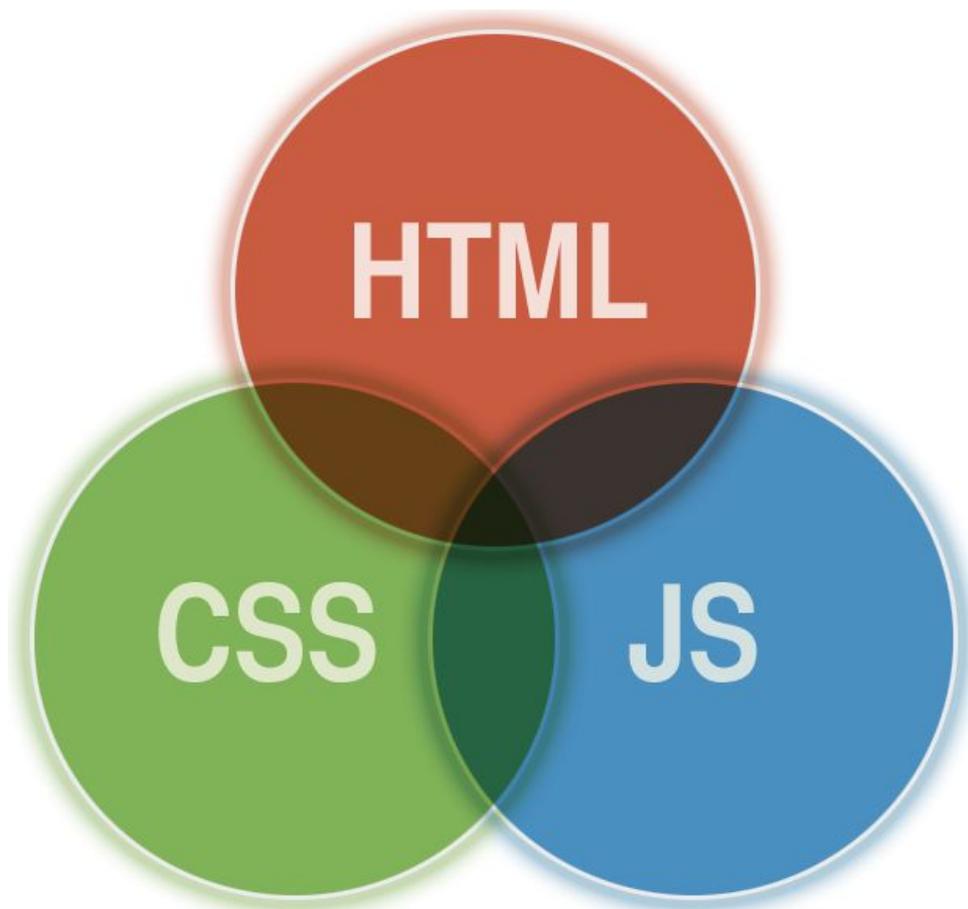


# Общие сведения. Назначение и применение **JavaScript**.

Ст. преподаватель Еремеев А.А.  
YeremeevAA@mpei.ru



# Что необходимо знать для создания сайта?



# Кратко о HTML

- ▣ **HTML** (*HyperText Markup Language*)— «язык гипертекстовой разметки».
- ▣ HTML не является языком программирования, он предназначен для разметки текстовых документов.
- ▣ То, как будут выглядеть картинки, текст и другие элементы на вашей странице, и как они будут располагаться относительно друг друга, определяют *HTML*-контейнеры - тэги.

# Структура HTML-документа

```
<html>
```

```
<head>
```

```
<title>Мой первый шаг </title>
```

```
</head>
```

```
<body>
```

Здравствуйте, это моя первая страница.

```
<br>
```

Добро пожаловать! :)

```
</body>
```

```
</html>
```

# Тэги

- ▣ Все тэги, расположенные между `<head>` `</head>`, это что-то вроде служебной информации.
- ▣ Все тэги, расположенные между `<body>` `</body>` - непосредственное содержание документа.

# JavaScript

- ▣ Язык управления сценариями просмотра гипертекстовых страниц *Web* на стороне клиента.
- ▣ Основная идея JavaScript состоит в возможности изменения значений атрибутов *HTML*-контейнеров и свойств среды отображения в процессе просмотра *HTML*-страницы пользователем. При этом перезагрузки страницы не происходит.

