



Казанский федеральный  
УНИВЕРСИТЕТ



Казанский  
федеральный  
университет

ИНСТИТУТ  
вычислительной математики  
и информационных технологий

# Web-программирование

## Лекция 2

# Три правила создания изображений

1

СОХРАНЯЙТЕ  
ИЗОБРАЖЕНИЯ  
В ПРАВИЛЬНОМ  
ФОРМАТЕ.

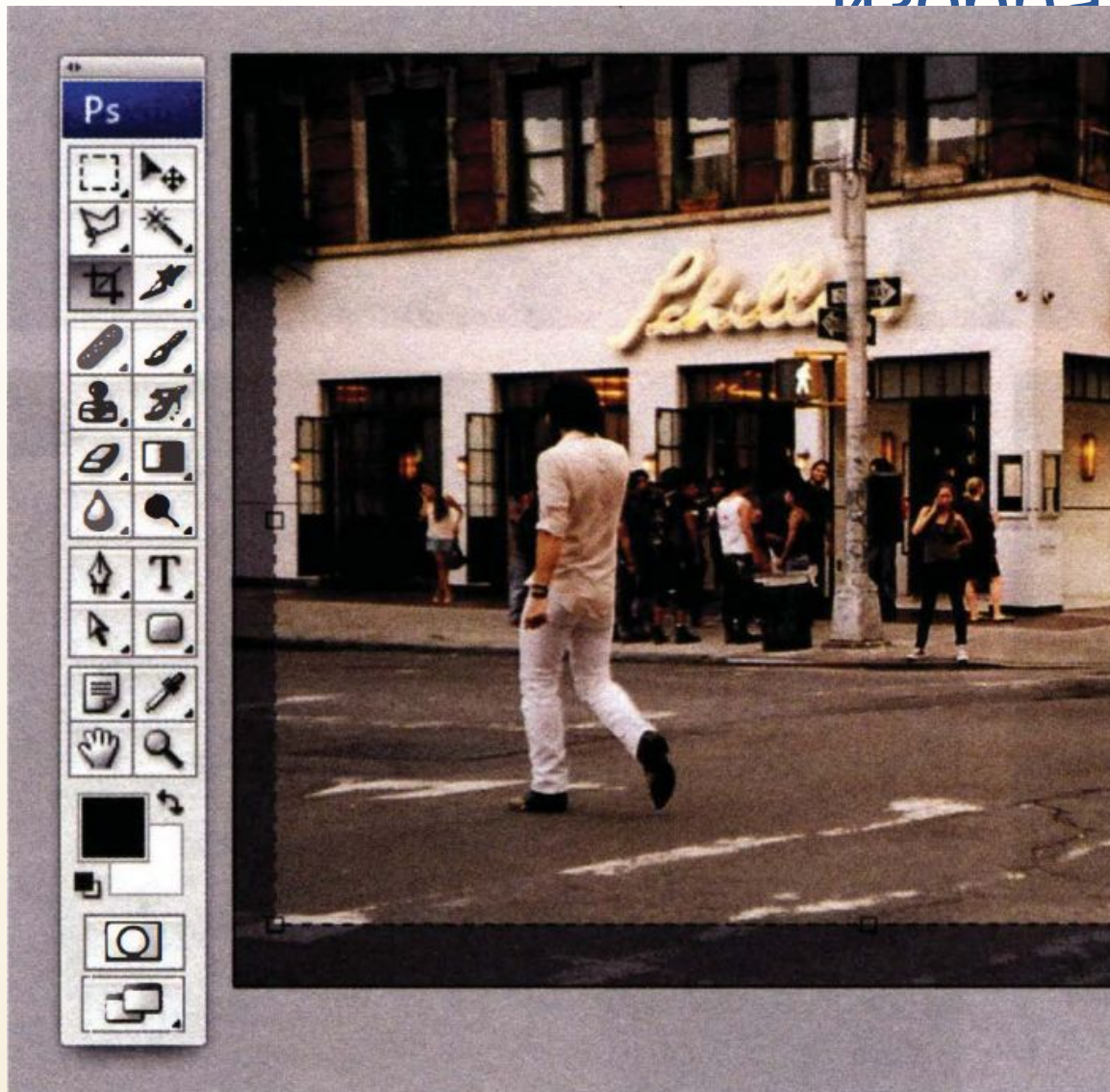
2

СОХРАНЯЙТЕ  
ИЗОБРАЖЕНИЯ  
С ПРАВИЛЬНЫМИ  
ЗНАЧЕНИЯМИ  
ШИРИНЫ  
И ВЫСОТЫ.

