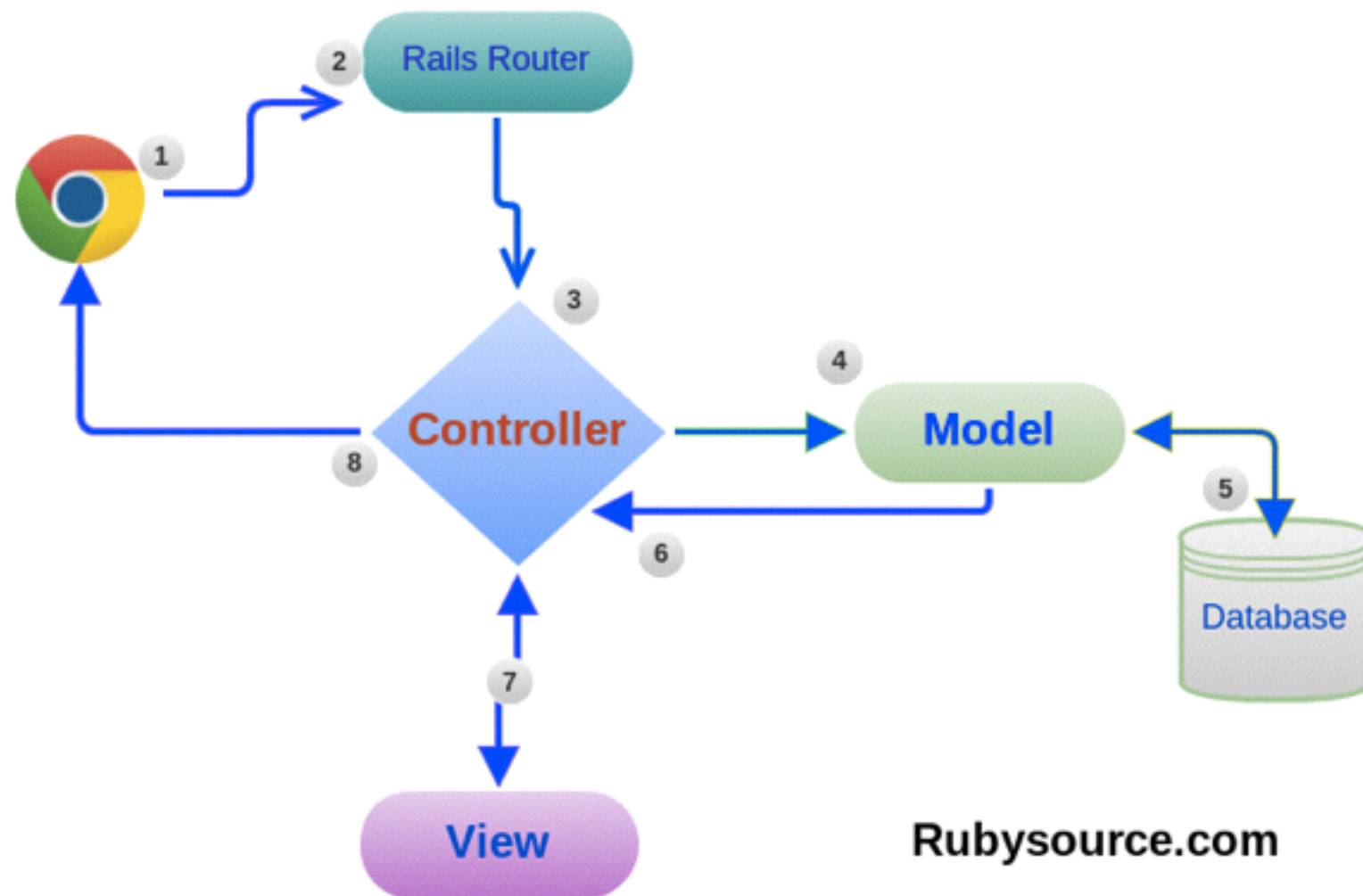
# Https Verbs

When GET is called.

- Visiting a website
- Clicking a link

# Model View Controller

aka MVC





# MVC

## Routes to Controller

### Controllers:

- Render a View
- Redirect to another view

# Controllers

## Controllers are Objects!

- Responsible for parsing user request
- It can display some text to the browser, redirect to another path or send an error message.
- Contains actions (aka functions)

