ANSWER 1-

In JavaScript, synchronous and asynchronous refer to different ways of executing code and handling operations.

Synchronous operations in JavaScript are executed one after another in a blocking manner. When a synchronous operation is encountered, the program execution pauses until that operation completes before moving on to the next line of code. Imagine a line of people waiting for their turn to use a single computer. Each person has to wait for the previous person to finish before they can use the computer. Synchronous operations can potentially cause delays and block other code execution until the operation is complete.

Asynchronous operations in JavaScript, on the other hand, do not block the execution of other code. They allow the program to continue running while the operation is being processed in the background. Instead of waiting for the operation to finish, an asynchronous operation initiates the process and then moves on to the next line of code without pausing. When the operation completes, a callback function is executed to handle the result. Imagine a group of people using different computers simultaneously. Each person can work independently without waiting for others to finish. Asynchronous operations are useful for tasks that may take some time to complete, such as making API requests or reading files.

In summary, synchronous operations in JavaScript execute one after another, blocking the program until each operation is complete. Asynchronous operations, on the other hand, allow the program to continue running while operations are processed in the background, using callback functions to handle results once they are ready.