

Тема: JavaScript

1. Что такое JavaScript
2. Синтаксис языка
3. Объектная DOM-модель JavaScript
4. Включение JavaScript в HTML
5. Примеры использования JavaScript

1. Что такое JavaScript

- **JavaScript** – объектно-ориентированный *скриптовый*, *интерпретируемый* язык программирования.
- JavaScript обычно используется как *встраиваемый язык* для программного доступа к объектам приложений.
- Наиболее широкое применение находит в браузерах как язык сценариев для придания *интерактивности* веб-страницам.
- Является языком выполняемым на стороне клиента – браузера.
- JavaScript язык со слабой типизацией и с автоматическим управлением памятью.

1. История JavaScript

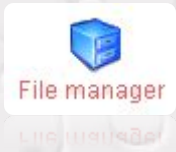
- **1992** год компания **Nombas** начала разработку нового языка Cmm. Этот язык должен был: быть похожим на Си, автоматически работать с памятью, быть мощным и мог заменить макросы.
- **1995** год язык переименован в **Cenvi** и теперь встраивается в веб страницы (но работает только подо Netscape Navigator).
- Конец **1995** года язык переименован с **LiveScript**, а затем в **JavaScript**.
- **1996** год компания Microsoft выпускает язык **Jscript** и включает его поддержку в браузер Internet Explorer 3.0.
- с **2005** года начинает развиваться AJAX который обеспечивает языку JavaScript популярность.

2. Синтаксис языка

- Синтаксис похож на Си и Java
- **JavaScript регистрозависимый язык**
- В *имени переменных* можно использовать латинские буквы, цифры, знак подчеркивания и знак доллара
- *Комментарии* // однострочные и /*многострочные*/
- Поддерживает объектную модель браузера: окно – строка заголовка...
- Поддерживает объектную модель документа: страница – таблица первая – ячейка двенадцатая...
- Управляющие конструкции очень схожи с языком Си (условия, циклы, функции и т.д.)
- Нет модульности программирования, т.е. не поддерживается управление областью видимости переменной
- Используют атрибут **id** для уникальной идентификации элементов html-документа.

2. Объектная модель браузера

- Основное предназначение – управление окнами браузера и обеспечение их взаимодействия. Каждое из окон браузера представляется объектом window, центральным объектом BOM (Browser Object Model).
- Объектная модель браузера не стандартизирована, но спецификация находится в разработке.
- Кроме управления окнами есть:
 - управление фреймами,
 - поддержка задержки в исполнении кода и зацикливания с задержкой,
 - системные диалоги,
 - управление адресом открытой страницы,
 - управление информацией о браузере,
 - управление информацией о параметрах монитора,
 - ограниченное управление историей просмотра страниц,
 - поддержка работы с HTTP cookie.



3. Объектная модель документа

- Интерфейс программирования HTML и XML-документов.
- Любой html-документ с помощью DOM представлен в виде дерева, в котором каждый узел представляет собой текстовый, графический или другой объект. Узлы связаны между собой родительский-дочерний.
- Модель DOM позволяет производить различные манипуляции с узлами: добавление узлов, изменение узлов, удаление узлов.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ARTICLES SYSTEM
"D:\Project\clients\XML\Contents\Templarticelst.dtd">
<?xml-stylesheet type="text/xsl"
href="D:\xmltohtml.xsl"?>
<ARTICLES>
  <ARTICLE>
    <ARTICLEDATA>
      <TITLE>XML Demystified</TITLE>
      <AUTHOR>Jaidev</AUTHOR>
    </ARTICLEDATA>
  </ARTICLE>
  <ARTICLE>
    <ARTICLEDATA>
      <TITLE>XSLT Demystified</TITLE>
      <AUTHOR>X S Cel Tea </AUTHOR>
    </ARTICLEDATA>
  </ARTICLE>
  <ARTICLE <div>
    <ARTI    <p>Текст абзаца</p>
    <    <p>Текст абзаца</p>
    <    <p>Текст абзаца</p>
    </ARTI    <p>Текст абзаца</p>
  </ARTICL    <p>Текст абзаца</p>
</ARTICLES>    <p>Текст абзаца</p>
  </div>
```

Пример объектной модели DOM

The image shows a browser's developer tools interface. The left pane displays the DOM tree with the following structure:

