**Частное учреждение образования**

**«Гродненский колледж бизнеса и права»**

**Лабораторная работа № 15**

**по дисциплине**

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

**Тема:** «Создание Использование иерархии классов JavaScript для доступа к элементу HTML - страницы»

для учащихся 4 курса специальности

2-40 01 01 «Программное обеспечение информационных технологий»

**Тема:** Использование иерархии классов JavaScript для доступа к элементу HTML -   
страницы.

**Цель:** обучить способам доступа к элементам HTML – страницы в соответствии с иерархией классов JavaScript.

**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. Дайте характеристику иерархии объектов JavaScript
2. Перечислите методы объекта window предназначенные для вывода сообщений на экран. В чем их различие?
3. Опишите правило организации доступа к объектам JavaScript.
4. **Информационно-подготовительный** **этап занятия**

В языке JavaScript все элементы на web-странице выстраиваются в иерархическую структуру. Каждый элемент предстает в виде объекта. И каждый такой объект может иметь определенные свойства и методы. В свою очередь, язык JavaScript позволит Вам легко управлять объектами web-страницы, хотя для этого очень важно понимать иерархию объектов, на которые опирается разметка HTML.

**Задание 1.** Изучить HTML-код, описать, как будет выглядеть страница.

*<html>*

*<head>*

*</head>*

*<body bgcolor=#ffffff>*

*<center>*

*<img src="home.gif" name="pic1" width=200 height=100>*

*</center>*

*<p>*

*<form name="myForm">*

*Name:*

*<input type="text" name="name" value=""><br>*

*e-Mail:*

*<input type="text" name="email" value=""><br><br>*

*<input type="button" value="Push me" name="myButton" onClick="alert('Yes')">*

*</form>*

*<p>*

*<center>*

*<img src="ruler.gif" name="pic2" width=300 height=15>*

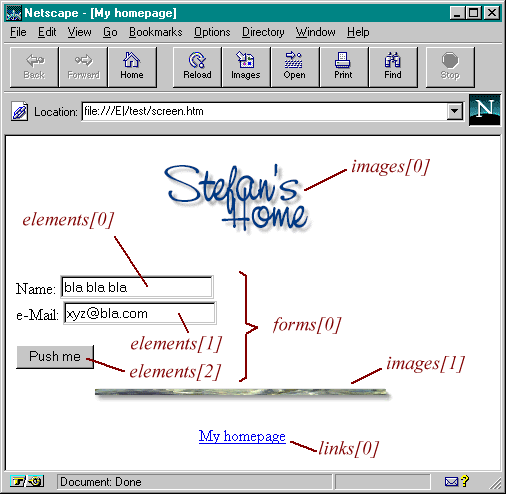
*<p>*

*<a href="http://rummelplatz.uni-mannheim.de/~skoch/">My homepage</a>*

*</center>*

*</body>*

*</html>*



Итак, мы имеем два рисунка, одну ссылку и некую форму с двумя полями для ввода текста и одной кнопкой. С точки зрения языка JavaScript окно браузера - это некий объект window. Внутри окна мы можем разместить документ HTML. Такая страница является объектом document. Это означает, что объект document представляет в языке JavaScript загруженный на настоящий момент документ HTML. Примерами объекта HTML являются, к примеру, ссылка или заполняемая форма. На следующем рисунке иллюстрируется иерархия объектов, HTML-страницей из примера:



К свойствам объекта document относятся, например, цвет фона для web-страницы. Однако для нас гораздо важнее то, что все без исключения объекты HTML являются свойствами объекта document.

Разумеется, мы должны иметь возможность получать информацию о различных объектах в этой иерархии и управлять ею. Для этого мы должны знать, как в языке JavaScript организован доступ к различным объектам. Как видно, каждый объект иерархической структуры имеет свое имя.

