

# Web-страницы. Язык HTML

HTML-формы

# Форма

Форма создаётся с помощью тега `<form>`, внутри неё могут быть любые необходимые теги, и характеризуется она следующими необязательными параметрами:

- адрес программы на веб-сервере, которая будет обрабатывать содержимое данных формы;
- элементами формы, которые представляют собой стандартные поля для ввода информации пользователем;
- кнопку отправки данных на сервер.

# Пример формы

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Формы</title>
  </head>
  <body>
    <form>
      <p><input name="a">
      <input type="submit"></p>
    </form>
  </body>
</html>
```

# Атрибут action

```
<!DOCTYPE HTML>
<html>
<head>
  <meta charset="utf-8">
  <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>
```

# Указание метода передачи данных

- Для указания в форме метода передачи данных используется атрибут “method”, принимающий значения “GET” или “POST”.

# Передача данных методами GET и POST

## Метод GET

- Сохраняется в строке URL-адреса; адрес с запросом GET можно сохранить в закладках
- Кэшируется браузером
- Данные передаются в открытом виде и могут быть перехвачены

## Метод POST

- «Невидим» для пользователя при отправке
- Передача данных происходит в теле запроса
- Способен передавать большие объёмы данных
- Способен передавать файлы

# Метод GET

`http://yandex.ru/yandsearch?text=cats+photo&lang=ru`

CGI-сценарий

Запрос

В данном примере CGI-сценарию *yandsearch* передаётся строка запроса, внутри которой содержатся переменные “*text*” со значением “*cats photo*” и “*lang*” со значением “*ru*”.

Заголовок запроса начинается так:

```
GET /yandsearch?text=cats+photo&lang=ru HTTP 1.1
```

# Чередование методов GET и POST

POST /passport?mode=auth HTTP/1.1

Host: passport.yandex.ru

User-Agent: Mozilla/5.0 (Windows NT 6.1)

Gecko/20100101 Firefox/23.0

Content-Type: application/x-www-form-urlencoded

Content-Length: 62

from=passport&login=john&passwd=mypass&timestamp  
=1379013756458

# Связывание формы с полями

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <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>
```

# Отправка формы

```
<body>  
  <form>  
    <p><input placeholder="Ваше имя"  
name="user"></p>  
    <p><input type="submit" value="Отправить"  
formaction="handler.php"  
formmethod="post"></p>  
  </form>  
</body>
```

# Однострочное текстовое поле

```
<input type="text" maxlength="25"  
size="40" name="username" value="  
Вася">
```

Параметры:

**disabled** – заблокировать поле для ввода

**maxlength** – максимальное число символов, которое возможно ввести в поле.

**name** – имя поля

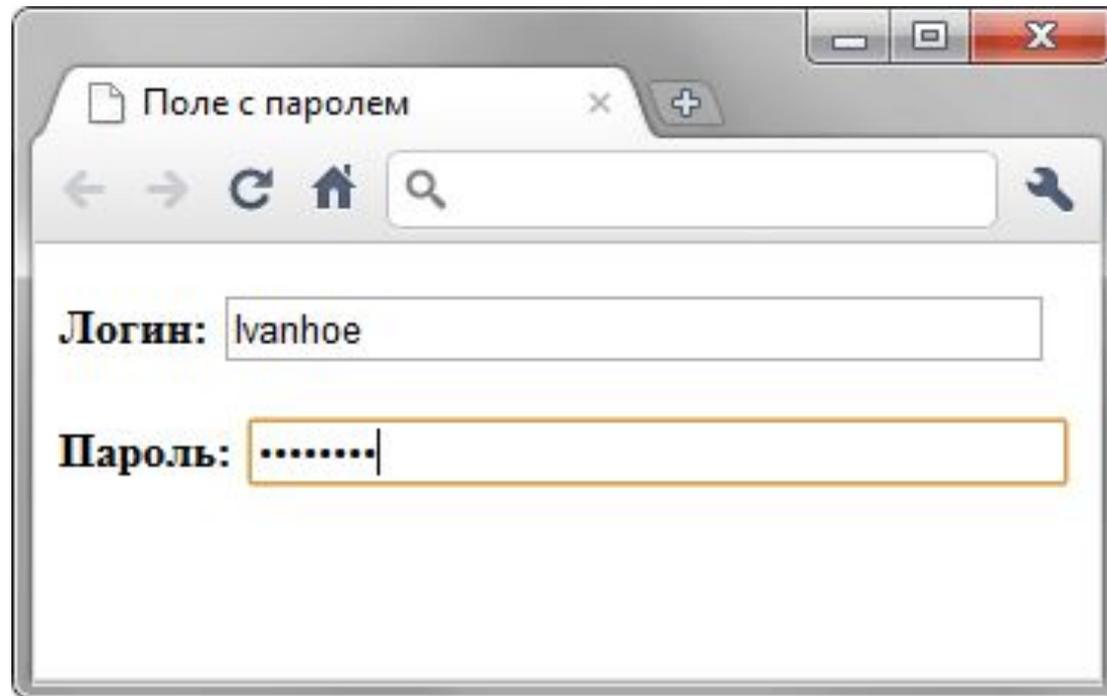
**size** – ширина поля в символах

**value** – начальный текст в поле

# Поле для пароля

`<input type="password" атрибуты>`

Используются те же самые атрибуты, что и у `type="text"`.



# Многострочный текст

`<textarea` атрибуты>

**Текст**

`</textarea>`

Атрибуты :

**cols** – ширина поля в символах

**disabled, maxlength, name** – также, как и в `input`.

**readonly** – поле не может изменяться пользователем

**rows** – высота поля в строках текста

# Поле только для чтения

```
<body>  
<form action="handler.php">  
  <p>Скопируйте приведённый текст и  
вставьте его в  
  поле запроса пароля.</p>  
  <p><textarea name="comment"  
readonly>Мухаха</textarea></p>  
</form>  
</body>
```

# Многострочный текст

Заблокировано

Заблокировано

Только для чтения

Только для чтения

*Internet Explorer*

*Opera*

Заблокировано

Заблокировано

Только для чтения

Только для чтения

*Chrome*

*Firefox*

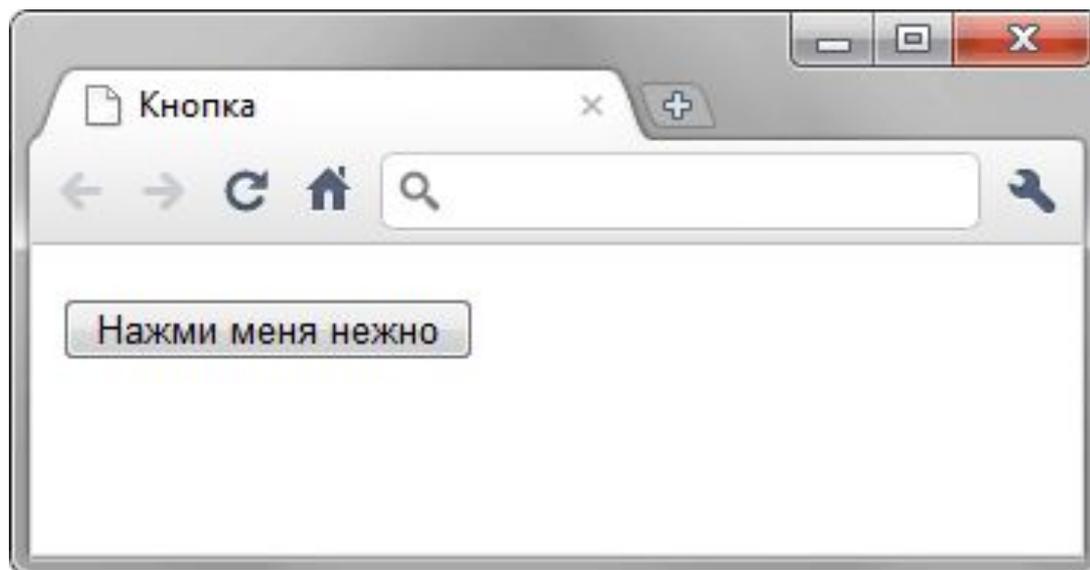
# Кнопки

Существует два способа создания кнопок:

- `<input type="button" атрибуты>`
- `<button атрибуты>Надпись на кнопке</button>`