```
div.thumb < div#bodyContent < div#content < div#column-content < div#global
  <a id="top"/>
  <div id="siteNotice">
    <h1 id="firstHeading" class="firstHeading">Document
    Object Model </h1>
    <div id="bodyContent">
      <span class="editsection">
        <h3 id="siteSub">Материал из Википедии – свободной
        энциклопедии </h3>
        <div id="contentSub"/>
      <div id="jump-to-nav">
      <div class="thumb tright">
        <div class="thumbinner" style="width: 122px;">
      </div>
    <p>
```

The right pane shows the DOM object model for the selected `div#bodyContent` node:

ownerDocument	Document Document_Object_Model
parentNode	div#content
prefix	null
previousSibling	<TextNode textContent="\n"
scrollLeft	0
scrollTop	0
scrollWidth	552
spellcheck	false
style	CSSStyleDeclaration length=0
tabIndex	-1
tagName	"DIV"
textContent	" [править] \n Материал из \nПример иерархии объектов в HTML"

4. Включение JavaScript в HTML

- Расположение внутри html-страницы:

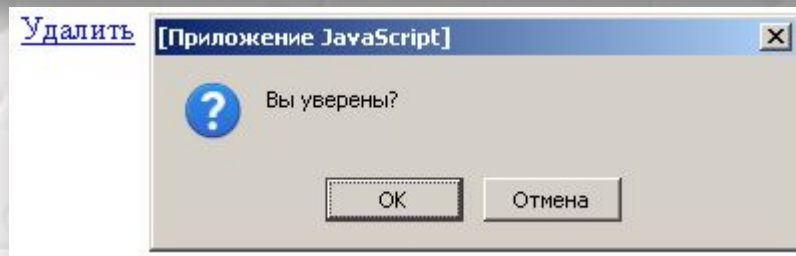
```
<script type="text/javascript">  
alert('Hello, World!');  
</script>
```



Действия
выполняются в
момент загрузки
кода в браузер

- Расположение внутри тега:

```
<a href="delete.php" onclick="return confirm('Вы уверены?');">Удалить</a>
```



Действия
выполняются в
момент щелчка на
ссылке

- Подключение JavaScript вынесенного в отдельный файл:

```
<script type="text/javascript" src="http://Путь_к_файлу_со_скриптом"></script>
```


5. События элементов DOM (DHTML)

- Событиями принято считать какие-либо действия со стороны пользователя посредством клавиатуры или мыши, а также такие операции с различными документами (например формы), как загрузка и их отправка.
- У каждого события в языке DHTML предусмотрена соответствующая команда для обработки, которая имеет вид названия события с префиксом ' on '.
- Например:...

События и команды обработчики событий

событие	команда	описание события
mouseover	onmouseover	наведение указателя мыши на объект.
mouseout	onmouseout	указатель мыши покидает область, отведенную под объект.
mousemove	onmousemove	динамическое движение указателя мыши по заданной области объекта.
click	onclick	"клик" мышкой, когда её указатель наведен на объект
mousedown	onmousedown	нажатие кнопки мыши, независимо от того, где находится указатель мыши в документе. <i>Примечание</i> : команду нужно применять в "связке" с обратной командой onmouseup .
mouseup	onmouseup	применяется в ответ на действие события mousedown , т.е. отпускание кнопки мыши. Обработка команды onmouseup произведётся только в "связке" с командой onmousedown .
keypress	onkeypress	"клик" клавиатурной клавиши.
keydown	onkeydown	нажатие на клавишу клавиатуры, при раскладке "клика" на два события. Соответственно команда onkeydown используется с противоположной командой onkeyup .
keyup	onkeyup	"логическое" завершение события keydown , т.е. отпускание клавиатурной клавиши. Команда onkeyup обрабатывается в "связке" с onkeydown .
keycode	onkeycode	любые события, связанные с нажатием клавиш на клавиатуре, будут работать в Юникодах.
focus	onfocus	на объект наведен фокус.
blur	onblur	с объекта убран фокус.
load	onload	документ загружен полностью и без ошибок.
error	onerror	документ загружен с ошибкой.
unload	onunload	отправка или выгрузка документа.
select	onselect	выбор текста или его фрагмента из формы для последующей манипуляции с ним.
submit	onsubmit	отправка формы.
reset	onreset	сброс содержимого формы, при использовании кнопки "RESET"
resize	onresize	данное событие наступает при изменении габаритов окна браузера.
move	onmove	перемещение окна браузера.
scroll	onscroll	прокрутка содержимого окна браузера.
dragdrop	ondragdrop	перетаскивание какого-либо элемента (например ссылки) в окно браузера.
dragstart	ondragstart	перетаскивание элемента куда-либо из окна браузера.
abort	onabort	документ загружен не полностью.

5. Примеры использования JavaScript

- Обработка события `click` для объекта кнопка. Есть форма:



Ваше имя

- Необходимо что бы при щелчке на кнопку выводилось сообщение содержащее текст из поля ввода.

С использованием PHP

