

Технологии веб-UI Tizen (Tizen v.2.3)

Весна 2015 г.

Ким Су Донг, *доктор философии.*

Профессор кафедры информатики
Лаборатория программирования
Университет Сунгсил

Office 02-820-0909

Mobile 010-7392-2220

sdkim777@gmail.com

<http://soft.ssu.ac.kr>

Веб-технологии для веб-UI Tizen

- **HTML**

- **Язык разметки, предлагающий строительные блоки для создания сайтов**

- **CSS**

- **Таблицы стилей, определяющие способ отображения элементов HTML**

- **JavaScript**

- **Язык программирования, определяющий поведение веб-страниц**

- **JQuery**

- **Библиотека JavaScript для упрощения программирования на JavaScript**

Часть 5-1. HTML и CSS

Что такое HTML?

- Означает «язык гипертекстовой разметки» — Hyper Text Markup Language
- Язык разметки для описания веб-документов
 - Описание HTML-документов
 - Каждый тэг HTML описывает разный контент документов.

```
<html>
  <head>
    <title>Page title</title>
  </head>
  <body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

Тэги HTML

- **Основное использование тэгов HTML**
 - Раскладка
 - Стиль текста
 - Элемент
 - Форма
 - Список

Тэги HTML — Структура

- **Определяют структуру документа HTML**
- **Основные тэги**
 - **<header>**
 - **Представляет собой контейнер для вводного контента**
 - **<section>**
 - **Определяет раздел документа**
 - **<div>**
 - **Определяет раздел или главу документа HTML**
 - **<footer>**
 - **Определяет нижний колонтитул документа или раздела**

Тэги HTML — Стиль текста

- **Определяют стили текста документов HTML**
- **Основные тэги**
 - **<p>**
 - **Определяют абзацы**
 - **<h1> - <h6>**
 - **Определяют заголовки HTML**
 - **<i>**
 - **Определяют часть текста от другого лица или другого настроения**

Тэги HTML — Элемент

- **Определяют виджеты в документе HTML**
- **Основные тэги**
 - **<a>**
 - **Определяет гиперссылку, используемую для перехода с одной страницы на другую**
 - **<button>**
 - **Определяет действующую кнопку**
 - **<canvas>**
 - **Для рисования через скрипты JavaScript**
 - ****
 - **Для определения изображения на HTML странице**

Тэги HTML — Форма

- **Определяют поле, где пользователь может вводить данные**
- **Key Tags**
 - **<form>**
 - **Содержит один или более элементов формы**
 - **<input>**
 - **Указывает поле ввода данных пользователем**
 - **<select>**
 - **Создает выпадающий список**
 - **<option>**
 - **Определяет опцию в выделенном списке**
 - **<textarea>**
 - **Для определения контроля за вводом многострочного текста**

Тэги HTML — Список

- **Отображают пункты в виде списка**
- **Основные тэги**
 - ****
 - **Определяют неупорядоченный (маркированный) список**
 - ****
 - **Определяют упорядоченный список**
 - ****
 - **Определяют пункт списка**
 - **Используются в упорядоченных списках (), неупорядоченных списках ()**

Что такое CSS?

- Означает Cascading Style Sheets — Каскадные Таблицы Стилей
- Определяет способ отображения элементов HTML
- Связанные ссылками таблицы стилей

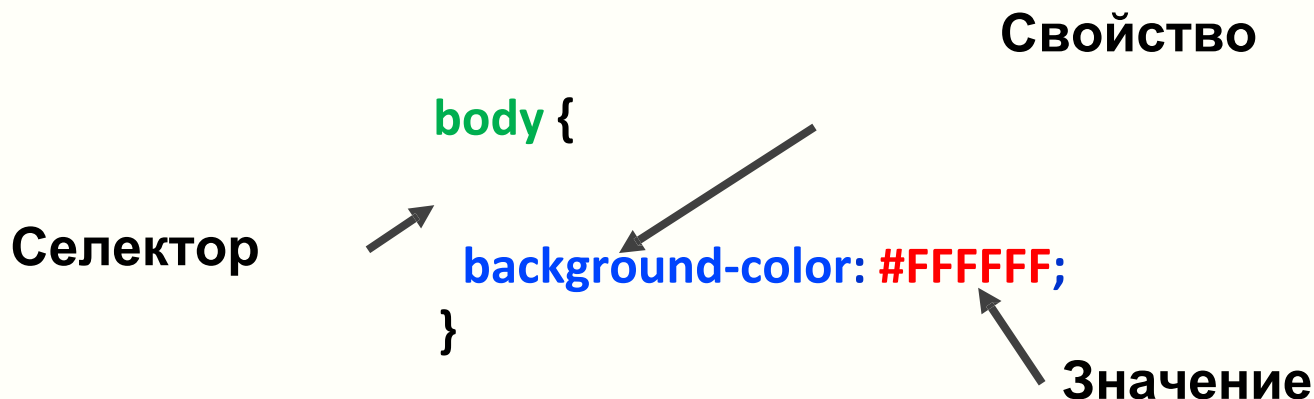
```
<link rel="stylesheet" type="text/css" href="mystyle.css">
```

- Встроенные таблицы стилей

