



JAVASCRIPT.



MATERIAL DE APOIO

Disciplina: JavaScript

Nome da aula: Funções

Professor(a): Marcelo Estruc

Introdução

Uma função é uma unidade de código que geralmente é definida por seu papel dentro de uma estrutura de código maior. Especificamente, uma função contém uma unidade de código que funciona em várias entradas, muitas das quais são variáveis, e produz resultados concretos envolvendo alterações em valores de variáveis ou operações reais com base nas entradas.

Objetivos da aula

- Definir os conceitos de Funções.
- Demonstrar a implementação de Funções na linguagem Javascript.

Resumo

Uma função é um bloco em JavaScript que faz a mesma coisa repetidamente, sempre que você chamar a função. Ele economiza codificação repetitiva e torna seu código mais fácil de ser compreendido. Ou seja, as funções são blocos de construção fundamentais em JavaScript.

Uma função é um procedimento em JavaScript - um conjunto de instruções que executa uma tarefa ou calcula um valor, por exemplo. Para usar uma função, você deve defini-la em algum lugar no escopo do qual você quiser chamá-la.

Em seu site, suponha que haja várias páginas nas quais você deseja exibir um alerta que informa ao usuário a hora atual. O código pode ficar assim

```
var now = new Date();  
var theHr = now.getHours();  
var theMin = now.getMinutes();  
alert("time: " + theHr + ":" + theMin);
```

Fonte: Autoral

Você pode escrever esse bloco de código repetidamente, sempre que precisar. Ou você poderia escrevê-lo uma vez como uma função, nomeando-o, digamos, tellTime. Depois disso, este é o único código que você precisará chamar para fazer todo o bloco executar.

A definição da função (também chamada de declaração de função) consiste no uso da palavra-chave function, seguida por:

- Nome da Função;
- Lista de argumentos para a função, entre parênteses e separados por vírgulas;
- Declarações JavaScript que definem a função, entre chaves { }.

Abaixo um exemplo de uma função:

```
function tellTime() {  
    var now = new Date();  
    var theHr = now.getHours();  
    var theMin = now.getMinutes();  
    alert("Current time: " + theHr + ":" + theMin);  
}
```

Fonte: Autoral

Sempre que o JavaScript vê essa instrução sendo chamada, o bloco de contagem de tempo é executado. É assim que você escreve um código único, como uma função, que pode ser usada diversas vezes:

```
tellTime();
```

Fonte: Autoral

Uma das coisas realmente úteis sobre funções é que podemos passar dados para dentro de uma função. Se você colocar alguns dados entre os parênteses, você pode passar esses dados para a função que ela usará quando for executado.

```
function greetUser(greeting) {  
    alert(greeting);  
}
```

Fonte: Autoral

Para usar essa função poderíamos fazer da seguinte forma:

```
greetUser("Olá pessoal")
```

Então, agora nós preenchemos os parênteses do código de chamada e da definição da função. Os parênteses do código de chamada contêm um argumento. No exemplo, o argumento é o "Olá pessoal". E, como você pode ver no exemplo acima, os parênteses da função agora contêm uma variável.

Uma variável dentro dos parênteses em uma instrução de função é conhecida como parâmetro. O nome do parâmetro é você que define. Você pode dar a ela qualquer nome que seja legal para uma variável. Então, você pode usá-lo para realizar algo no corpo da função.

Na linha 2 no exemplo acima, usei para especificar a mensagem em um alerta.

Você não precisa declarar uma variável quando ela é usada como parâmetro em uma função. Quando a variável é informada como parâmetro, a declaração é implícita.

O argumento na instrução que chama a função —"Olá pessoal"—são os dados que são passados para a função. O parâmetro dentro dos parênteses da função captura os dados que são passados, ele agora contém "Olá pessoal." Em outras palavras, a string `greeting`, especificado na chamada de função, é atribuído à saudação na função. Então essa variável, `greeting`, é usada para especificar a mensagem na instrução de alerta.

O valor passado para uma função é usado no corpo da função para realizar algo. Nesse caso, o valor é usado para especificar a mensagem no alerta.

No exemplo, o argumento é uma string, mas também pode ser uma variável. No código a seguir, declaro a variável `greeting` e atribuo a ela o valor "Olá pessoal". Então, em vez de usar a própria string como argumento na chamada da função, uso a variável.

A declaração de função não é a única maneira de definir uma nova função. Você pode definir sua função dinamicamente usando o construtor **Function()** junto com o novo operador.

A seguir está a sintaxe para criar uma função usando o construtor `Function ()` junto com o novo operador:

```
var variablename = new Function (Arg1, Arg2 ..., "corpo da função");
```

Fonte: Autoral

O construtor `Function ()` espera qualquer número de argumentos de string. O último argumento é o corpo da função, ele pode conter instruções JavaScript arbitrárias, separadas umas das outras por ponto e vírgula.

Observe que o construtor `Function ()` não recebe nenhum argumento que especifique um nome para a função que ele cria. As funções não nomeadas criadas com o construtor `Function ()` são chamadas de **funções anônimas**.

```
var func = new Function("x", "y", "return x*y;");
function secondFunction() {
    var result;
    result = func(10,20);
    document.write ( result );
}
```

Fonte: Autoral

Referência Bibliográfica

FLANAGAN, David. **JavaScript: O Guia Definitivo**. 6ª Ed. Porto Alegre: Bookman, 2013.

FREEMAN, Eric. **Use a cabeça!**: programação JavaScript. 1ª Ed. São Paulo: Alta Books, 2016.

Exercícios

- 1) Sobre as funções, é correto afirmar que:
 - a) permitem que se escreva o código em um único módulo.
 - b) podem transformar um grande programa em várias funções pequenas e gerenciáveis.
 - c) são chamadas apenas uma vez durante a execução do programa.
 - d) não podem ser criadas pelo programador, que utiliza apenas as já existentes na linguagem.
 - e) não existem sem parâmetros.

- 2) Ainda sobre as funções no JavaScript, assinale o item correto:
 - a) podem existir mais de uma com o mesmo nome.
 - b) possuem no máximo dois parâmetros.
 - c) no JavaScript, utilizamos o nome function para defini-las.
 - d) usam colchetes para definir um bloco de instruções.
 - e) podem ser chamadas de qualquer parte do programa.

- 3) Sobre os parâmetros de uma função, é correto afirmar que:
 - a) são separados por ponto e vírgula.
 - b) a lista de parâmetros vem dentro de aspas.
 - c) o nome da função inicia com um número.
 - d) podemos operar e manipular os parâmetros de qualquer forma.
 - e) parâmetros devem ser sempre numéricos.

4) Na declaração de retorno de uma função podemos ter as características de...

Assinale o item correto:

- a) ser obrigatório para todas as funções.
- b) deve ser a primeira declaração em uma função.
- c) pode retornar o resultado de uma expressão numérica.
- d) sempre opera com números.
- e) nenhuma das respostas anteriores.

Gabarito

1. A **Letra B** é a alternativa correta, pois as funções são utilizadas para modularizar o código em partes menores e mais fáceis de manter.

2. A **Letra C** é a alternativa correta, pois `function` é o comando utilizado para definir uma função em JavaScript.

3. A **Letra D** é a alternativa correta, pois os parâmetros de uma função podem ser manipulados de várias maneiras diferentes, como números, por exemplo, podem ser utilizados em operações aritméticas ou lógicas.

4. A **Letra C** é a alternativa correta, pois o retorno de uma chamada a uma função pode ser o resultado de uma expressão.