# Кнопка 1

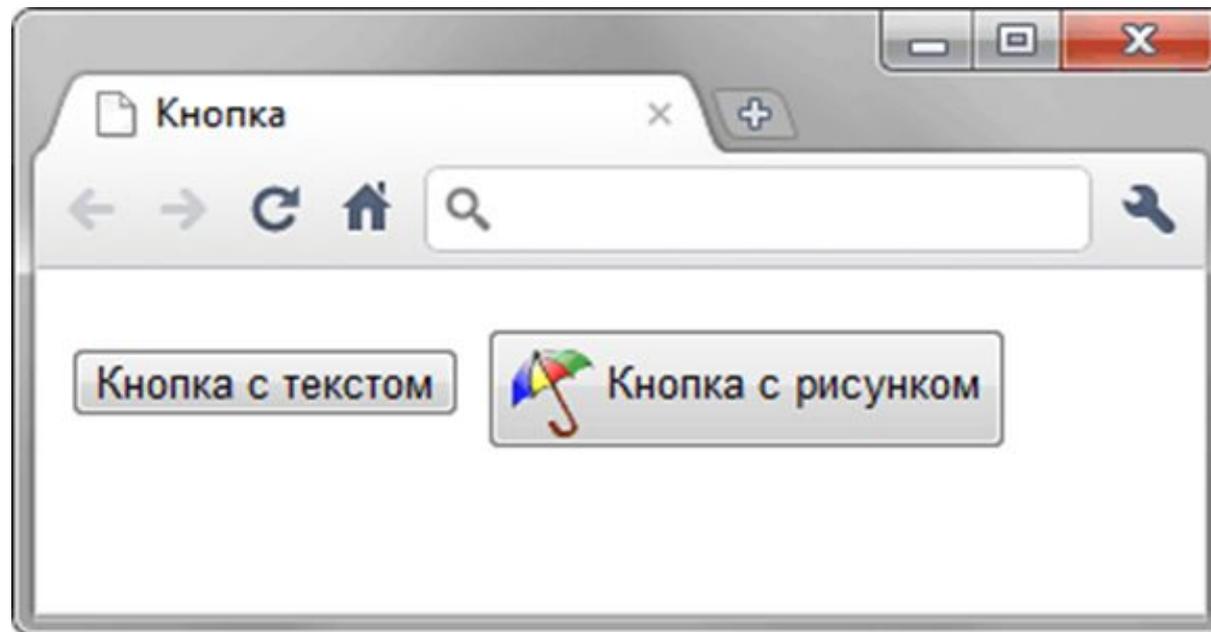
```
<body>  
  <form>  
    <p><input type="button" value=" Нажми меня нежно "></p>  
  </form>  
</body>
```



# Кнопка 2

```
<body>
  <form>
    <p><button>Обычная кнопка</button>
    <button>
      
      Кнопка с рисунком
    </button></p>
  </form>
</body>
```

# Кнопка 2



# Кнопка Submit

- `<input type="submit" атрибуты>`
- `<button type="submit">Надпись  
на кнопке</button>`

Специальная кнопка Submit отправляет данные формы на сервер.

# Отправка данных на сервер

```
<body>  
  <form>  
    <p><input name="login"></p>  
    <p><input type="submit"></p>  
  </form>  
</body>
```

# Кнопка Reset

Синтаксис:

- `<input type="reset" атрибуты>`
- `<button type="reset">Надпись на кнопке</button>`

Кнопка Reset сбрасывает данные формы в первоначальное состояние.

# Кнопка Reset

```
<body>
  <form>
    <p><input value="Введите текст"></p>
    <p><input type="submit" value="Отправить">
      <input type="reset" value="Очистить"></p>
  </form>
</body>
```

# Переключатели

Переключатели (жарг. радиокнопки) используют, когда необходимо выбрать один единственный вариант из нескольких предложенных.

- `<input type="radio" name="имя" атрибуты>`

Атрибуты :