```
...  
<style>  
body {  
  background-color: #d0e4fe;  
}  
</style>  
...
```

Синтаксис CSS

- **Выделитель-селектор**
 - Выделение и работа с элементами HTML
- **Свойство**
 - Указывают, какой аспект элемента изменен
- **Значение**
 - Значение свойства



Выделители

- **Выделитель элементов**
- **Выделитель ID**
- **Выделитель класса**
- **Выделитель атрибута**

Выделители (Выделитель Элемента)

- Для выделения элементов на основе имени элемента (HTML-тэг)
- Пример) выделить все элементы `<p>` на странице

```
p {  
  text-align: center; color: red;  
}
```

- Пример) Выделить все элементы `<p>`, `<h1>` и `<h2>` на странице

```
p, h1, h2 {  
  text-align: center;  
  color: red;  
}
```

Выделители (Выделитель ID)

- Использует id элемента HTML для выделения определенного элемента
- Использует символ #
- Пример) Выделить элемент по данному ID, 'id_1'

```
#id_1 {  
text-align: center; color: red;  
}
```

Выделители (Выделитель класса)

- Выделяет элементы с определенным именем класса
- Использует символ «.»
- Пример) выделить элементы по данному имени класса — center

.center {

```
text-align: center;
color: red;
}
```

- Пример) выделить элементы `<p>` с именем класса center
- p.center {**

```
text-align: center; color: red;
}
```


Выделители (Выделитель атрибута)

- Выделение элемента с определенным атрибутом
- Использует символ []
- Пример) выделить элементы <a> с атрибутом target

```
a[target] {  
    background-color: yellow;  
}
```

- Пример) выделить элементы <a> в которых атрибут target — это '_blank'