3

ИСПОЛЬЗУЙТЕ  
ПРАВИЛЬНОЕ  
РАЗРЕШЕНИЕ.

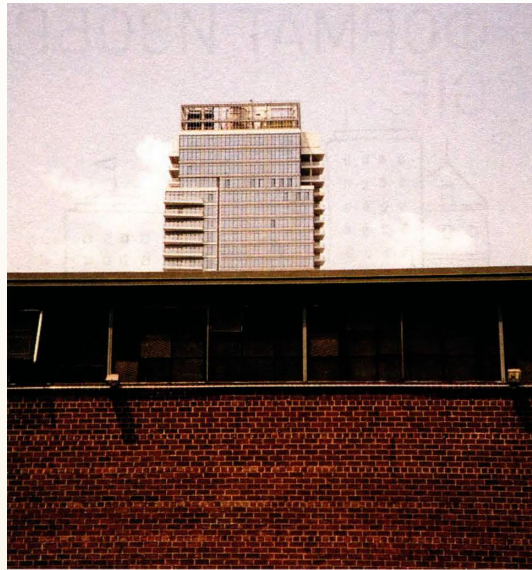
# Инструменты для редактирования и сохранения изображений



ДРУГИЕ РЕДАКТОРЫ:  
Adobe Fireworks  
Pixelmator  
PaintShop Pro  
Paint.net

ОНЛАЙН-РЕДАКТОРЫ:  
[www.photoshop.com](http://www.photoshop.com)  
[www.pixlr.com](http://www.pixlr.com)  
[www.splashup.com](http://www.splashup.com)  
[www.ipiccy.com](http://www.ipiccy.com)

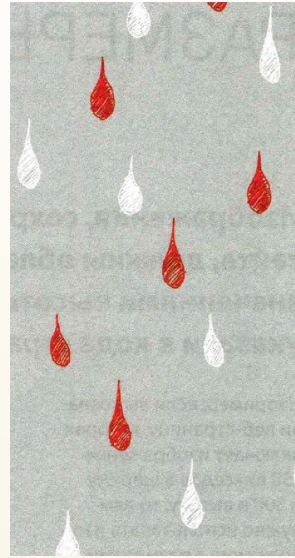
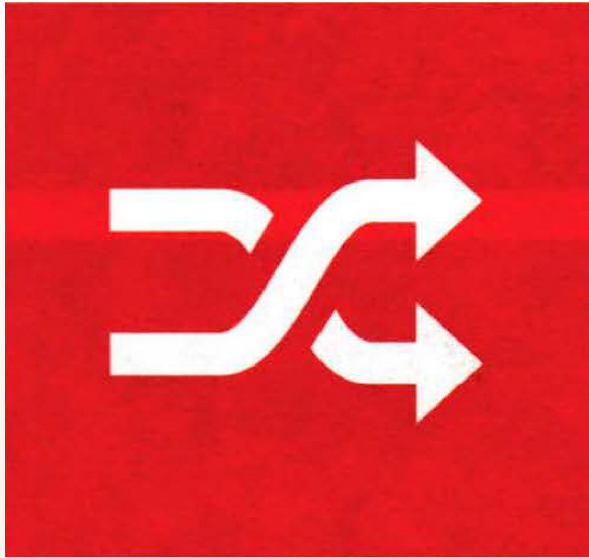
# Формат изображений JPEG



