Today was my tenth day as an intern at Surfboard Payments. The day was full of learning, especially about how a compiler works and how it converts a programming language into machine language step by step.The morning session started with an explanation of tokens and expressions. I learned that symbols like letters (a-z, A-Z), numbers (0-9), and special characters (+, -, .) are the basic parts of a programming language. A token is the smallest unit in a program, similar to a word in a sentence. Tokens combine to form statements, which are like sentences in programming. A group of statements forms a complete program, just like a paragraph in a written document.The compilation process has several stages. First, tokenization breaks the program into small parts called tokens. Then, syntax analysis checks if the code follows the correct rules. Next, semantic analysis ensures that the code makes sense. Finally, code generation converts the program into machine language that a computer can understand. This process helps identify errors and makes the program run efficiently.Understanding how a compiler works is important because it helps in writing better programs. Knowing this process allows me to find mistakes easily and improve my coding logic. Connecting theory with practice will help me understand key concepts like parsing, tokenization, and syntax rules, which are useful in real-world programming.

In the afternoon session, I worked on JavaScript programming in Visual Studio Code. I successfully completed 15 coding problems, which helped me improve my problem-solving skills and gain more confidence in JavaScript. Solving these problems allowed me to apply logical thinking, improve debugging skills, and write efficient code.After completing the JavaScript problems, I pushed my completed code to a Git repository. This step is important for version control and collaboration. Using Git helps track changes, share code with others, and maintain a record of progress. Learning how to use Git is essential for working in a team and managing code efficiently.Later in the day, I focused on writing PostgreSQL queries. I had already completed 10 queries in MySQL, so my task was to rewrite them using PostgreSQL in Visual Studio Code. This helped me understand the differences between MySQL and PostgreSQL and improve my database management skills. I ensured that all queries were properly structured and executed them in PostgreSQL to verify their accuracy.

By the end of the day, I had gained valuable knowledge about compilers, programming languages, JavaScript, Git, and PostgreSQL. Each task helped me develop important skills that will be useful in my future projects. I am excited to continue learning and applying these concepts in real-world scenarios.