

## Практическое занятие № 9 «Оформление HTML-форм»

- Цель работы:**
1. Изучение особенностей задания и оформления форм.
  2. Получение навыков по созданию Web-страниц с формами.
  3. Закрепление теоретического материала.

**Оборудование и ПО:** ПК, операционная система Windows7, офисный пакет Microsoft.

**Порядок выполнения работы:**

1. Ознакомление с методическими рекомендациями по заданию и оформлению форм.
2. Выполнение практической части задания **в соответствии с разделом 4**.
3. Ответы на контрольные вопросы.
4. Оформление отчета.

### 1. Понятие формы. Способы задания и оформления.

HTML-формы. Это те самые формы, которые позволяют пользователю регистрироваться на сайте, входить на сайт, отправлять e-mail администратору и т.д. Через такие формы на многих сайтах посетители могут общаться с владельцем сайта (отсылать ему какие-то данные).

Форма - это набор полей для ввода, при этом каждое поле имеет свое уникальное имя. Сама по себе она ничего не значит - это обычный объект HTML.

Для получения данных из формы, нужен какой-то обработчик. Так что имейте в виду, что форма состоит из двух частей: одна - это внешняя часть, которая видна на html-странице куда пользователь вводит данные. Вторая часть - это обработчик, который принимает на себя данные из каждого поля формы и совершает над ними какие-то действия.

Обработчик пишется на каком-то языке программирования, простым html не обойтись. В нашем случае это будет язык PHP (в него мы вникать не будем).

Перед тем как мы приступим к созданию формы давайте создадим блокнотом еще одну html-страницу, в которой будем записывать html-код нашей формы. Пусть это будет formpage.html.

В этой странице нам понадобится одна таблица с одной строкой и одной ячейкой.

**Пример:**

```
<!DOCTYPE html public "-//W3C//DTD html 4.01 Transitional//EN">
<html>
<head>
<title>страница изучения форм</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<meta name="Keywords" content="формы">
<meta name="Description" content="создание форм в html">
</head>
<body>
<h1 align="center">Формы</h1>
<table border="1" width="800" bgcolor="white" align="center" bordercolor="black" cellpadding="10">
<tr>
<td>
</td>
</tr>
</table>
</body>
</html>
```

Проверьте в браузере как выглядит наша страничка для изучения написания форм.

Форма создается с помощью парного тега `<form>`, который имеет несколько основных атрибутов. Это атрибут **action** - обработчик (какой файл будет обрабатывать данные формы), атрибут **method** - как будут пересылаться данные от формы к обработчику. Значение "GET" - открытая передача через строку браузера или "POST" - закрытый метод (пользователи не увидят какие данные передаются в обработчик). Обычно используется метод POST. Еще один атрибут - **name**: мы должны указать имя формы.

**Пример:**

```
<h1 align="center">Формы</h1>
<table border="1" width="800" bgcolor="white" align="center" bordercolor="black" cellpadding="10">
<tr>
<td>
<form action="obrabotka.php" method="POST" name="form1">
</form>
</td>
</tr>
</table>
```

Тег формы у нас готов, далее будем вставлять в него элементы (поля для ввода данных). Для примера создадим форму заказа какого-нибудь товара. Первым сделаем поле ввода ФИО. Прописываем в html-коде абзац с указаниями для пользователя.

```
<p>Введите свое ФИО</p>
```

В абзац вставим тег переноса строки и на следующей строке пропишем поле для ввода текстовой информации. Для этого используем одинарный тег `<input>` с указанием необходимых атрибутов.

**Пример:**

```
<h1 align="center">Формы</h1>
<table border="1" width="800" bgcolor="white" align="center" bordercolor="black" cellpadding="10">
<tr>
<td>
<form action="obrabotka.php" method="POST" name="form1">
<p>Введите свое ФИО<br>
<input type="text" name="fio">
</p>
</form>
</td>
</tr>
</table>
```

## 2. Поля. Типы, способы задания и изменения

Атрибут `type` со значением `text` - поле для ввода текста. Атрибут `name` - обязательно дать имя полю. Следующим абзацем создадим поле ввода пароля.

**Пример:**

```
<h1 align="center">Формы</h1>
<table border="1" width="800" bgcolor="white" align="center" bordercolor="black" cellpadding="10">
<tr>
<td>
<form action="obrabotka.php" method="POST" name="form1">
<p>Введите свое ФИО<br>
<input type="text" name="fio">
</p>
<p>Введите пароль<br>
<input type="password" name="pass">
</p>
</form>
</td>
</tr>
</table>
```

Как мы видим - изменились тип и имя поля. Тип `"password"` позволяет нам вводить пароль скрытый для чтения.

Добавим поле ввода email:

```
<p>Введите свой email<br>
<input type="text" name="email">
</p>
```

Далее появляется новый элемент - `radio` переключатель, который позволяет выбрать только одну позицию из нескольких. В нашем примере - это выбрать формат диска: CD или DVD.

```
<p>Выберите формат диска<br>
<input type="radio" name="format" value="cd">CD<br>
<input type="radio" name="format" value="dvd">DVD<br>
</p>
```

У нас появился новый тип тега `<input>` это `type="radio"`. И еще мы видим что оба поля имеют одинаковое имя. Это говорит браузеру о том, что можно выбрать только один элемент. Если же мы зададим каждому элементу разное имя, то клиент сможет выбрать сразу два элемента.

