As someone with limited coding experience, creating this memory-matching game was both a challenging and semi-rewarding experience. While my academic background has given me strong analytical and problem-solving skills, applying those skills to a practical coding project is completely useless really. It has been hard to embrace a new way of thinking. This project deepened my understanding of web development but also taught me valuable lessons about perseverance, debugging, and the importance of breaking down complex problems into baby steps.

One of the most significant challenges I faced was understanding how to structure the game logic. While the concept of a memory-matching game is simple, translating it into code requires careful planning. I had to think about how to track the state of the game, such as which cards were clicked, whether they matched, and how to handle the timing of revealing and hiding images. Breaking the problem into smaller components—like shuffling the cards, handling clicks, and checking for matches—made the task less overwhelming.

Debugging was another area where I learned a lot. At first, I struggled to figure out why certain parts of the game weren't working as expected. For example, I initially had issues with the cards not flipping back correctly after a mismatch. Using `console.log` to track the values of variables and the flow of the program was incredibly helpful. It reminded me of the importance of testing and iteration. Just as I would revise a thesis based on feedback, I found myself revising and refining the code until it worked as intended. Over and over again.

One of the most satisfying aspects of this project was seeing the game come to life. When I first started, the idea of creating an interactive web application seemed daunting. However, as I built each component—starting with the grid layout, then adding the click functionality, and finally implementing the game logic—I began to see how the pieces fit together. The moment I clicked

on two cards and saw them flip to reveal matching images, I felt a sense of accomplishment that my project looked like I might actually complete it.

This project also highlighted the importance of collaboration and learning from others. While I wrote the code myself, I relied heavily on online resources, tutorials, and examples to understand concepts like event listeners, array manipulation, and DOM manipulation. This experience reinforced the idea that coding is a continuous learning process. It also made me appreciate the value of clear documentation and community support in the programming world.

For the AI portion, I asked chat gpt to create a memory matching game, but to use colors instead of images, I thought it would actually work that way, and it does. It is wild to see AI casually put together something that took me a significant amount of time. It still is a bit buggy but I thought for a first run that was pretty good!

In conclusion, creating this memory-matching game was a learning experience. It taught me the importance of patience, problem-solving, and iterative development. While I still have much to learn, this project has given me some confidence to tackle more complex coding challenges in the future.