



Disciplina: Introdução ao DOM

Prof.: Arcanjo Miguel Mota Lopes

1) A partir do código HTML e CSS abaixo, faça:

```
/* CSS Questao 1 */
.container {
  display: grid;
  grid-template-columns: 49% 49%;
  grid-template-rows: 10em 10em;
  gap: 1%;
  background-color: AliceBlue;
  margin: 2em;
  padding: 2em;
}
.cell {
  display: flex;
  align-items: center;
  justify-content: center;
  background-color: blue;
  font-size: 3em;
}
```

```
<!-- HTML Questao 1 -->
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Document</title>
</head>

<!-- HTML Questao 1 -->
<body>
  <div class="container">
    <div id="divA" class="cell">A</div>
    <div id="divB" class="cell">B</div>
    <div id="divC" class="cell">C</div>
    <div id="divD" class="cell">D</div>
  </div>
</body>
</html>
```

- a. Implemente um evento 'click' que alterne a div 'D' entre as cores de fundo (192,192,192) para (218,165,32).



#DICA: use propriedade `style` e `event.target`

- b. Implemente um evento 'mouseover'. A cada movimento do mouse na div 'C', sua cor de fundo muda de "Blue " para "#00FF00", "#FF4500", "(255,215,0)", "(211,211,211)" e de volta para ""Blue ".

#DICA: pode utilizar switch-case

- c. Implemente dois manipuladores de eventos para mouse na div 'B'. Quando o ponteiro do mouse entra na área, sua cor de fundo muda para "vermelho"; ao sair da área, sua cor de fundo muda para "verde".

2) Copie o seguinte código HTML. Adicione os seguintes elementos ao container usando APENAS JavaScript e os métodos DOM.

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <title>DOM</title>
</head>
<!-- The DOM -->
<body>
  <h1>
    WEB ACADEMY
  </h1>
  <div id="container">
    <div class="content">
      Introdução ao DOM (document object model)
    </div>
  </div>
</body>
</html>
```

- a. Crie um elemento <p> com texto de cor **vermelho** escrito "Hierarquia de Objetos"
- b. Crie um elemento <h3> com texto de cor **azul** escrito "Windows Object"
- c. Crie uma <div> com borda 2px, solid, black e background: pink. Além disso, adicione os seguintes **filhos**:
- Elemento <h1> que contém "Título da DIV";
 - Elemento <p> que contém "Parágrafo da DIV";

3) Copie o seguinte código HTML.



```
<!DOCTYPE html>
<html>
<head>
  <meta charset = "UTF-8">
  <title>
    DOM
  </title>
  <style>
    .body-data {
      border:#3CB371 2px solid;
      background-color: #03a9f400;
      text-align: left;
      padding-left: 20px;
      padding-bottom: 20px;
      height: auto;
      width: auto;
    }
    .resultText {
      margin: 0 0 3px 0;
      padding: 0px;
      display: block;
      font-weight: bold;
    }
    .heading {
      font-weight: bold;
      border-bottom: 2px solid #708090;
      font-size: 15px;
      width: 98%;
    }
  </style>
</head>
<body>
  <div class = "body-data" >
    <div class = "heading" >
      <h2> Evendo com 'blur' </h2>
      <span>Click no input para disparar o Evento </span>
    </div>
    <div class = "resultText" >
      <br>
      <p> Campo de Validação </p>
      <textarea id="myText" rows="4" cols="50"></textarea>
      <p id = "result" ></p>
    </div>
  </div>
</body>
</html>
```



- a. Adicione um evento 'blur' no campo <textarea> que faça uma verificação que aceite somente texto. Se o usuário inserir números, seja sozinho ou junto com a string mude o background para vermelho, caso tenha somente texto, mude para verde.
- b. Escreva no id= 'result' os seguintes feedbacks, exemplo:
[abc123] -> "Input contém números"
[abcd] -> "Input não contém número",
[] -> "Input está vazio"
#Dica: use a função `RegExp()`¹ para verificar o padrão do texto.
- c. Adicione um evento que não permita que o usuário realizar a ação CTRL+V na <textarea>.
#Dica: pode utilizar Eventos de Teclado (combinando teclas) ou Bloqueio de ação padrão (`preventDefault()`)

4) Copie o seguinte código HTML.