Если нужно узнать, какой текст ввел читатель в первый элемент формы, то сперва необходимо выяснить, как получить доступ к этому объекту. И снова начинаем мы с вершины нашей иерархии объектов. Затем прослеживаем путь к объекту с именем *elements[0]* и последовательно записываем названия всех объектов, которые минуем. В итоге выясняется, что доступ к первому полю для ввода текста можно получить, записав:

document.forms[0].elements[0]

Там Вы найдете, что элемент, соответствующий полю для ввода текста, имеет свойство value, которое как раз и соответствует введенному тексту. Итак, теперь мы имеем все необходимое, чтобы прочитать искомое значение. Для этого нужно написать на языке JavaScript строку:

name= document.forms[0].elements[0].value;

name= document.myForm.name.value;(регистр)

Полученная строка заносится в переменную name. Следовательно, теперь мы можем работать с этой переменной, как нам необходимо. Например, мы можем создать выпадающее окно, воспользовавшись командой *alert("Hi " + name)*. В результате, если читатель введет в это поле слово 'Stefan', то по команде *alert("Hi " + name)* будет открыто выпадающее окно с приветствием ' Hi Stefan '.

В JavaScript многие свойства объектов доступны не только для чтения. Вы также имеете возможность записывать в них новые значения.

<form name="myForm">

<input type="text" name="input" value="hello hi привет">

<input type="button" value="Write"

onClick="document.myForm.input.value= 'Моё имя'; ">

Исходный код скрипта:

<html>

<head>

<title>Objects</title>

<script language="JavaScript">

function first()

{alert("The value of the textelement is: " +

document.myForm.myText.value);}

function second()

{ var myString= "The checkbox is ";

if (document.myForm.myCheckbox.checked) myString+= "checked"

else myString+= "not checked";

alert(myString);}

</script>

</head>

<body bgcolor=lightblue>

<form name="myForm">

<input type="text" name="myText" value="bla bla bla">

<input type="button" name="button1" value="Button 1" onClick="first()">

<br>

<input type="checkbox" name="myCheckbox" CHECKED>

<input type="button" name="button2" value="Button 2" onClick="second()">

</form>

<p><br><br>

<script language="JavaScript">

document.write("The background color is: ");

document.write(document.bgColor + "<br>");

document.write("The text on the second button is: ");

document.write(document.myForm.button2.value);

</script></body></html>

Программа выдачи простого предупреждения при выборе гипертекстовой ссылки. Схема URL - JavaScript

<HTML> <HEAD>

<title>Самый первый пример JavaScript</title>

</HEAD>

<BODY>

<center>

<h1>Мой первый пример JavaScript</h1>

<hr>

</center>

В текст этого документа внедрена гипертекстовая ссылка

<a href="javascript:window.alert('Do you speak English?')">

"Don`t click here" </a>.

Любопытно, что будет, если все-таки выбрать?

</BODY>

</HTML>

Запись содержимого в документ. Это выполняется с помощью метода write объекта document. Ниже приведена простая программа, которая записывает строку *"Hello, World!"* в HTML-документ:

<HTML>

<HEAD>

<TITLE>Hello, World!</TITLE>

</HEAD>

<BODY>

<SCRIPT LANGUAGE="JavaScript">

document.write("<H1>Hello, World!</H1>")

/ Запись строки "Hello, World!" в документ.

</SCRIPT>

</BODY>

</HTML>

<HTML>

<SCRIPT LANGUAGE="JavaScript">

// Генерирует весь документ на основе данного сценария.

document.write("<HEAD><TITLE>My Document</TITLE></HEAD>");

document.write("<BODY><H1>This is my page.</H1></BODY>");

</SCRIPT>

</HTML>

Доступ к элементам формы

<html >  
<head>  
  <title>JavaScript</title>  
  <script language="JavaScript" type="text/javascript">  
    function ShowText(f) {  
      alert("Entered text: " + f.elements["TextBox1"].value);  
    }  
  </script>  
</head>  
<body>  
  <form action="">  
    <input type="text" name="TextBox1" />  
    <input type="button" value="Show text" onclick="ShowText(this.form);" />  
  </form>  
</body>  
</html>

1. **Исполнительский этап занятия**

*Уровень 1.* Перечислите сокращённые операторы присвоения.

