Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Филиал

«Минский радиотехнический колледж»

Учебная дисциплина «Программные средства создания Internet-приложений»

**Инструкция**

по выполнению лабораторной работы

«Создание форм на HTML-странице и изменение свойств элементов формы»

Минск

2017

**Лабораторная работа № 4**

**Тема работы: «Создание форм на HTML-странице и изменение свойств элементов формы»**

**1. Цель работы**

Формирование практических умений создания форм на HTML-странице.

**2. Задание**

Создать HTML-форму регистрации.

**3. Оснащение работы**

ПК, текстовый редактор, браузер.

**4. Основные теоретические сведения**

Формы являются одним из важных элементов любого сайта и предназначены для обмена данными между пользователем и сервером. Область применения форм не ограничена отправкой данных на сервер, с помощью клиентских скриптов можно получить доступ к любому элементу формы, изменять его и применять по своему усмотрению.

Элементы форм представлены в таблице:

|  |  |  |
| --- | --- | --- |
| **Название** | **Описание** | **Пример** |
| Однострочное текстовое поле | Предназначено для ввода строки символов с помощью клавиатуры. |  |
| Поле для пароля | Обычное текстовое поле, но отличается тем, что все символы отображаются звездочками. Предназначено для того, чтобы никто не подглядел вводимый пароль. |  |
| Многострочный текст | Область, в которой можно вводить несколько строк текста. |  |
| Кнопка | Элемент формы, на который нужно нажимать. |  |
| Кнопка SUBMIT | После нажатия на эту кнопку данные формы отправляются на сервер и обрабатываются программой, указанной параметромaction тега **FORM**. |  |
| Кнопка RESET | При нажатии на эту кнопку, данные формы возвращаются в первоначальное значение. |  |
| Переключатели (radiobutton) | Используются для выбора только одного варианта из предложенных. |  |
| Флажки (checkbox) | Используются для выбора одного и более вариантов из предложенных. |  |
| Поле со списком | Предназначен для выбора одного или нескольких значений из списка. |  |
| Скрытое поле | Скрытое поле не отображается на странице и прячет свое содержимое от пользователя. |  |
| Поле с изображением | Аналогично по действию кнопке Submit, но представляют собой рисунок. |  |
| Отправка файла | Создает поле для ввода имени файла, который пересылается на сервер. |  |

Для создания форм используется парный тег <FORM></FORM>.

Тег <FORM> может содержать следующие атрибуты таблица 1.

Таблица 1 – Атрибуты тега Form

|  |  |
| --- | --- |
| Атрибут | Назначение |
| NAME | Имя формы |
| ACTION | URL, по которому следует передать введенную информацию для последующей обработки |
| METHOD | Метод передачи данных из формы |

Документ может содержать любое количество форм, но одновременно на сервер может быть отправлена только одна форма. По этой причине данные форм должны быть независимы друг от друга.

Когда форма отправляется на сервер, управление данными передается программе, заданной атрибутом **action** тега **<form>**.

Если для отправки данных используется метод **GET**, то адресная строка может принимать следующий вид.

[**http://www.htmlbook.ru/cgi-bin/handler.cgi?nick=%C2%E0%ED%FF+%D8%E0%EF%EE%F7%EA%E8%ED&page=5**](http://www.htmlbook.ru/cgi-bin/handler.cgi?nick=%C2%E0%ED%FF+%D8%E0%EF%EE%F7%EA%E8%ED&page=5)

Если управляющие элементы указаны вне содержимого тега FORM, то они не создают форму, а используются для построения пользовательского интерфейса на веб-странице, то есть для привнесения в нее различных кнопок, флажков, полей ввода. Имена элементам формы присваиваются через их атрибут NAME.

Сама форма обычно предназначена для получения от пользователя информации для дальнейшей пересылки её на сервер, где данные формы принимает программа-обработчик. Такая программа может быть написана на любом серверном языке программирования вроде PHP, Perl и др. Адрес программы указывается в атрибуте action тега <form>.

*Пример:*

**<!DOCTYPE HTML>**

**<html>**

**<head>**

**<meta charset=" windows-1251">**

**<title>Данные формы</title>**

**</head>**

**<body>**

**<form action="/example/handler.php">**

**<p><input name="login"> <input type="password" name="pass"></p>**

**<p><input type="submit"></p>**

**</form>**

**</body>**

**</html>**

Допускается использовать несколько форм на странице, но они не должны вкладываться одна в другую.

Передача на сервер происходит двумя разными методами: GET и POST, для задания метода в теге <form> используется атрибут method, а его значениями выступают ключевые слова get и post. Если атрибут method не задан, то по умолчанию данные отправляются на сервер методом GET. В таблице показаны различия между этими методами.

|  |  |  |
| --- | --- | --- |
|  | **GET** | **POST** |
| Ограничение на объём | 4 Кб | Ограничения задаются сервером. |
| Передаваемые данные | Видны сразу всем. | Видны только при просмотре через расширения браузера или другими методами. |
| Кэширование | Страницы с разными запросами считаются различными, их можно кэшировать как отдельные документы. | Страница всегда одна. |
| Закладки | Страницу с запросом можно добавить в закладки браузера и обратиться к ней позже. | Страницы с разными запросами имеют один адрес, запрос повторить нельзя. |

Какой метод используется легко определить по адресной строке браузера. Если в ней появился вопросительный знак и адрес стал похож на этот, то это точно GET.

