

Aidan Gardner

How to start a new code project

Create a new GitHub repo to have the new portfolio code to be running in.

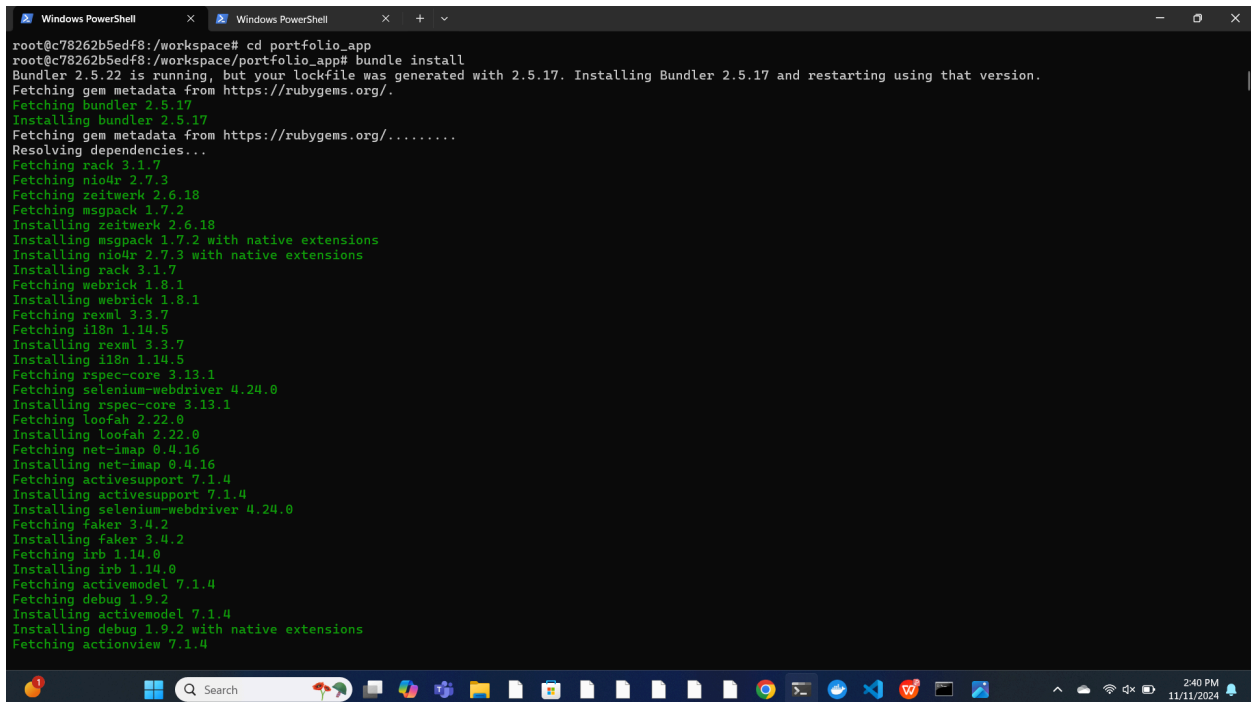
Create a new docker image: **docker build -t dockerusername/dockerimagename .**

Run the image: **docker -it -p 3000:3000 -v \${PWD} (for people using windows):/workspace dockerusername/imagename**