```
a[target="_blank"] {  
    background-color: yellow;  
}
```

Свойства

- **Ключевые категории свойств**
 - **Фон**
 - **Рамка**
 - **Форматирование текста**
 - **Поля и заполнение**

Свойства (Фон)

- **Определение эффектов фона элемента**
- **Основные свойства**
 - **Background-color**
 - Определяет цвет фона элемента
 - **Background-image**
 - Определяет изображение для использования в качестве фона элемента
 - **Background-repeat**
 - Определяет повторение изображения, служащего фоном
 - **Background-position**
 - Определяет положение фонового изображения

Свойства (Рамка)

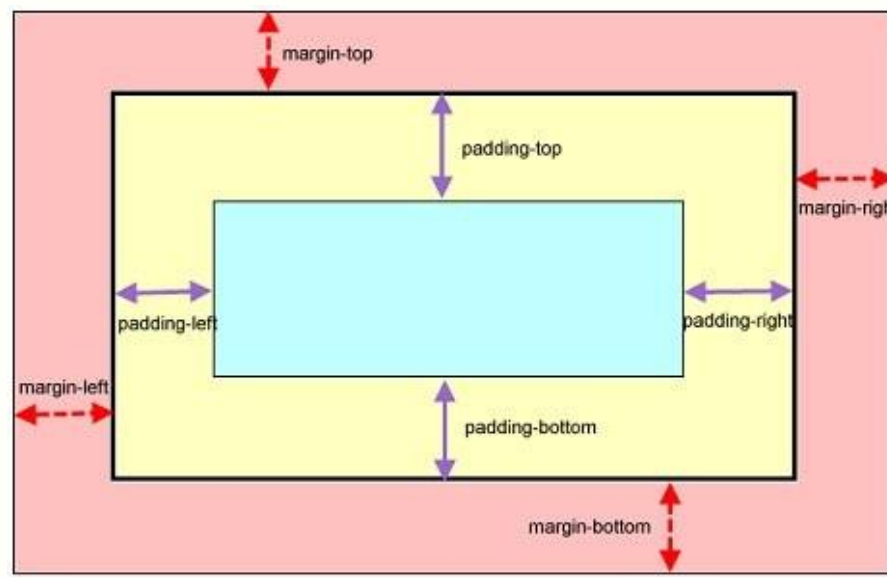
- **Определяет эффекты границ всех элементов**
- **Основные свойства**
 - **Border-style (Стиль рамки)**
 - Указывает, какую рамку показывать
 - Точечный пунктир, пунктир, сплошная, и т.д.
 - **Border-width (ширина рамки)**
 - Настраивает ширину рамки
 - **Border-color (цвет границы)**
 - Устанавливает цвет рамки

Свойства (Форматирование текста)

- **Устанавливает формат текста**
- **Основные свойства**
 - **Цвет**
 - **Задаёт цвет текста**
 - **Выравнивание текста**
 - **Задаёт горизонтальное выравнивание текста**
 - **Украшения текста**
 - **Устанавливает или удаляет украшения из текста**
 - **В основном используется для удаления подчёркиваний из ссылок, для эстетики дизайна**
 - **Трансформация текста**
 - **Устанавливает верхний и нижний регистр для букв в тексте**

Свойства (Поля и Заполнение)

- **Определяет место, отведенное элементам HTML**
- **Основные свойства**
 - **Поля**
 - Определяют пространство вокруг элемента
 - **Заполнение**
 - Определяет пространство между рамкой элемента и его контентом



Пример

- Встроенный код CSS

```
<!DOCTYPE html>
<html>
<head>
<style> body {
  background-color: #d0e4fe;
}

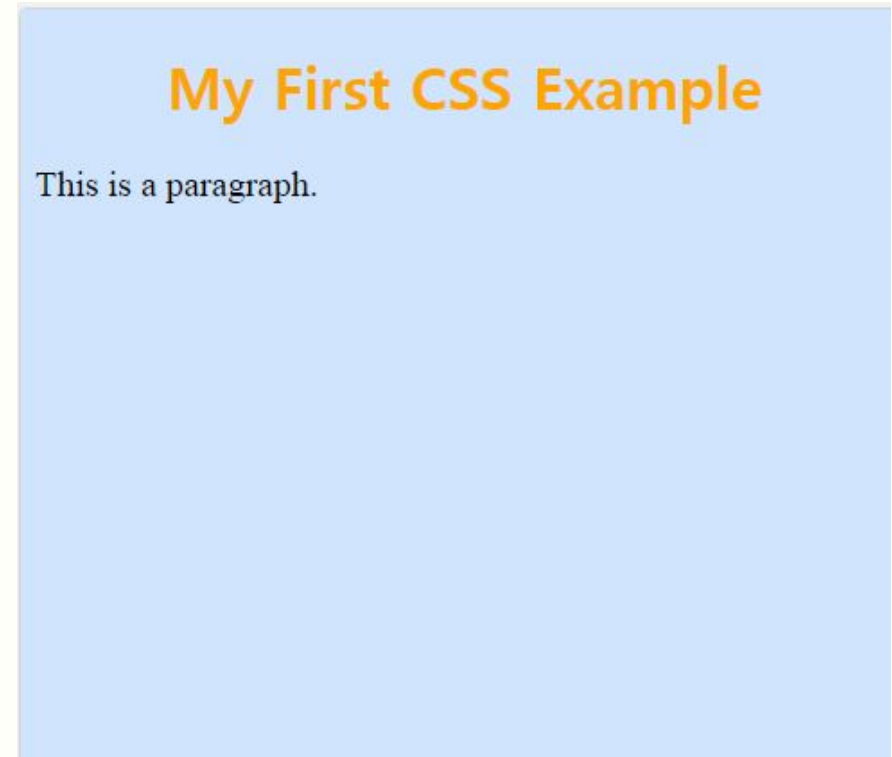
h1 {
  color: orange;
  text-align: center;
}

