ЛАБОРАТОРНАЯ РАБОТА

Тема: «Реализация анимации на странице средствами *JQuery*»

**Теоретический материал**

Ключевым методом, на которой базируются все остальные, является метод *animate()*, с помощью которого можно задавать плавное изменение различных *CSS*-свойств:

***.animate(properties, [duration], [easing], [callback])***

***properties*** — список *CSS*-свойств, участвующих в анимации и их конечных значений. Задаются объектом, в формате {ключ:значение}, например:{*opacity*: 50, *width*: 100, *height*: 200}.

***duration*** — продолжительность выполнения анимации. Может быть задана в миллисекундах или строковым значением *'fast'* или *'slow'* (200 и 600 миллисекунд).

***easing*** — изменение скорости анимации (будет ли она замедляется к концу выполнения или наоборот ускорится). Задается строковым значением: "*linear*" и "*swing*" (для равномерной анимации и анимации с ускорением). Другие варианты можно найти в плагинах.

***callback*** — функция, которая будет вызвана после завершения анимации.

**Пример**. Пусть у нас есть элемент div с каким-нибудь текстом. Мы хотим плавно скрыть этот элемент, заменить текст, и плавно сделать элемент видимым:

*$("#mydiv")*

*.animate({height: "hide"}, 300)*

*.text("Новый текст")*

*.animate({height: "show"}, 300);*

значения "*hide*", "*show*" означают исчезновение и появление элемента, за счет параметра, к которому они применены.

Существуют отдельные методы, позволяющие скрывать и показывать элементы, а так же, изменять их прозрачность. Все эти методы можно найти в разделе эффекты. Необходимо также заметить, что при одновременном вызове нескольких эффектов, применительно к одному элементу, их выполнение будет происходить не одновременно, а поочередно. Например, при выполнении следующих команд:

*$("#my-div").animate({height: "hide"}, 1000);*

*$("#my-div").animate({height: "show"}, 1000);*

элемент с идентификатором *my-div*, в начале будет плавно исчезать с экрана, а затем начнет плавно появляться вновь. Однако, анимации, заданные на разных элементах, будут выполняться одновременно:

*$("#my-div-1").animate({height: "hide"}, 1000);*

*$("#my-div-2").animate({height: "show"}, 1000);*

Чтобы скрыть или показать элемент не сразу, а с анимацией, то можно воспользоваться другими функциями. Чтобы плавно переместить элемент вверх до полного исчезновения, можно воспользоваться методом *slideUp*. Например, мы хотим скрыть элемент <p> с атрибутом *updown* с анимацией вверх:

$("updown").slideUp("slow");

Для обратной задачи - переместить элемент вниз с анимацией, нужно воспользоваться схожей методом *slideDown*:

*$("updown").slideDown("slow");*

Метод *slideUp* прячет указанные элементы при помощи анимации высоты до полного уменьшения, делая элементы невидимыми. Когда высота элемента достигнет значения 0, то свойство стиля display устанавливается в значение none. Тем самым элемент как бы перестает существовать в разметке страницы, и на его место наезжают нижние элементы

Синтаксис:

***.slideUp([duration], [callback])***

Параметры:

***Duration -*** cтрока или число, определяющие продолжительность анимации. В качестве строк можно использовать слова fast (200 мсек), slow (600 мсек). По умолчанию используется интервал 400 миллисекунд.

***Callback*** - функция для вызова после отработанной анимации

**Пример**.

<!DOCTYPE html>

<html>

<head>

<script src="jquery.js"></script>

</head>

<body>

<div>

<button>Кнопка 1</button>

</div>

<div>

<button>Кнопка 2</button>

</div>

<div>

<button>Кнопка 3</button>

</div>

<div id="msg"></div>

<script>

$("button").click(function () {

$(this).parent().slideUp("slow", function () {

$("#msg").text($("button", this).text() + " скрыта.");

});

});

</script>

</body>

</html>

**Появление и исчезновение элементов**

.show() .hide() - с помощью этих функций можно плавно показывать и скрывать выбранные элементы на странице, за счет изменения размера и прозрачности.

Примеры использования:

*$("#leftFit").hide( )* мгновенно скроет элемент с идентификатором *leftFit*.

