**Частное учреждение образования**

**«Гродненский колледж бизнеса и права»**

**Лабораторная работа № 18**

**по дисциплине**

**«Программные средства создания Internet - приложений»**

**Тема:** «Обработка событий, связанных с текущим состоянием страницы или действиями пользователя»

для учащихся 4 курса специальности

2-40 01 01 «Программное обеспечение информационных технологий»

**Тема:** Обработка событий, связанных с текущим состоянием страницы или действиями пользователя.

**Цель:** обучить приемам обработки событий на HTML – странице

**Вид занятия:** Практическая работа.

**Тип занятия:** Формирование навыков.

**Время выполнения:** 2 часа

**Оборудование и методическое обеспечение:** IBM PC, приложение Блокнот, MS Internet Explorer карточки с заданием.

**Литература:**

1. Java Script.. Г. В. Гудман СПб.: Питер, 2004.

**План работы:**

1. Организационный этап занятия
2. Проверка домашнего задания
3. Информационно-подготовительный этап занятий
4. Исполнительский этап занятий
5. Постановка домашнего задания
6. Оценочно-рефлексивный этап занятия

**Ход занятия**

1. **Организационный этап занятия**

Проверка готовности учащихся к занятию, отметка отсутствующих, объявление темы и цели урока.

1. **Проверка домашнего задания**

Устный опрос учащихся по следующим вопросам:

1. Что такое функции? Почему они полезны?
2. Как определяется функция?
3. Как информация передается в функцию? Зачем это нужно? И наоборот, как можно получить информацию из функции?
4. Что такое событие?
5. В чем различие между перехватом события и всплытием события?
6. Можно ли управлять выполнением события, т.е. останавливать поведение по умолчанию. Как?
7. **Информационно-подготовительный** **этап занятия**

В качестве примера давайте создадим функцию, которая генерирует случайный цвет фона для элемента на странице:

function setElementBackground() {

var red = Math.floor(Math.random() \* 256);

var green = Math.floor(Math.random() \* 256);

var blue = Math.floor(Math.random() \* 256);

var obj = document.getElementById('element\_to\_change');

if ( obj ) {

obj.style.background = 'rgb(' + red + ',' + green + ',' + blue + ')'; }

Рассмотрим, например, такую задачу. У нас есть три квадратика, при щелчке по каждому должно появляться окно сообщений с указанием цвета квадратика, по которому щелкнули (пощелкайте по квадратикам ниже).

Итак, сначала напишем код функции, которая будет открывать окно сообщений, назовем ее "message". Как вы помните, вызов окна сообщений осуществляет стандартная функция javascript - *alert*, а текст сообщений мы будем передавать, как параметр, обозначим его как "m":

function message(m){

alert (m); }

Теперь в html-странице напишем код самих квадратиков:

<html>

<head>

<title>javascript параметры</title>

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

<script type="text/javascript" src="script.js"></script>

</head>

<body>

<table><tr>

<td><div id="red" onClick="message('Вы щелкнули по красному квадратику');"> </div></td>

<td><div id="green" onClick="message('Вы щелкнули по зеленому квадратику');"> </div></td>

<td><div id="blue" onClick="message('Вы щелкнули по синему квадратику');"> </div></td>

</tr></table>

</body>

</html>

Мы указали, что при щелчке по нашим квадратикам должна срабатывать функция "message", а в качестве параметров указали текст нужных сообщений (они будут подставляться вместо параметра "m" при обращении к функции).  
Осталось только оформить наши квадратики. Для этого на странице style.css напишем стили для них:

#red, #green, #blue{

width:100px;

height:100px;

margin:10px;}

#red{

background:red;

}

#green{

background:green;

}

#blue{

background:blue;}

Все, осталось только проверить работу сценария в браузере.

Рассмотрим пример:

<script language="JavaScript">

function rev(btn)

{ btn.value = btn.value.split("").reverse().join(""); }

</script> </p>

<form> <p>

<input type="button" value="Click Me" onclick="rev(this)">

<input type="button" value="Don't Click Me" onclick="rev(this)">

</p> </form>

*Передача информации о событиях*

До сих пор нас интересовал факт совершения события как таковой. Нас не интересовали подробности, такие как, координаты курсора, какая именно кнопка мышки или клавиша была нажата и т.д. Конечно, во многих случаях без такой информации не обойтись.

Источником такой информации может служить объект Event. Свойства этого объекта содержат все сведения связанные с событием.

