

# Василевский Игорь

## Сам себе веб-мастер: шпаргалка для фрилансера

### Содержание

Василевский Игорь	
Сам себе веб-мастер: шпаргалка для фрилансера.....	1
Введение.....	2
Определение целей проекта.....	2
Выбор и регистрация доменного имени.....	2
Инструменты для работы.....	3
Устройство веб-сайта.....	5
Drupal.....	5
Базы данных.....	6
Хостинг.....	6
Настройка домена и почты.....	7
Установка Drupal на сервер.....	7
Начальная настройка Drupal.....	8
Расширение функциональности Drupal.....	11
Публикация нового контента.....	11
Язык HTML.....	12
Использование мультимедиа.....	19
Google для веб-мастера.....	22
Обновление Drupal.....	23

## Введение

Свой первый сайт я создал в 2000-м году с помощью редактора Microsoft FrontPage. Сайт был посвящен компании, в которой я тогда работал, и размещался на бесплатном хостинге. Слава богу, даже на [www.archive.org](http://www.archive.org) этот сайт уже не найти. Я не стал крутым веб-дизайнером и даже простым веб-дизайнером тоже не стал. Мои профессиональные интересы лежат в другой области, что не мешает мне совершенно самостоятельно и достаточно свободно развивать свой личный сайт «Искусство БОЛЬШИХ продаж». Я решил составить краткое пособие по запуску своего сайта во всемирной сети интернет, надеюсь, кому-то из братьев и сестер-фрилансеров это будет полезно. Не ищите глубокого погружения в тему веб-разработки и веб-продвижения, моя цель помочь сделать первые шаги в Сети. Почему я назвал данный текст шпаргалкой? Потому что писать я буду кратко, как пишутся шпаргалки перед экзаменом. Дополнительную информацию можно получить из источников, ссылки на которые я обязательно дам.

Время от времени я буду вносить в текст изменения. Последнюю версию вы всегда сможете [загрузить с моего сайта](#).

## Определение целей проекта

Прежде чем приступать к освоению просторов интернет, надо определиться со своими целями.

Вопрос первый: **Зачем вам сайт?** Возможные ответы:

- Заработать.
- Прославиться.
- Систематизировать и сохранить полезную информацию.
- Научиться навыкам веб-разработки и интернет-маркетинга.

Конечно, можно попытаться совместить несколько задач, главное — осознавать, чего именно вы хотите достичь с помощью своего проекта.

Вопрос второй: **Кого вы хотите видеть в числе посетителей своего сайта?**

Вопрос третий: **Что вы предложите этим людям?** С помощью сайта вы можете:

- Развлекать,
- Информировать,
- Обучать,
- Связывать пользователей между собой,
- Предоставлять какой-либо сервис,
- Продавать товары и услуги.

От ответов на эти вопросы зависит все остальное.

## Выбор и регистрация доменного имени

Определившись с целями проекта и целевой аудиторией можно переходить к регистрации доменного имени. Я бы сказал, что с этим вопросом стоит поторопиться, поскольку, хорошие имена — дефицитный товар. Какой домен можно назвать хорошим? Вот несколько критериев, по которым можно оценить и выбрать домен:

- Краткость,
- Запоминаемость,
- Легкость написания,
- Однозначность восприятия на слух,
- Использование в имени ключевых слов.

Выбор доменной зоны - .com, .net, .org, .ru или любой другой определяется следующим:

- Характер ресурса. Домен в зоне .com предназначен для коммерческих организаций; .net — для ресурсов, связанных с коммуникациями; .biz — для коммерческих организаций, .org — общественных.
- Регион и язык целевой аудитории. Для русскоязычных ресурсов оптимальной является зона .ru.
- Легкость регистрации и администрирования домена. Например, в зоне .ua зарегистрировать домены могут только владельцы соответствующих торговых марок, а смена регистратора доменов в зоне .ru сопряжена с определенными сложностями. В свое время, потеряв очень интересный домен, я решил не связываться с доменами в зоне .ru.
- Стоимость регистрации доменов.

Зона .com является универсальным вариантом, но, к сожалению, найти здесь хороший незанятый домен с каждым днем все сложнее.

Возможно, вы сумеете найти вариант, который на 100% соответствует всем критериям, если нет, выбирайте тот, который максимально близок к идеалу.

