University of Sheffield

COM1001 SPRING SEMESTER

Professor Phil McMinn

p.mcminn@sheffield.ac.uk

Controllers

Controllers

A Sinatra application can have several routes – as many as we want

A set of routes for an application are collectively referred to as a **Controller**.

The job of a controller is to generate HTTP responses for incoming HTTP requests.

HTTP requests are matched to an appropriate route via the HTTP method and the resource identifier that it specifies.

Live Demonstration:

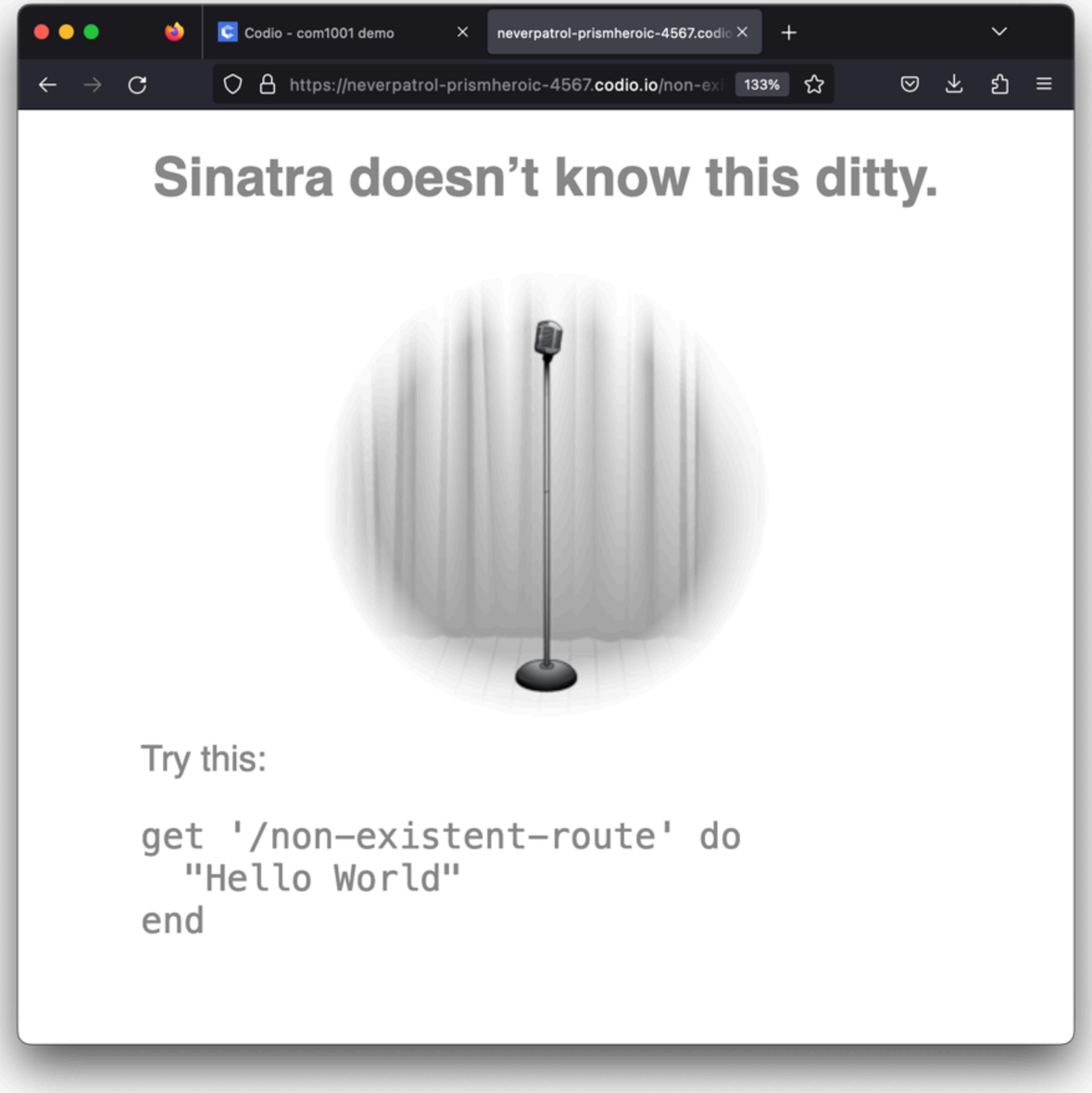
the `controllers_example` example

(from the com1001-examples GitHub repository)

Featuring:

- Multiple routes (for multiple resource IDs)
- Multiple controller files and their inclusion by `app.rb`
- Using the `rerun` command to avoid re-starting the web server every time a change is made
- What happens when there is no route matching the HTTP request – the HTTP 404 error page
- What happens when the code fails with an error – the HTTP 500 error page