p {
  font-family: "Times New Roman"; font-size: 20px;
}
</style>
</head>
<body>

<h1>My First CSS Example</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

- Результат



Часть 5-2. JavaScript и jQuery

Что такое JavaScript?

- **Язык программирования HTML и Web**
- **Определяет поведение веб-страниц**
 - **Изменяет отображаемое содержание документа**
 - **Взаимодействует с пользователем**

Размещение JavaScript

- **Связанный JavaScript**

- **В документе HTML;**

```
<head>  
<script src="myjavascript.js"></script>  
</head>
```

- **Встроенный JavaScript**

- **В документе HTML;**

```
<head>  
<script>  
  //JavaScript Statements  
</script>  
</head>
```

Переменные (1)

- **5 типов данных, которые могут содержать значения;**
 - **String** — строковая переменная
 - **Number** — число
 - **Boolean** — булева переменная
 - **Object** — объект
 - **Function** — функция

Переменные (2)

- **Используется ключевое слово `var`**

```
var str = "Hello JS" // Assigns string
var i = 5; // Assigns number
var b = true; // Assigns boolean var obj = {name:'John',
age:'28'} // Assigns object var date = new Date()// Assigns
object
var arr = [1,2,3,4] // Assigns object
var func = function(){...} // Assigns function
```

- **Используется `typeof` для поиска типа данных**

```
typeof str // Returns string
typeof i // Returns number
typeof b // Returns boolean
typeof obj // Returns object
typeof date // Returns object
typeof arr // Returns object
typeof func // Returns
function
```

Операторы (1)

- **Арифметические операторы**
 - Для арифметических действий с числами
- **Операторы присвоения**
 - Присваивают значения переменным JavaScript

Оператор	Описание
+	Сложение
-	Вычитание
*	Умножение
/	Деление
%	Модуль
++	Инкремент
--	Декремент

Оператор	Пример	Эквивалент
=	$x = y$	$x = y$
+=	$x += y$	$x = x + y$
-=	$x -= y$	$x = x - y$
*=	$x *= y$	$x = x * y$
/=	$x /= y$	$x = x / y$
%=	$x \% = y$	$x = x \% y$

Операторы (2)

- **Операторы Сравнения**
- **Определяют равенство или отличие переменных или значений**
- **Пусть $x=5$**

Оператор	Описание	Сравнение	Результат
==	Равно	$x == 8$	false
		$x == 5$	true
===	Одинаковые значение и тип	$x === "5"$	false
		$x === 5$	true
!=	Не равно	$x != 8$	true
!==	Тип один, значения не равны	$x !== "5"$	true
		$x !== 5$	false
>	Больше	$x > 8$	false
<	Меньше	$x < 8$	true
>=	Больше или равно	$x >= 8$	false
<=	Меньше или равно	$x <= 8$	true

Примеры JavaScript

- **Изменение контента элементов HTML**
- **Изменение CSS элементов HTML**
- **Реакция на события веб-страницы**

Пример — изменение контента

- **Заменяет определенный текст**
- **Исходный код**

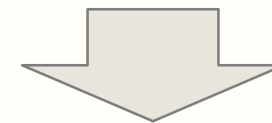
```
...  
<script>  
function myFunction() {  
  var paragraph =  
document.getElementById("demo");  
  paragraph.innerHTML = "Hello  
  JavaScript!";  
}..  
<body>  
<p id="demo">Click the button to replace  
text.</p>  
  
<button onclick="myFunction()">Try  
it</button>  
</body>  
...
```

Находит элементы с id 'demo'

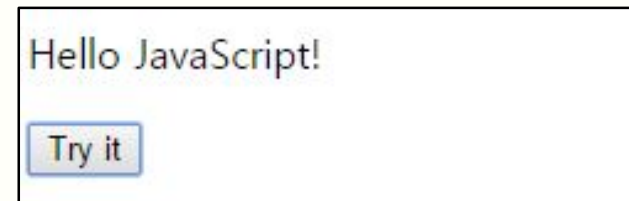
Настраивает текст элемента на "Hello, JavaScript!"