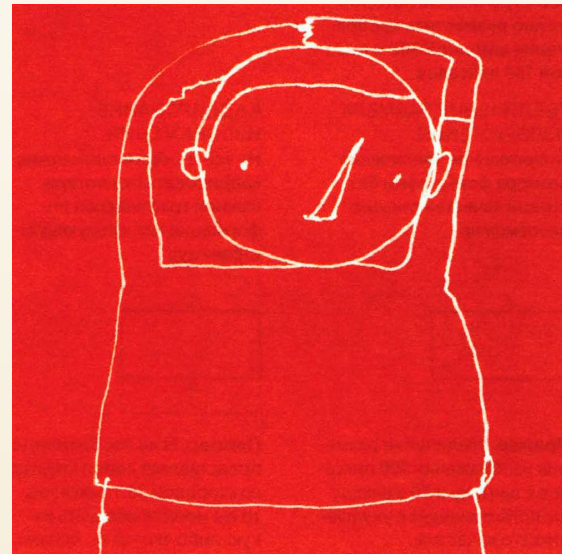
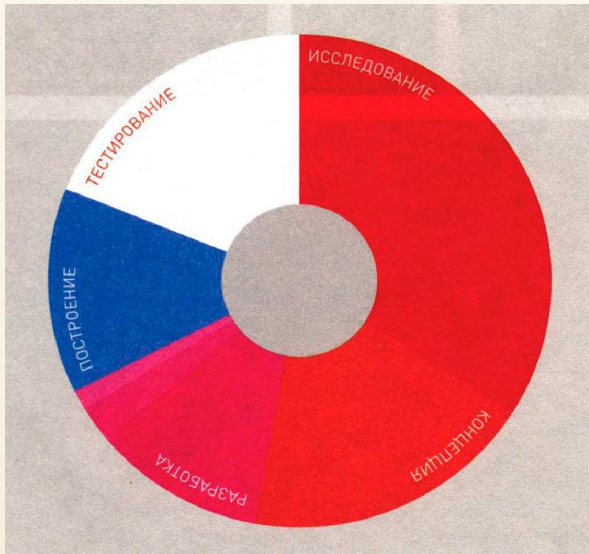
В случае если ваше изображение содержит большое количество оттенков, всегда используйте формат JPEG. Может показаться, что фотографии с изображением снега или пасмурного неба состоят из больших областей однотонного белого или серого цвета, но это не так: обычно фотографии содержат огромное число оттенков, едва различимых невооруженным глазом.



# Формат изображений PNG

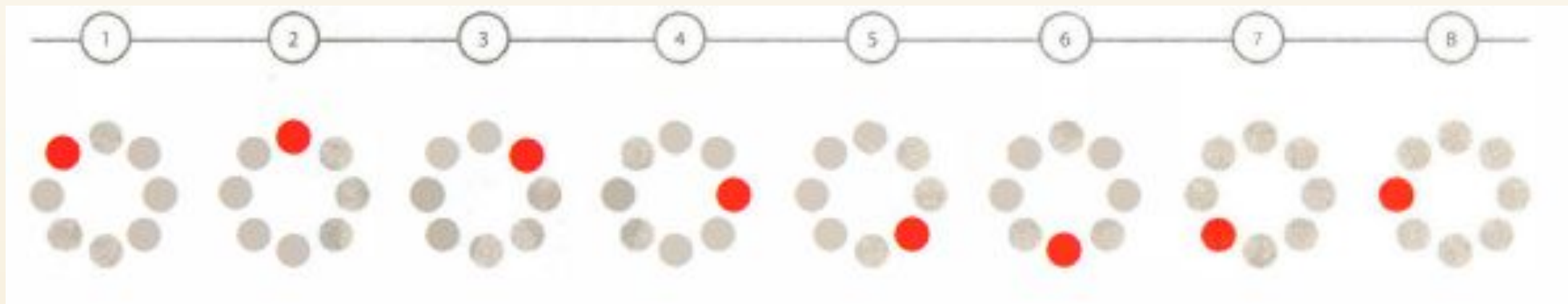


Если на рисунке есть область, окрашенная одним цветом, она называется «однотонной». Логотипы, иллюстрации и схемы чаще всего имеют однотонные области. (Обратите внимание, что фотографии снега, неба или травы не однотонные: они состоят из большого количества оттенков, различия между которыми едва заметны человеческому глазу, и не подходят для сохранения в формате GIF или PNG.)



