



Desenvolvimento Web Completo 2022


Por Jorge Sant Anna, Programador
Jamilton Damasceno, Analista de Sistema e Professor




Seção 4: JavaScript




Aula 17: Funções


- 
- Uma função tem como objetivo, encapsular um determinado bloco de códigos que possui um objetivo bem definido e, através das chamadas das funções, podemos recuperar essas instruções para reutilizá-las em diversas partes das nossas aplicações.


- 
- A declaração de uma função é bastante simples. Ela começa com a palavra reservada “function” seguida do nome da função. Após o nome, precisamos abrir e fechar parênteses, onde podemos encaminhar os parâmetros, que também são chamados de argumentos. Na sequência, abrimos chaves, para a inclusão do bloco de códigos da função e, por fim, fechamos chaves, indicando o término do bloco de códigos da função.


Exemplo

```
//Função  
function calcularAreaTerreno(largura, comprimento) {  
    var area = largura * comprimento;  
  
    return area;  
}
```

- 
- Os parâmetros, ou argumentos, funcionam como uma espécie de entrada de dados para a função.
 - Podemos não passar nenhum parâmetro, ou mesmo, passar quantos parâmetros forem necessários para a função, basta separar os parâmetros por vírgula.

- 
- Vale lembrar que os parâmetros são variáveis, portanto, a definição do nome de um parâmetro deve seguir a regra que já conhecemos para a definição de nome de variáveis.
 - Perceba que os parâmetros estão definidos na função, constituindo a sua respectiva assinatura e os mesmos são trabalhados dentro do bloco de código da função. Por isso que os parâmetros funcionam como entrada de dados para as funções.

- 
- E já dentro do escopo da função, podemos criar variáveis, criar estruturas de repetição, criar estruturas condicionais, ou seja, podemos trabalhar com todos os recursos da linguagem de programação para a criação da lógica da nossa função. Inclusive, é possível chamar outras funções. Por isso, também, falamos que funções são subprogramas dos nossos programas.

- 
- Além disso, as funções podem ser de dois tipos diferentes, elas podem ser do tipo “void”, ou seja, quando a chamada irá apenas processar alguma lógica, ou do tipo “com retorno”, basicamente falando, quando a função é chamada, além de processar alguma lógica, ela terá um retorno de um valor pré-definido ou, até mesmo, uma variável de qualquer tipo, para o script que efetuou a chamada da função.
 - Nesse segundo caso, o script solicitante poderá ter a posse do valor retornado pela função, continuar o seu respectivo processamento.