Pwd Once your in workspace

Cd into portfolio_app

Run bundle install

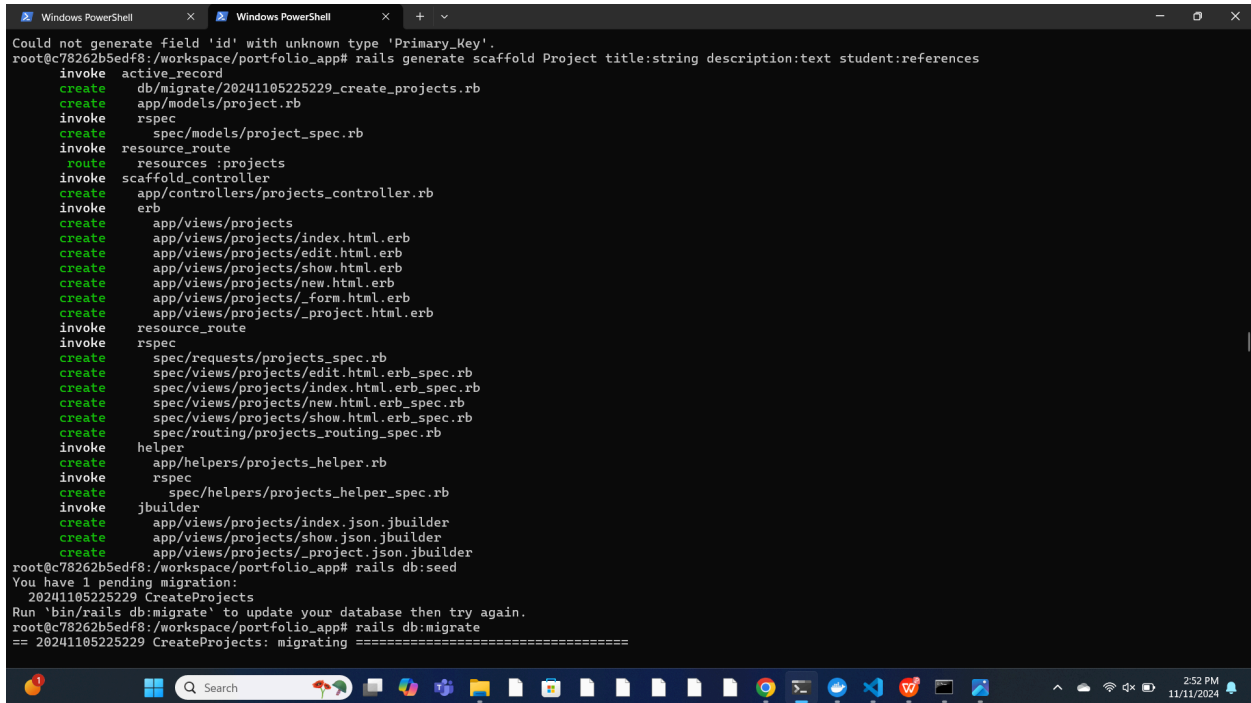


```
root@c78262b5edf8:/workspace# cd portfolio_app
root@c78262b5edf8:/workspace/portfolio_app# bundle install
Bundler 2.5.22 is running, but your lockfile was generated with 2.5.17. Installing Bundler 2.5.17 and restarting using that version.
Fetching gem metadata from https://rubygems.org/.
Fetching bundler 2.5.17
Installing bundler 2.5.17
Fetching gem metadata from https://rubygems.org/.....
Resolving dependencies...
Fetching rack 3.1.7
Fetching nio4r 2.7.3
Fetching zeitwerk 2.6.18
Fetching msgpack 1.7.2
Installing zeitwerk 2.6.18
Installing msgpack 1.7.2 with native extensions
Installing nio4r 2.7.3 with native extensions
Installing rack 3.1.7
Fetching webrick 1.8.1
Installing webrick 1.8.1
Fetching rexml 3.3.7
Fetching i18n 1.14.5
Installing rexml 3.3.7
Installing i18n 1.14.5
Fetching rspec-core 3.13.1
Fetching selenium-webdriver 4.24.0
Installing rspec-core 3.13.1
Fetching loofah 2.22.0
Installing loofah 2.22.0
Fetching net-imap 0.4.16
Installing net-imap 0.4.16
Fetching activesupport 7.1.4
Installing activesupport 7.1.4
Installing selenium-webdriver 4.24.0
Fetching faker 3.4.2
Installing faker 3.4.2
Fetching irb 1.14.0
Installing irb 1.14.0
Fetching activemodel 7.1.4
Fetching debug 1.9.2
Installing activemodel 7.1.4
Installing debug 1.9.2 with native extensions
Fetching actionview 7.1.4
```

Run rails db:migrate for any given saved files.