**checked** – переключатель включён изначально

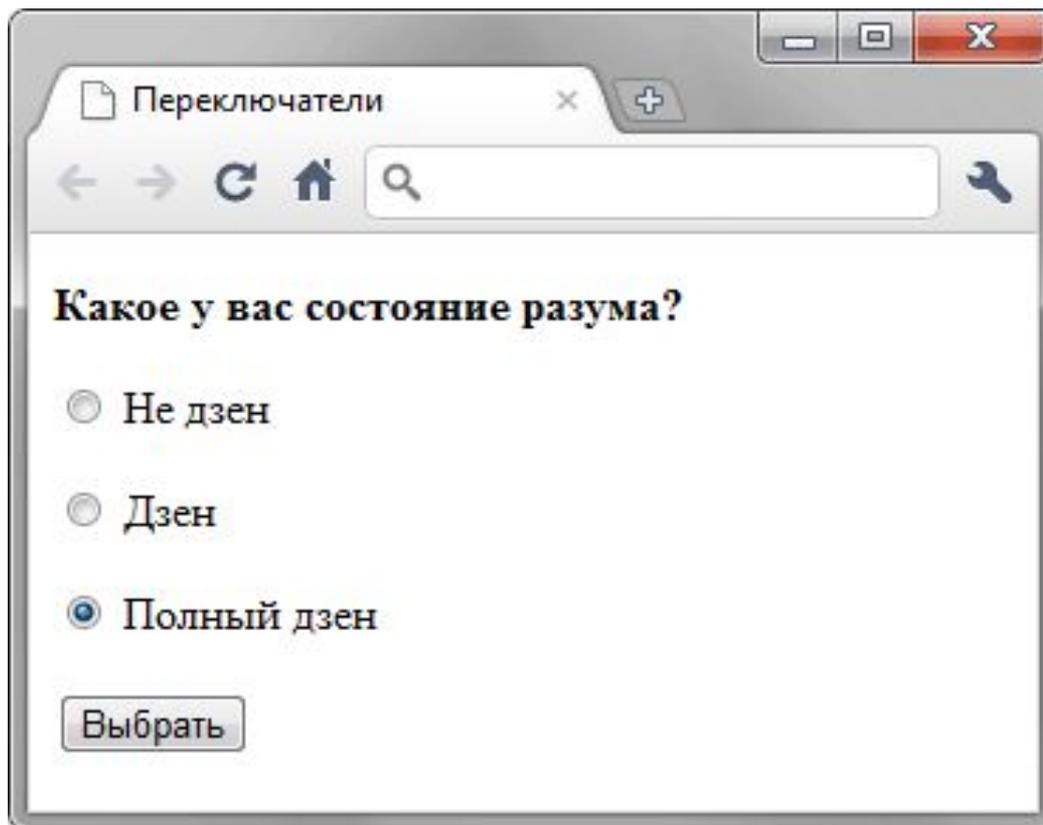
**name** – поскольку переключатель является групповым элементом, имя должно быть у всей группы одинаковое.

**value** – значение, отправляемое на сервер.

# Переключатели

```
<form>
  <p><b>Какое у вас состояние
разума?</b></p>
  <p><input name="dzen" type="radio"
value="nedzen"> Не дзен</p>
  <p><input name="dzen" type="radio"
value="dzen"> Дзен</p>
  <p><input name="dzen" type="radio"
value="pdzen" checked> Полный дзен</p>
  <p><input type="submit" value="
Выбрать "></p>
</form>
```

# Переключатели



Переключатели

← → ↻ 🏠 🔍 🛠

**Какое у вас состояние разума?**

Не дзен

Дзен

Полный дзен

Выбрать

# Флажки

Флажки (жарг. чекбоксы) используют, когда необходимо выбрать любое количество вариантов из предложенного списка.

`<input type="checkbox" атрибуты>`

Атрибуты те же самые, что и у переключателей.

# Флажки

```
<form>
```

```
  <p>В каких годах произошли самые известные извержения вулкана  
Кракатау?</p>
```

```
  <p><input type="checkbox" name="a" value="1417"> 1417</p>
```

```
  <p><input type="checkbox" name="a" value="1680" checked> 1680</p>
```

```
  <p><input type="checkbox" name="a" value="1883" checked> 1883</p>
```