```
/* CSS Questão 4 */
html, body {
  margin: auto 0;
  font-size: 18px;
  font-family: 'BioRhyme', serif;
}

body {
  background-size: cover;
  display: grid;
  justify-items: center;
}

.header {
  display: inline-flex;
}

h1 {
  padding: 0 1.5rem;
  font-family: 'Arial', cursive;
  letter-spacing: 3px;
  font-weight: 300;
  font-size: 2.8em;
}

#first { display: flex; align-self: baseline;}
```

¹ <http://devfuria.com.br/javascript/regexp/>



```
.list {
  background: #FFDAB9;
  padding: 2rem;
  width: 30vw;
  display: flex;
  flex-direction: column;
  align-items: flex-end;
}
.list--header { align-self: center; padding-top: 1rem;}

#userinput, #enter {
  padding: .5rem;
  font-size: .8em;
  border-radius: 5%;
}

ul { text-align: right; list-style-type: none;}

li { padding-top: .5rem; cursor: pointer;}

button {
  margin-left: 1rem;
  margin-bottom: .5rem;
  text-transform: uppercase;
  letter-spacing: 1px;
  padding: .5rem .3rem;
  cursor: pointer;
  border-radius: 30px;
}

.done { text-decoration: line-through;}
```

```
<!-- HTML Questao 4 -->
<!DOCTYPE html>
<html>
<head>
  <title>Javascript + DOM</title>
  <link href="https://fonts.googleapis.com/css?family=BioRhyme:300|Monoton" rel="stylesheet">
</head>
<body>
  <div class="container">
    <div class="header">
      <h1>Lista da Festa</h1>
    </div>
    <div class="list">
      <div class="list--header">
```

```

        <input id="userinput" type="text" placeholder="...">
        <button id="enter">Enter!</button>
    </div>
    <ul id="item--List">
        <li>
            <span>Chocolate</span>
            <button class="delete--button">Delete</button>
        </li>
        <li><span>Balões</span>
            <button class="delete--button">Delete</button>
        </li>
        <li>
            <span>Cupcake</span>
            <button class="delete--button">Delete</button>
        </li>
        <li>
            <span>Bolo de Aniversário</span>
            <button class="delete--button">Delete</button>
        </li>
    </ul>
</div>
</body>
</html>

```

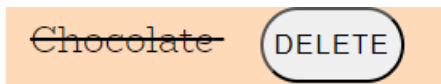
- a. Adicione um evento de click (DELETE) para os botões <button class= 'delete--button'>

#DICA: não esquecer de observar o nó pai e o nó filho da lista

- b. Adicione evento para criar novos objetos na lista. Observação: cada elemento deve ser criado com seu respectivo botão DELETE, conforme HTML acima.

#DICA: não esquecer que cada botão tem seu evento de DELETE

- c. Adicione um evento que quando o usuário clicar sobre o elemento da lista , seja adicionada a classe **'done'** do CSS. **Resultado** deve ser:



#Dica: veja a propriedade `classList`.

- d. Adicione um evento para que o novo elemento da lista seja adicionado quando o usuário pressionar a tecla ENTER.

#Dica: uma boa prática, é verificar o tamanho da string do input antes de adicionar, pois é possível que o evento adicione um elemento vazio na lista. Além disso, pode ser utilizada tabela ASCII para encontrar o código da tecla.



- e. Adicione evento para não permitir que o usuário adicione números no <input> da lista e, caso ocorra, acione um windows.alert como feedback.

#Dica: faça teste com 'blur' e 'focus',

5) Copie o HTML e CSS abaixo e, faça:

```
/* CSS Questao 5 */
.btn, .remove-btn {
  text-transform: capitalize;
  width: 100px;
  height: 10px;
  border: 1px solid black;
  padding: 5px;
  border-radius: 5px;
  cursor: pointer;
}
img { transition: display .2ms ease; cursor: pointer; }

.js-hidden { display: none; }

.js-show { display: block; }
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Questao 5</title>
</head>
<body>
  <span class='btn'>Visualizar Imagem</span>
</body>
</html>
```

- a. Adicione um eventos de 'click' ao botão '.btn', quando for clicado, uma imagem de **sua escolha** seja exibida abaixo do botão. **A imagem pode estar presente no HTML (body)** antes que o botão seja clicado, mas deve ser ocultada (veja o CSS display).
- b. Adicione um eventos à imagem. Quando a imagem for clicada, **oculte** a imagem novamente.
- #DICA:** veja o CSS js-hidden.
- c. **Crie** um segundo botão (também criado a partir de um elemento span) com a **classe= 'remove-btn'** (Veja CSS).



Quando este botão for clicado, o event listener deverá **remover** completamente a imagem da página. Não se esqueça **de remover o evento anexado do Selector 'remove-btn'**.

Praticar:

<https://www.crocoder.dev/blog/the-dom-exercises/>