**SCOPE:** Es el alcance que va a tener una variable dentro del código. En otras palabras, el Scope se encargará de decidir a que bloques de código va a acceder una variable.

**GLOBAL SCOPE :** No están dentro de funciones o bloques, por lo tanto se puede acceder a ellas de manera global.

* Con var podemos re-asignar una variable pero es una mala práctica.
* Con let y const no podemos, aparecerá un error.
* Es una mala práctica crear una variable sin las palabras reservadas: var, let y const.
* Si se asigna una variable dentro de una función sin las palabras reservadas será una variable global.
* La doble asignación de una variable también es una mala práctica.

**EL LOCAL SCOPE:** se refiere a la variable o funcion que esta dentro de un bloque o funcion especifica. Solo se pueden acceder a ellas (ejecutar o llamar) dentro del entrono en donde conviven.

Puede darse en una función o en un bloque de código

**EL AMBITO LEXICO:** se refiere a que una funcion puede acceder a una funcion o variable fuera de ella.Cada nivel interno puede acceder a sus niveles externos hasta poder alcanzarlas.

LET Y CONST son de bloque a diferencia de VAR que es de función. Si no están en el mismo bloque por mas que estén dentro de la función no va a poder acceder y solo podemos ver VAR

**Variables privadas con Closures:** JS por su naturaleza no fomenta el uso de datos privados pero por medio de los Closures podemos crear valores que solo puedan ser accedidos por medio de métodos, que no van a estar disponibles fuera de esta función.

**HOISTING:** Eleva las declaraciones, esto pasa en el momento en que se compila nuestro código antes de ser interpretado por el navegador.. solo en las declaraciones

De esta forma podemos asignar nuestros valores o acceder a un valor que previamente no ha sido declarado dentro de esta estructura.