

Personal website instructions

- I. Create your website
 - A. Download personal website
 1. Visit https://github.com/smgood/demo_website
 2. Click Green button that says **Code**
 3. Click **Download Zip**
 4. Unzip file
 5. Open index.html in Chrome
 - B. Open index.html file in code editor of your choice (Visual studio, TextEdit, etc)
 - C. Follow directions and replace text for every TODO note
 1. Once finished, the text on the website should be updated with your name and info.
 - D. Replace the photo in the folder with a photo of yourself
 1. Remove the existing photo.jpg in the personal website folder
 2. Find a photo of yourself and crop it to a square.
 3. Save your photo in the personal website folder as photo.jpg
 4. You should now see a photo of yourself on the personal website.
 - E. Replace resume
 1. Remove the **resume.pdf** file in the personal website folder
 2. **If you do not have a resume**, create a new google doc and write "This is a placeholder resume for [your name]".
 3. Download and save your resume in the personal website folder as resume.pdf
 4. When you click on the resume button, you should see your resume now.
- II. Host your website on github pages
 - A. Visit github.com
 - B. Sign up for new account
 - C. click the plus sign (+) icon in the top-right corner of the page and select "New repository"
 - D. For the **Repository name**, use your GitHub username followed by .github.io (e.g., yourusername.github.io). This is a special name that tells GitHub to turn your code into a website.
 - E. Add a description "HTML website written at Vision Hacks 1"
 - F. Set the repository visibility to public. This is a requirement for GitHub to display your website.
 - G. Click the green **"Create repository"** button.
 - H. On the repository's main page, click the **"Add file"** dropdown menu.
 - I. Select **"Upload files"**.
 - J. **Drag and drop** the following files:
 1. Index.html
 2. resume.pdf
 3. portrait_photo.jpg
 - K. Once the files are uploaded, scroll down. A **"Commit changes"** box will appear.

- L. In the **"Commit message"** field, type a short note about the files that were just uploaded such as "first version of my personal website"
- M. Click the green **"Commit changes"** button.
- N. Visit your website at [https://www.\[yourusername\].github.io](https://www.[yourusername].github.io). Note it might take a few minutes for your website to go live.

III. Get a free .me domain name

- A. Get the github student pack
 - 1. Go to the GitHub Education page: <https://education.github.com/pack>.
 - 2. Click on **"Get the Pack"** or **"Sign up for Student Developer Pack."**
 - 3. Provide your school email address to confirm student status
- B. **Step 2: Claim Your Free Domain from Namecheap**
 - 1. Scroll down to the **"Namecheap"** offer. It will mention a free **.me** domain for one year.
 - 2. Click the link or button to redeem the offer.
 - 3. You may need to sign in with your GitHub account to authorize the offer.
 - 4. On the Namecheap page, search for a domain name you like (e.g., [yourname.me](https://www.namecheap.com/domain/yourname.me)).
 - 5. Add it to your cart, and when you check out, the price should be **\$0.00**. You may need to create a Namecheap account if you don't have one.
 - 6. Complete the registration.

III. Connect your new domain to your github pages website

- 1) Go to Your GitHub Repository Settings
 - a) Go to your personal website repository on GitHub ([yourusername.github.io](https://www.[yourusername].github.io)).
 - b) Click the **"Settings"** tab near the top of the page.
 - c) In the left-hand menu, click **"Pages."**
- 2) Add Your Custom Domain in GitHub
 - a) Find the section for **"Custom domain."**
 - b) In the text box, type the full custom domain you just got from Namecheap (e.g., www.yourname.me).
 - c) Click **"Save."**
- 3) Configure DNS Settings in Namecheap
 - a) Go to Namecheap and log in.
 - b) From your dashboard, go to your **"Domain List"** and find the domain you just registered.
 - c) Click the **"Manage"** button next to it.
 - d) Click on the **"Advanced DNS"** tab.
 - e) You need to create two different types of records to make your domain work with and without the "www."

- f) Add a CNAME Record for "www":
 - Under "Host Records," click **"Add New Record."**
 - **Type:** CNAME Record
 - **Host:** www
 - **Value:** yourusername.github.io (This should be the name of your GitHub repository. Do NOT include http:// or https://.)
 - Click the green checkmark to save.
 - g) Add A Records for the Root Domain (without "www"):
 - Click **"Add New Record"** again.
 - **Type:** A Record
 - **Host:** @
 - **Value:** 185.199.108.153
 - Click the green checkmark to save.
 - **Repeat this step three more times** to add the remaining GitHub Pages IP addresses. Use the same "A Record" and "@" host, but with these values:
 - 185.199.109.153
 - 185.199.110.153
 - 185.199.111.153
- 4) Confirm changes
- a) Wait for the DNS changes to complete. This can take anywhere from a few minutes to a few hours.
 - b) Go back to your GitHub Pages settings page.
 - c) After a while, you should see a green checkmark next to your custom domain, and the option to **"Enforce HTTPS"** will appear.
 - d) **Make sure to check the "Enforce HTTPS" box** for security. This makes sure your website has the secure lock icon in the browser.