**LEARN RUBY ON RAILS**

**Portfolio**

*Note: This project is currently giving you the option to deploy the application using* [*Heroku*](https://www.heroku.com/)*.* [*Heroku has announced*](https://blog.heroku.com/new-low-cost-plans) *that they are deprecating their free offering on November 28th, 2022. We are aware of this issue and are working hard on creating content that uses a free alternative. Thank you for your patience.*

You’re going to make a portfolio app to feature your projects.

**Please note that apps created here on site will NOT be deployable to Heroku. If you want to use Heroku to deploy your app, work on this project locally.**

Check out the instructions on how to deploy your app to Heroku locally [here](https://www.codecademy.com/articles/deploy-rails-to-heroku" \t "_blank).

To build this app, you’ll use:

* **HTML** and **CSS** for the front end
* **Rails** for the back end
* **Git** to manage your code
* **Heroku** to deploy your app (optional)
* **GitHub** to share your code with others

Here are three resources with tips on what goes into a good portfolio:

* [10 Steps To The Perfect Portfolio Website](https://www.smashingmagazine.com/2013/06/workflow-design-develop-modern-portfolio-website/" \t "_blank)
* [How To Build A World-Class Design Portfolio](https://www.fastcompany.com/3035190/designers-at-facebook-dropbox-and-pinterest-on-how-to-build-a-world-class-portfolio" \t "_blank)
* [Guide to building a web dev portfolio](https://discuss.codecademy.com/t/guide-how-to-build-a-web-dev-portfolio/394816" \t "_blank)

To preview your app at any point:

1. In the terminal, type rails server
2. In the browser, visit http://localhost:4001

**Tasks**

7/36 Complete

Mark the tasks as complete by checking them off

**Setup**

**1.**

Create a new Rails app named portfolio.

**2.**

After you create the new app, switch to its folder. In the terminal, type:

cd portfolio

**3.**

Install the gems specified in the **Gemfile**.

**4.**

Generate a controller named Pages.

**Make a welcome page**

**5.**

In the Pages controller, make an action named welcome.

**6.**

In the routes file, set the welcome action as the root route.

**7.**

Set up the layout file (**app/views/layouts/application.html.erb**). The layout file lets you build a base view that contains all the common elements of your site, such as CSS files, the header, and the footer. The <%= yield %> defines the portion of the layout that child templates can fill in.

* + In the <head> below the <title>, add CSS for the Roboto web font. Follow the [instructions here](http://www.google.com/fonts" \l "UsePlace:use/Collection:Roboto" \t "_blank). Choose the styles Thin 100, Light 300, Medium 500, and Bold 700.
  + Add CSS for Bootstrap. Follow the instructions [here](http://getbootstrap.com/getting-started/" \l "download-cdn" \t "_blank). Use the latest compiled and minified CSS, and not the optional theme.
  + Make sure both CSS URLs start with https.

**8.**

Create a view named **app/views/pages/welcome.html.erb** and style it in **app/assets/stylesheets/pages.css.scss**.

* + Add a header section with a navigation menu and your name.
  + Add a main section with a tagline summarizing what you do and a ‘View my Work’ button. Add styles to pages.css.scss that targets the main section and sets the background to your own image.
  + Add a footer with links to your GitHub, LinkedIn, Twitter, and a copyright. Flaticon has [free social icons here](https://www.flaticon.com/search?word=social%20media" \t "_blank).

To use images, first, upload your images to a hosting service of your choice. Then link to the images in your HTML and CSS.

**9.**

Start a Rails server to preview the app in the browser. In the terminal, type:

rails server

This command starts a Rails server listening on port 4001. View the result by running the local server and visiting http://localhost:4001 in the browser.

**10.**

Put your app under version control with Git. Then push it up to GitHub. [Check out the instructions here](https://www.codecademy.com/articles/push-to-github" \t "_blank).

**11.**

Deploy your app to Heroku if you’re working on this locally. [Check out the instructions here](https://www.codecademy.com/articles/deploy-rails-to-heroku" \t "_blank). Otherwise, check off this step and proceed with the next one.

**Make a portfolio page**

**12.**

Take screenshots of the apps you’ve built in this course to showcase some of your work. Upload them to an image hosting service of your choice.

**13.**

Make a new controller action and a new route for the portfolio page.

**14.**

Create a view for the portfolio page and style it:

* + The portfolio page has the same header and footer as the welcome page. Instead of writing the same code twice, refactor the header and footer from the welcome page and place them into **application.html.erb** so that they are available by all pages in the app.
  + Add a main section at the top of the page with an image of your best work.
  + Below the main section, add a supporting section that contains a grid of items for the apps you’ve built.
  + For each item, include the screenshot of the app, a title and description, and a link to the live app on Heroku.

**15.**

In the header, use the link\_to helper to generate links to the welcome page and portfolio page.

**16.**

In the welcome page, use link\_to to generate a link to the portfolio page.

**17.**

Run the local web server to view the app at http://localhost:4001.

**18.**

Commit the changes using Git, and push up to GitHub.

**19.**

Deploy your app to Heroku if you’re working on this locally. Otherwise, check off this step and proceed with the next one.

**Link to your blog**

**20.**

A blog is a good way to share your expertise and interests with others. It helps promote your portfolio as well as increase your online presence.

In the header, use link\_to to generate a link to your blog if you have one.

If you don’t have a blog, [Medium](https://medium.com/" \t "_blank) is a great blog-publishing platform to distribute your content and get discovered.

If you prefer not to blog about your portfolio, check off this step and proceed with the next one.

**21.**

Run the local server to view the result at http://localhost:4001.

**22.**

Commit the changes using Git, and push up to GitHub.

**23.**

Deploy your app to Heroku if you’re working on this locally. Otherwise, check off this step and proceed with the next one.

**Make an about page**

**24.**

Make a new controller action and a new route for the about page.

**25.**

Create a view for the about page and style it:

* + Add a picture of yourself.
  + Write a description of yourself. Describe your background, your goals, and your technical skills.

**26.**

In the header, use link\_to to generate a link to the about page.

**27.**

Run the local server to view the result at http://localhost:4001.

**28.**

Commit the changes using Git, and push up to GitHub.

**29.**

Deploy your app to Heroku if you’re working on this locally. Otherwise, check off this step and proceed with the next one.

**Make a contact page**

**30.**

Make a new controller action and a new route for the contact page.

**31.**

Create a view for the contact page and style it. Add links to places people can contact you online. A few options are:

* + email
  + Facebook
  + GitHub
  + Google+
  + LinkedIn
  + Twitter

Flaticon has [free social icons here](https://www.flaticon.com/search?word=social%20media).

**32.**

In the header, use link\_to to generate a link to the contact page.

**33.**

Run the local server to view the result at http://localhost:4001.

**34.**

Commit the changes using Git, and push up to GitHub.

**35.**

Deploy your app to Heroku if you’re working on this locally. Otherwise, check off this step and proceed with the next one.

**Map a custom domain**

**36.**

Great job! You’ve built your own portfolio app in Rails!

If you’ve been deploying your app to Heroku thus far, you might be interested in adding your own custom domain (eg http://www.yourname.com). [Check out the instructions here](https://www.codecademy.com/article/going-beyond-with-heroku" \t "_blank).