When a HTTP Request Doesn’t Map to a Route



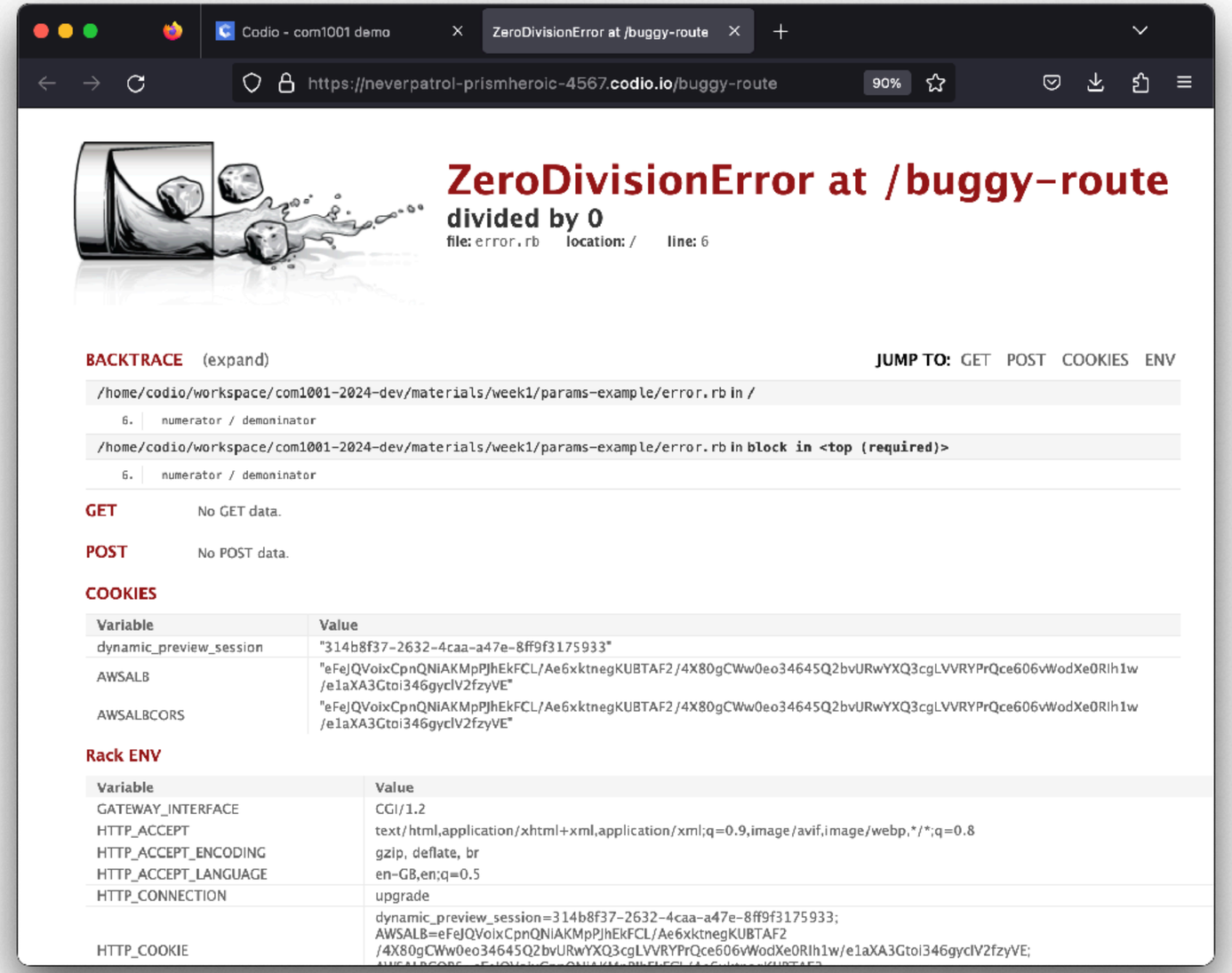
If Sinatra cannot map the HTTP request to a block with a matching route, it triggers a `HTTP 404 Not Found` status and returns this page.

This might be due to a problem with our Ruby code, or it could be that a user mistyped the URL into their browser.

When the Application Contains a Fatal Bug

```ruby
get "/buggy-route" do
  numerator = 5
  denominator = 0
  numerator / denominator
end
```

Sinatra encounters an error and so triggers a HTTP 500 Internal Server Error status message. In this instance, Sinatra usefully presents us with diagnostic information that may help with debugging.



Controllers – Summary

- Sinatra applications consist of a series of **routes**.
- A set of routes is called a **controller**.
- **Sinatra works by matching incoming HTTP requests to a route.** The matched route generates the body of the HTTP response.
- If Sinatra cannot match a route, it generates a 404 HTTP error.