Вы также можете зарегистрировать несколько доменов, сделав один из них основным, а с другие настроить так, чтобы посетитель автоматически перенаправлялся на ваш сайт.

Важно! Убедитесь, чтобы домен был зарегистрирован на вас, а не на компанию-регистратора! Кроме этого, в число требований к регистратору домена я бы включил легкость управления зарегистрированным доменом, характер поддержки со стороны регистратора и его репутация на рынке. Избегайте сомнительно дешевых предложений. Цена вопроса не соответствует уровню рисков, связанных с неверным выбором регистратора.

Надо понимать, что зарегистрированный домен не переходит в вашу собственность, вы получаете возможность использования домена и приоритетное право на продление, когда подойдет срок окончания «аренды».

Завершая доменную тему, надо сказать, что один из источников поиска подходящего домена — доменные аукционы, например, [GoDaddy](#).

## Инструменты для работы

Веб-строительство, как и любое другое строительство, помимо знаний и навыков требует специальных инструментов под рукой. Вот, что нам может понадобиться.

### Браузер

Естественно, что для работы в интернет нам, в первую очередь, потребуется веб-браузер. Поскольку неизвестно, каким именно браузером будет пользоваться потенциальный посетитель, для тестирования сайта желательно иметь под рукой все наиболее популярные варианты: [Firefox](#), [Opera](#), [Chrome](#), [Internet Explorer](#). Для проверки того, как сайт будет

выглядеть на экране коммуникатора или смартфона, достаточно популярного браузера [Opera Mini](#).

В качестве основного браузера я использую Firefox. Благодаря принципу расширяемости едва ли не любую необходимую функцию в Firefox можно получить путем установки небольшой программы-расширения (Add-ons). Таких расширений уже создано [несколько сотен](#), некоторые из них очень полезны веб-мастеру, например, FireFTP.

Расширения для Firefox можно установить двумя путями:

- щелкнув мышкой по ссылке загрузки, если компьютер подключен к интернету;
- запустив сохраненный .xpi-файл нужного расширения.

## FTP-клиент

Физически сайт находится на сервере. Сервер — это компьютер, постоянно подключенный к сети интернет. Хостинг — предоставление места на сервере и оказание сопутствующих услуг. О вариантах хостинга мы поговорим позже.

Обмен данными с сервером, как правило, происходит по FTP-протоколу, а это значит, нам необходима программа для связи с сервером — FTP-клиент. Рекомендую [FireFTP](#), бесплатный FTP-клиент, устанавливаемый как дополнение к Firefox. FireFTP имеет простой и интуитивно понятный интерфейс. Уверен, вы быстро поймете, что к чему. Для настройки соединения у провайдера необходимо узнать адрес FTP-сервера, логин и пароль.

FireFTP дает возможность перемещать файлы со своего компьютера на сервер и обратно, создавать и удалять папки и файлы, а также устанавливать права доступа (для этого выберите нужный файл или папку правой кнопкой мыши).

При обмене файлами с сервером надо учитывать следующее. Двоичные файлы (архивы, программы, изображения, аудио, видео, pdf и doc файлы) передаются и загружаются без каких-либо изменений. Закачанный или скачанный файл является точной копией исходного. Другая картина наблюдается при передаче текстовых файлов. Большая часть серверов в интернете работает под управлением операционных систем Unix и Linux (так называемых \*nix систем). В \*nix и Windows системах для обозначения конца строки используются разные символы ("line-feed" (LF) в \*nix и пара "carriage-return" (CR) и "line-feed" (LF) в Windows), поэтому при передаче текстового файла из одной системы в другую, искажается разбиение текста на строки. Это может стать серьезной проблемой, поскольку текстовыми файлами являются файлы со сценариями, например, PHP и Perl.

Чтобы этого не случилось, необходимо выбрать способ передачи, соответствующий типу файла. Режим **Binary** — для передачи двоичных файлов, и режим **ASCII** — для передачи текстовых файлов (в этом режиме заменяются символы конца строки). К счастью, многие FTP-клиенты могут автоматически выбирать необходимый режим передачи на основании расширения и содержания файла, однако будет не лишним проверить соответствующие настройки FTP-клиента: для файлов с расширениями .asp, .cgi, .css, .hqx, .htm, .html, .shtml, .js, .php, .pl, .cnf, .forward, .htaccess, .map, .pwd, .grp, .ctl, .text, .txt должен быть выбран режим ASCII (такие файлы, как правило и в отличие от двоичных файлов, открываются в текстовом редакторе).