если мы хотим передать информацию о событии неявно заданному обработчику, то, вместо:

<input type="button" value="Click Me"

onclick="buttonClicked()">

мы должны писать:

<input type="button" value="Click Me"

onclick="buttonClicked(event)">

тогда функция **buttonClicked()** может выглядеть так:

<script language="JavaScript">

function buttonClicked(e) {

....}

где **e** - передаваемый нам браузером параметр - экземпляр объекта **Event**. Именно из него мы и будем получать подробную информацию о событии.

***Пример.*** Событиепроисходит при нажатии левой кнопки мыши

<html>

<head>

<title></title><script>

function clicks()

{ alert("Вы думали это ссылка?")}

</script>

</head>

<body>

<a href="#" onclick="clicks()">Ссылка</a>

</body>

</html>

1. **Исполнительский этап занятия**

***Вариант 1***

*Уровень 1.* Что является событием в JavaScript?

Какие события, связанные с действиями указателя мыши, вы знаете?

*Уровень 2.* Создать скрипт: появление окна сообщений с фамилией и номером варианта после полной загрузки документа.

Создайте web-документ и установите в нём текст *Уровень 1*. При одинарном щелчке по тексту, в текущее окно должен загрузиться документ с ответом на вопросы.

При наведении указателя мыши на текст, текст в документе должен становиться красного цвета.

*Уровень 3.* Создать форму, состоящую из флажков:

Что установить в заголовке окна?

1. Уровень 3
2. Самостоятельная работа
3. Это JavaScript

При установлении пользователем флажка, заголовок окна должен измениться.

При наведении указателя на соответствующий вариант ответа, флажок должен установиться автоматически.

*Уровень 4.* Создать документ, в котором при перемещении указателя мыши, в строке заголовка должны отображаться текущие координаты курсора.

Создать фреймовую структуру, состоящую из 2 фреймов. В первый фрейм загрузите пустую страницу. Во второй фрейм загрузите страницу, на которой должен находиться рисунок. При двойном щелчке по рисунку, во второй фрейм должен загрузиться новый документ.

***Вариант 2***

*Уровень 1.* Какое место занимают события в JavaScript?

Какие события, связанные с web-страницей, вы знаете?

*Уровень 2.* Создать скрипт: появление окна сообщений с фамилией и номером варианта после завершения работы с документом.

Создайте web-документ и установите в нём текст *Уровень 1*. При двойном щелчке по тексту, в новое окно должен загрузиться документ с ответом на вопросы.

При наведении указателя мыши на текст, цвет фона документа должен стать синим.

*Уровень 3.* Создать форму, состоящую из переключателей:

Что установить в строке состояния?

1. В данном документе нет ссылок
2. Документ содержит форму
3. В документе установлен скрипт

При выборе переключателя, в строке состояния должен появиться выбранный текст.

При наведении указателя на соответствующий вариант ответа, переключатель должен включиться автоматически.

*Уровень 4.* Установить в новом документе рисунок. При наведении мыши на рисунок должно появиться окно сообщений, где будут показаны координаты той точки, где в этот момент находится мышь.

Создать фреймовую структуру, состоящую из 2 фреймов. В первый фрейм загрузите пустую страницу. Во второй фрейм загрузите страницу, на которой должен находиться один радиопереключатель, играющий роль ссылки, при выборе которой, во второй фрейм загрузиться новый документ.

Самостоятельно:

1. После того как веб-страница полностью загрузиться должно появиться окно сообщений: "Веб-страница загрузилась на 100%".
2. Когда пользователь закроет веб-страницу должно появиться окно сообщений: «До встречи!!!».
3. Наберите одно предложение. При наведении на него указателя мыши цвет текста должен стать красным, когда мышь уводят от данного текста его цвет должен измениться на синий.
4. Вставьте в документ рисунок. При перемещении указателя мыши по данному рисунку, в строке состояния должны отображаться его координаты.
5. При двойном щелчке левой кнопки мыши по данному рисунку должен измениться цвет фона документа.
6. При вызове контекстного меню в строке заголовка браузера должен возникнуть текст: «Произошел вызов контекстного меню».
7. При вызове меню помощи должно появиться новое окно браузера с документом, содержащим справочную информацию.
8. При изменении размеров окна цвет фона документа должен измениться.
9. При прокрутке окна цвет текста документа должен замениться на зелёный.
10. Вставьте в документ рисунок. При перемещении указателя мыши по документу, в строке заголовка пишем на рисунке мы или вне его
11. Выведете предложение. При щелчке на нем в строке заголовка укажите его длину.
12. **Постановка домашнего задания**

Из литературы [2], с.39-42

1. **Оценочно-рефлексивный этап занятия**

Учащимся необходимо ответить на поставленные вопросы:

1. Что нового Вы узнали на занятии?
2. Что вызвало затруднение при выполнении задания?
3. Было ли Вам интересно на занятии?