# Анимированный GIF

**Анимированный GIF-файл позволяет отображать последовательность из нескольких изображений и поэтому может быть использован для создания простой анимации.**



# Векторная графика (SVG)

**Векторные изображения отличаются от растровых и не зависят от разрешения. Создавать векторные изображения можно в таких программах, как Adobe Illustrator.**



# Иллюстрации и подписи

`<figure>`

`<figcaption>`

```
<figure>
  
  <img />
  <br />
  <figcaption>Во время сна выдры держат друг
  друга за лапки, чтобы их не разнесло
  течением.</figcaption>
</figure>
```



Во время сна выдры держат друг друга за лапы, чтобы их не разнесло течением.



# Таблицы

Таблица представляет информацию в виде сетки. Примерами таблиц могут служить финансовые отчеты, телепрограммы и результаты спортивных соревнований.

Сетка помогает воспринимать большие объемы данных путем соотнесения их с подписями по обеим осям.

Отдельный блок сетки называется **ячейкой таблицы**. С помощью языка HTML таблицы пишутся построчно.

The screenshot shows the Reuters website interface for Russia and CIS countries. It features a navigation menu on the left, a main content area with a currency converter and a table of exchange rates, and a market news section on the right.

**REUTERS** РОССИЯ И СТРАНЫ СНГ

ПОСЛЕДНИЕ НОВОСТИ **ВЕНЕСУЭЛА НЕ УСПЕВАЕТ ЗАБАЛЬЗАМИРОВАТЬ ТЕЛО ЧАВЕСА**

Quote

Вы здесь: [На главную страницу](#) > [НОВОСТИ](#) > [ВАЛЮТЫ](#) Четверг, 14 марта 2013 | 23:28 MSK

**Валюты**

ВАЛЮТНЫЙ КАЛЬКУЛЯТОР

1.00  Russia (new) Ruble (RUB)

US Dollar (USD)

КУРСЫ ВАЛЮТ

	RUB	Euro	US \$	UK £	Yen	SFranc
1 RUB	1.0	0.025029	0.032555	0.021583	3.1279	0.030820
1 Euro	39.944	1.0	1.3004	0.88211	124.96	1.2312
1 US \$	30.717	0.76899	1.0	0.66295	96.090	0.94860
1 UK £	48.330	1.1600	1.5083	1.0	144.94	14278
1 Yen	0.31967	0.0080029	0.010407	0.0068983	1.0	0.0098543
1 SFranc	32.450	0.81229	1.0563	0.70028	101.5	1.0

ПОСЛЕДНИЕ НОВОСТИ РЫНКОВ

Russia  US  UK  Europe

RTS 1,539.12 +3.39

MICEX 1,502.52 -3.55

Get a Share Quote

910 11234567890 22

1545  
1540  
1535  
1530  
1525

Все котировочные данные с задержкой 15 минут

# Базовая структура таблицы

`<table>`

`<tr>`

`<td>`

```
<table>
  <tr>
    <td>15</td>
    <td>15</td>
    <td>30</td>
  </tr>
  <tr>
    <td>45</td>
    <td>60</td>
    <td>45</td>
  </tr>
  <tr>
    <td>60</td>
    <td>90</td>
    <td>90</td>
  </tr>
</table>
```

15	15	30
45	60	45
60	90	90

# Заголовки таблиц



```
<table>
  <tr>
    <th></th>
    <th scope="col">Суббота</th>
    <th scope="col">Воскресенье</th>
  </tr>
  <tr>
    <th scope="row">Продано билетов</th>
    <td>120</td>
    <td>135</td>
  </tr>
  <tr>
    <th scope="row">Выручка</th>
    <td>18 000 руб.</td>
    <td>20 250 руб.</td>
  </tr>
</table>
```

	Суббота	Воскресенье
Продано билетов	120	135
Выручка	18000 руб.	20250 руб.