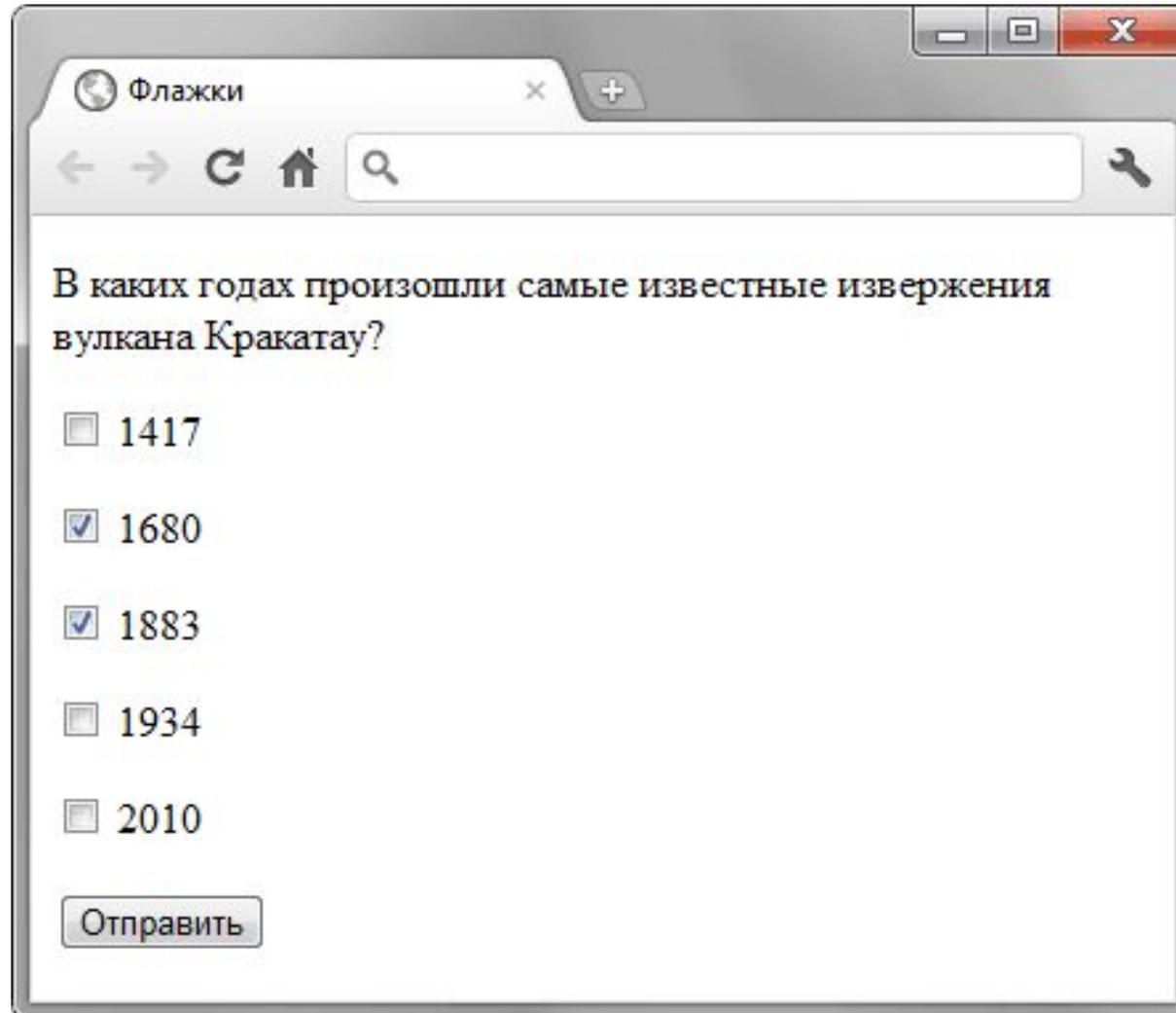
```
  <p><input type="checkbox" name="a" value="1934"> 1934</p>
```

```
  <p><input type="checkbox" name="a" value="2010"> 2010</p>
```

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

```
</form>
```

# Флажки



Флажки

← → ↻ 🏠 🔍

В каких годах произошли самые известные извержения вулкана Кракатуа?

- 1417
- 1680
- 1883
- 1934
- 2010

Отправить

# Поле со списком

Синтаксис:

```
<select атрибуты>
```

```
  <option атрибуты>Пункт 1</option>
```

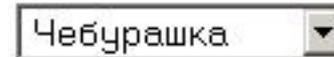
```
  <option атрибуты>Пункт 2</option>
```

```
</select>
```

# Поле со списком



*Список множественного выбора*



*Раскрывающийся список*

# Тег <select>

Тег <select> выступает контейнером для пунктов списка и определяет его вид.

