

## 2. Aplicar efeitos em páginas web com jQuery

A biblioteca jQuery fornece várias técnicas para adicionar efeitos de animação a uma página da web. Isso inclui animações simples e padrão que são usadas com frequência e a capacidade de criar efeitos personalizados sofisticados, são alguns métodos para criar efeitos de animações usando esta biblioteca:

- ✓ **.hide()**, Oculte os elementos correspondentes;
- ✓ **.show()**, Exibir os elementos correspondentes;
- ✓ **.toggle()**, Exiba ou oculte os elementos correspondentes;
- ✓ **.animate()**, Execute uma animação personalizada de um conjunto de propriedades CSS;
- ✓ **.slideUp()**, Oculte os elementos correspondentes com um movimento deslizante;
- ✓ **.slideDown()**, Exiba os elementos correspondentes com um movimento deslizante;
- ✓ **.slideToggle()**, Exiba ou oculte os elementos correspondentes com um movimento deslizante;
- ✓ **.stop()**, Pare a animação em execução nos elementos correspondentes;
- ✓ **.finish()**, Pare a animação em execução no momento, remova todas as animações que estão na fila e conclua todas as animações para os elementos correspondentes.

### 2.1. Animações com métodos de Mostrar e de Ocultar

Quando há necessidade de criar os efeitos na vertente de **mostrar** e/ou **ocultar** elementos, com recurso a jQuery, é implementado pelas funções: **show()** e **hide()**, cuja s sua sintaxe é:

```
$(selector).hide({params}, speed, callback);  
$(selector).show({params}, speed, callback);
```

ONDE: **params** *é necessário e define as propriedades CSS para ser animado.*

**speed** *é opcional especifica a duração do efeito. Ele pode ter os seguintes valores: "slow", "fast", ou milissegundos.*

**callback** é opcional é uma função a ser executada após a animação completa.

Estes nomes em inglês dos métodos, correspondentes à ação, ocultar para o primeiro e mostrar para o segundo, eles podem ser aplicados a qualquer elemento em nosso DOM, Portanto, eles são bastante práticos e úteis.

**Exercício Exemplo1:** criar uma pagina web que tenha uma imagem e debaixo dessa imagem tenha um descrito da mesma, usando a função *hide* esconder os elementos, seguidamente com recurso ao método *show* animar a visualização por forma a mostrar imagem lentamente e de seguida o respectivo descrito.

```
<> exercicio_ec2_exemplo_1.html X <> animacao_css_jquery_exemplo1.html
<> exercicio_ec2_exemplo_1.html > ...
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Animacoes CSS e jQuery</title>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <script type="text/javascript" src="jquery.js"></script>
8      <style>
9        body{
10          background-color: lightblue;
11        }
12        #meuPariso{
13          border: 1px solid green;
14          width: 600px;
15        }
16      </style>
17      <script type="text/javascript" charset="utf-8">
18        $(document).ready(function(){
19          $("#meuParaiso").hide();
20          $("#meuTexto").hide();
21          $("#meuParaiso").show(3000, function() {
22            $("#meuTexto").show(2000);
23          });
24        });
25      </script>
26    </head>
27    <body>
28      <div >
29        
30        <h2 id="meuTexto">
31          Esta animação é feito pela funções hide e show do jQuery
32        </h2>
33      </div>
34    </body>
35  </html>
~
```

**Exercício Exemplo2:** Refazer o exercício anterior, agora ocultando a imagem e o texto de forma de uma animação, seguidamente mostrando os dois elementos numa animação.

```
<> exercicio_ec2_exemplo_1.html <> exercicio_ec2_exemplo_2.html X <> animacao_css_jquery_exemplo
<> exercicio_ec2_exemplo_2.html > ...
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Animacoes CSS e jQuery</title>
5          <meta charset="UTF-8">
6          <meta name="viewport" content="width=device-width, initial-scale=1.0">
7          <script type="text/javascript" src="jquery.js"></script>
8          <style>
9              body{
10                 background-color: lightblue;
11             }
12             #meuPariso{
13                 border: 1px solid green;
14                 width: 600px;
15             }
16         </style>
17         <script type="text/javascript" charset="utf-8">
18             $(document).ready(function(){
19                 $("#meuParaiso").hide(3000, function(){
20                     $("#meuTexto").hide(2000);
21                 });
22                 $("#meuParaiso").show(3000, function() {
23                     $("#meuTexto").show(2000);
24                 });
25             });
26         </script>
27     </head>
28     <body>
29         <div >
30             
31             <h2 id="meuTexto">
32                 Esta animação é feito pela funções hide e show do jQuery
33             </h2>
34         </div>
35     </body>
36 </html>
--
```

## 2.2. Mostrar e ocultar, implementado pelo o método toggle()

O que esse método faz é avaliar o estado actual do elemento e, com ele, se estiver visível, oculta e, se estiver oculto, mostra, assim como indicamos o comportamento do tipo de comutador. O melhor é que, neste método, podemos adicionar as diferentes opções de animação, com a duração e a possibilidade de fazer o callback, a sua sintaxe é:

**`$(selector).toggle(speed,callback);`**

Onde: **params** é necessário e define as propriedades CSS para ser animado.

