

Конспект «Формы. Знакомство».

Раздел 1

Чтобы создать форму, нужно использовать парный тег `<form>`, внутри которого размещаются поля формы. У тега `<form>` есть два важных атрибута:

- `action` задаёт `URL`, адрес для отправки формы;
- `method` задаёт метод отправки формы.

Пример:

```
<form action="https://echo.htmlacademy.ru/courses" method="get">
```

поля формы

```
</form>
```

Для отправки формы обычно используют методы `get` или `post`. Если не указать атрибут `method`, то будет использован `get`.

Метод `get` посылает данные формы в строке запроса, то есть они видны в адресной строке браузера и следуют после знака вопроса.

Метод `post` посылает данные в теле HTTP-запроса и используется, когда нужно отправить много данных и ссылка на результат обработки этих данных не нужна.

Текстовое поле ввода

Большинство полей форм создаётся с помощью одиночного тега `<input>`. У этого тега два обязательных атрибута:

- `type` задаёт тип поля;
- `name` задаёт имя поля.

От типа поля зависит то, как оно будет отображаться и вести себя. Самый распространённый тип — это `text`, который обозначает *текстовое поле*. Данный тип используется по умолчанию.

Для того, чтобы правильно обработать данные на сервере нужно имя поля. Обычно, имя поля должно быть уникальным в пределах формы. Для задания имени поля используют латинские буквы и цифры.

Идентификатор и значение по умолчанию

Атрибут `id` поля ввода обозначает *идентификатор*. Он должен быть уникальным на всей странице.

Обычно идентификаторы используют для повышения удобства работы с формой, например, создают подписи, связанные с мелкими полями. При нажатии на такие подписи активируется связанное поле. Также идентификаторы используют в JavaScript для работы с полями.

Идентификатор не передаётся на сервер. Лучше использовать идентификаторы, отличающиеся от имени поля.

Атрибут `value` задаёт значение поля ввода по умолчанию.

Подпись для поля ввода

Парный тег `<label>` создаёт подпись к полю формы.

Он связывает текст и поле ввода логически. Если нажать на текст в такой подписи, то курсор переместится в соответствующее поле.

Первый способ создать подпись — просто обернуть текст подписи и тег поля в тег `<label>`, вот так:

```
<label>
  Подпись <input type="text" name="username">
</label>
```

При оборачивании текста в тег `<label>` он визуально никак не меняется.

Связь подписи и поля по id

Можно связать подпись с полем с помощью атрибута `id`. Алгоритм такой:

1. Добавляем к полю ввода идентификатор с помощью атрибута `id`;
2. Оборачиваем текст подписи в тег `<label>`;
3. Добавляем тегу `<label>` атрибут `for`;
4. В атрибут `for` записываем такое же значение, что и в атрибуте `id` у поля.

Например:

```
<label for="user-field-id">Имя пользователя</label>
```

...

много-много других тегов

...

```
<input id="user-field-id" type="text" name="username">
```

Поле ввода пароля

Чтобы сделать поле для ввода пароля, в котором текст будет отображаться «звёздочками», нужно изменить значение атрибута `type` на `password`.

Кнопка отправки формы

Кнопка для отправки формы создаётся с помощью тега `<input>` с типом `submit`.

Надпись на кнопке можно задать с помощью атрибута `value`. Для кнопки отправки формы задавать имя необязательно, но если оно задано, то на сервер будут отправляться имя и значение кнопки. Обычно имя для кнопки отправки задают, когда в форме несколько кнопок, отвечающих за разные действия.

Многострочное поле ввода

Многострочное текстовое поле создаётся с помощью парного тега `<textarea>`. У него есть атрибуты `name` и `id`, аналогичные атрибутам текстового поля.

Атрибут `rows` принимает целочисленное значение и задаёт высоту многострочного поля в строках. Атрибут `cols` задаёт ширину поля в символах. В качестве ширины символа берётся некоторая «усреднённая ширина».

Атрибут `value` у многострочного поля отсутствует. Текст, расположенный между открывающим и закрывающим тегом `<textarea>` является значением по умолчанию.

Чекбокс

Чекбокс — это тег `<input>` с типом `checkbox`.

Он работает по принципу «либо да, либо нет». Если галочка стоит, то браузер посылает переменную с именем поля на сервер, если галочки нет, то не посылается ничего. Атрибут `value` не является обязательным.

Чтобы поле было отмечено по умолчанию, нужно добавить к тегу атрибут `checked`.

Имена чекбоксов в рамках одной формы должны быть *разными*, так как чекбокс не подразумевает выбор одного элемента из нескольких.

Конспект «Формы. Знакомство».

Раздел 2

Радиокнопка

Радиокнопка — это тег `<input>` с типом `radio`.

Значение радиокнопки задаётся с помощью `value`. Атрибут `value` является для радиокнопок обязательным. Браузер отправляет на сервер значение `value` выбранной радиокнопки.

Группа радиокнопок

Если добавить ещё несколько `<input>` с таким же именем, но другими значениями `value`, то мы получим группу радиокнопок. Имя поля у радиокнопок одной группы должно быть одинаковым, но идентификаторы всегда должны быть уникальными.

Чтобы сделать какой-либо вариант выбранным по умолчанию, нужно добавить к соответствующему тегу `<input>` атрибут `checked`.

Раскрывающийся список, или «селект»

Раскрывающийся список позволяет выбрать один вариант ответа из нескольких. Такой список создаётся с помощью парного тега `<select>`, у которого есть атрибуты `name` и `id`.

Варианты ответов задаются с помощью парных тегов `<option>`, которые должны располагаться внутри тега `<select>`. Например:

```
<select name="theme">

  <option value="light">Светлая тема</option>

  <option value="dark">Тёмная тема</option>

  ...

</select>
```

В атрибуте `value` тега `<option>` задаётся значение варианта ответа, а внутри этого тега располагается подпись варианта ответа.

Если при отправке формы у выбранного варианта задан `value`, то на сервер отправится значение этого атрибута. В противном случае будет отправлен текст подписи.

«Мультиселект»

Раскрывающийся список можно превратить в так называемый «мультиселект», то есть список, в котором можно выбрать несколько вариантов. Для этого нужно добавить к тегу `<select>` атрибут `multiple`.

Высоту мультиселекта можно изменять с помощью атрибута `size` тега `<select>`.

Чтобы отметить как выбранные по умолчанию одно или несколько значений, нужно к соответствующим тегам `<option>` добавить атрибут `selected`.

При отправке данных мультиселекта на сервер с PHP после имени в значении атрибута `name` ставятся символы квадратных скобок `[]`. Например, `<select name="days[]">`. Это необязательное требование для имени мультиселекта, оно нужно только для корректной обработки данных в PHP.

Поле для загрузки файлов

Поле для загрузки файлов — это тег `<input>` с типом `file`. Для этого поля обязательным атрибутом является имя.

Чтобы поле заработало и браузер смог передать выбранный файл на сервер, необходимо добавить всей *форме* атрибут `enctype` со значением `multipart/form-data`. Этот атрибут указывает браузеру, в каком виде пересылать данные. Если вы хотите отправить на сервер файл, данные из формы эффективнее всего передавать по частям. За это и отвечает значение `multipart/form-data`.

Внешний вид полей с типом `file` очень сильно отличается в зависимости от операционной системы и очень плохо изменяется с помощью стилей.

Скрытое поле

Скрытое поле используют, когда в форме нужно отправить какие-то дополнительные служебные данные, которые не вводятся пользователем.

Скрытое поле — это тег `<input>` с типом `hidden`.