

## **Лабораторная работа №5**

### **Обработка событий в HTML.**

**1. Цель:** Изучить и научиться использовать обработчики событий в HTML.

#### **2. Методические указания**

- 1) При изучении конструкций языка HTML можно использовать любой текстовый редактор. Для получения HTML файла, сохраняйте свои изменения как текстовые, для файла используйте расширение \*.htm или \*.html.
- 2) Более опытные пользователи могут воспользоваться любым специализированным редактором HTML страниц (MacromediaDreamweaver, HomeSite, FrontPage, AceHTML, WebDevelopmentStudio, Notepad++)
- 3) Создаваемые файлы необходимо тестировать в основных браузерах InternetExplorer, MozillaFirefox, Opera, GoogleChrome, Safari.
- 4) Отлаженные файлы необходимо сохранять в отдельном каталоге.
- 5) По окончании работы сохраните все созданные файлы на своих носителях.

#### **3. Теоретические сведения**

Популярность HTML во многом обусловлена именно тем, что написанный на нем сценарий может реагировать на действия пользователя и другие внешние события. Каждое из событий связано с тем или иным объектом: формой, гипертекстовой ссылкой или даже с окном, содержащим текущий документ.

В качестве примеров внешних событий, на которые могут реагировать объекты JavaScript, можно привести следующие.

- окончание загрузки документа в окно (или окончание загрузки документов во все фреймы окна). Это событие связано с объектом window;
- щелчок мышью на объекте. Это событие может быть связано с интерактивным элементом формы или с гипертекстовой ссылкой;
- получение объектом фокуса ввода. Это событие может быть связано с объектами типа Text, Password и с другими интерактивными элементами;
- передача на сервер данных, введенных пользователем с помощью интерактивных элементов. Связывается с формой.

Обработка события производится с помощью специально предназначенного для этого фрагмента кода, называемого обработчиком события. Для каждого события предоставляет свой обработчик. Однако при построении сценария можно создавать собственный обработчик события и использовать его вместо обработчика, заданного по умолчанию.

Имя обработчика определяет, какое событие он должен обрабатывать. Так, для того чтобы сценарий нужным образом отреагировал на щелчок мышью, используется обработчик с именем **onClick**, для обработки события, заключающегося в получении фокуса ввода, - обработчик **onFocus**.

Для того чтобы указать интерпретатору JavaScript на то, что обработкой события должен заниматься обработчик, необходимо включить в HTML-дескриптор следующее выражение:

```
имя_обработчика="команды_обработчика"
```

Это выражение включается в тэг, описывающий объект, с которым связано событие.

Например, если необходимо обработать событие, заключающееся в получении фокуса полем ввода, дескриптор, описывающий этот интерактивный элемент, должен иметь примерно следующий вид:

```
<input type="text" name="Inform" onFocus="alert('Hello World');">
```

Имя обработчика является одним из атрибутов HTML-дескриптора, а команды, предназначенные для обработки события, выступают в роли значения этого атрибута. В данном случае обработка события производится функцией которая выводит сообщение alert(). В принципе, обработчиком может быть не только функция, но и любая последовательность команд JavaScript в виде составного оператора.

Следующий пример демонстрирует обработку события, связанного с наведением курсора мыши на гиперссылку:

```
<a href = "http://www.ya.ru"  
onmouseover="alert('An onMouseOver event'); return false">
```

```
Такой вот обработчик!!!</a>
```

Список атрибутов, которые определены для HTML тэгов приводится ниже:

IE: Internet Explorer, F: Firefox, O: Opera, W3C: стандарт

Атрибут	Описание	Номер версии браузера			W3C
		IE	F	O	
onabort	Прерванная загрузка изображения	4	1	9	Да
onblur	утрата фокуса элементом	3	1	9	Да
onchange	Изменение содержимого в поле ввода	3	1	9	Да
onclick	Щелчок мыши на объекте	3	1	9	Да
ondblclick	Двойной щелчок мыши на объекте	4	1	9	Да
onerror	Ошибка при загрузке изображения или документа	4	1	9	Да
onfocus	Получение фокуса элементом	3	1	9	Да
onkeydown	Нажатие клавиши	3	1	Нет	Да
onkeypress	Клавиша нажата	3	1	9	Да
onkeyup	Отжатие клавиши	3	1	9	Да
onload	Завершение загрузки страницы или изображения	3	1	9	Да
onmousedown	Нажатие кнопки мыши	4	1	9	Да
onmousemove	Перемещение курсора мыши	3	1	9	Да
onmouseout	Смещение курсора мыши с объекта	4	1	9	Да
onmouseover	Наведение курсора мыши на объект	3	1	9	Да
onmouseup	Отжатие кнопки мыши	4	1	9	Да
onreset	Кнопка "Reset" нажата	4	1	9	Да
onresize	Изменение размера окна	4	1	9	Да
onselect	Выделение текста	3	1	9	Да
onsubmit	Кнопка "Submit" нажата	3	1	9	Да
onunload	Уход с веб-страницы	3	1	9	Да