**http://www.google.ru/search?q=%D1%81%D0%B8%D1%81%D1%8C%D0%BA%D0%B8&ie=utf-8**

Ниже перечислены типовые области применения этих методов на сайтах.

**GET:**

Передача небольших текстовых данных на сервер, поиск по сайту: поисковые системы, формы поиска по сайту всегда отправляются методом GET, это позволяет делиться результатами поиска с друзьями, слать ссылку по почте или выкладывать её на форуме.

**POST:**

Пересылка файлов (фотографий, архивов, программ и др.); отправка комментариев; добавление и редактирование сообщений на форуме, блоге. Работа с формой по умолчанию происходит в текущей вкладке браузера, при этом допустимо при отправке формы изменить этот параметр и открывать обработчик формы в новой вкладке или во фрейме. Такое поведение задаётся через «имя контекста», которое выступает значением атрибута target тега <form>. Популярные значения это \_blank для открытия формы в новом окне или вкладке, и имя фрейма, которое задаётся атрибутом name тега <iframe>.

*Пример (открытие формы во фрейме):*

**<!DOCTYPE html>**

**<html>**

**<head>**

**<meta charset="windows-1251">**

**<title>Использование фрейма</title>**

**</head>**

**<body>**

**<p><iframe name="area" width="500" height="200"></iframe></p>**

**<form action="handler.php" target="area">**

**<p><input placeholder="Введите текст" name="text">**

**<p><input type="submit" value="Отправить"></p>**

**</form>**

**</body>**

**</html>**

в HTML5 есть возможность отделить форму от её элементов. *Это сделано для удобства и универсальности, так, сложный макет может содержать несколько форм, которые не должны пересекаться меж собой или к примеру, некоторые элементы выводятся с помощью скриптов в одном месте страницы, а сама форма находится в другом.* Связь между формой и её элементами происходит в таком случае через идентификатор формы, а к элементам следует добавить атрибут form со значением, равным этому идентификатору.

*Пример (связывание формы с полями):*

**<!DOCTYPE html>**

**<html>**

**<head>**

**<meta charset=" windows-1251">**

**<title>Форма</title>**

**</head>**

**<body>**

**<form id="auth" action="handler.php" method="post"></form>**

**<p>...</p>**

**<p><input name="login" form="auth">**

**<input type="password" name="pass" form="auth"></p>**

**<p><input type="submit" form="auth"></p>**

**</body>**

**</html>**

Для определения элементов формы могут использоваться следующие теги, представленные в таблице 2.

Таблица 2 – Теги для определения элементов формы

|  |  |
| --- | --- |
| Тег | Назначение |
| <TEXTAREA> </TEXTAREA> | Многострочное текстовое поле ввода |
| <INPUT> | Поля ввода |
| <SELECT> </SELECT> | Меню-список |

Тег <TEXTAREA> имеет следующие атрибуты, представленные в таблице 3.

Таблица 3 – Атрибуты тега Textarea

|  |  |
| --- | --- |
| Атрибут | Назначение |
| NAME | Имя поля ввода |
| ROWS | Число строк в поле ввода |
| COLS | Ширина поля ввода в символах |

Тег <INPUT> имеет следующие атрибуты, представленные в таблице 4

Таблица 4 – Атрибуты тега Input

|  |  |
| --- | --- |
| Атрибут | Назначение |
| CHECKED | Элемент формы CHECKBOX или RADIO будет отмечен |
| SIZE | Размер поля ввода в символах |
| MAXLENGTH | Количество символов, которое можно ввести в поле ввода |
| NAME | Имя поля ввода |
| SRC | Указывает путь к изображению (используется вместе со значением IMAGE атрибута TYPE) |
| VALUE | Устанавливает текст по умолчанию для поля ввода текста или пароля. Для флажка или переключателя указывает значение, возвращаемое серверу в случае выбора флажка или переключателя. Для кнопок определяет надпись |
| TYPE | Определяет тип поля ввода; по умолчанию создается однострочное текстовое поле ввода |

Возможные значения атрибута TYPE представлены в таблице 5.

Таблица 5 – Значения атрибута Type

|  |  |
| --- | --- |
| Атрибут | Назначение |
| CHECKBOX | Флажок; может принимать значение ON (отмечен) или  OFF(не отмечен) |
| HIDDEN | Скрытое поле |
| IMAGE | Изображение |
| TEXT | Однострочное поле ввода |
| PASSWORD | Модифицированное текстовое поле (при вводе текста вместо символов отображаются звездочки) |
| RADIO | Переключатель (используется для выбора одного вари- анта из нескольких)\* |
| RESET | Кнопка, при нажатии на которую поля формы принимают значения по умолчанию |
| SUBMIT | Кнопка отправки данных |
| BUTTON | Кнопка, определенная пользователем (т. е. конкретного действия за данной кнопкой не закреплено, оно задается пользователем) |
| FILE | Поле ввода и кнопка «Обзор» для поиска файла на диске |
| \* Если переключателям даны одинаковые имена (в атрибуте NAME), они объединяются в группу; из группы переключателей можно выбрать только один. | |

Тег <SELECT> имеет следующие атрибуты, представленные в таблице 6.