Текстовые файлы, которые содержат интернациональные, например, кириллические, символы, также должны передаваться на сервер в двоичном режиме. Это объясняется тем, что стандартная 7-ми битная кодировка ASCII имеет ограниченную поддержку расширенных кодировок.

Если текстовые html-файлы можно без проблем загрузить и в текстовом и в двоичном формате, то загрузка в двоичном формате текстовых файлов со скриптами может вызвать серьезные проблемы. Это частая причина ошибки «Server 500 Error - Malformed Headers», а также других ошибок и отказов в работе. При передаче сценариев Perl и CGI убедитесь, что ftp-клиент находится в текстовом режиме.

## Устройство веб-сайта

Любой веб-сайт представляет собой набор взаимосвязанных документов. В первую очередь, это самые обыкновенные текстовые файлы, которые могут включать в себя ссылки на разнообразные мультимедиа файлы: изображения, музыку, видео.

Самый простой способ создания сайта — подготовить несколько файлов в текстовом редакторе (соблюдая определенную структуру документа) и загрузить их на сервер. Способ самый простой, но не самый эффективный, (например, чтобы изменить внешний вид сайта, придется править каждую существующую страницу).

Другой вариант — воспользоваться специализированной программой.

- Во-первых, можно установить программу на свой компьютер, а создаваемые или редактируемые с ее помощью файлы регулярно загружать на сервер. Одна из самых известных программ такого рода — [Adobe Dreamweaver](#).
- Во-вторых, можно установить программу сразу на сервер; в этом случае, подготовленные документы сразу же будут доступны посетителю сайта. Такие программы часто называют «движками» веб-сайта.

У каждого варианта есть свои преимущества, но мы их рассматривать не будем и сразу пойдем по пути программы на сервере.

Прежде чем мы перейдем к рассмотрению Drupal, надо упомянуть блог-сервисы, которые позволяют создать сайт без покупки домена и хостинга, а также избавляют от необходимости устанавливать и настраивать программное обеспечение. Пример такого сервиса от Google — [Blogger](#). Минусы такого подхода к созданию своего сайта:

- Полная зависимость от разработчиков сервиса,
- Необходимость соблюдать определенные поставщиком правила,
- Меньшие возможности управления сайтом и расширения его функционала.

## Drupal

Существующие программы отличаются друг от друга различными характеристикам: требованиями к хостингу и технической компетентности пользователя, функциональными возможностям и стоимостью, условиями распространения и популярностью среди веб-разработчиков.

Требования к «движку»:

- Легкость установки, освоения и использования,
- Безопасность,
- Нетребовательность к ресурсам хостинга,
- Функциональные возможности.

Все эти параметры очень тесно связаны с уровнем популярности программы. Чем популярнее программа, чем больше сообщество разработчиков и пользователей, тем:

- раньше выявляются и исправляются ошибки,
- проще получить совет и помощь,
- активнее развивается функционал.

На мой взгляд, для большего числа сайтов прекрасно подойдут популярные программы с открытым программным кодом (open source software). Немаловажно, что распространяются такие программы бесплатно. (Что не отменяет возможности выразить финансовую благодарность создателям этих программ.)

Не буду утверждать, что протестировал большое число программ, но из того, что пробовал и о чем читал, я рекомендую Drupal. Опыт работы с этой программой оставляет самые приятные впечатления.

Альтернативы Drupal: [Joomla!](#), [PHP-Nuke](#), [PostNuke](#), [WordPress](#),

## Базы данных

[Drupal](#) позволяет создавать и управлять содержимым и внешним оформлением сайта, при этом контент (наполнение сайта) хранится в расположенной на сервере базе данных.

При создании или редактировании страницы сайта Drupal записывает информацию в базу данных, а когда на сайт заходит посетитель, извлекает из базы нужную информацию и отображает страницу на экране.

Drupal может работать с различными базами данных, однако разработчиками рекомендуется бесплатная и быстрая база данных MySQL.