Атрибуты:

**multiple** – будет отображать список со множественным выбором.

**name** – определяет имя списка.

**size** – устанавливает высоту списка.

# Список множественного выбора

```
<form>  
  <p><select name="select" size="3" multiple>  
    <option selected value="s1">Чебурашка</option>  
    <option value="s2">Крокодил Гена</option>  
    <option value="s3">Шапокляк</option>  
    <option value="s4">Крыса Лариса</option>  
  </select>  
  <input type="submit" value="Отправить"></p>  
</form>
```

# Тег <option>

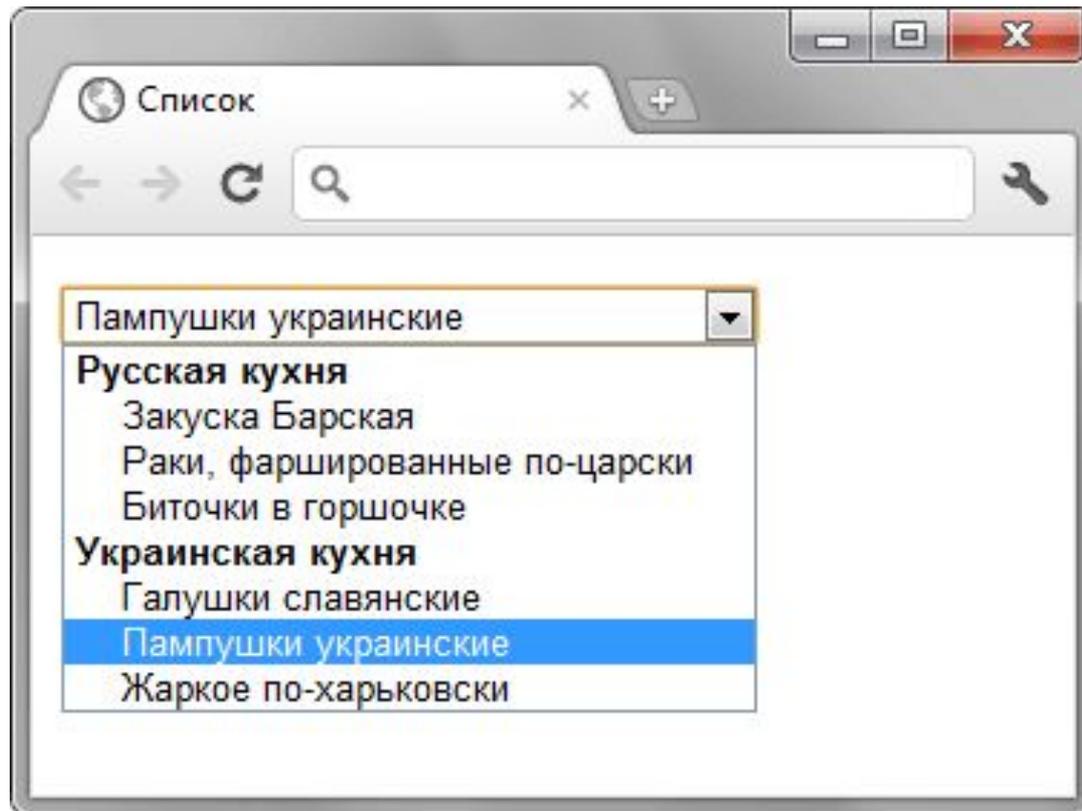
Тег <option> также имеет атрибуты, влияющие на вид списка:

- **selected** – делает текущий пункт списка выделенным
- **value** – определяет значение пункта списка, которое будет отправлено на сервер.

# Группирование элементов списка

```
<form>
  <p><select name="food">
<optgroup label="Русская кухня">
  <option value="r1">Закуска Барская</option>
  <option value="r2">Раки, фаршированные по-царски</option>
  <option value="r3">Биточки в горшочке</option>
</optgroup>
<optgroup label="Украинская кухня">
  <option value="u1">Галушки славянские</option>
  <option value="u2">Пампушки украинские</option>
  <option value="u3">Жаркое по-харьковски</option>
</optgroup>
  </select></p>
  <p><input type="submit" value="Отправить"></p>
</form>
```

# Группирование элементов списка



# Скрытое поле

Предназначено для передачи промежуточных технических данных между страницами. Поле не видимо для пользователя.

