

Podcast

Disciplina: **Linguagem e padrões Web**

Título do tema: **Implementação de Páginas Web com HTML, CSS e JavaScript.**

Autoria: **Anderson da Silva Marcolino**

Leitura crítica: **Gabriela Silveira Barbosa**

Abertura:

Olá, ouvinte! No *podcast* de hoje vamos falar um pouco sobre boas práticas quando vamos programar em JavaScript.

As boas práticas existem para nos ajudar a errar menos. No contexto de uma equipe de desenvolvimento, auxiliará na melhoria da manutenção, na redução de esforços para realizar tais manutenções, além de gerar um código mais limpo. Em alguns casos, algumas dessas boas práticas auxiliam no desempenho, na execução, ou interpretação do código, como é o caso do JavaScript. Ouçamos algumas destas boas práticas para JavaScript. Reduza o uso de variáveis globais, incluindo todos os tipos de dados, objetos e funções, uma vez que variáveis globais podem ser sobrescritas por outro código no mesmo script. Dê preferência para variáveis locais e aprenda a utilizar *closures*. Lembre-se que para declarar uma variável local você deverá usar a palavra *var* ou *let*. Ainda no contexto das variáveis, sempre as declare no início do código. Isso permitirá um código mais limpo, possibilitará a centralização de declarações para uma eventual verificação, tornará facilmente verificável se alguma variável foi declarada de modo global e reduzirá a possibilidade de uma reutilização de um mesmo nome de variável mais uma vez no código. E não podemos esquecer que assim evitamos problemas com o *hoisting*! Uma outra boa prática é sempre inicializar as variáveis! O que deixará o código limpo e evitará valores indefinidos (*undefined*). Quanto ao desempenho, busque nunca declarar objetos do tipo números, caracteres (*strings*) e booleano. Sempre trate números, caracteres e booleanos como valores primitivos. Cuidado também com a questão das variáveis poderem alterar seu tipo de dado, o que pode gerar uma conversão de um caractere em um número e, conseqüentemente, na indicação de que não se trata de um número (*not a number*). Outra boa prática é sempre utilizar para comparação três sinais de igual e nunca dois, visto que dois sempre converte os dados para permitir que os tipos sejam iguais. O operador formado pelos três sinais de igual força a comparação entre o tipo e também entre o valor. Finalmente, uma última boa prática refere-se a sempre declarar um valor padrão para os argumentos ou parâmetros passados em uma função. Argumentos que não recebem valores são tidos como indefinidos (*undefined*), o que pode gerar problemas no seu código. Nestes casos você pode indicar como valor o zero. Espero que tenha gostado destas

boas práticas! Apesar de simples são essenciais para garantir melhor qualidade quando programamos para a web.

Fechamento:

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