*$("#leftFit").show( )* мгновенно покажет элемент с идентификатором *leftFit*.

*$("#leftFit").hide(300)* в течении 1/3 секунды скроет элемент с идентификатором *leftFit*.

*$("#leftFit").show("slow")* в течении 600 миллисекунд вернет видимость элементу с идентификатором *leftFit*.

**Пример**: При нажатии на надпись, скроем картинку, а после завершения этой анимации выведем текст:

<!DOCTYPE html>

<html>

<head>

<style>

#clickme{width:300px; border:1px solid gray; padding:3px; cursor:pointer}

</style>

<script src="jquery.js"></script>

</head>

<body>

<div id="clickme">

Нажми на надпись

</div>

<img id="robot" src="robot.jpg" alt=""/>

<div id="result"></div>

<script>

$('#clickme').click(function(){

$('#robot').hide('slow', function() {

$('#result').html("....примерный текст <br>");

;

});

});

</script>

</body>

</html>

**Поочередное разворачивание и сворачивание элементов**

Синтаксис:

***.slideToggle([duration],[callback])***

***duration*** — продолжительность выполнения анимации (появления или скрытия). Может быть задана в миллисекундах или строковым значением *'fast'* или *'slow'* (200 и 600 миллисекунд). По умолчанию, анимация будет происходить за 400 миллисекунд.

***callback*** — функция, заданная в качестве обработчика завершения анимации (появления или скрытия). Ей не передается никаких параметров, однако, внутри функции, переменная *this* будет содержать *DOM*-объект анимируемого элемента. Если таких элементов несколько, то обработчик будет вызван отдельно, для каждого элемента.

Примеры использования:

*$("#leftFit").slideToggle()* свернет/развернет элемент с идентификатором *leftFit* за 400 мс.

*$("#leftFit").slideToggle(300)* в течении 1/3 секунды свернет/развернет элемент с идентификатором *leftFit*.

*$("#leftFit").slideToggle("slow")* в течении 600 мс свернет/развернет элемент с идентификатором *leftFit*.

**Пример**: При нажатии на кнопку элемент с картинкой и текстом будет поочередно сворачиваться и разворачиваться:

<!DOCTYPE html>

<html>

<head>

<script src="http://code.jquery.com/jquery-1.4.4.js"></script>

</head>

<body>

<button>Пуск</button><br><br>

<div>

<img src="/tags/images/yo.jpg" style="height:150px; float:right; margin:5px 5px">

<b>Правило.</b> Ё должна использоваться: в случаях возможных разночтений; в словарях;

в книгах для изучающих русский язык (т. е. детей и иностранцев); для правильного

прочтения редких топонимов, названий или фамилий. Во всех остальных случаях

наличие буквы ё только затрудняет чтение. Она плохо выглядит, зато хорошо звучит.

</div>

<script>

$("button").click(function () {

$("div").slideToggle("slow");

});

</script>

</body>

</html>

**Появление и исчезновение элементов за счет прозрачности**

*.fadeIn([duration],[callback])*

*.fadeOut([duration],[callback])*

Примеры использования:

*$("#leftFit").fadeOut( )* элемент с идентификатором *leftFit* "растворится" за 400 мс.

*$("#leftFit").fadeIn( )* элемент с идентификатором *leftFit* "прояснится" за 400 мс.

*$("#leftFit").fadeOut(300)* в течении 1/3 секунды элемент с идентификатором *leftFit* исчезнет.

*$("#leftFit").fadeIn("slow")* в течении 600 мс появится элемент с идентификатором *leftFit*.

**Пример**: При нажатии на текст "растворим" картинку, а после завершения этой анимации выведем текст:

<!DOCTYPE html>

<html>

<head>

<style>

#clickme{width:160px; border:1px solid gray; padding:3px; background-color:#eee; cursor:pointer}

</style>

<script src="jquery.js"></script>

</head>

<body>

<div id="clickme">

Получите уникальную методику обогащения, всего за 100 долларов

</div>

<img id="robot" src="100usd.jpg" alt=""/>

<div id="result"></div>

<script>

$('#clickme').click(function(){

$('#robot').fadeOut('slow', function() {

$('#result').html("Вот результат...");

});

});

</script>

</body>

</html>

**Установка заданной прозрачности у элемента**

*.fadeTo(duration, opacity, [callback])*

***duration*** — продолжительность изменения прозрачности. Может быть задана в миллисекундах или строковым значением *'fast'* или *'slow'* (200 и 600 миллисекунд).