# Controllers

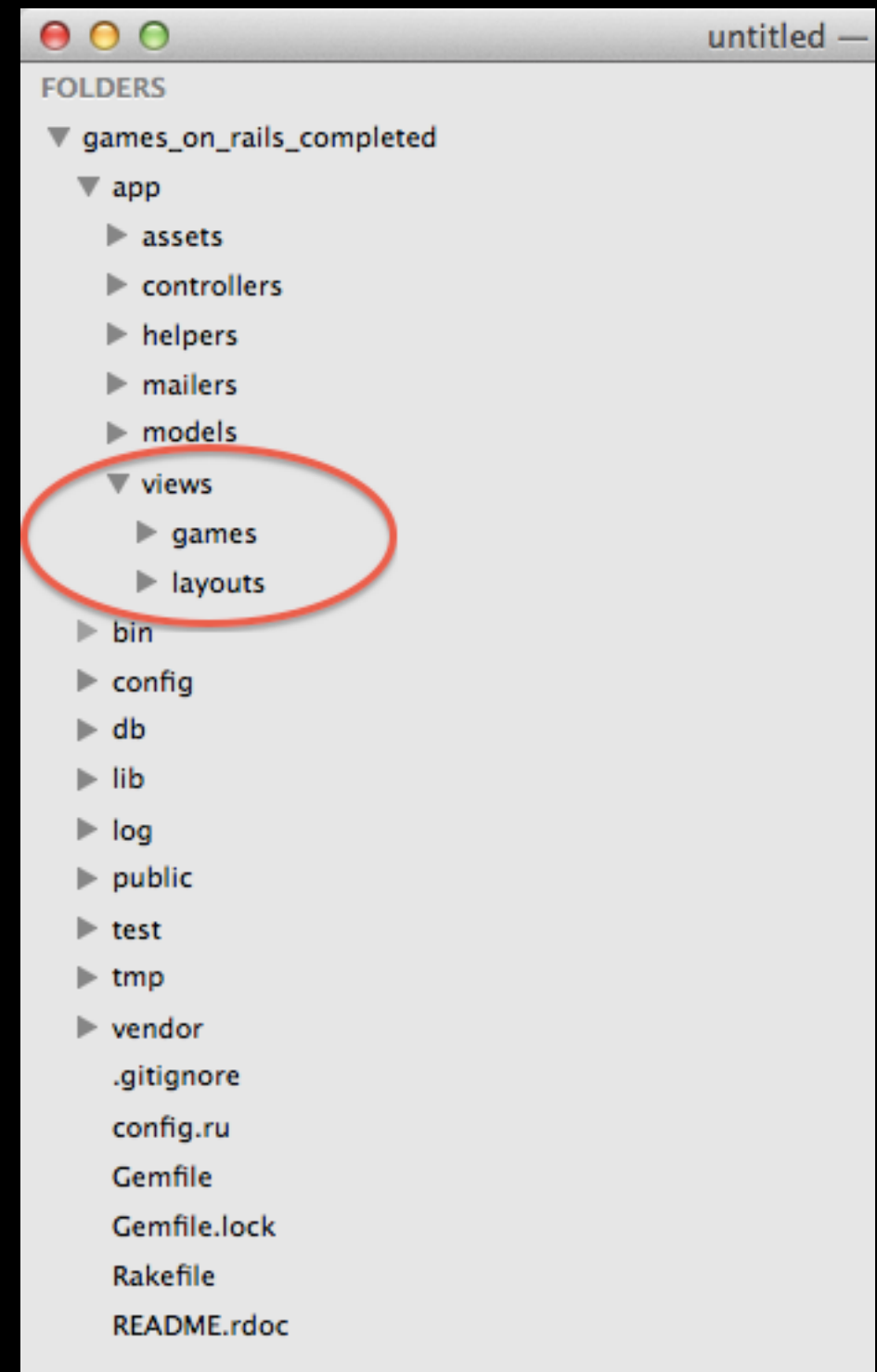
## Methods

- Recall: methods are "def blocks" in Ruby classes.
- Controllers are also classes, like any other.
- However, methods inside a controller are not like any other method; they are special (called Controller Actions)
- A Rails action gets invoked when a route matches to that controller/action combination
- Note: a private/protected method in a controller is NOT an action (you cannot hit that method via a URL). You can use private/protected methods to have helper methods inside a controller without being actions.

# MVC

## Controllers to Views

- Views can be rendered directly by the controller (i.e. return string from action)
  - Note: Not scalable when HTML file gets more complex
- By default your controller renders the view named `controller_name/action_name.html.erb`