Generating a scaffold in an existing scaffold

Example: To create new code files for project in a student portfolio: **rails generate scaffold Project title:string description:text student:references**



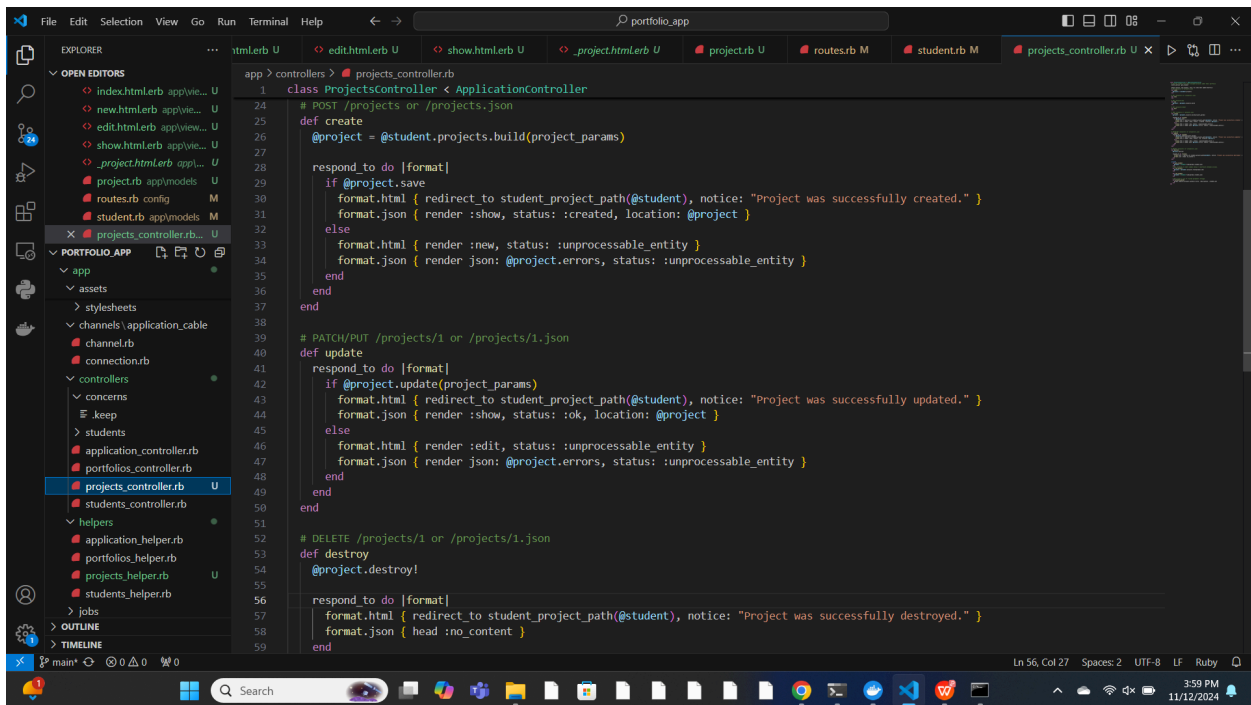
```
Could not generate field 'id' with unknown type 'Primary_Key'.
root@c78262b5edf8:/workspace/portfolio_app# rails generate scaffold Project title:string description:text student:references
invoke   active_record
create   db/migrate/20241105225229_create_projects.rb
create   app/models/project.rb
invoke   rspec
create   spec/models/project_spec.rb
invoke   resource_route
route    resources :projects
invoke   scaffold_controller
create   app/controllers/projects_controller.rb
invoke   erb
create   app/views/projects
create   app/views/projects/index.html.erb
create   app/views/projects/edit.html.erb
create   app/views/projects/show.html.erb
create   app/views/projects/new.html.erb
create   app/views/projects/_form.html.erb
create   app/views/projects/_project.html.erb
invoke   resource_route
invoke   rspec
create   spec/requests/projects_spec.rb
create   spec/views/projects/edit.html.erb_spec.rb
create   spec/views/projects/index.html.erb_spec.rb
create   spec/views/projects/new.html.erb_spec.rb
create   spec/views/projects/show.html.erb_spec.rb
create   spec/routing/projects_routing_spec.rb
invoke   helper
create   app/helpers/projects_helper.rb
invoke   rspec
create   spec/helpers/projects_helper_spec.rb
invoke   jbuilder
create   app/views/projects/index.json.jbuilder
create   app/views/projects/show.json.jbuilder
create   app/views/projects/_project.json.jbuilder
root@c78262b5edf8:/workspace/portfolio_app# rails db:seed
You have 1 pending migration:
  20241105225229 CreateProjects
Run 'bin/rails db:migrate' to update your database then try again.
root@c78262b5edf8:/workspace/portfolio_app# rails db:migrate
== 20241105225229 CreateProjects: migrating =====
```