- `<input type="hidden" атрибуты>`

Атрибуты:

- **name, value.**

# Скрытое поле

```
<form action="handler.php" method="post">
  <p><b>Напишите любимое слово и нажмите кнопку
Отправить
(никакие данные не будут передаваться на
сервер!) :</b></p>
  <p><input size="25" name="word">
  <input type="hidden" name="name"
value="Vasya">
  <input type="hidden" name="password"
value="pupkin"></p>
  <p><input type="submit" value="Отправить"></p>
</form>
```

# Группирование элементов форм

При создании сложной формы можно визуальнo группировать некоторые её элементы.

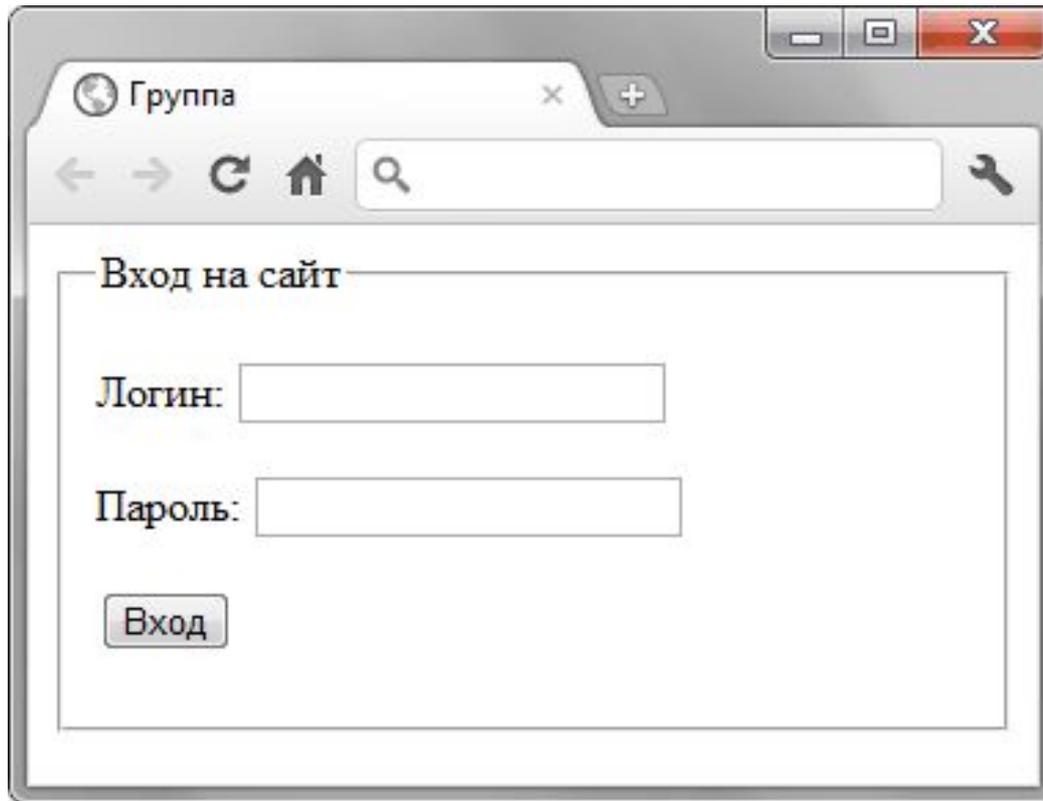
Структура:

```
<fieldset>  
  <legend>Заголовок</legend>  
  ...  
</fieldset>
```

# Группирование элементов форм

```
<form>
  <fieldset>
    <legend>Вход на сайт</legend>
    <p>Логин: <input name="login"></p>
    <p>Пароль: <input type="password"
name="pass"></p>
    <p><input type="submit" value="
Вход"></p>
  </fieldset>
</form>
```

# Группирование элементов форм



Группа

← → ↻ 🏠 🔍 🛠

Вход на сайт

Логин:

Пароль:

The image shows a browser window with a single tab titled "Группа". The address bar contains navigation icons (back, forward, refresh, home, search) and a search icon. The main content area displays a login form titled "Вход на сайт". The form consists of two text input fields: "Логин:" and "Пароль:". Below these fields is a "Вход" button. The entire form is enclosed in a thin grey border, illustrating the concept of grouping elements.

# Лабораторная работа

- Сверстать произвольного вида форму с семантическим использованием всех пройденных элементов форм.