



Overview

Homework 1 involves building a professional home page for yourself. It's the first iteration of building a personal tech blog and online professional presence. Your goal is to make a personal homepage of this caliber – or better.

MODEL SOLUTION: michaelpb.github.io/personalhomepage/

Feel free to dig into the source code of the model solution to see how it's put together. However, when you submit something, it should be all yours.

Solo project topic

- The recommended topic is a **Personal Tech Blog + Online Resume**

If you have an idea that you think might fit for the solo project, let the instructors know. Subject to instructor approval, we can work to tweak the requirements for the page to be about this idea, as opposed to a personal tech blog / professional profile (this default project). Keep in mind you will likely be working on the same project for the next 5 homework assignments.

1 Requirements

1. Must be built using Bootstrap or a Bootstrap-based template
2. Must contain 3 pages:
 - For example: *Blog*, *Projects*, and *Contact*, or alternatively *Bio*, *Resume*, and *About*
3. Each page must be in a separate file, so there must be at least 3 HTML files, all of which provide links to navigate to each other
 - *Note:* This means that one long scrolling page *does not* qualify.
4. Must have at least 10 active custom styling rules to add extra styling, possibly further customizing or overriding the template or bootstrap theme
 - *Note:* Put these in a separate CSS file, distinct from your theme or anything else you downloaded (easier to grade, and also a good habit regardless)

2 Steps

Here are recommended steps for achieving these results.



2.1 Gathering media

Gather stock photos OR your own original photography (if you have taken photos good enough) to fulfill the following:

1. Your profile picture (should be a nice, professional photo of you)
2. At least 1 neutral, striking background photo
3. A nice repeating background (optional)
4. A striking CSS gradient (optional)

HINT: For free stock photos, you might try: pixabay.com/

HINT: For great free backgrounds, check out: www.toptal.com/designers/subtlepatterns/

HINT: For gorgeous background gradients, check out: www.webcore-it.com/colorful-background/

2.2 Picking a color scheme

Don't *only* use the color scheme that comes with Bootstrap or the template you selected. You need *at least* 1 more color of your choosing.

1. Pick 1 or 2 complementary colors codes to use in conjunction with or replacing the colors that came with the template. Colors in CSS are often specified with “hex codes”, for example: `#ae0000` is the code for a dark red, and `#f6ffa0` the code for a light yellow.
2. Write down these codes into your README.md file

2.2.1 Design Tips: Color Scheme

- When choosing a color scheme, pick several colors that go well together, and not clash with any background photos you will be using.
- Not sure where to begin? Look at color scheme sharing sites such as Color Hunt or Colour Lovers.
- Try ColorMind.io to pick out a color scheme from a photograph.

2.3 Pick two fonts

Using Google Fonts, pick two nice looking fonts to be used on your page.

- Pick a font to be used for all header text
- Pick a body text to be used for everything else

2.3.1 Design Tips: Fonts

- Pay attention to serif vs sans-serif. For most things you will want to choose sans-serif, but serif can add a certain “print-like” quality.
- When it comes to picking fonts, caution is key. While handwritten fonts or funky looking “display” fonts might be fun, they should rarely be used for a professional site.¹

¹Remember: Text is meant to convey information, not just to decorate. Keep your font choice modest and clear.



2.4 Brainstorming your new style

1. Google for other websites as inspiration. For example, here are bunch of links showcasing different designs:
 - speckyboy.com/creative-portfolios-of-50-designers/
 - medium.muz.li/50-user-profile-page-design-inspiration-5c45aeeda400
 - www.mockplus.com/blog/post/personal-website-design-examples
2. Sketch out on paper how you want your page to be laid (using a “wireframe” sketch).

2.4.1 Design Tips: Site style

- Look for something you like, but also stay grounded. You need to get this done by the deadline!²
- If you want, you may use the same layout and structure as the solution, but don’t copy code from it.