# Объединение столбцов

```
<table>
  <tr>
    <th></th>
    <th>9:00</th>
    <th>10:00</th>
    <th>11:00</th>
    <th>12:00</th>
  </tr>
  <tr>
    <th>Понедельник</th>
    <td colspan="2">География</td>
    <td>Музыка</td>
    <td>ИЗО</td>
  </tr>
  <tr>
    <th>Вторник</th>
    <td colspan="3">Физкультура</td>
    <td>География</td>
  </tr>
</table>
```

	9am	10am	11am	12am
Понедельник	География		Музыка	ИЗО
Вторник	Физкультура			География

# Объединение строк

```
<table>
  <tr>
    <th></th>
    <th>TV1000</th>
    <th>ТНТ</th>
    <th>СТС</th>
  </tr>
  <tr>
    <th>20:00-21:00</th>
    <td rowspan="2">Фильм</td>
    <td>Ток-шоу</td>
    <td>Юмор</td>
  </tr>
  <tr>
    <th>21:00-22:00</th>
    <td>Ток-шоу</td>
    <td>Мультфильм</td>
  </tr>
</table>
```

	TV1000	ТНТ	СТС
20:00-21:00	Фильм	Ток-шоу	Юмор
21:00-22:00		Ток-шоу	Мультфильм

# Длинные таблицы

```
<table>
  <thead>
    <tr>
      <th>Дата</th>
      <th>Прибыль</th>
      <th>Расход</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>1-е января</th>
      <th>250</th>
      <th>36</th>
    </tr>
    <tr>
      <th>2-е января</th>
      <th>285</th>
      <th>48</th>
    </tr>
    <!-- прочие строки как предыдущие две -->
    <tr>
      <th>31-е января</th>
      <th>129</th>
      <th>64</th>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td></td>
      <th>7824</th>
      <th>1241</th>
    </tr>
  </tfoot>
</table>
```

Дата	Прибыль	Расход
1-е января	250	36
2-е января	285	48
3-е января	260	42
4-е января	290	38
5-е января	310	115
6-е января	168	14
7-е января	226	20
8-е января	253	37
9-е января	294	33
10-е января	216	46
11-е января	244	29
12-е января	297	32
13-е января	328	86
14-е января	215	38
15-е января	254	30
16-е января	256	27
17-е января	311	68
18-е января	212	39
19-е января	234	36
20-е января	221	43
21-е января	259	38
22-е января	246	31
23-е января	248	17
24-е января	229	45
25-е января	263	34
26-е января	258	41
27-е января	283	22
28-е января	256	30
29-е января	278	47
30-е января	251	15
31-е января	129	64
	7824	1241

<thead>

<tbody>

<tfoot>

# Формы

**Наверное, самая известная веб-форма — это поле ввода поискового запроса, находящаяся в центре главной страницы сайта поисковой системы Google.**



# Элементы формы

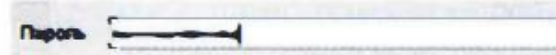
## ВВОД ТЕКСТА

Поле ввода текста (одна строка)  
Позволяет ввести строку текста, например адрес электронной почты или имя пользователя.



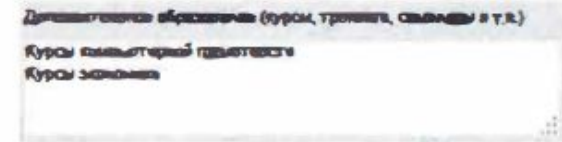
Предлагаемая вакансия \*  
Генеральный директор

Поле ввода пароля  
Оно аналогично полю ввода текста с той лишь разницей, что поле ввода пароля скрывает вводимые символы от посторонних глаз.



Пароль

Текстовая область (многострочное поле ввода текста)  
Данная разновидность поля используется для отправки более длинного текста, например сообщения или комментария.



Дополнительные комментарии (курсы, тренинги, семинары и т.д.)  
Курсы компьютерной грамотности  
Курсы экономики

## ВЫБОР ВАРИАНТОВ

Переключатели  
Используются, когда пользователь должен выбрать только один вариант из нескольких представленных.



НАИБОЛЕЕ РАБОТЫ С КОМПЬЮТЕРОМ  
Укажите уровень компьютерных навыков \*  
 Начальный уровень  
 Промышленный компьютерщик  
 Специалист по компьютерным технологиям

Флажки  
Используется, когда пользователь может выбрать несколько вариантов из предложенных либо не выбрать ни одного.