Вызывается при нажатии на кнопку

- **Результат**



Щелкните по кнопке!



Пример — изменение CSS

- Изменение стиля CSS определенного элемента
- Исходный код
- Результат

```
...  
<script>  
function changeTextColor(color)  
{ var paragraph = document.getElementById("p1");  
  paragraph.style.color= color;  
}  
...  
<body>  
<p id="p1">  
This is a text.  
</p>  
<button  
onclick="changeTextColor('red')">Red</button>  
<button onclick="changeTextColor('blue')"  
>Blue</button>  
</body>  
...
```

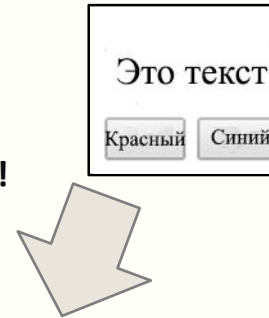
Находит элемент с id 'p1'

Меняет цвет текста

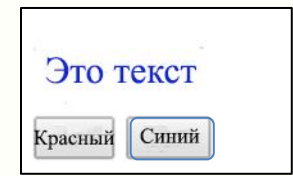
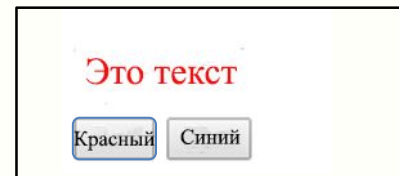
Вызывается для смены цвета текста на красный

Вызывается для смены цвета текста на синий

Щелкните по Красный!



Щелкните по Синий!



Пример — Обработка событий

- Реакция на событие нажатия на кнопку при нажатии на определенную кнопку
- Исходный код

- Результат

```
...  
<script>  
function onLoad() {  
  var b1= document.getElementById("b1");  
  b1.onclick= function(){  
    alert('The element is clicked!');  
  }  
}  
</script>  
...  
<body onload="onLoad()">  
<button id="b1">Hello JavaScript</button>  
</body>
```

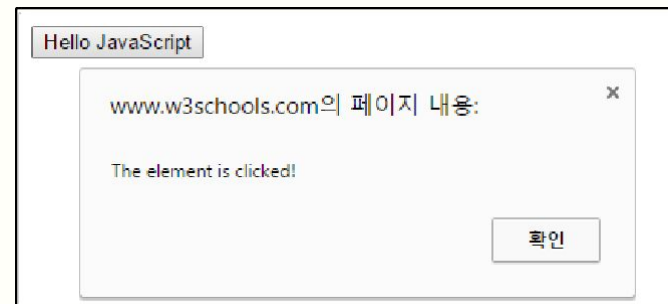
Находит элементы с id 'b1'

Показывает страницу предупреждения с сообщением

Вызывается при загрузке страницы



Щелкните по кнопке!



ООП в JavaScript

- Для определения классов, частных атрибутов/функций и публичных атрибутов/функций

```
function MyClass() {  
    var _privateAttr = "value"; function _privateMethod1() { }  
    var _privateMethod2 = function() {};  
  
    var self = {  
        publicAttr: "value",  
        publicMethod: function() {  
            console.log("publicAttr: "+self.publicAttr);  
        }  
    };  
    return self;  
}
```

Что такое jQuery?