stop

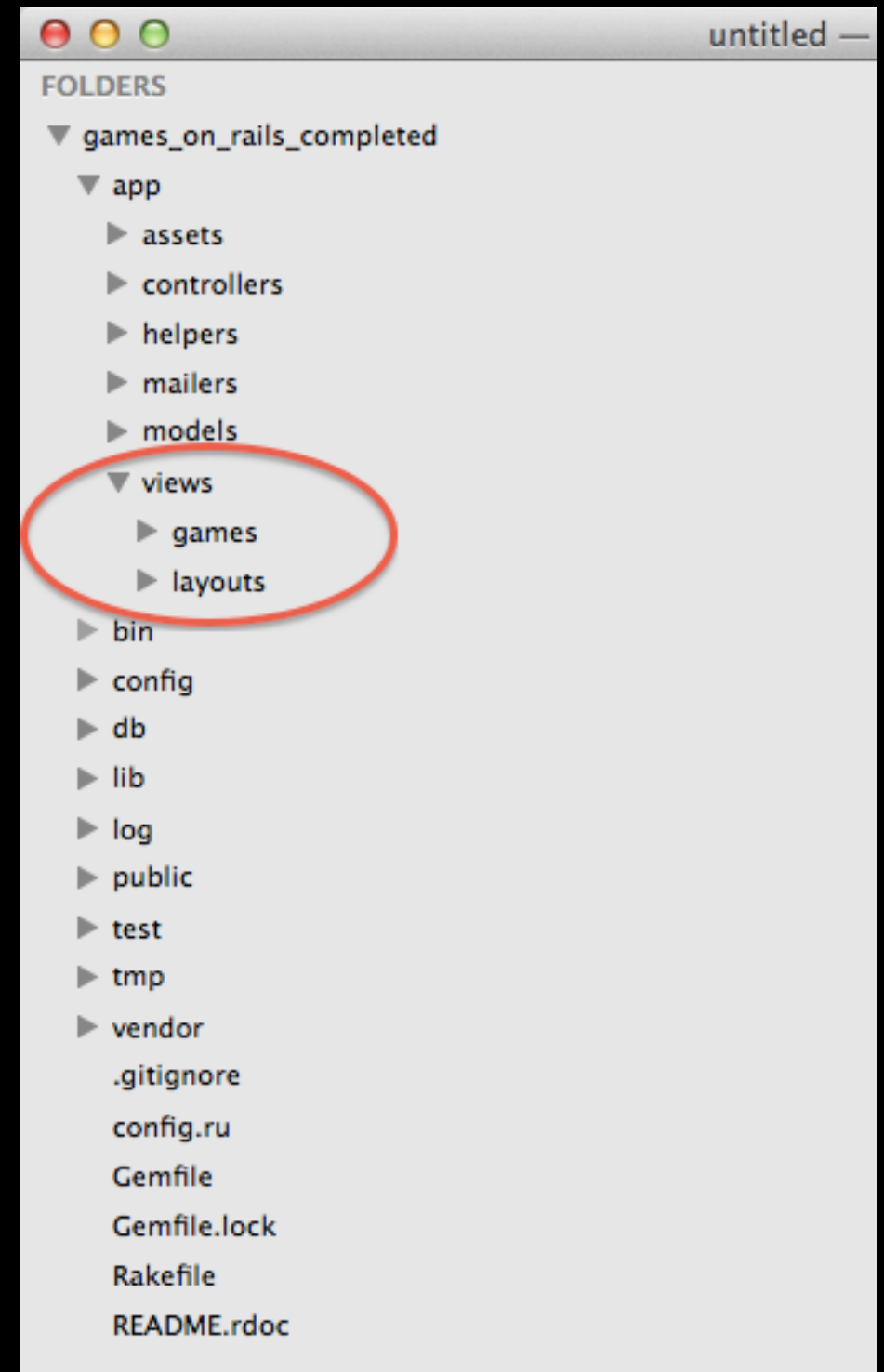
# Views

## html.erb

- Can write html
- Can write pure ruby.

```
<% @games.each do |game| %>  
  <%= game %>  
<% end %>
```

```
<% 1 + 1 %> // won't show  
<%= 2 + 2 %> // will show!
```



# Views

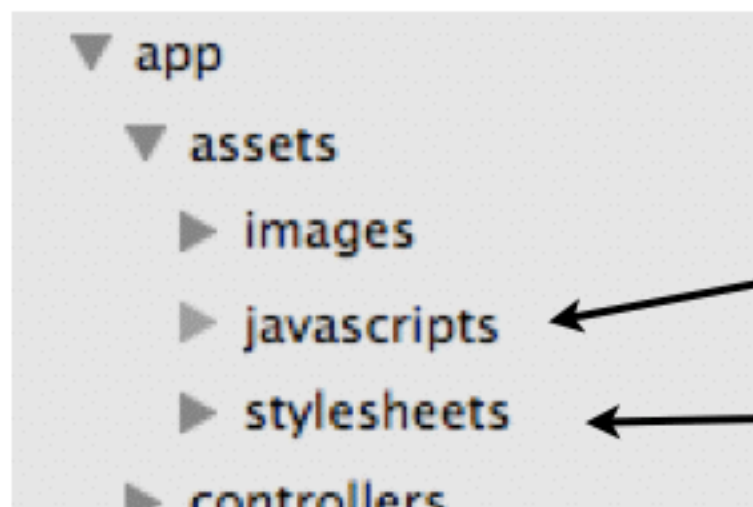
application.html.erb

- All views are wrapped inside application.html.erb
  - It loads the assets
  - sets up the meta tags, titles etc...
  - You can render no layout Or a different layout

# Views

## Assets Folder

- Keeps javascript files, stylesheets, and images



JavaScript files

css files



# Lab Time & Homework

- Create a Rails application. The app should have 2 controller and 2 views. The views should link back and forth to each other. One of the controllers should send some random data to its view.
- Submit a proposal for a final project – due next week
- EXIT TICKET!!!!