Историческая смена работы  
 Нет  
 IT и телекоммуникации  
 Торговля  
 Здравоохранение, спорт, туризм  
 Промышленность, инженерия

Раскрывающиеся списки  
Используются, когда пользователь должен выбрать один вариант из списка



Дата рождения \*  
15 03 2013



# Элементы формы

## ОТПРАВКА ФОРМ НА СЕРВЕР

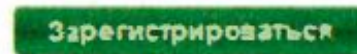
### Кнопка подтверждения

Кнопка подтверждения используется для отправки введенных данных формы на другую веб-страницу.



### Графические кнопки

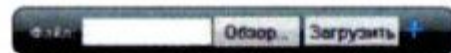
Они аналогичны обычным кнопкам подтверждения, но вместо стандартного изображения можно использовать собственное.



## ЗАГРУЗКА ФАЙЛОВ НА САЙТ

### Загрузка файлов

Данный элемент формы позволяет пользователям загружать на сайт файлы, например изображения.



Посетитель сайта заполняет поля формы, после чего щелкает мышью по кнопке подтверждения для отправки введенных данных на сервер.

1

ПРОГОЛОСУЙТЕ ЗА ЛЮБИМОГО ДЖАЗОВОГО МУЗЫКАНТА

Имя:

Я голосую за:

- Ella Fitzgerald
- Herbie Hancock
- John Coltrane
- Miles Davis
- Thelonius Monk

2

На сервер отправляется имя каждого элемента формы и значение, введенное или выбранное посетителем

3

Сервер обрабатывает полученную информацию с помощью языков программирования, таких как PHP, C#, VB.net или Java. Кроме того, он может сохранить информацию в базе данных

4

Спасибо, Анна!

Вы проголосовали за Herbie Hancock.

Основываясь на полученной информации, сервер создает новую веб-страницу и возвращает ее браузеру

`username=Анна`

`vote=Herbie`

Если элемент формы позволяет посетителю ввести какой-либо текст, то он и будет являться его значением.

Если элемент формы позволяет выбрать один пункт из представленного списка ответов (например, с помощью группы переключателей, флажков или раскрывающегося списка), то веб-дизайнеру необходимо ввести код, автоматически присваивающий значение каждому пункту.

# <form>

## Структура формы

### action

Каждый элемент `<form>` требует указания атрибута `action`, значением которого является URL-адрес страницы-получателя введенных данных при отправке формы на сервер.

### method

Формы можно отправить одним из двух методов: `get` или `post`.

При использовании метода `get` введенные данные будут добавлены в конец URL-адреса, указанного в атрибуте `action`. Метод `get` идеален:

- для коротких форм (таких как поля для ввода поискового запроса);
- при простом получении данных с веб-сервера (без отправки сведений, добавляемых в базу данных).

При использовании метода `post` данные отправляются на сервер с помощью так называемых HTML-заголовков. Хорошей практикой будет использование метода `post` в случае, если ваша форма:

- позволяет пользователям загружать файлы на сайт;
- очень длинная;
- содержит конфиденциальные данные (например, пароли);
- добавляет сведения в базу данных или удаляет их оттуда.

Если атрибут `method` не указан, то форма будет отправлена на сервер с помощью метода `get`.

```
<form action="http://www.primer.ru/subscribe.php"
  method="get">
  <p>Здесь будут отображены элементы формы.</p>
</form>
```

**Здесь будут отображены элементы формы.**

# Ввод текста

`<input>`

`type="text"`

`name`

`maxlength`

`size`

```
<form action="http://www.primer.ru/login.php">
  <p>Имя пользователя:
    <input type="text" name="username" size="15"
      maxlength="30" />
  </p>
</form>
```

Имя пользователя:

# Ввод пароля

`<input>`

**`type="password"`**

Когда атрибуту `type` присваивается значение `password*`, на странице создается однострочное поле ввода текста, которое функционирует как обычное поле ввода текста с той лишь разницей, что вводимые в него символы маскируются. Это сделано, чтобы не допустить утечки таких важных данных, как пароль, в случае, если кто-то подглядывает из-за спины посетителя сайта.

```
<form action="http://www.primer.ru/login.php">
  <p>Имя пользователя:
    <input type="text" name="username" size="15"
      maxlength="30" />
  </p>
  <p>Пароль :
    <input type="password" name="password" size="15"
      maxlength="30" />
  </p>
</form>
```

Имя пользователя:

Пароль:

# Текстовая область

**<textarea>**

```
<form action="http://www.primer.ru/profile.php">  
  <p>Что вы думаете об этом выступлении?</p>  
  <textarea name="comments" cols="20"  
    rows="4">Введите свой комментарий...</textarea>  
</form>
```

Атрибут `cols` устанавливает ширину текстовой области (измеряется в количестве символов в одной строке). Атрибут `rows` задает высоту поля в строках.