Перед установкой Drupal необходимо создать новую базу данных и пользователя с административными правами (если база не создана для вас хостером). Для этого воспользуйтесь средствами контрольной панели (ищите в панели ссылку «Базы данных MySQL» или ей подобную). Процесс создания новой базы может отличаться в зависимости от программного обеспечения на сервере, и если не сможете разобраться, обращайтесь в службу поддержки своего хостера.

## Хостинг

Пришло время выбрать провайдера услуг хостинга и пакет услуг. Разные программы предъявляют различные требования к хостингу. Требования Drupal версии 6.14 таковы:

- веб-сервер Apache;
- поддержка языка программирования PHP 4 или PHP 5;
- база данных MySQL;
- включенный модуль `mod_rewrite` и возможность использования локального файла `.htaccess` для реализации произвольных адресов страниц (так называемые "Clean URLs").

Drupal не занимает много места на сервере, чего не скажешь о контенте. Если вы планируете размещать на сайте мультимедиа, выбирайте пакет с большим дисковым пространством. В любом случае, позднее можно будет докупить место у провайдера.

Детально с требованиями можно ознакомиться по ссылке: <http://drupal.org/requirements>.

Один из вариантов хостинга — бесплатный хостинг. Я бы рассматривал его только в случае, если вам не важен имидж сайта. Для бизнес-сайта этот бесплатный хостинг, равно как и домен вида [myzen.narod.ru](http://myzen.narod.ru) категорически противопоказан. Кроме этого, бесплатный хостинг,

как правило, не соответствует техническим требованиям «движков».

## Настройка домена и почты

Перед установкой Drupal надо настроить домен. Так называемый DNS-сервер превращает зарегистрированное доменное имя в IP-адрес сервера, на котором находится сайт. Проще говоря, когда пользователь вводит в обозреватель адрес сайта вида `www.example.com`, браузер обращается к DNS-серверу, который определяет необходимый IP-адрес и передает его браузеру. Для настройки DNS необходимо в панели управления доменом указать первичный и вторичный адреса DNS-серверов хостера, а в панели управления хостингом указать имя сайта.

Также в контрольной панели хостинга можно настроить почту, получив возможность писать письма со своего домена. Ищите в панели ссылку «Управление почтой», или подобную ей.

## Установка Drupal на сервер

1. [Скачиваем последнюю доступную версия Drupal](#). Распаковываем архив и переносим содержимое с помощью FTP-клиента к себе на сервер в корневую папку (либо совпадает с именем домена, либо `public_html`, либо другой вариант, который вам укажет провайдер). Обратите внимание, что переносить надо не папку, которая получилась после распаковывания архива, а содержимое этой папки.
2. Создаем файл конфигурации:
  - находим в папке `sites/default` файл `default.settings.php`;
  - под именем `settings.php` сохраняем его копию;
  - разрешаем веб-серверу вносить в него изменения, присваивая файлу `settings.php` на время установки права доступа `0666`, а папке `sites/default` — `0777`.
3. В папке `sites/default` создаем папку `files` для хранения пользовательских файлов, например, изображений; в папке `sites/all` создаем папку `modules` для установки дополнительных модулей и папку `themes` для загрузки тем оформления сайта.
4. Для запуска программы установки переходим по адресу своего сайта и вносим необходимые данные:
  - название базы данных, имя пользователя и пароль;
  - название сайта и адрес, с которого будут отправляться служебные сообщения;
  - имя, почта и пароль пользователя (обратите внимание, что первый пользователь сайта становится его администратором);
  - часовой пояс;
  - активация «чистых ссылок» (если сервер поддерживает такую возможность, появится сообщение: «Your server has been successfully tested to support this feature») и автоматической проверки обновлений.
5. После завершения установки возвращаем конфигурационному файлу `settings.php` права доступа `0644`, а папке `sites/default` — `0755`.

Поздравляю! Установка Drupal завершена, можно приступить к увлекательному процессу настройки сайта и его наполнению контентом.

На старом хостинге я столкнулся с такой проблемой. Установка прерывалась с сообщением об ошибке «register\_globals on». Решил проблему, поместив в файл `.htaccess` строку кода:

«php\_flag register\_globals off».

## Начальная настройка Drupal

### Информация о сайте

Новый сайт встретит вас страницей приветствия и предложением приступить к настройке. Сделать это можно с помощью меню, которое находится в боковой панели сайта. Путь к странице, на которой нужно ввести информацию о сайте, такой:

**Administer > Site configuration > Site information.**

Перейти на нужную страницу можно, зная ее адрес, в данном случае:

<http://sample.com/admin/settings/site-information>

Здесь и дальше вместо sample.com указывайте свой домен).

