Files

IO (Input/Output)

File is a type of class

Create a new file:

My\_file = File.new

Open a file

My\_file = File.open

My file = File.open(“file name”, “r”)

^ (mode)

a = File.open(“app.rb”, “w”)

* #if there is no existing file it will create it, must be in w (write) mode

open(filename, mode="r" [, opt]) → file

Standard Out = quasi temporary file we are seeing in the terminal

my\_file.read 🡪 turns file intro string

IO.readlines(".testfile")[0] #=> “this is line one\n”

file\_as\_string = my\_file.read

file\_as\_string.

file\_as\_string.split(”\n”)

^array ^ (new line character)

students = file\_as\_string.split(“\n”)

students.push “Joel”

new\_string = students.join(“\n”)

File 🡪 String 🡪 Array 🡪 String 🡪 File

Making files and directories in Ruby:

#=> in a file titled Sinatra\_builder.rb

project\_dir =Dir.mkdir(“my\_sinatra\_project”)

public\_dir = Dir.mkdir(“./my\_sinatra\_project/public”)

views\_dir = Dir.mkdir(“./my\_sinatra\_project/views”)

style\_css = File.open(“./my\_sinatra\_project/public/style.css”, “w”)

layout\_erb = File.open(“./my\_sinatra\_project/views/layout.erb”, “w”)

layout.erb puts <<-HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="description" content=" ">

<meta name="keywords" content=" ">

<link rel="stylesheet" href="styles.css">

<title>Untitled</title>

</head>

<body>

<%= erb :nav %> #=> allows you to create sep nav.erb file for links

<% = yield %> #=> notion is PARTIALS, pulls every ERB tag in app

</body>

</html>

HTML

index\_erb = File.open(“./my\_sinatra\_project/views/index.erb”, “w”)

app\_rb = File.open(“app.rb”, “w”)

app.rb.puts <--APPFILE #=> when you call puts on a file, it writes into a file

require ‘sinatra’ #=> APPFILE can be any text, simply starts/ends puts

require ‘sinatra/reloader’

get ‘/’ do

“Hello World”

end

APPFILE

app\_rb.close

cp –R <path/file>

mv <file> <end path>

CRUD (Create, Read, Update, Destroy)

Run array loop in index and print list of students in unordered list form:

<ul class =”students”>

<% @students\_array.each do |student| %>

<li><%= student % ><\li> #=> ‘=’ in ERB prints return value in browser

<% end %> #=> no ‘=’ just runs the code only

</ul> #=> ‘<%= %>’ only understands strings

<h1>New Student Form</h1>

<form>

<input type=”text” placeholder =”new student name” name=”student\_name” />

<input type=”submit” value=”Add Student” />

</form>

<h1>New Student Form</h1>

<form action =”/students” method=”POST”>

<input type=”text” placeholder =”new student name” name=”student\_name” />

<input type=”submit” value=”Add Student” />

</form>

post ‘students’ do #=> new route that process form input

students\_file = File.open and accepts POST requests

students\_string = students\_file.read # convert to string

students\_array students\_string.split(“\n”) # convert to array

student\_name = params [:student\_name] # retrieve st\_name from POST param

students\_array << student\_name # push new student to array

students\_string = students\_array.join(“\n”) # make array into a string

students\_file = File.open(“./db/students/txt”, “w”) # reopen database file in w

students\_file.puts(students\_string) # put string into original file

students\_file.close # close the file

redirect “/students” # redirect to list

end

Updating 🡪 open with ‘a’ mode or ‘add’ mode

post ‘students’ do #=> new route that process form input

students\_file = File.open(“./db/students/txt”, “a”) & accepts POST requests

student\_name = params [:student\_name] # append new student to file

students\_file.write “#{students\_name} # writes new line w/ student

students\_file.close # close the file

redirect “/students” # redirect to list

end

get “/students/:id” #=> show page that shows only one student

student\_id = params[:id]

students\_file = File.open(“./db/students/txt”, “a”)

student\_string = student\_file.read

students\_array = students\_string.split(“\n”)

students\_array[student\_id]

erb:show

end

<ul class =”students”>

<% @students\_array.each do |student| %>

<li><a href= “/students/<%= student %></a><\li> #print ruby value into html

<% end %>

</ul>

# Show.erb

<h1><%= @student\_name %></h1>

<p>

<

<\p>

readlines is alternative to read that converts to an array.

# EDIT Route

get ‘students/:id/edit’ do

erb :edit

end

Edit.erb

<h1><%= @student\_name %></h1>

<form action=”/students/<%= @student\_id%>/update” method=”POST”>

<input type=”text” placeholder=”student\_name” name=”student\_name” value=”<$= @student\_name %>”/>

<input type =”submit” value=”Edit” />

</form>

<form> #=> w/out ‘action 🡪 reload page, get request w/ params