Что вы думаете об этом выступлении?

Введите свой  
комментарий. . .

# Переключатели

## <input>

### type="radio"

Переключатели позволяют посетителям сайта выбрать только один вариант из предложенных.

### value

Атрибут `value` устанавливает значение выбранного переключателя, отправляемое серверу. Значение каждого переключателя должно быть уникальным внутри группы (чтобы сервер знал, какой вариант ответа выбрал посетитель).

### checked

Атрибут `checked` указывает, какой из предлагаемых вариантов должен быть выбран по умолчанию при загрузке страницы (если это необходимо). Атрибут принимает единственное значение — `checked*`. Данный атрибут должен быть установлен только у одного переключателя из группы.

```
<form action="http://www.primer.ru/profile.php">
  <p>Пожалуйста, выберите любимый жанр музыки:
  <br />
  <input type="radio" name="genre" value="rock"
checked="checked" /> Рок
  <input type="radio" name="genre" value="pop" />
Поп
  <input type="radio" name="genre" value="techno"
/> Электроника
</p>
</form>
```

Пожалуйста, выберите любимый жанр музыки:  
☉ Рок ○ Поп ○ Электроника

# Флажки

## <input>

### type="checkbox"

С помощью флажков посетители сайта могут выбирать из нескольких вариантов ответа, а также отменять выбор.

```
<form action="http://www.example.com/profile.php">
  <p>Пожалуйста, выберите, где вы слушаете музыку:
  <br />
  <input type="checkbox" name="service"
  value="house" checked="checked" /> Дома
  <input type="checkbox" name="service"
  value="work" /> На работе
  <input type="checkbox" name="service"
  value="inroad" /> В дороге
  </p>
</form>
```

Пожалуйста, выберите где вы слушаете музыку:

Дома  На работе  В дороге



# Раскрывающийся список

## <select>

## <option>

Элемент `<option>` используется для указания вариантов ответа, которые может выбрать посетитель сайта. Текст, помещенный между тегами `<option>` и `</option>`, будет выведен на экран как пункт раскрывающегося списка.

## value

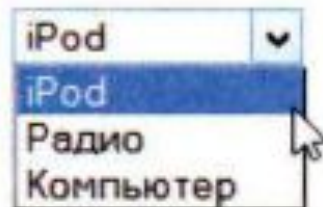
Атрибут `value` используется в элементе `<option>` для установки значения, отправляемого на сервер вместе с именем выбранного элемента.

## selected

Атрибут `selected` устанавливает вариант ответа, который должен быть выбран по умолчанию во время загрузки страницы. Данный атрибут принимает единственное значение — `selected`.

```
<form action="http://www.example.com/profile.php">
  <p> На каком устройстве вы слушаете музыку чаще всего?</p>
  <select name="ustroistva">
    <option value="ipod">iPod</option>
    <option value="radio">Радио</option>
    <option value="computer">Компьютер</option>
  </select>
</form>
```

На каком устройстве вы слушаете музыку чаще всего?



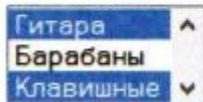
# Список множественного выбора

## <select>

### multiple

Добавив атрибут `multiple` со значением `multiple`, вы позволите посетителям вашего сайта выбирать сразу несколько вариантов ответа (не забудьте написать предупреждение об этой возможности).

Играете ли вы на одном из инструментов, перечисленных ниже? (Вы можете выбрать сразу несколько вариантов ответа, нажав и удерживая клавишу `Ctrl` (Windows) или `Cmd` (OS X)).