#### 4. Практическое задание

1) Наберите следующий код:

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
<title>Лабораторная работа №6</title>
<style>
div {
font-size:14px;
```

```

font-family:Tahoma;
color:red;
border:1px solid #0000FF;
font-weight:bold;
width:450px;
text-align:center;
height:40px;
vertical-align:middle;
padding-top:20px;
margin-bottom:20px
}
</style>
<script language="javascript" type="text/javascript">
functionPrimerClik(event){
    switch (event.button){
        case 0: alert("Вы щелкнули левой кнопкой мыши!");
break;
        case 1: alert("Вы щелкнули средней кнопкой мыши!");
break;
        case 2: alert("Вы щелкнули правой кнопкой мыши!");
break;
    }
}
functionPrimerMove(ElementId){
    var s = document.getElementById(ElementId).style.color;

    if (document.getElementById(ElementId).style.color == "") s = 'red';

    if (s == 'red'){
        document.getElementById(ElementId).style.color = 'green';
    } else {

        document.getElementById(ElementId).style.color = 'red';
    }

}
</script>
</head>
<body onLoad="alert('Запускается лабораторная работа №5');"
onUnload="alert('Вы закрыли лабораторную работу №5. Не забудьте сдать
отчёт!!!');">
    <div onMouseDown="PrimerClik(event);">Щелкните любой кнопкой мыши по
тексту и проверьте "onMouseDown"</div>
    <div onMouseUp="PrimerClik(event);">Щелкните любой кнопкой мыши по
тексту и проверьте "onMouseUp"</div>

    <div id="Moveld" onMouseMove="PrimerMove('Moveld');">Наведите курсор
на элемент и проверьте "onMouseMove"</div>
    <div id="MoveldO" onMouseOver="PrimerMove('MoveldO');">Наведите курсор
на элемент и проверьте "onMouseOver"</div>
    <div id="MoveldS" onMouseOut="PrimerMove('MoveldS');">Наведите курсор
на элемент и проверьте "onMouseOut"</div>

    <div id="MoveldC" onclick="PrimerMove('MoveldC');">Кликните на элементе и
проверьте "onclick"</div>
    <div id="MoveldDC" onDbClick="PrimerMove('MoveldDC');">Кликните
дважды на элементе и проверьте "onDbClick"</div>
    <div id="KeyPresElementId" style="padding-bottom:20px;">
        Введите символы с клавиатуры и проверьте "onKeyPress"<br>

```

```

        <input type="text" name="KeyPresElement" id="KeyPresElement"
onKeyPress="PrimerMove('KeyPresElementId');">
    </div>
    <div id="KeyUpElementId" style="padding-bottom:20px;">
        Введите символы с клавиатуры и проверьте "onKeyUp"<br>
        <input type="text" name="KeyUpElement" id="KeyUpElement"
onKeyUp="PrimerMove('KeyUpElementId');">
    </div>
    <div id="KeyDownElementID" style="padding-bottom:20px;">
        Введите символы с клавиатуры и проверьте "onKeyDown"<br>
        <input type="text" name="KeyDownElement" id="KeyDownElement"
onKeyDown="PrimerMove('KeyDownElementID');">
    </div>

</body>
</html>

```

Проставьте комментарии. Проверьте все обработчики событий из таблицы.

- 2) Примените 10 любых событий к элементам формы для расширенной переписи населения, созданной в лабораторной работе №5.

## 5. Содержание отчёта

Отчёт должен содержать название и цель. Выполненные задания. Выводы.

## 6. Контрольные вопросы

- 1) Для чего нужны обработчики событий.
- 2) Перечислить обработчики событий клавиш мышки.
- 3) Перечислить обработчики событий курсора мышки.
- 4) Перечислить обработчики событий клавиатуры.
- 5) Перечислить обработчики событий документа.