Итак, заполняем поля:

- **Name:** Название проекта будет отображено в заголовках страниц сайта. Хорошо, если в название будут использованы ключевые слова, отражающие основную тематику сайта.
- **E-mail address:** Указанный здесь электронный адрес будет указан в сообщениях, направляемых пользователям сайта.
- **Slogan:** Слоган отображается под названием проекта. Здесь также желательно использовать ключевые слова.
- **Mission:** Заявление о целях сайта выводится на главную страницу в некоторых темах оформления.
- **Footer message:** Нижний колонтитул страниц сайта можно использовать для указания авторских прав.
- **Anonymous user:** Незарегистрированные пользователи оставляют комментарии под указанным здесь именем, хороший вариант — Гость.
- **Default front page:** С помощью этого поля любую страницу сайта можно указать в качестве главной.

### Настройка модулей

Drupal состоит из модулей, каждый из которых обеспечивает выполнение какой-либо функции. Установленный Drupal уже включает в себя самые необходимые модули. Если какой-либо функции вам не хватает, этот недостаток можно восполнить с помощью модулей, разработанных сообществом пользователей. Установку дополнительных модулей мы рассмотрим ниже. Сейчас зайдите на страницу управления модулями и включите те, что вам необходимы:

**Administer > Site building > Modules**

<http://sample.com/admin/build/modules>

### Управление пользователями

Drupal предоставляет широкие возможности для управления пользователями:

- Различные варианты регистрации новых пользователей.

- Возможность создания разнообразных ролей, присваивая каждой свой уровень доступа к возможностям сайта.
- Создать правила для учетных записей пользователей.

Выполнить необходимые настройки можно на странице:

**Administer > User management > Users**

<http://sample.com/admin/user/user>

## Настройка блоков

Drupal выводит информацию в виде блоков. Меню — пример блока. За создание блоков отвечают те или иные модули. Блоки можно настраивать, определяя месторасположение и другие опции, в зависимости от характера самого блока. Кроме этого, можно создавать новые блоки. Страница управления блоками:

**Administer > Site building > Blocks**

<http://sample.com/admin/build/block>

## Настройка внешнего оформления сайта

Система тем Drupal позволяет легко изменить дизайн всего сайта. Это возможно благодаря тому, что содержание страниц отделено от их оформления. Свежеустановленный Drupal предлагает на выбор несколько стандартных тем.

Выбрать и настроить тему оформления можно на соответствующей странице:

**Administer > Site building > Themes**

<http://sample.com/admin/build/themes>

Настраивать можно как каждую тему в отдельности, так и определить настройки по умолчанию для сайта и всех тем оформления. Они будут использоваться, пока вы не переопределите эти настройки для какой-либо темы

Если стандартные темы вас не устраивают, вы можете найти и скачать себе что-то более подходящее из десятков доступных в Сети вариантов.

На заре веб-дизайна было принято создавать макет страницы с помощью HTML-таблиц. Это не лучший метод уже потому, что таблица предназначена для показа табличных данных, а не всей веб-страницы. Выбирая тему, отдайте предпочтение варианту без таблицы в ее основе.

Также надо определиться, что вам больше подходит: макет с фиксированной шириной (fixed) или «резиновый» (liquid). Резиновый макет заполняет все окно браузера, независимо от его размера.

Обратите внимание, что административную тему надо определять отдельно:

**Administer > Site configuration > Administration theme**

<http://sample.com/admin/settings/admin>

## Таксономия

Изначально работа с веб-сайтом строилась следующим образом. На локальном диске создавалась директория. В пределах этой директории специальными HTML-редакторами создавались и редактировались документы, которые периодически передавались на веб-сервер. Внутри корневой директории создавались вложенные директории, которые фактически представляли собой категории сайта. Как правило одна директория отводилась под изображения. Изначально заданную структуру сайта было сложно изменить. В Drupal все

страницы находятся «на одном уровне». Структура задается с помощью тегов (меток), которые присваиваются каждой странице. Теги организуются в систему словарей.