# Размещение кода JavaScript на HTML-странице

- ▣ JavaScript-код исполняет браузер.

Три способа функционального применения JavaScript:

- ▣ гипертекстовая ссылка (*схема URL*);
- ▣ обработчик события (в атрибутах, отвечающих событиям);
- ▣ вставка (контейнер `<SCRIPT>` ).

# Оператор alert()

- Оператор *alert*(строка) выводит эту строку на экран в *окне предупреждения*



# Оператор `document.write()`

- Оператор `document.write(строка)` записывает указанную строку в текущий *HTML*-документ.

Простой HTML-документ	Использование <code>document.write()</code>
<pre>&lt;HTML&gt; &lt;BODY&gt; &lt;H1&gt;Заголовок&lt;/H1&gt; &lt;/BODY&gt; &lt;/HTML&gt;</pre>	<pre>&lt;HTML&gt; &lt;BODY&gt; &lt;SCRIPT&gt; document.write('&lt;H1&gt;Заголовок&lt;/H1&gt;'); &lt;/SCRIPT&gt; &lt;/BODY&gt; &lt;/HTML&gt;</pre>

# Способ 1: URL-схема "JavaScript:"

- ▣ *Схема URL (Uniform Resource Locator)* - один из основных элементов Web-технологии.
- ▣ В Web-технологии стандартной программой, вызываемой при гипертекстовом переходе, является программа загрузки страницы (т.е. при *клике* по ссылке загружается страница с указанным URL). JavaScript позволяет поменять стандартную программу на программу пользователя.

# Способ 1: URL-схема "JavaScript:"

- Для того, чтобы отличить стандартный переход по протоколу HTTP от перехода, программируемого на JavaScript, разработчики языка ввели новую схему URL - JavaScript:

*<A HREF="JavaScript:код\_программы">...</A>*

*<FORM ACTION="JavaScript:код\_программы" ...> ...  
</FORM>*

# Пример 1

- `<A HREF="JavaScript:alert('Внимание!!!');">`  
Клики здесь`</A>`



# Пример 2

- `<FORM METHOD=post NAME="form"`
- `ACTION="JavaScript:form.e.value='Нажали кнопку: Заполнить';void(0);">`
- `<INPUT TYPE=text NAME=e SIZE=30`  
`VALUE=""><BR>`
- `<INPUT TYPE=submit VALUE="Заполнить">`
- `<INPUT TYPE=reset VALUE="Очистить">`
- `</FORM>`

# Способ 2: обработчики событий

- ▣ *Обработчики событий*, указываются в атрибутах теги, с которыми эти события связаны. Например, при нажатии на кнопку происходит событие *Click* и соответственно вызывается обработчик этого события *onClick*:
- ▣ `<FORM>`
- ▣ `<INPUT TYPE=button VALUE="Кнопка"`
- ▣ `onClick="alert('Вы нажали кнопку');">`  
`</FORM>`

# Способ 2: обработчики событий

- В момент завершения полной загрузки документа (он связан с контейнером `<BODY>` ) происходит событие *Load* и, соответственно, будет вызван обработчик этого события `onLoad`:
- `<BODY onLoad="alert('Приветствуем!');">`
- ...
- `</BODY>`

# Способ 3: вставка (тег<SCRIPT>)

□ Два способа:

Внутри контейнера:

□ <SCRIPT>

□ a = 5;

