



INTRODUÇÃO AO DESENVOLVIMENTO WEB

PROF^a. M.Sc. JULIANA H Q BENACCHIO

Definição de funções

- Encapsular o código em funções é uma importante funcionalidade para reaproveitamento de código e organização do mesmo. No javascript uma função é definida da seguinte forma:

```
<script>  
    function bemVindo() {  
        alert("Seja bem vindo");  
    }  
</script>
```

Chamada de funções

- Para chamar a função, podemos associar um evento qualquer e passar o nome da função que será executada.
- Por exemplo, para chamar a função **bemVindo()**:

```
<body>  
  <input type="button" value="Clique aqui"  
        onclick="bemVindo()">  
</body>
```

Função com parâmetros

- É possível passar valores para a função na forma de argumentos. Por exemplo:

```
<script>
  function bemVindo(nome) {
    alert("Seja bem vindo " + nome + "!");
  }
</script>

<body>
  <input type="button" value="Clique aqui"
    onclick="bemVindo('João')">
</body>
```

Retorno de valores

- Funções podem retornar valores. Por exemplo:

```
<script>
    function soma(a,b){
        var resultado = a + b;
        return resultado;
    }
    var total = soma(2,2);
    document.write("Valor total = " + total);
</script>
```

Exemplo

```
<html>
<head>
  <title>Exemplo Funções</title>
  <meta charset="utf-8">
  <script>
    function altera(tag){
      tag.style.backgroundColor = "yellow";
    }
    function reinicia(tag){
      tag.style.backgroundColor = "white";
    }
    function altera2(tag, corFundo, corTexto){
      tag.style.backgroundColor = corFundo;
      tag.style.color = corTexto;
    }
  </script>
  <style>
    div{
      border: 1px solid black;
```

Exemplo

```
        width: 150px;
        height: 150px;
    }
</style>
</head>

<body>

    <div onmouseover="altera(this)" onmouseout="reinicia(this)">
</div>

    <p onmouseover="altera(this)" onmouseout="reinicia(this)">
    Texto no paragrafo</p>

    <div onmouseover="altera2(this, 'black', 'white')"
    onmouseout="altera2(this, 'white', 'black')">Texto na div</div>

</body>
</html>
```

Exercícios

- 1) Faça uma função javascript que monitora o clique do usuário sobre uma imagem, e a cada novo clique aumenta em 10 pixel o tamanho da mesma. (dica: use a propriedade **width**). Quando o mouse sai de cima da imagem, a mesma volta ao tamanho original.

Clique na imagem para ampliá-la



Exercícios

2) Faça uma função javascript que quando o usuário clica na imagem da estrela, a mesma é rotacionada 15 graus em sentido horário. A cada clique o ângulo é incrementado em 15 graus sucessivamente. Dica: use a propriedade: **transform = "rotate(15deg)";**

Clique na imagem para rotacionar



Exercícios

3) Faça uma função javascript associada a um botão, que quando clicado gera a seguinte tabela:

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Use **for** para montar essa tabela.

Exercícios

4) Faça uma função javascript associada a um botão, que quando clicado gera a seguinte tabela:

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Use **for** para montar essa tabela.

Posições pares tem fundo amarelo

Posições ímpares tem fundo azul claro.

Exercícios

5) Faça uma função javascript associada a um botão, que quando clicado gera a seguinte tabela. Dica: use dois for para gerar as linhas e as colunas.

1	2	3	4	5	6	7	8	9	10
2	3	4	5	6	7	8	9	10	11
3	4	5	6	7	8	9	10	11	12
4	5	6	7	8	9	10	11	12	13
5	6	7	8	9	10	11	12	13	14
6	7	8	9	10	11	12	13	14	15
7	8	9	10	11	12	13	14	15	16
8	9	10	11	12	13	14	15	16	17
9	10	11	12	13	14	15	16	17	18
10	11	12	13	14	15	16	17	18	19

Exercícios

6) Faça uma função que gera a matriz abaixo. As células com valores ímpares devem ser pintadas com fundo amarelo.

1	2	3	4	5	6	7	8	9	10
2	3	4	5	6	7	8	9	10	11
3	4	5	6	7	8	9	10	11	12
4	5	6	7	8	9	10	11	12	13
5	6	7	8	9	10	11	12	13	14
6	7	8	9	10	11	12	13	14	15
7	8	9	10	11	12	13	14	15	16
8	9	10	11	12	13	14	15	16	17
9	10	11	12	13	14	15	16	17	18
10	11	12	13	14	15	16	17	18	19

Exercícios

7) Faça uma função que gera a tabuada de 1 à 10.

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

Exercícios

8) Faça uma função no javascript que gera a tabela ao lado. Observe o padrão de evolução dos números, onde a cada nova linha, diminui-se uma célula.

2	3	4	5	6	7	8	9	10
3	4	5	6	7	8	9	10	
4	5	6	7	8	9	10		
5	6	7	8	9	10			
6	7	8	9	10				
7	8	9	10					
8	9	10						
9	10							
10								