2.5 Getting started

Time to actually get started! You have two options: (1) Either hacking away at a pre-built template, or (2) build it up from scratch using the Bootstrap grid. There are pros and cons to both:

1. Using a pre-built Bootstrap template could end up looking really, really cool. However, to customize it to your liking you will have to delete a lot of code, and spend time hacking away at it turning it into *yours*.
2. Starting from scratch with the Bootstrap grid will allow you to have more control and create cleaner, clearer code. But it takes more effort and an eye for design to make something that approaches the design quality of a template someone else wrote.

2.5.1 Using Bootstrap from scratch

Using the Bootstrap skills you learned in class, start from an empty file and use the sketch you drew as reference to build the structure of your page. You might consider starting from the solution to an in-class activity as the basis.

2.5.2 Using a pre-built Bootstrap-based template

Using the sketch you drew as a reference, look through the available Bootstrap starter templates for one that resembles what you want. There are many sources for these (try Googling “Free Bootstrap 5 Template”). To list a few:

- mdbootstrap.com/freebies/
- mobirise.com/bootstrap-template/
- startbootstrap.com/template-categories/all/
- colorlib.com/wp/cat/bootstrap/

Note that some (such as bootstrapmade.com) cost money to to “license” it: bootstrapmade.com/license/

²For some, this is the most fun step, so avoid the rabbit hole of “picking the perfect style”.



2.6 Fill in your content

Time to roll up your sleeves and dive in to add in your own info! Start by creating the “About” page.

1. Add in your bio
2. Add in a link to your email
3. Add in navigation that includes your name and profile page
4. Generally customize your page so that it contains *at least* the same amount of complexity as the example solution

2.7 Start tweaking your style

1. Change all your header elements to use the **font-family** that you chose from Google Fonts
2. If the color you chose does not match the template or default Bootstrap color, you’ll have to create new classes that modify the default style. Check look at the source code to the solution and check out the CSS for **btn-red** and **nav-red** to see how you might do this.

2.8 Duplicate for the extra pages

- Copy the HTML file 2 more times to create the **projects.html** page, and the **blog.html** page
- It’s okay to have placeholder content for now on these pages if you don’t have time to finish the rest.
- Make sure to change the links on ALL the pages to link to each other. The links might look like:

```
<a href="./projects.html">Click here for my projects</a>
```

3 Submission

Submit in the same way you uploaded your website in the last stage of 1.1 - HTML - Activity 4. Refer to that activity & videos for further details. In summary:

1. Create a new Git repo. Name it whatever you’d like to call your site (for example, **tech-blog** or **resume**)
2. Ensure “Initialize with README.md” is checked
3. Select “Add file” and then “Upload files”
4. Copy over files into the newly created directory
5. Save & commit
6. Submit the URL to your published site as your homework submission

NOTE: If you have already completed the 2.2-Git lesson, and are comfortable using the **git** command-line tool, feel free to use that instead. Use of **git** will be required for future all assignments. This is the only assignment that you are permitted to use the “Add files” button in order to submit.



4 Bonus

You have a space for a blog now. Consider writing a few articles, maybe about the start of your coding journey with Kickstart Coding or about some technology you find interesting. Up to you!

4.1 Use your Pework Journal for ideas

As an idea for blog posts, consider incorporating text you wrote for your Pework Journal. Many of these “personal musings” could form the basis of an interesting blog post about your journey into the world of coding. You could use the summary you wrote describing the Hacker News article. Include the link to the article, and then the body of the text might be useful as the basis of a blog post.

4.2 Search Engine Optimization

SEO, or Search Engine Optimization, is how “Google-friendly” is your site. You can go REALLY deep with this, but if you have some more time, consider researching “meta” tags, in order to customize the display of your site if/when someone finds it on Google.

4.3 Mix-in extras

If you have extra time, see if you can integrate one or more of the following free services to spice things up:

- JotForm for a nice “contact me” form: jotform.com
- Disqus for a comment section on your blog: disqus.com