Таблица 6 – Атрибуты тега Select

|  |  |
| --- | --- |
| Атрибут | Назначение |
| MULTIPLE | Дает возможность выбора нескольких пунктов меню при удержании клавиши Ctrl |
| NAME | Определяет имя меню |
| SIZE | Определяет количество видимых пунктов меню |

Между тегами <SELECT> </SELECT> находятся значения, которые может выбирать пользователь. Они перечисляются с помощью тега <OPTION>.

Тег <OPTION> имеет следующие атрибуты, представленные в таблице 7.

Таблица 7 – Атрибуты тега Option

|  |  |
| --- | --- |
| Атрибут | Назначение |
| VALUE | Значение, присваиваемое выбранному элементу списка и отсылаемое серверу |
| SELECTED | Означает, что данный элемент списка будет выбран |

Пример создания формы в документе form.html приведен ниже, результат – на рисунке1

<HTML>

<HEAD> <TITLE> Анкета </TITLE> </HEAD>

<BODY>

<FORM action=["http://question.by/result.cgi](http://question.by/result.cgi)">

Имя <BR>

<INPUT type="text" name="name" size="20"><BR> Фамилия <BR>

<INPUT type="text" name="surname" size="20">

<BR><BR>

Пол <INPUT type="radio" name="pol"

value="m" checked> &nbsp;&nbsp;

<INPUT type="radio" name="pol" value="w"><BR><BR> Ваши увлечения:<BR>

<INPUT type="checkbox" name="choice1" value="1">

Спорт<BR>

<INPUT type="checkbox" name="choice2" value="2">

Программирование<BR>

<INPUT type="checkbox" name="choice3" value="3">

Вышивание<BR>

<INPUT type="checkbox" name="choice4" value="4">

Чтение<BR>

<INPUT type="checkbox" name="choice5" value="5">

Прочее

<INPUT type="text" name="hobby" size="20">

<BR><BR>

<INPUT type="submit" name="send"

value="Отправить данные">

</FORM>

</BODY>

</HTML>

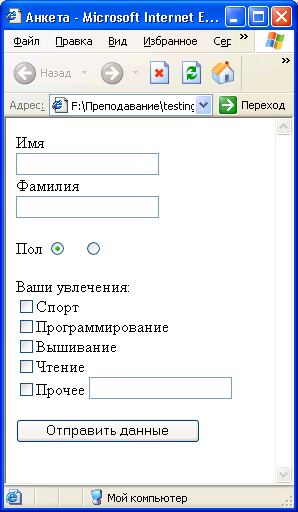


Рисунок 1 – Документ form.html в Internet Explorer

При отправке данных создаются пары параметр=значение, где параметру соответствует имя элемента (атрибут NAME), а значению – данные, введенные пользователем (в частности атрибут VALUE).

**5. Порядок выполнения работы**

1. Создайте документ Lab8.html.
2. Установите название документа, цвет фона и текста
3. Создайте регистрационную форму, содержащую:

* текстовое поле для ввода имени;
* текстовое поле для ввода фамилии;
* поле для ввода пароля;
* меню-список для выбора страны проживания;
* группу выпадающих списков для указания даты рождения;
* переключатель для указания пола;
* переключатели-флажки для указания интересов;
* кнопку очистки формы;
* кнопку отправки данных формы;
* многострочное текстовое поле для ввода примечания.

1. Установите надписи на кнопках.
2. В поле для примечания укажите текст по умолчанию.
3. Для полей ввода имени и фамилии укажите размер поля и максимальное количество вводимых символов.
4. Добавьте на страницу графическое изображение, используя значение IMAGE атрибута TYPE.

**6. Форма отчета о работе**

*Лабораторная работа № \_\_\_*

*Номер учебной группы \_\_\_\_\_\_\_\_\_\_\_\_*

*Фамилия, инициалы учащегося \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Дата выполнения работы \_\_\_\_\_\_\_\_\_\_\_\_\_*

*Тема работы: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Цель работы: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Оснащение работы: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Результат выполнения работы: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

**7. Контрольные вопросы и задания**

1. Перечислите атрибуты тега <FORM>.

2. Перечислите известные вам теги определения отдельных элементов формы.

3. Какие атрибуты тега <TEXTAREA> вы знаете?

4. Как можно создать выпадающий список значений?

5. Для чего используется атрибут MULTIPLE?

**8. Рекомендуемая литература**

**Прохоренок, Н.А**. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера / Н.А.Прохоренок. – СПб.: БХВ-Петербург, 2010.

**Никсон, Р.** Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 / Р. Никсон. — СПб.: Питер, 2016.