**Курс: Разработка Web-приложений на Python,**

**с применением фреймворка Django**

**Дисциплина: Разработка веб-страниц на языке разметки HTML5 с использованием каскадных таблиц стилей CSS3**

**Тема занятия №8:** **Формы**

**HTML- формы**

Форма HTML используется для сбора пользовательского ввода. Пользовательский ввод чаще всего отправляется на сервер для обработки.

Элемент HTML <form>используется для создания формы HTML для пользовательского ввода.

Элемент <form>представляет собой контейнер для различных типов элементов ввода, таких как: текстовые поля, флажки, переключатели, кнопки отправки и т. д.

Атрибут действия

Атрибут **action** определяет действие, которое должно быть выполнено при отправке формы. Обычно данные формы отправляются в файл на сервере, когда пользователь нажимает кнопку отправки. В приведенном ниже примере данные формы отправляются в файл с именем «action\_page.php». Этот файл содержит серверный скрипт, который обрабатывает данные формы

Атрибут **target** указывает, где отображать ответ, полученный после отправки формы.

Атрибут targetможет иметь одно из следующих значений:

\_blank - Ответ отображается в новом окне или вкладке

\_self - Ответ отображается в текущем окне

\_parent - Ответ отображается в родительском фрейме

\_top - Ответ отображается в полном теле окна

*Framename -* Ответ отображается в именованном iframe

Атрибут **method** указывает метод HTTP, который будет использоваться при отправке данных формы. Данные формы могут быть отправлены как переменные URL (с помощью method="get") или как почтовая транзакция HTTP (с помощью method="post"). Метод HTTP по умолчанию при отправке данных формы — GET.

Атрибут **autocomplete** указывает, должна ли форма включать или выключать автозаполнение. Когда автозаполнение включено, браузер автоматически дополняет значения на основе значений, которые пользователь ввел ранее.

Элемент HTML <input>является наиболее часто используемым элементом формы. Элемент <input>может отображаться разными способами, в зависимости от type атрибута. Вот некоторые примеры:

<input type="button">

<input type="checkbox">

<input type="color">

<input type="date">

<input type="datetime-local">

<input type="email">

<input type="file">

<input type="hidden">

<input type="image">

<input type="month">

<input type="number"> и т.д.

<input type="text"> - Отображает однострочное поле ввода текста

<input type="radio"> - Отображает переключатель (для выбора одного из множества вариантов). Для работы этого элемента формы, ваш HTML - код должен сгруппировать список переключателей вместе. Это достигается с помощью одного и того же значения для атрибута name:

<input type="radio” name= "сheck">

<input type="checkbox"> - Отображает флажок (для выбора нуля или более из множества вариантов). По умолчанию флажок выключен. Вы можете пометить его включенным, просто используя атрибут checked.

<input type="submit"> - Отображает кнопку отправки (для отправки формы)

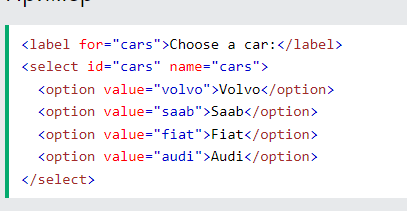
<input type="button"> - Отображает нажимаемую кнопку

*Элемент <label>*

Тег <label>определяет метку для многих элементов формы. Этот <label>элемент полезен для пользователей программ чтения с экрана, поскольку программа чтения с экрана будет читать метку вслух, когда пользователь фокусируется на элементе ввода. Этот <label>элемент также помогает пользователям, которым трудно нажимать на очень маленькие области (например, переключатели или флажки), потому что, когда пользователь щелкает текст внутри элемента <label>, он переключает переключатель/флажок. Атрибут for тега <label>должен быть равен id атрибуту элемента <input> , чтобы связать их вместе.

*Элемент <select>*

Элемент <select>определяет раскрывающийся список:



Элементы <option>определяют параметр, который можно выбрать. Чтобы определить предварительно выбранный параметр, добавьте selected атрибут к параметру:



Используйте multiple атрибут, чтобы позволить пользователю выбрать более одного значения:



Используйте size атрибут, чтобы указать количество видимых значений:

**

*Элемент <datalist>*

Элемент <datalist>определяет список предопределенных опций для <input>элемента.

При вводе данных пользователи увидят раскрывающийся список предопределенных параметров.

Атрибут list элемента <input>должен ссылаться на id атрибут элемента <datalist>.