**speed** é opcional especifica a duração do efeito. Ele pode ter os seguintes valores: "slow", "fast", ou milissegundos.

**callback** é opcional é uma função a ser executada após a animação completa.

**Exercício Exemplo3:** Criar uma pagina web que com um elemento div, que contém um conteúdo, e um botão associado a uma animação de tal forma que quando clicar no botão mostra ou oculta o conteúdo.

```
<> exercicio_ec2_exemplo_1.html <> exercicio_ec2_exemplo_2.html <> exercicio_ec2_exemplo_3.html X
<> exercicio_ec2_exemplo_3.html > html > head > script > ready() callback > click() callback
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Animacoes CSS e jQuery</title>
5          <meta charset="UTF-8">
6          <meta name="viewport" content="width=device-width, initial-scale=1.0">
7          <script type="text/javascript" src="jquery.js"></script>
8          <script type="text/javascript" src="Chart.js"></script>
9          <style>
10             body{
11                 background-color: lightblue;
12             }
13
14             #display{
15                 width:720px;
16                 height:500px;
17                 background-color: AliceBlue;
18                 border:thin solid black
19             }
20         </style>
```

```

21     <script type="text/javascript" charset="utf-8">
22         $(document).ready(function(){
23             $("#btnHideShow").click(function(){
24                 $("#display").toggle();
25             });
26         });
27     </script>
28 </head>
29 <body>
30     <div id="display">
31         <canvas id="myChart" style="width:100%;max-width:600px"></canvas>
32         <script>
33             var xValues = ['Maputo Cidade','Maputo Provinvia','Gaza','Inhambane'];
34             var yValues = [9875, 8352, 5850, 6925];
35             var barColors = ["#b91d47", "#00aba9", "#2b5797", "#e8c3b9"];
36             new Chart("myChart", {
37                 type: "pie",
38                 data: {
39                     labels: xValues,
40                     datasets: [{
41                         backgroundColor: barColors,
42                         data: yValues
43                     }]
44                 },
45                 options: {
46                     title: {
47                         display: true,
48                         text: "Representação do emprego na Zona Sul de Moçambique"
49                     }
50                 }
51             });
52         </script>
53     </div>
54     <input type="button" name="btnHideShow" id="btnHideShow"
55         value="HideShow" style="width:100px; height:30px;"/>
56 </body>
57 </html>

```

## 2.3. Animação com método animar

Um método bem interessante que pode ser muito bem explorado é o `.animate`, ele controla propriedades do CSS para realizar animações em elementos HTML. Sendo assim seu uso se torna bem simples, e possibilita ao desenvolvedor criar desde pequenos efeitos até grandes animações.

Sintaxe:

**`$(selector).animate({params},speed,callback);`**

Onde: **params** é necessário e define as propriedades CSS para ser animado.

**speed** é opcional especifica a duração do efeito. Ele pode ter os seguintes valores: "slow", "fast", ou milissegundos.

**callback** é opcional é uma função a ser executada após a animação completa.

**Exercício Exemplo4:** Criar uma pagina web que com um elemento div, que contém três complementos representando uma superfície terrestre, e duas bolas, estilizar de tal forma que as suas bolas estejam assentes na superfície terrestre, seguidamente animar um movimento de esquerda para direita a primeira bola, quando esta atinge a posição da segunda animar a segunda num movimento esquerda para direita, e a primeira de direita para esquerda, simulando um choque causa estes movimentos das bolas.

```
<> exercicio_ec2_exemplo_1.html <> exercicio_ec2_exemplo_4.html X <> exercicio_ec2_exemplo_2.html
<> exercicio_ec2_exemplo_4.html > html > head > style > #ball2
1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
4 <script src="jquery.js" type="text/javascript"></script>
5 <title>Animação com jQuery</title>
6
7 <style type="text/css">
8 #box{
9 border:1px solid #a74a0d;
10 height:250px;
11 left:10px;
12 position:relative;
13 top:20px;
14 overflow:hidden;
15 width:720px;
16 }
17
```

```

18     #earth{
19         background: #RosyBrown;
20         border-top: 20px solid #090;
21         bottom: 0;
22         height: 50px;
23         position: absolute;
24         width: 720px;
25     }
26
27     #ball1, #ball2{
28         border-radius: 50px;
29         height: 100px;
30         position: absolute;
31         top: 80px;
32         width: 100px;
33     }
34     #ball1{
35         background: #RoyalBlue;
36         left: 0px;
37     }
38     #ball2{
39         background: #DimGray;
40         left: 450px;
41     }
42 </style>
43
44 <script>
45     $(document).ready(function(){
46         $("#ball1").animate({
47             "left": "400px"
48         }, 400, null, function(){
49             $("#ball2").animate({
50                 "left": "600px"
51             }, 400);
52             $("#ball1").animate({
53                 "left": "400px"
54             }, 400);
55         });
56     });
57 </script>
58
59 </head>
--