```
<form action="file.php" method="get">  
  <input type="text" name="name" />  
  <input type="submit" value="Click me!" />  
</form>
```

Выполняется *на сервере*

Тег *form* **обязательный**

Input идентифицируется атрибутом *name*

Кнопка имеет *type=submit*

Выполняется *в браузере* (на клиенте)

Тег *form* **не обязательный**

Input идентифицируется атрибутом *id*

Кнопка имеет *type=button*

С использованием JavaScript

```
<input type="text" id="name" />  
<input type="button" value="Click me!"  
  onclick="a=document.getElementById('name').value;alert(a);" />
```

Кнопка содержит обработчик *onclick*

```
<input type="text" id="name" />  
<input type="button" value="Click me!"  
  onclick=" " />
```

- document.

-

5. Примеры использования JavaScript

- Обработка события onclick для объекта кнопка



A screenshot of a web form containing three empty text input fields followed by a button labeled "Click me!".

- После того как все данные «вроде как» введены необходимо проверить заполнения текстовых полей ввода на пустоту. Т.е. Если в поле ввода нет текста (атрибут value пустой), то окрасить ячейки в красный цвет, иначе окрасить ячейки в белый цвет.

```
<input type="text" id="name1" />
<input type="text" id="name2" />
<input type="text" id="name3" />
<input type="button" value="Click me!"
  onclick="doit();" />
```


Пример: бегущая строка

```
<html>
<head>
<script type="text/javascript">
  var scrollCounter = 0;
  var scrollText    = "IT-студия SoftTime приветствует вас!!!";
  var scrollDelay   = 70;
  var i = 0;
  while (i ++ < 140)
  {
    scrollText = " " + scrollText;
  }
  function Scroller()
  {
    window.status = scrollText.substring(scrollCounter++,
                                          scrollText.length);
    if (scrollCounter == scrollText.length)
    {
      scrollCounter = 0;
    }
    setTimeout("Scroller()", scrollDelay);
  }

