Javascript:

Focus on the name value pairs.

Every object is just a collection of name value pairs.

Javascript Engine - parses the data

Base Execution context is the Global Execution Context.

Global standard definition

Global Execution creates 2 things: Global object and the \*this\* variable when it runs.

When you load the javascript, the javascrip engine stars, it creates the Global Base Execution for the environment.

It creates a Global Javascript object for everything to run in, and it creates the \*this\* variable to act as a pointed to point back in on itself.

\*This\* refers to the Window object, which then contains the key value pairs for the object. \*this\* points to the existing object being referenced.

Global is defined as : “Not inside a function”

Invocation: - to run a function

Function ()

Function myFunction(){

}

Function yourFunction (){

myFunction()

}

Every function creates an execution context phase:

Every function call creates an execution invocation for the execution stack.

The item on the top of the stack is the one currently running and being interpreted.