Гитара  
Барабаны  
Клавишные

```
<form action="http://www.primer.ru/profile.php">
  <p> Играете ли вы на одном из инструментов,
    перечисленных ниже? (Вы можете выбрать сразу
    несколько вариантов ответа, нажав и удерживая
    клавишу <b>Ctrl</b> (Windows) или <b>Cmd</b>
    (OS X)).</p>
  <select name="instrumenty" size="3"
    multiple="multiple">
    <option value="gitara"
      selected="selected">Гитара</option>
    <option value="barabany">Барабаны</option>
    <option value="klavishnye"
      selected="selected">Клавишные</option>
    <option value="bas">Бас-гитара</option>
  </select>
</form>
```

# Загрузка файлов на сервер

## <input>

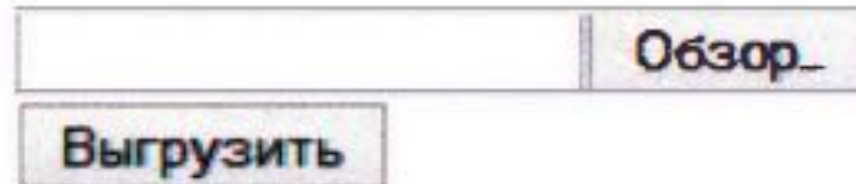
Если вы хотите позволить посетителям размещать файлы на вашем сайте, например изображения, видеоролики, MP3-записи или документы в формате PDF, вам понадобится специальное поле для загрузки файлов.

## type="file"

Данное значение атрибута `type` создает поле, по внешнему виду напоминающее поле ввода текста, после которого помещается кнопка **Обзор** (Browse). Когда посетитель щелкает по ней мышью, браузер открывает диалоговое окно, позволяющее выбрать нужный файл на компьютере.

```
<form action="http://www.primer.ru/zagruzka.php"
  method="post">
  <p>Выгрузите свою песню в формате MP3:</p>
  <input type="file" name="pesnya" /><br />
  <input type="submit" value="Загрузить" />
</form>
```

**Выгрузите свою песню в формате MP3:**



The screenshot shows a web form with the following elements:

- A text input field for the file name.
- A button labeled "Обзор\_" (Browse) next to the input field.
- A button labeled "Выгрузить" (Upload) below the input field.

# Кнопки подтверждения

## <input>

### type="submit"

Кнопка подтверждения используется для отправки формы с данными на сервер.

### name

Данному элементу формы может быть присвоен атрибут name, но его использование необязательно.

### value

Атрибут value используется для указания текста надписи на кнопке. Устанавливать текст надписи самостоятельно нужно потому, что по умолчанию в некоторых браузерах на кнопке пишется **Отправить** (Submit), что подходит не для всех форм.

```
<form action="http://www.primer.ru/podpiska.php">
  <p>Подпишитесь на рассылку наших новостей:</p>
  <input type="text" name="email" />
  <input type="submit" name="podpisatsya"
    value="Подписаться" />
</form>
```

**Подпишитесь на рассылку наших новостей:**

# Графическая кнопка

```
<form action="http://www.primer.ru/podpiska.php">  
  <p>Подпишитесь на рассылку наших новостей:</p>  
  <input type="text" name="email" />  
  <input type="image" src="images/subscribe.jpg"  
    width="100" height="20" />  
</form>
```

`<input>`  
`type="image"`

Подпишитесь на рассылку наших новостей:

Подписаться

# Кнопки и скрытые элементы формы

## <button>

Элемент `<button>` был добавлен в язык HTML, чтобы позволить разработчикам сайтов получить более полный контроль над внешним видом кнопок, а также сделать возможным появление внутри кнопки других элементов. Это значит, что теперь вы можете сочетать на кнопке текст и изображение, располагая их между открывающим `<button>` и закрывающим `</button>` тегами.

## <input>

### `type="hidden"`

В этом примере мы также продемонстрируем вам элемент формы `hidden*`. Скрытые элементы не отображаются, хотя вы и можете увидеть их при просмотре исходного кода страницы. Они позволяют веб-дизайнерам размещать в форме данные, невидимые посетителям. Например разработчик может сохранить в скрытом поле информацию о том, с какой страницы посетитель отправил форму на сервер.

```
<form action="http://www.example.com/add.php">
  <button>Добавить</button>
  <input type="hidden" name="zakladka"
    value="tekst_pesni" />
</form>
```