- Библиотека JavaScript для упрощения программирования на JavaScript
- Чтобы далее упростить использование JavaScript на вашем сайте
 - Чтобы взяться за большое количество задач, требующих много строк кода JavaScript

Синтаксис jQuery

- **Основной синтаксис**

```
$(selector).action()
```

- **Знак '\$' для доступа к jQuery**
- **(выделитель) для запроса или поиска элементов HTML**
- **action() действие для выполнения над элементом**

- **Примеры**

```
$("#p").hide(); //Скрыть все элементы <p>.
```

```
$(".test").hide(); // Скрыть все элементы с class="test".
```

```
$("#test").hide(); // Скрыть элемент с id="test".
```

Событие ДОКУМЕНТ ГОТОВ

- Не допускает запуск кода jQuery или JavaScript до момента готовности документа HTML

```
$(document).ready(function(){  
    //Здесь помещается код jQuery или JavaScript ...  
});
```

```
$(function () {  
    //Здесь помещается код jQuery или JavaScript ...  
});
```

Выделители jQuery (1)

- Поиск элементов HTML
- Основан на существующих выделителях CSS
 - Выделитель элементов
 - Выделитель ID
 - Выделитель класса
 - Выделитель атрибута
 - Выделитель значения

Выделители jQuery (2)

- **Выделитель элементов**
 - **Выделение элементов HTML по имени элемента**

```
$("#p").hide(); //Скрывает все элементы <p> на странице
```

- **Выделитель ID**
 - **Использование атрибута id тэга HTML для поиска определенного элемента**
 - **Использует символ #**

```
$("#test").hide(); // Скрывает элемент с id = "test"
```

http://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_hide_show

Выделители jQuery (3)

- **Выделитель класса**

- Поиск элементов определенного класса
- Использует символ «.»

```
$(".test").hide(); // Скрывает все элементы класса "test"
```

- **Выделитель атрибута**

- Выделение элемента с определенным атрибутом

```
$("p[attr=hello]").hide(); //Скрывает все элементы <p> с атрибутом "hello"
```

Выделители jQuery (4)

- Прочие выделители

Синтаксис	Описание
<code>\$("*")</code>	Выделяет все элементы
<code>\$(this)</code>	Выделяет текущий элемент HTML
<code>\$("#p.intro")</code>	Выделяет все элементы <code><p></code> с классом="intro"
<code>\$("#p:first")</code>	Выделяет первый элемент <code><p></code>
<code>\$("#ul li:first")</code>	Выделяет первый элемент <code></code> первого <code></code>
<code>\$("#ul li:first-child")</code>	Выделяет первый элемент <code></code> всех <code></code>
<code>\$("[href]")</code>	Выделяет все элементы с атрибутом href
<code>\$("#a[target='_blank']")</code>	Выделяет все элементы <code><a></code> со значением целевого атрибута, равным "_blank"
<code>\$("#a[target!='_blank']")</code>	Выделяет все элементы <code><a></code> со значением целевого атрибута, НЕ равным "_blank"
<code>\$(":button")</code>	Выделяет все элементы <code><button></code> и элементы <code><input></code> типа "button"
<code>\$("#tr:even")</code>	Выделяет все четные элементы <code><tr></code>
<code>\$("#tr:odd")</code>	Выделяет все нечетные элементы <code><tr></code>

Действие jQuery Action

- **Выполняется на выделенных элементах**
 - **Получение/задание контента и атрибутов**
 - **Добавление/удаление элементов HTML**
 - **Обработка CSS**
 - **Обработка событий**

jQuery Action — Контент и Атрибуты (1)

- **text()**
 - **Задание/получение текста элементов**
- **html()**
 - **Задание/получение html-контента элементов**
- **val()**
 - **Задание/получение значений полей форм**

```
//Возвращает текст элемента с id='id'  
var text = $("#id").text();  
  
//Возвращает HTML-контент элемента с id='id'  
var html = $("#id").html();  
  
//Возвращает значение полей формы с id='id'  
var val = $("#id").val();
```

```
//Задает текст элемента с id='id' как "Hello jQuery!"  
$("#id").text("Hello jQuery!");  
  
//Задает HTML-контент элемента с id='id' как "Hello jQuery!"  
$("#id").html("<p>Hello jQuery!</p>");  
  
//Возвращает значение полей формы с id='id' как  
"value"  
$("#id").val("value");
```

jQuery Action — Контент и Атрибуты (2)

- attr()
- Задание/получение значения атрибута

```
//Возвращает значение атрибута href
var attr = $("#id").attr("href");

//Задает значение атрибута href
$("#id").attr("href", "http://www.soft.ssu.ac.kr");

//Задает несколько атрибутов
$("#id").attr(
  {"href": http://www.soft.ssu.ac.kr,
  "title": "Software Engineering Laboratory"}
);
```

jQuery Action — Элементы HTML (1)