***opacity*** — значение требуемой прозрачности. Должно быть задано числом от 0 до 1 (0 - полная прозрачность, 1 - отсутствие прозрачности).

***callback*** — функция, заданная в качестве обработчика завершения изменения прозрачности. Ей не передается никаких параметров, однако, внутри функции, переменная *this* будет содержать *DOM*-объект элемента, у которого изменяют прозрачность. Если таких элементов несколько, то обработчик будет вызван отдельно, для каждого элемента.

Примеры использования:

*$("#leftFit").fadeTo(0, 0.5)* элемент с идентификатором *leftFit* мгновенно станет прозрачным наполовину.

*$("#leftFit").fadeTo(500, 0.1)* элемент с идентификатором *leftFit* за полсекунды станет практически невидимым.

**Поочередное появление и скрытие элементов**

*.fadeToggle([duration],[easing],[callback])*

***duration*** — продолжительность выполнения анимации (появления или скрытия). Может быть задана в миллисекундах или строковым значением 'fast' или 'slow' (200 и 600 миллисекунд). По умолчанию, анимация будет происходить за 400 миллисекунд.

***easing*** — изменение скорости анимации (будет ли она замедляется к концу выполнения или наоборот ускорится). (см. описание)

***callback*** — функция, заданная в качестве обработчика завершения анимации (появления или скрытия). Ей не передается никаких параметров, однако, внутри функции, переменная this будет содержать DOM-объект анимируемого элемента. Если таких элементов несколько, то обработчик будет вызван отдельно, для каждого элемента.

Примеры использования:

*$("#leftFit").fadeToggle()* скроет/покажет элемент с идентификатором *leftFit* за 400 мс.

*$("#leftFit").fadeToggle(300)* в течении 1/3 секунды скроет/покажет элемент с идентификатором leftFit.

*$("#leftFit").fadeToggle("slow")* в течении 600 мс скроет/покажет элемент с идентификатором *leftFit*.

**Пример**: При нажатии на кнопку элемент с картинкой и текстом будет поочередно появляться и исчезать:

<!DOCTYPE html>

<html>

<head>

<script src="jquery.js"></script>

</head>

<body>

<button>Пуск</button><br><br>

<div>

<img src="/tags/images/yo.jpg" style="height:150px; float:right; margin:5px 5px">

<b>Правило.</b> Ё должна использоваться: в случаях возможных разночтений; в словарях;

в книгах для изучающих русский язык (т. е. детей и иностранцев); для правильного прочтения редких топонимов, названий или фамилий. Во всех остальных случаях наличие буквы ё только затрудняет чтение.

</div>

<script>

$("button").click(function () {

$("div").fadeToggle("slow");

});

</script>

</body>

</html>

ВЫПОЛНИТЬ:

В html-страницу, созданную в первой лабораторной работе, добавить следующие эффекты анимации:

ВАРИАНТ 1

1. элемент div с текстом плавно скрыть, изменить текст, и плавно показать
2. скрыть элемент <p> с анимацией вниз
3. При нажатии на надпись, скрыть картинку, а после завершения этой анимации вывести текст.
4. При нажатии на кнопку элемент с картинкой и текстом разворачивается
5. При нажатии на текст "растворить" картинку, а после завершения этой анимации вывести текст «Завершение анимации».
6. При нажатии на пункт списка элемент с картинкой и текстом будет поочередно появляться и исчезать

ВАРИАНТ 2

1. элемент div с текстом плавно скрыть, изменить цвет, и плавно показать
2. скрыть элемент <p> с анимацией вверх
3. При нажатии на надпись, показать картинку, а после завершения этой анимации вывести текст.
4. При нажатии на кнопку элемент с картинкой и текстом сворачивается
5. При нажатии на пункт списка "растворить" картинку, а после завершения этой анимации вывести текст «Завершение анимации».
6. При нажатии на кнопку элемент с картинкой и текстом будет поочередно появляться и исчезать.