

Podcast

Disciplina: **Linguagem e padrões Web**

Título do tema: **Linguagem JavaScript: do básico ao avançado.**

Autoria: **Anderson da Silva Marcolino**

Leitura crítica: **Gabriela Silveira**

Abertura:

Olá, ouvinte! No *podcast* de hoje vamos falar um pouco sobre *hoisting* ou içamento das declarações de variáveis em JavaScript.

Para àqueles que estão iniciando em JavaScript ou que já conhecem a linguagem o *hoisting*, em um momento ou outro sempre acaba gerando transtornos, por isso precisamos saber do que se trata. Toda as declarações de variáveis, objetos ou função são içadas, por padrão, para o topo. Ou seja, ainda que declaremos uma variável global no meio de nosso código, ela será levada ao topo do código e declarada antes da interpretação do restante do código. Para quem se perguntou, sim! Uma variável em JavaScript pode ser usada antes de ser declarada. Isso se deve ao comportamento de *hoisting* que garante que a variável seja içada e iniciada antes da interpretação do código JavaScript. Por exemplo, nós poderíamos começar um código com x recebendo 5 e imprimir por meio do innerHTML o conteúdo da variável x e, por último declarar a variável com var x. É importante mencionar que, se a variável foi declarada no escopo global, ela será levada para o topo do código. Se for local – dentro do corpo de uma função, por exemplo – ela será movida para o começo do corpo da respectiva função. O mesmo ocorre com as variáveis declaradas com let e const, mas diferente de uma variável declarada com var ela não será inicializada até ser declarada. Utilizar uma variável declarada com let antes de sua declaração e atribuição resultará em um erro de referência, o *ReferenceError* (em inglês). Já, ao utilizar uma const antes de sua declaração e atribuição não resultará em erros, mas o código não irá executar. Finalmente, é importante mencionar que uma variável declarada com var pode ser inicializada, ou seja, pode receber um valor posteriormente, ou o contrário. Porém, isso implica em possíveis erros ou comportamentos inesperados. Para se livrar da possível dor de cabeça em se preocupar com o *hoisting* a melhor solução é sempre declarar as variáveis no início de cada escopo. Agora que sabe sobre o içamento das declarações em JavaScript, ou na ECMA Script – quando chamada pelo seu nome técnico, nunca mais terá tais problemas! Lembre-se, nada melhor do que testar e ver os içamentos na prática para identificar se há exceções para seu uso em seus projetos!

Fechamento:

Este foi nosso *podcast* de hoje! Até uma próxima oportunidade e bons estudos!