- **append()**
 - Вставка контента в конце выделенного элемента
- **prepend()**
 - Вставка контента в начале выделенного элемента
- **after()**
 - Вставка контента после выделенного элемента
- **before()**
 - Вставка контента перед выделенным элементом

```
$("#p").append("Добавляемый в конце текст .");  
$("#p").prepend("Добавляемый в начале текст .");  
$("#p").after("Текст после.");  
$("#p").before(".");
```

jQuery Action — Элементы HTML (2)

- **remove()**

- **Удаление выделенных элементов и их дочерних элементов**

```
$("#div1").remove();
```

- **empty()**

- **Удаление дочерних элементов выделенных элементов**

```
$("#div1").empty();
```

jQuery Action — обработка CSS (1)

- **addClass()**
 - Добавление одного или более классов к выделенным элементам
- **removeClass()**
 - Удаление одного или более классов из выделенных элементов
- **toggle(class)**
 - Переключение между добавлением/удалением классов выделенных элементов

```
$(“div”).addClass(“important”) // Добавляет значение ‘important’ к атрибуту класса
$(“div”).addClass(“important blue”) // Добавляет значения ‘important’ и ‘blue’ к атрибуту
классу

$(“div”).removeClass(“important”) // Удаляет значение ‘important’ из атрибута класса
$(“div”).removeClass(“important blue”) // Удаляет значения ‘important’ и ‘blue’ из атрибута класса
$(“div”).toggleClass(“important”) // Добавляет или удаляет значение ‘important’ из атрибута
классу
$(“div”).toggleClass(“important blue”) // Добавляет или удаляет значения ‘important’ и ‘blue’ из
атрибута класса
```


jQuery Action — обработка CSS (2)

- `css()`
- Задание/получение одного или более свойства стилей для выбранного элемента (элементов)

```
// Возвращает значение цвета фона
var bgColor = $("p").css("background-color")

// Задаёт значение цвета фона как желтый
$("p").css("background-color", "yellow")

// Задаёт несколько свойств стиля
$("p").css(
  {"background-color": "yellow"},
  {"font-size": "200%"}
)
```

jQuery Action — Обработка событий

- Синтаксис метода событий

```
$(selector).eventMethod(function(){  
  
    // Здесь код обработки события!  
});
```

- Основные методы событий

События мыши	События клавиатуры	События формы
click()	keypress()	submit()
dblclick()	keydown()	change()
mouseenter()	keyup()	focus()
mouseleave()		blur()

http://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_event_blur_alert

Пример (1)

- **Обработка событий нажатия на кнопку**
- **Изменение стиля CSS и текста элемента HTML**
- **Исходный код**

- **JavaScript**

```
<script>
$(document).ready(function(){
  $("#btn1").click(function(){
    $("#p1").text("Hello world!");
  });
  $("#btn2").click(function(){
    $("#p1").css("color", "blue");
  });
});
</script>
```

Вызывается при нажатии на кнопку '#btn1'

Задает текст "Hello world"

Вызывается при нажатии на кнопку '#btn2'

Меняет цвет текста на синий

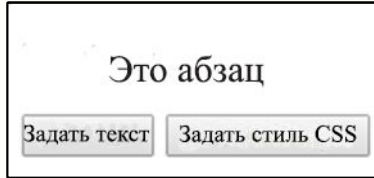
- **HTML**

```
<body>
<p id="p1">This is a paragraph.</p>
<button id="btn1"> Set Text </button>
<button id="btn2"> Set CSS Style </button>
</body>
```

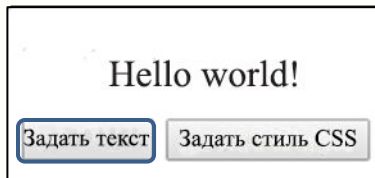
Пример (2)

- **Вывод**

- **Исходный экран**



- **Случай 1: после нажатия на кнопку Set Text**



- **Случай 2: после нажатия на кнопку Set CSS Style**