Атрибут **value** определяет значение элемента формы, для точного установления серверной программой какой вариант выбрал клиент.

Теперь давайте добавим еще один переключатель, который позволит пользователю выбрать оба элемента. Тип такого переключателя - **checkbox**. Имена элементам задаем разные.

**Пример:**

```
<p>Какие курсы должны быть на диске<br>
<input type="checkbox" name="site">Курс по созданию сайтов<br>
<input type="checkbox" name="book">Курс по созданию книг<br>
</p>
```

### 3. Выпадающий список

Еще один элемент формы - это выпадающий список. Для его создания так же прописываем новый абзац и используем другой тег - это парный тег `<select>` и уже внутри этого тега прописываем пункты нашего выпадающего списка, вписывая каждый пункт списка в парный тег `<option>`.

**Пример:**

```
<p>Выберите тип доставки<p><br>
<select name="delivery">
<option>Срочная</option>
<option>Обычная</option>
</select>
<p></p>
```

Если мы хотим что бы были видны оба пункта можно к тегу `<select>` применить атрибут `size`, указав в значении количество пунктов. Но выбрать можно только один. Если же надо сделать так, что-бы пользователь мог выбрать несколько пунктов, используем атрибут `multiple` присвоив ему такое-же значение - `multiple`.

Следующий элемент формы - текстовая область. Применяем тег `<textarea>`.

**Пример:**

```
<p>Введите адрес доставки<br>
<textarea name="address" cols="40" rows="5">
</textarea>
<p></p>
```

В значении атрибута `cols` указываем количество символов в ширину, а в значении атрибута `rows` - количество рядов (строк). Если будет больше рядов, чем прописано - появится прокрутка.

И последний элемент нашей формы - кнопка "отправить". Открываем новый абзац и через тег `<input>` создаем кнопочку. Только тип этого тега будет уже "submit" - это говорит о том, что эта кнопка будет отправлять данные в обработчик.

**Пример:**

```
<input type="submit" name="button" value="Отправить">
<p></p>
```

Слово, прописанное в значении атрибута `value`, будет видно на нашей кнопке отправления.

Вот так выглядит весь html-код нашей формы:

```
<title>страница изучения форм</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<meta name="Keywords" content="формы">
<meta name="Description" content="создание форм в html">
<h1 align="center">Формы</h1>
<table border="1" width="800" bgcolor="white" align="center" bordercolor="black" cellpadding="10">
<tbody><tr>
<td>
<form action="obrabotka.php" method="POST" name="form1">
<p>Введите свое ФИО<br>
<input type="text" name="fio">
</p>
<p>Введите пароль<br>
<input type="password" name="pass">
</p>
<p>Введите свой email<br>
<input type="text" name="email">
</p>
<p>Выберите формат диска<br>
<input type="radio" name="format" value="cd">CD<br>
<input type="radio" name="format" value="dvd">DVD<br>
</p>
<p>Какие курсы должны быть на диске<br>
<input type="checkbox" name="site">Курс по созданию сайтов<br>
```

```

<input type="checkbox" name="book">Курс по созданию книг<br>
</p>
<p>Выберите тип доставки<br>
<select name="delivery">
<option>Срочная</option>
<option>Обычная</option>
</select>
</p>
<p>Введите адрес доставки<br>
<textarea name="address" cols="40" rows="5"></textarea>
</p>
<p>
<input type="submit" name="button" value="Отправить">
</p>
</form>
</td>
</tr>
</tbody></table>

```

Можно проверить в браузере. Кнопка "Отправить" пока не работает, так как обработчика у нас нет. И еще один момент - как сделать чтобы при загрузке формы в переключателях был выбран какой-то пункт.

Для этого прописываем этому элементу ключевое слово checked - что означает - отмеченный.

**Пример:**

```

<p>Выберите формат диска<br>
<input type="radio" name="format" value="cd">CD<br>
<input type="radio" name="format" value="dvd" checked>DVD<br>

```

#### 4. Практическая часть

4.1. Выберите из разделов 1 - 3 по одному примеру и разместите в соответствии с требованиями к построению и структуре html-документа:

- Главная страница должна иметь форму одного из типов обработки (из примеров раздела 1)
- Разместите в главной странице форму для ввода какого-либо типа (согласно раздела 2)
- Добавьте список по вашему выбору, используя примеры раздела 3

4.2. Если требуется создание дополнительных html-документов (медиа-объектов) для обеспечения полноценной и безошибочной загрузки страницы, создайте их и оформите соответствующие ссылки (теги загрузки, атрибуты размещения и выравнивания)

#### Контрольные вопросы

1. Что такое «форма» и каково ее назначение в HTML?
2. Каким тегом в HTML задаются формы?
3. Что означает атрибут METHOD, каковы применяемые к нему значения?.
4. Что означает понятие «поле ввода»?
5. Каким образом можно изменить тип поля и его имя?
6. Что означает понятие «текстовая область»? Каким тегом задается?
7. Каким образом можно создавать (задавать) кнопки в HTML?