

Resource Recommendations - from Angela

Blog:

<https://www.freecodecamp.org/news/how-i-went-from-newbie-to-software-engineer-in-9-months-while-working-full-time-460bd8485847/>

<https://code.likeagirl.io/thoughts-on-becoming-a-self-taught-software-engineer-c8d8e7bde704>

Courses:

- [CS50's Introduction to Computer Science](#) from Harvard University
- [Divide and Conquer, Sorting and Searching, and Randomized Algorithms](#) from *Stanford University*
- [Graph Search, Shortest Paths, and Data Structures](#) from *Stanford University*
- [Greedy Algorithms, Minimum Spanning Trees, and Dynamic Programming](#) from *Stanford University*
- [Shortest Paths Revisited, NP-Complete Problems and What To Do About Them](#) from *Stanford University*
- [Modern HTML & CSS From the Beginning \(Including Sass\)](#) by Brad Traversy
- [Modern React with Redux](#) by Stephen Grider
- [Web development program](#) from freeCodeCamp
- [CS61B Data Structure](#) from UC Berkeley

Preparing for the Job Application, Search, and Interview Process - from Marie

- Resume
 - List technical skills, education, and experience
 - Include non technical experience and education to show the other skills and experience you can bring to the job.
 - Check online for examples and advice. Have multiple people review and give feedback
- LinkedIn Profile
 - Complete all sections that you have an experience or skills for.
 - Make sure to link your projects and portfolio (add thumbnail images)
 - Include an up to date and professional profile photo of your face.
 - Your education, skills, experience should be as similar to your resume as possible.
 - Read examples and advice online about writing the About section
 - Make sure your settings show you are open to work and list several roles. May be helpful to do a few job searches to make sure the roles you list are available and match your interests and skills.
 - If possible, ask for recommendations from colleagues, supervisors, or clients. Make sure to ask politely and be specific about what skills you would like them to include in the recommendation
- Portfolio website
 - Include live website links, descriptions, images and maybe demo videos of your projects as well as links to the github repositories.
 - Link your resume, blog, and any other online work you have.
 - You can use a form instead of your contact information for privacy.
- Github (GitLab, etc)
 - Make sure you have several well organized repositories that are pinned and public.
 - Have at least one project that is a collaboration with someone to show that you can work with git with multiple people (this is a very important skill!)
 - Make sure there are links to Github on your resume, LinkedIn, portfolio website.
- Tracking spreadsheets and documents
 - Setup a spreadsheet to keep track of which jobs you have applied to, how you applied, and if there is a contact person you should follow up with (and when/how).
 - Setup a document to draft all your cover letters and any essay or short answer questions for applications. Document the role and company (and question prompt if there is one) for each one so that when you write cover letters for similar roles or application questions you have previous drafts to improve on. Saves times and each draft improves!
- Mock interviews

- Non-technical - practice questions about why you are interested in the role and company, your strengths, your story, etc. You can get a career coach or may know someone who can do this for you (maybe research questions and give your mock interviewer a list to ask)
 - Technical - There are paid services that provide mock technical interviews with live coding such as Skilled.
 - Elevator pitch - have a short pitch that you can comfortably share with anyone while networking, conducting phone screens, or more formal interviews.
- Make a list of the strengths and experiences that you have that will make you stand out from other entry level software engineers. Emphasize those in all of the above and be confident pitching yourself with those strengths.
- Make a list of job boards to check each week. Also list industries you are most interested in, but keep your options open.
- Keep taking classes (and listing them on LinkedIn) and practicing algorithms using Udemy classes, Leetcode, Codewars, Cracking the Code, etc.