

Variáveis

Transcrição

Uma variável pode guardar praticamente o que você quiser: um número, uma string ou um outro pedaço de código. Vamos guardar a informação da soma das idades em uma variável:

```
var eu = 25;
var adriano = 26;
var paulo = 32;

var total = eu + adriano + paulo;
document.write("A soma das idades é: " + total);
```

A `var` é uma palavra especial no JavaScript e esse tipo de palavras chama-se **palavras-chave** de uma linguagem. Ela possui um tratamento diferenciado, nesse caso, porque cria uma variável. Não é necessário preocupar-se agora com a sintaxe, no decorrer das lições isso ficará cada vez mais claro.

O operador igual (`=`) não funciona exatamente como na matemática, por exemplo, `2012 = ano` não é o mesmo que `ano = 2012`. Dizemos que o operador `=` **atribui** o valor `2012` à variável, que fica a esquerda do `=`.

O uso das aspas é o que define aquilo que será impresso. Se não há aspas dentro dos parênteses do `document.write(...)`, o JavaScript busca o valor da variável. Caso sejam utilizadas aspas, ele lê o que está dentro delas acompanhado de uma sequência de caracteres (`string`). É importante que o aluno teste e veja essa diferença:

```
var ano = 2012;
document.write("ano");
document.write(ano);
```

O que acontece?

A ideia é utilizar variáveis o tempo inteiro. Portanto, vamos praticar!

Com mudanças no código, é possível calcular a média de idade de sua família de uma forma organizada e sem copiar os números pra dentro do `document.write`. Por exemplo:

```
document.write((eu + adriano + paulo) / 3);
```

Outra forma de calcular a média seria quebrar esse processo em passos. É bastante comum criar variáveis a mais para ajudar a legibilidade. Um programador costuma trabalhar em equipe, outros colegas estarão sempre lendo, modificando e trabalhando com as mesmas linhas de código, portanto, é necessário sempre criar códigos organizados e legíveis.

Vamos criar uma variável para a soma e outra para a média:

```
var total = eu + adriano + paulo;
var media = total / 3;
document.write(media);
```

Na seção anterior, escrevemos o código de forma legível. Caso queira, crie o arquivo `testando_idades_com_variaveis.html` de forma a praticar o código uma última vez, depois passaremos para o próximo problema. Inicialmente é preciso declarar a idade de cada pessoa:

```
<script>
var eu = 25;
var adriano = 26;
var paulo = 32;
//Depois calculamos os dados que precisamos: o total e a média.

var total = eu + adriano + paulo;
var media = total / 3;
//Por último imprimimos a média e fechamos a tag de `script`:

document.write("A média de idade é " + media);
</script>
```

Seu código está funcionando?

É preciso ter atenção com os nomes das variáveis. Você precisa utilizá-las da mesma forma como as declarou. O JavaScript diferencia, inclusive, maiúsculas de minúsculas. Para entender como isso funciona erre propositalmente o nome de uma variável para verificar qual mensagem de erro aparece no JavaScript Console. Por exemplo, escreva `med` em vez de `media`. Exemplo com problema:

```
document.write("A média de idade é " + med);
```

É comum digitar o nome de uma variável de forma errada. Portanto, fique atento e habitue-se a utilizar o JavaScript Console para auxiliar na busca por erros. As mensagens de erro notificam as falhas em inglês e algumas vezes elas não serão tão específicas quanto gostaríamos, mas sempre dizem com exatidão em que linha do código houve o problema.

Para arredondar a média de idades podemos utilizar o `Math.round(numero)`. Altere seu código para:

```
document.write("A média de idade é " + Math.round(media));
```

O `Math.round` pega o valor que está dentro dos parênteses e arredonda ele. Nós utilizamos o novo valor para juntar (concatenar) com o restante da frase que queremos mostrar.

Agora, vamos fazer um exercício completamente novo:

Vimos que utilizar variáveis ajudou de algumas formas:

- não precisamos mais copiar e colar tantas informações;
- alguns trechos ficam mais legíveis;
- as variáveis podem ir além, pois, economizam na repetição de linhas de código;

Atenção! Nessa seção não é necessário alterar seu código. Basta acompanhar os recursos novos que estamos aprendendo! A prática será feita durante a revisão, que virá logo a seguir.