Coding with AI to this extent was a very enlightening experience on how useful it can be. It saved me a very great deal of time, giving me a very good styling so that I could focus on making the POST requests work. I made sure to read over every piece of code that Github Copilot generated, and it helped me come to a much better understanding of website design, especially the code for a green success message and an orange validation error. One point where the AI generated code I had some trouble understanding was when I asked it to make a function for loading comments. It included some functions I wasn't familiar with, but after looking over it more closely, I have a much better understanding of how it works and how to implement it in my future projects.

Thanks to these experiences, I think I have a good understanding of the correct uses of AI. Just how a calculator and/or google won't turn someone into a genius mathematician, AI won't make someone who doesn't understand anything about coding into a programmer. What it can do, however, is help generate a good framework when given thoughtful prompts, which can be checked and refined by someone who knows what they are doing in order to massively speed up time. For example, the comments display section wasn't working, and the AI couldn't troubleshoot the error which was ultimately a small syntax difference. AI is good at drawing an outline, but not particularly great at small details, which is why it's important that the one using AI understands the code they want it to write. To those who do understand it, or at the very least can follow what it writes, it can be extremely helpful for doing the time-consuming things that one knows how to do, such as styling, or help learn different ways of doing things. If AI tools weren't available, I would have certainly still been able to create the comments page, but it would have taken me a much longer time, particularly with styling since I'm not very good at it. With the AI helping with that and with the other functions, I can work a lot faster while still knowing what I'm doing. If I was to use AI for something completely unfamiliar, it's unlikely I would have been able to troubleshoot any possible errors that could occur from it missing the small details. LLM's are, ultimately, repetition machines, and they work well with creating things similar to what already exists. It's up to the human controlling it to refine and troubleshoot those things into the unique implementation that is needed. While some think the rise of AI might make people less intelligent, in the field of programming, at least, it might be different. In the future, the most important skill

for a programmer will be the ability to think, rather than the ability to create that which has already been created.