```

```

60
61     <body>
62         <div >
63             <h2>Animação com jQuery usando função animate</h2>
64             <br/>
65         </div>
66         <div id="box">
67             <div id="ball1"></div>
68             <div id="ball2"></div>
69             <div id="earth"></div>
70         </div>
71     </body>
72 </html>

```

&lt;&gt; exercicio\_ec2\_exemplo\_4.html

&lt;&gt; exercicio\_ec2\_exemplo\_6.html 1 X

&lt;&gt; exercicio\_ec2\_exemplo\_2.html

&lt;&gt; exercicio\_ec2\_exemplo\_6.html &gt; html &gt; body &gt; button#toggle

```

1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>funcoes de deslizes em jQuery</title>
5          <script type = "text/javascript" src = "jquery.js"> </script>
6          <script type = "text/javascript" language = "javascript">
7              $(document).ready(function() {
8                  $("#toggle").click(function(){
9                      $(".target").slideToggle( 'slow', function(){
10                         $(".log").text('Deslize slideToggle foi concluido com sucesso');
11                     });
12                 });
13             });
14          </script>
15          <style>
16              h2{
17                  background-color: #bca;
18                  width:600px;
19                  border:5px solid green;
20                  padding: 10px;
21              }
22              button{
23                  padding: 10px;
24              }
25          </style>
26      </head>

```



```
27 <body style="">
28   <h2>
29     Executa as funções de deslizes com click de botões abaixo
30   </h2>
31   <button id = "toggle"> Deslize com slideToggle </button>
32   <div class = "target">
33     <img src = "imagens/paraiso4.jpg" alt = "jQuery" />
34   </div>
35   <div class = "log"></div>
36 </body>
37 </html>
```

## 2.4. Animações com métodos de deslizes

A biblioteca jQuery providencia mecanismo de mover os objectos de cima para, bem como de baixo para cima, assim como executar os dois movimentos, com recurso as funções: `slideDown`, `slideUp` e `slideToggle`, respectivamente, usando as sintaxes:

```
$(selector).slideDown(speed,callback);
$(selector).slideUp(speed,callback);
$(selector).slideToggle(speed,callback);
```

Onde:

**speed** é opcional especifica a duração do efeito. Ele pode ter os seguintes valores: *"slow"*, *"fast"*, ou *milissegundos*.

**callback** é opcional é uma função a ser executada após a conclusão do deslizamento.



## 2.5. Animações com métodos de fade

A biblioteca jQuery providencia mecanismo de criar efeitos bastante atrativos, onde se produz um fundido através da mudança da propriedade “opacity” do arquivo “CSS”. O efeito “Fade” engloba três principais métodos, designadamente `fadeIn`, `fadeOut` e `fadeTo`, e as suas sintaxes são:

```
$(selector).fadeIn(speed,callback);
$(selector).fadeOut(speed,callback);
$(selector).fadeTo(speed,opacity,callback);
```

exercicio\_ec2\_exemplo\_1.html    exercicio\_ec2\_exemplo\_7.html    exercicio\_ec2\_exemplo\_4.h

exercicio\_ec2\_exemplo\_7.html > html > head > style > button

```

1  <html>
2  <head>
3      <title>Efeitos JQuery</title>
4      <script type = "text/javascript" src = "jquery.js"></script>
5      <script type = "text/javascript" language = "javascript">
6          $(document).ready(function() {
7              $("#in").click(function(){
8                  $(".target").fadeIn( 3500, function(){
9                      $(".log").text('Transação com função FADEIN concluido');
10                 });
11             });
12             $("#out").click(function(){
13                 $(".target").fadeOut( 'slow', function(){
14                     $(".log").text('Transação com função FADEOUT concluido');
15                 });
16             });
17         });
18     </script>
19     <style>
20         h2{
21             background-color: #bca;
22             width:600px;
23             border:5px solid green;
24             padding: 10px;
25         }
26         button{ padding: 10px; }
27     </style>
28 </head>

```

```
29     <body>
30         <h2>CLIQUE NOS BOTÕES E EXECUTE EFEITOS FADE EMjQUERY</h2>
31         <button id = "out"> Execute função FadeOut </button>
32         <button id = "in"> Execute função FadeIn</button>
33         <div class = "target">
34             <img src = "imagens/paraiso2.jpeg" alt = "jQuery" />
35         </div>
36         <div class = "log"></div>
37     </body>
38 </html>
```

## Exercício de Aplicação.

Criar uma pagina web com um elemento **<nav>** com os elementos: Inicio, Sobre nós, Nossos Serviços, Parceiros, Contacte-nos. Estes devem estarem na horizontal formando um menu da pagina. Cada item menu deve estar associado a uma imagem previamente oculta. Criar efeito/animação usando jQuery, de tal forma que quando recebe o foco em cada um dos itens, visualize a imagem que possa guiar a navegação dos internautas que não sabem ler, ao perder o foco a imagem também deve estar oculta visualizando somente menu textual;

Inclua elemento que ao seu critério que deixe a sua pagina agradável. O menu deve permite ter acesso a outras paginas da sua criatividade.