

Report on Today's Learning

Today was my seventh day of internship at Surfboard Payments. the day I learned important concepts about computing and JavaScript, which helped me improve my understanding of programming and problem-solving.

One of the key topics I explored today was the Eulerian cycle. It is a concept in graph theory that helps solve certain types of problems. An Eulerian cycle is a path in a graph that goes through every edge exactly once and returns to the starting point. Understanding this concept helps solve problems that require visiting all paths efficiently. I learned different types of problems and their solutions. The first type includes problems that have only one correct solution. The second type consists of problems that have multiple possible solutions. The third type involves problems that require trial and error, where solutions are found through testing and experimentation instead of a fixed method.

I also learned about how computers work. A computer processes information in a structured way : Input -> Memory + Process -> Output. This means data is first received as input, then stored temporarily and processed, and finally, an output is produced. Computers understand everything in binary form, which uses only 0s and 1s. These numbers represent electrical signals, where 1 means high voltage (5V) and 0 means no voltage (0V). All types of text, images, and code are eventually converted into this binary language so that the computer can process them.

Another major topic I learned was how JavaScript reads and runs code. My mentor explained JavaScript functions in detail. Functions are blocks of code that perform specific tasks. In JavaScript, functions can take parameters, which are input values used in the function. I also learned about function invocation, which is the process of executing a function. There are different ways to call functions in JavaScript, such as call, apply, and bind methods. The call method allows a function to be executed with a specific `this` value and individual arguments. The apply method works similarly but takes arguments as an array. The bind method creates a new function with a permanently set `this` value, which helps control how functions are used.

Another important concept I learned today was closures. A closure is a function that remembers the variables from its parent function even after the parent function has finished running. This is a useful feature in JavaScript because it allows functions to keep their data, which is helpful for creating private variables and maintaining state.

I also learned about the event loop in JavaScript. This is an important part of how JavaScript handles tasks that take time, such as fetching data from a server or setting a timer. Unlike other programming languages that execute code line by line, JavaScript uses the event loop to handle multiple tasks efficiently without blocking other code from running. This makes JavaScript good for building fast and interactive web applications.

Another topic my mentor explained was the difference between `var`, `let`, and `const` in JavaScript. The `var` keyword is function-scoped, which means it is only accessible within the function where it is defined. It can also be redeclared and updated. The `let` keyword is block-scoped, meaning it only exists inside the block where it is defined, which prevents errors caused by redeclaring variables. The `const` keyword is also block-scoped, but it has an additional rule—it cannot be changed once it is assigned. Understanding these differences helps in writing better and more efficient JavaScript code.

Overall, today's session helped me understand JavaScript and computing concepts more clearly. Learning how computers process information, along with JavaScript functions, closures, and the event loop, has given me more confidence in coding. These concepts build on what I have learned before and will help me tackle more advanced programming challenges in the future.