  Scroller();
</script>
</head>
<body>
</body>
</html>
```

Объектная модель документа или браузера используется в коде?

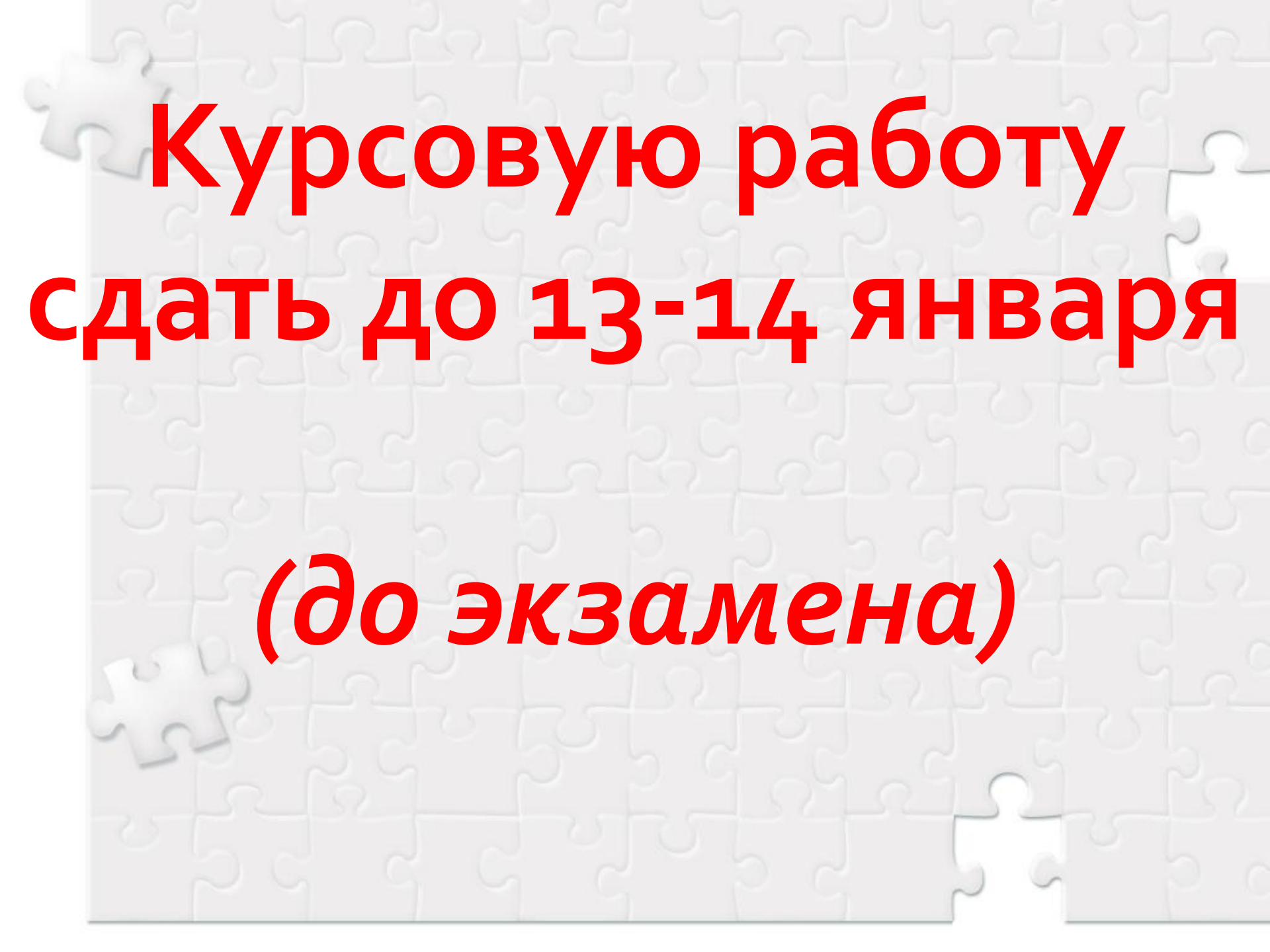
Пример: смена цвета фона

```
<html>
<head>
</head>
<body>
<FORM>
  <SELECT onChange=
    "document.bgColor=this.options[this.selectedIndex].value">
    <OPTION VALUE="red"> красный</OPTION>
    <OPTION VALUE="2E8B57"> морской волны</OPTION>
    <OPTION VALUE="87CEEB"> голубой</OPTION>
    <OPTION VALUE="brown"> коричневый</OPTION>
    <OPTION VALUE="yellow"> желтый</OPTION>
    <OPTION VALUE="blue"> синий</OPTION>
    <OPTION VALUE="FFFFFF" SELECTED> белый</OPTION>
  </SELECT>
</FORM>
</body>
</html>
```

Объектная модель документа или браузера используется в коде?

Требования к курсовой работе

Пояснительная записка	40
Верстка (HTML)	10
Использование CSS	10
Программирование PHP	13
Размещение на хостинге	7
База данных	5
Мета-теги	3
Авторизация в админку	3
Поисковики	3
Реклама	2
Счетчик	2
JavaScript	2



**Курсовую работу
сдать до 13-14 января**

(до экзамена)

Требования к СРС №2

Вид работы	Балл
Установка WordPress	2
Настройка индивидуальной темы	1
Наполнение 5-7 статей тематических	1
Указание каждой статье меток	1
Назначение статье рубрики	1
Создание тематического списка ссылок	1
Установка плагина 1	1
Установка плагина 2	1
Установка плагина 3	1
Добавление комментариев в 2-3 статьи	1
Настройка виджетов	1
Добавление RSS ленты работающей	1
Настройка ЧПУ	1
Добавление 3-х пользователей	1
Всего	15