Run rails db:migrate

Add any given modifications to new code files if needed

Run rails server -b 0.0.0.0

Nesting



```
1 class ProjectsController < ApplicationController
2   # POST /projects or /projects.json
3   def create
4     @project = @student.projects.build(project_params)
5
6     respond_to do |format|
7       if @project.save
8         format.html { redirect_to student_project_path(@student), notice: "Project was successfully created." }
9         format.json { render :show, status: :created, location: @project }
10      else
11        format.html { render :new, status: :unprocessable_entity }
12        format.json { render json: @project.errors, status: :unprocessable_entity }
13      end
14    end
15  end
16
17  # PATCH/PUT /projects/1 or /projects/1.json
18  def update
19    respond_to do |format|
20      if @project.update(project_params)
21        format.html { redirect_to student_project_path(@student), notice: "Project was successfully updated." }
22        format.json { render :show, status: :ok, location: @project }
23      else
24        format.html { render :edit, status: :unprocessable_entity }
25        format.json { render json: @project.errors, status: :unprocessable_entity }
26      end
27    end
28  end
29
30  # DELETE /projects/1 or /projects/1.json
31  def destroy
32    @project.destroy!
33
34    respond_to do |format|
35      format.html { redirect_to student_project_path(@student), notice: "Project was successfully destroyed." }
36      format.json { head :no_content }
37    end
38  end
39 end
```

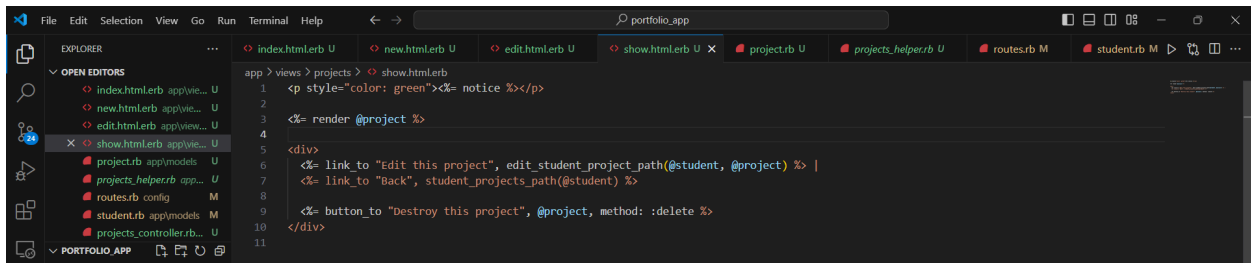
In this sample under the project_controller.rb, the student_project_path in the create, update, and destroy definitive actions indicates the project is being saved directly into the student profile.

The inclusion of the student in the project_path is useful for a few reasons:

- 1. The Student Scaffold serves as the parent function, so whatever is saved in the project scaffold, must be saved in the student.
- 2. The relationship serves as a one-to-many

Other nesting forms:

Edit_student_project_path



```
1 <p style="color: green"><%= notice %></p>
2
3 <%= render @project %>
4
5 <div>
6   <%= link_to "Edit this project", edit_student_project_path(@student, @project) %> |
7   <%= link_to "Back", student_projects_path(@student) %>
8   <%= button_to "Destroy this project", @project, method: :delete %>
9 </div>
```

In the show.html.erb in the projects file, the student can be able to edit a certain project for that student.

Instance Variables

Gives access to nested variables; (must have an @ for each variable to work)

These can only be used in code that is under the app/views file, as that gives the presentation of the string variables that are expected in the app uri.