**Hoisting**

**Hoisting** es un comportamiento en JavaScript donde las declaraciones de variables y funciones son elevadas al inicio de su contexto de ejecución antes de que el código sea ejecutado, solo la declaración es movida al inicio, mientras que la inicialización permanece en la ubicación original.

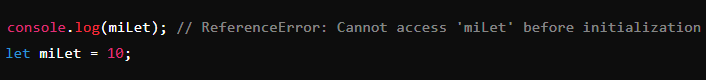
Internamente el motor de JavaScript procesa primero todas las declaraciones de variables y funciones antes de ejecutar el código, sin importar dónde se encuentren en el código fuente. Este comportamiento permite que puedas usar variables y funciones antes de declararlas en el código.

**Hoisting con var:** La declaración es elevada, pero la asignación de valor no. La variable es accesible, pero tiene el valor undefined hasta que se asigne.

**Texto

Descripción generada automáticamente**

**Hoisting con let y const:** Las declaraciones son elevadas, pero no puedes acceder a ellas antes de la línea de código donde se declaran, debido al Temporal Dead Zone (TDZ).

****

**Texto

Descripción generada automáticamente con confianza media**

**Hoisting en declaraciones de función:** En las declaraciones de función, no existe una asignación. Todo el bloque de la función es elevado. Esto quiere decir que podemos llamar a la función antes de declararla en el código y funcionará correctamente

Texto

Descripción generada automáticamente

**Hoisting en expresiones de función:** En las expresiones de funciones, existe una asignación, **el proceso de asignación está libre de hoisting**. En este caso, solo es elevado el identificador de la función, no el contenido de esta. Esto significa que la variable existe desde el inicio del contexto, pero no tiene un valor asignado hasta que la línea de código se ejecuta. Si intentas llamar a la función antes de la asignación, obtendrás un error.

Texto

Descripción generada automáticamente