Дайте характеристику иерархии объектов JavaScript. Опишите правило организации доступа к объектам JavaScript.

*Уровень 2.* Создаете HTML-документ и с помощью сценария запишите в него номер варианта вашей работы

Создайте псевдо ссылку: УРОВЕНЬ. При нажатии на неё номер выполняемого уровня должен появиться в заголовке окна браузера.

Создайте окно с запросом: «Ведите фамилию и имя». Текст по умолчанию: «Сидор Роман». Затем должно возникнуть оно сообщений с текстом: Меня зовут Фамилия Имя (введенные пользователем).

*Уровень 3.* Создайте HTML-документ. Наберите в этом документе текст: «Примеры сценариев на JavaScript».Создайте кнопку, при нажатии которой, в строке состояния должна прописаться сегодняшняя дата.

Создайте кнопку, при нажатии которой, появится окно сообщений с датой последнего изменения файла.

Создайте форму, состоящую из одной строки ввода. Создать ссылку, при нажатии которой, должно появляться диалоговое окно, отражающее содержимое созданной строки.

Создайте кнопку и строку ввода:

|  |  |
| --- | --- |
| **Цвет текста документа** |  |

При нажатии на кнопку должен установиться набранный пользователем цвет текста документа

*Уровень 4.* Создать две текстовые строки:

|  |  |
| --- | --- |
| **Заголовок** |  |

|  |  |
| --- | --- |
| **Строка состояния** |  |

и кнопку. При нажатии на кнопку должно измениться содержание заголовка окна браузера и строки состояния.

Создать HTML-документ, в котором будут появляться окна с запросом: первое окно – окно с запросом ввести название города, второе – с запросом ввести название улицы, третье окно – с запросом ввести номер дома. После выполнения запросов, должно появиться окно сообщений с содержимым: Вы живёте в ГОРОДЕ на УЛИЦЕ в ДОМЕ.

Создайте окно запроса: Введите число меньше 7 или больше 50. Данное окно должно возникать до тех пор, пока пользователь не введет нужное число. В случае правильного выполнения пользователем запроса должно возникнуть окно сообщений: Введено верное число.

Создать HTML-документ, в котором возникает окно запроса, предлагающее пользователю ввести любое число. Таким образом, пользователь должен ввести 9 чисел, а сценарий должен посчитать произведение всех введенных нечётных чисел и вывести полученное произведение через окно сообщений.

Создайте HTML-документ, в котором должно появиться окно с запросом: Вам нравиться наш сайт? Если пользователь введёт ответ на вопрос и выберет ОК, то должно появиться окно сообщений: Спасибо за ответ. Если пользователь выберет Отмена, то должно появиться окно с запросом: Может оставите тогда свои пожелания? Если пользователь введёт свои пожелания и выберет ОК, то должно появиться окно с сообщением: Спасибо за пожелания. Если пользователь выберет Отмена, то должно появиться окно сообщений: Всего хорошего!

Самостоятельно- сделать ВСЕ каждому!!!!

|  |  |
| --- | --- |
| 1 | Вывести в алерте название заголовка документа, поменять заголовок, вывести новый. |
| 2 | Выведите содержимое адресной строки браузера в теле документа. |
| 3 | Поле ввода, кнопка, по нажатию кнопки сделать заголовком страницы то, что ввели в поле ввода. |
| 4 | Флажок, если выбран вывести – выбор сделан, если не выбран – вы не выбрали ничего. |
| 5 | Ввести тест в воле ввода, по кнопке вывести его на 2 строки ниже, по центру. |
| 6 | Поменять кнопками фон страницы |
| 7 | 2 кнопки – рамблер и гугл, по каждой идет на соответствующие поисковики. |
| 8 | Поместить картинку, под ней вывести ссылку на эту картинку |

1. **Постановка домашнего задания**

Из литературы [2], с.44-47

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?