Execution context could be defined as an abstract concept which determines about the fact that how information about the function execution within a program is held.

It consist of all the data regarding the working of a program. Which include variables, functions and other necessary things.

Every time a function is called a new execution context is created and at the same time. It will be pushed inside the call stack.

There are two types of execution contexts in JavaScript:

1. Global Execution Context: It represents the default context when the code is executed outside of any function. The global execution context sets up the global scope, which includes global variables and functions accessible throughout the code.
2. Function Execution Context: Whenever a function is invoked, a new function execution context is created. It includes the function's arguments, local variables, and references to its outer environment (lexical scope). Each function execution context has its own scope, and it is pushed onto the execution stack.

[Diagram goes here](https://excalidraw.com/#json=Tgztk9uGRSHQtBiAc9_ua,2QFhF_AiQzRYsVlKxHSnow) : (Click to access the Link as well)

