**Частное учреждение образования**

**«Гродненский колледж бизнеса и права»**

**Лабораторная работа № 20**

**по дисциплине**

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

**Тема:** «Проверка корректности информации, введенной пользователем. Обработка данных формы»

для учащихся 4 курса специальности

2-40 01 01 «Программное обеспечение информационных технологий»

**Тема:** Проверка корректности информации, введенной пользователем. Обработка данных формы.

**Цель:** выработать умения обработывать данные формы

**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. Какие управляющие элементы формы вы знаете?
2. Как создать кнопку?
3. Как создать радиокнопку?
4. Как создать флажки?
5. **Информационно-подготовительный** **этап занятия**

Для обращения к самой форме и ее элементам используют именование:

 <FORM NAME=f\_name METHOD=post ACTION="javascript:...;">

<INPUT NAME=i\_name SIZE=30 MAXLENGTH=30>

</FORM>

 После именования, ко всей форме или ее элементам можно будет обратиться по имени, что сильно облегчает процесс написания скриптов:

 window.document.f\_name.i\_name.value="Изменяем значение текстового поля";

  конечно тоже самое мы можем сделать путем использования встроенного массива forms[0], но это не слишком удобно:

 window.document.forms[0].elements[0].value="Текстовое поле";

  Например, если на вашей странице используется более одной формы, то возникает риск просто запутаться с индексами.

Далее мы будем рассматривать основные методы и свойства работы с формами:

* [*action*](http://www.eltisbook.ru/dhtml/action.php)
* [*method*](http://www.eltisbook.ru/dhtml/method.php)
* [*target*](http://www.eltisbook.ru/dhtml/target.php)
* [*elements[]*](http://www.eltisbook.ru/dhtml/elements.php)
* [*encoding*](http://www.eltisbook.ru/dhtml/encoding.php)
* [*reset()*](http://www.eltisbook.ru/dhtml/reset.php)
* [*submit()*](http://www.eltisbook.ru/dhtml/submit.php)
* [*onReset*](http://www.eltisbook.ru/dhtml/onreset.php)
* [*onSubmit*](http://www.eltisbook.ru/dhtml/onsubmit.php)
* [*text field*](http://www.eltisbook.ru/dhtml/input-text.php)
* [*select*](http://www.eltisbook.ru/dhtml/select.php)
* [*buttons*](http://www.eltisbook.ru/dhtml/buttons.php)

 Первым делом, предлагаю вам изучить перехват события Action при помощи JavaScript.

Action - свойство HTML форм, которое предписывает браузеру, что он должен сделать после выполнения формы (нажатие на кнопку submit или на клавишу Enter). JavaScript с легкостью может перепрограмировать данное свойство по усмотрению программиста.

***Использование Action в javascript.***

 <FORM METHOD=post ACTION="javascript:window.alert('Здесь JavaScript'); void(0);">  
<INPUT TYPE=submit VALUE="Продемонстрировать JavaScript в action">  
</FORM>

Подобный пример использования *Action в javascript* мы уже рассматривали, поэтому подробно останавливаться на нем мы не будем.

Единственным важным моментом в нашем примере является использование метода POST для нашей формы. А связанно это с методом передачи данных из формы. Использование метода GET предназначено, как правило для языка PHP. Ведь при использовании этого метода в адрес URL добавляется символ "?". При обработке скрипта, попавшего в адресную строку JavaScript интерпритатор встретит символ ? и незамедлительно выдаст ошибку.

Для предотвращения подобных нежелательных событий и используется метод POST, который передает всю информацию из формы в теле HTTP протокола.

Свойство method позволяет определить, каким способом данные будут отправленны на сервер. Существуют всего два значения для этого свойства:

* **GET**
* **POST**

JavaScript позволяет динамически менять значение этого свойства и в этом уроке вы самостоятельно в этом убедитесь.

*Использование свойства method в JS.*

Всем нам уже давно знакомо свойство HTML форм – method. Как известно это свойство определяет способ передачи данных от формы обработчику.

JavaScript может самостоятельно определять и изменять значение этого свойства, как и множества других.

 Пример:

<FORM NAME=m ACTION="javascript:window.alert('Мы используем JavaScript – код в качестве an URL');void(0);" method="post">

<INPUT TYPE=button onClick="window.document.m.method='post'; window.alert('метод POST активирован');" VALUE="Метод POST">

<INPUT TYPE=button onClick="window.document.m.method='get'; window.alert('метод GET активирован'=);" VALUE="Метод GET">

<INPUT TYPE=submit VALUE="JavaScript в ACTION">

</FORM>

В данном примере мы создаем простую html форму, состоящую из трех кнопок. В первую кнопку под названием **Метод POST** встроен обработчик события onClick, который позволяет изменить значение метода на POST:

 window.document.m.method='post';

Во вторую кнопку с названием **Метод GET** аналогичным образом встроен обработчик onClick:

 window.document.m.method='get';

 Таким образом JS позволяет в режиме реального времени изменять метод передачи данных от формы, что может оказаться очень полезным при создании универсальных форм, способных встраиваться в различные html страницы.

***Использование метода submit()***

Использование метода JavaScript - submit() аналогично использованию элемента формы типа submit с одним лишь отличием, что можно инициировать передачу данных на сервер без ведома пользователя.

То есть, если вам необходимо получить о пользователе некоторую информацию после совершения им определенных действий ( например ввода номера кредитной карты)) ), то методом submit() можно передать эти данные на сервер и пользователь об этом ничего не узнает. Пример:

 <FORM NAME=s METHOD=post ACTION="javascript:window.alert('Данные отправлены');void(0);">  
Введите цифру или букву:

<INPUT SIZE=1 MAXLENGTH=1>

</FORM>

<A HREF="javascript:document.s.submit();">

Отправить данные

</A>

В этом простейшем примере мы создаем форму с полем для ввода, где пользователю предлагается ввести некоторый символ.

Далее мы создаем ссылку со схемой URL–JavaScript, при нажатии на которую данные отправляются на сервер:

 HREF="javascript:document.s.submit();">

***Событие onReset*** возникает в момент перезагрузки полей формы. JavaScript позволяет с легкостью управлять данным событием.

***Работа с кнопками в JavaScript.***

Среди всех элементов HTML форм есть один, который без вмешательства JavaScript или другого языка web-программирования не способен произвести каких–либо действий. Многие из вас уже догадались, что речь здесь идет именно об использовании кнопок в JavaScript.

Приведем простейший пример:

 <FORM>  
<INPUT TYPE=button VALUE="Просто кнопка" >

<INPUT TYPE=button VALUE="JavaScript кнопка" onClick="window.alert('кнопка была нажата');">

</FORM>

Этот простейший пример демонстрирует, как из безжизненного HTML элемента получается вполне работоспособная JavaScript кнопка.

Таким образом, добавляя к кнопкам скрипты JavaScript, мы можем инициировать действия любой сложности.

1. **Исполнительский этап занятия**

*Самостоятельно:*

Создайте форму. Поле ввода логина, поле ввода пароля, пол (выбирается м/ж), страна проживания вводится пользователем, любимый цвет (выбирается из флажков), кнопки «Отправить» и «Сброс». Первая отправляет данные и формирует новую страницу со всеми полученными данными, вторая очищает поля.

Создайте сами форму, с различными видами вопросов, полями ввода, флажками, радиокнопками.

1. **Постановка домашнего задания**

Из литературы [2], с.96-100

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?