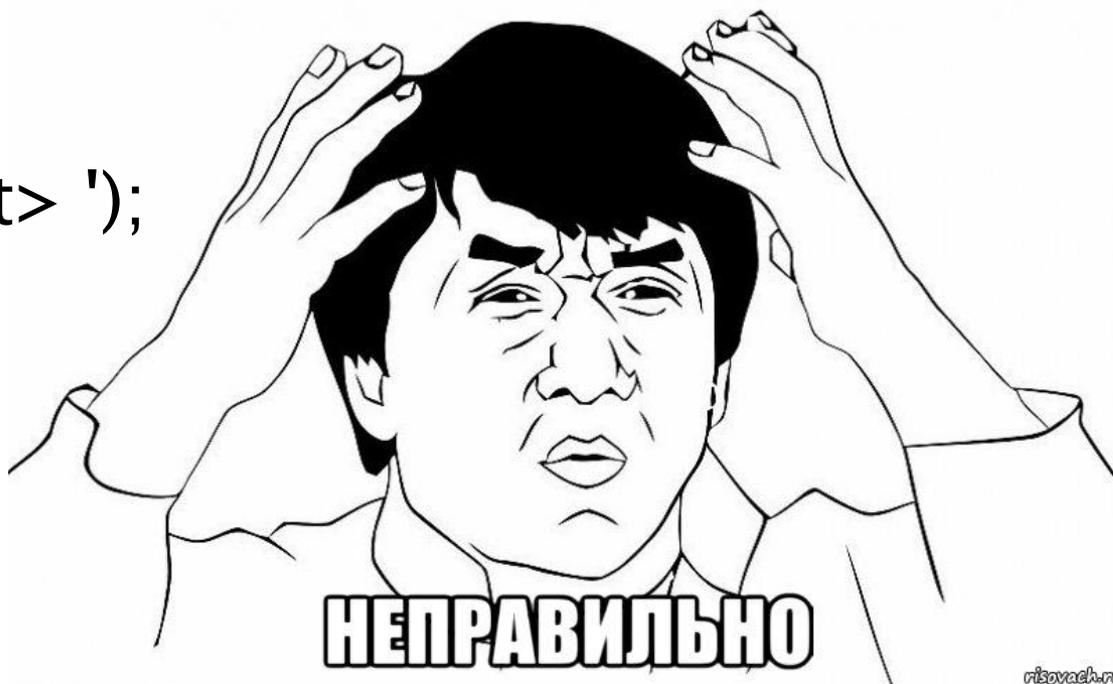
□ </SCRIPT>

Отдельный файл:

□ <SCRIPT SRC="myscript.js"></SCRIPT>

# Особенность

- ❑ Не использовать последовательность символов `</SCRIPT>` в коде:
- ❑ `<SCRIPT>`
- ❑ `alert(' </script> ');`
- ❑ `</SCRIPT>`



# Размещение кода внутри HTML-документа

- `<HTML><HEAD>`
- `<SCRIPT>`
- `function time_scroll()`
- `{ var d = new Date();`
- `window.status = d.getHours()`
- `+ ':' + d.getMinutes()`
- `+ ':' + d.getSeconds();`
- `setTimeout('time_scroll()',1000); }`
- `</SCRIPT>`
- `</HEAD>`
- `<BODY onLoad="time_scroll()">`
- `<H1>Часы в строке статуса</H1>`
- `</BODY> </HTML>`

# Условная генерация HTML-разметки на стороне браузера

- `<BODY>`
- ...
- `<SCRIPT>`
- `d = new Date();`
- `document.write('Момент загрузки страницы: '`
- `+ d.getHours() + ':'`
- `+ d.getMinutes() + ':'`
- `+ d.getSeconds());`
- `</SCRIPT>`
- ...
- `</BODY>`

# Комментарии в HTML и JavaScript

- `<SCRIPT>`

- 

- `a=5; //` однострочный комментарий

- 

- `/*` Многострочный

- комментарий `*/`

- 

- `</SCRIPT>`

# Скрытие JS от интерпретации старых браузеров

- ▣ `<SCRIPT>`
- ▣ `<!-- Скрываем JavaScript-код от старых браузеров`
- ▣
- ▣ `a = 5;`
- ▣
- ▣ `// -->`
- ▣ `</SCRIPT>`

# Указание языка сценария

Устаревший атрибут Language:

- ▣ `<SCRIPT LANGUAGE="JavaScript">`
- ▣ ...
- ▣ `</SCRIPT>`

Атрибут Type:

- ▣ `<SCRIPT TYPE="text/javascript">`
- ▣ ...
- ▣ `</SCRIPT>`

# Регистр символов

Язык HTML является **регистро-независимым**.

- `<SCRIPT>` и `<script>` ;
- `Type`, `LANGUage` и `src` ;
- `"JavaScript"` и `"TEXT/JavaScript"`.



Язык JavaScript - **регистро-зависимый**.

- `myText` и `MyText` ;
- `document.write()` и `Document.write()`.