Создать и отредактировать словари можно на соответствующей странице:

**Home > Administer > Content management**

<http://sample.com/admin/content/taxonomy>

## Настройка меню сайта

Содержание — главное, зачем посетитель приходит и возвращается на сайт, но без удобной навигации самый вкусный контент может банально остаться незамеченным. То, как устроена навигация по сайту, влияет на общее впечатление от работы с веб-ресурсом. С точки зрения навигации Drupal позволяет

- Организовать поиск по сайту.
- Настроить форму связи с веб-мастером.
- Построить понятную и последовательную системы навигации с помощью системы тегов и меню.

Drupal дает возможность создавать и редактировать разнообразные меню. В первую очередь и по мере наполнения сайта необходимо настроить Главное меню (Primary links):

**Administer > Site building > Menus**

<http://sample.com/admin/build/menu>

## Настройка сообщений об ошибках

Еще одну важную настройку нужно сделать на странице:

**Administer > Site configuration > Error reporting**

<http://sample.com/admin/settings/error-reporting>

- **Default 403 (access denied) page:** Здесь можно указать адрес страницы, которая будет демонстрироваться посетителю сайта в случае отказа доступа к запрашиваемому документу.
- **Default 404 (not found) page:** Здесь, соответственно, поле для адреса страницы, которая сообщает, что запрашиваемый документ не найден на сайте.
- **Error reporting:** вывод сообщения об ошибках, выберите вариант «*Write errors to the log*», это важно с точки зрения безопасности.

## Включение русскоязычного интерфейса

Drupal позволяет создавать мультиязычные сайты, интерфейс и содержание может быть на практически на любом языке. Пока ограничимся переводом интерфейса. Первым делом надо включить русский язык на странице

**Administer > Site configuration > Languages**

<http://sample.com/admin/settings/language>

После этого можно скачать с сайта Drupal.org .po-файл с переводом интерфейса и импортировать его себе на сайт. Не забудьте поставить русский языком по умолчанию.

**Administer > Site building > Translate interface**

<http://sample.com/admin/build/translate/import>

Либо перевести не весь интерфейс, а только необходимые строки:

**Administer > Site building > Translate interface**

<http://sample.com/admin/build/translate/search>

Я предпочитаю второй вариант.

Выше перечислены только базовые настройки сайта на базе Drupal. Уверен, что проделав первые шаги, вы поймете логику и сможете создать свой непохожий на другие сайт.

Посетите страницу <http://sample.com/admin> и изучите имеющиеся там возможности.

## Расширение функциональности Drupal

С помощью дополнительных модулей можно существенно расширить базовую функциональность Drupal. На сайте проекта можно найти [десятки разнообразных модулей](#), разработанных независимыми программистами.

Прежде всего необходимо убедиться, что выбранный модуль поддерживает установленную версию Drupal. Процедура установки модуля проста:

1. Скачать и распаковать архив с модулем себе на компьютер.
2. Папку, полученную в результате разархивирования файла, загрузить с помощью ftp-клиента в папку /sites/all/modules.
3. Активировать установленный модуль на странице управления модулями: Administer > Site building > Modules
4. Настроить модуль на странице модуля.

## Публикация нового контента

Англоязычные веб-мастера говоря: «Content is King». Посетитель приходит на сайт за полезной и информацией. Давайте посмотрим, как осуществляется в Drupal публикация контента.

В меню находим ссылку **Create content** (<http://sample.com/node/add>). По умолчанию Drupal предлагает два вида страниц: Page и Story. Некоторые модули при включении добавляют свои виды страниц. Вы также можете создать свой тип страницы, если имеющихся для вас недостаточно. Любой тип содержания имеет свои настройки, от которых, в частности зависит, будет ли новая страница выводиться на главную. Эти настройки можно изменять по своему усмотрению. По умолчанию страница вида Page не выводится на главную и не дает возможность оставлять комментарии (подходит для служебных страниц сайта), Story имеет противоположные настройки и подходит для статей по теме.

Пора заменить страницу приветствия на что-то более соответствующее названию сайта. Поля на странице создания содержания могут изменяться в зависимости от набора включенных моделей и прав, которыми имеет конкретный пользователь. По умолчанию доступны такие поля:

**Create content > Create Page**

<http://sample.com/node/add/page>

**Title:** Заголовок страницы обязателен к заполнению. Название важно как с точки зрения привлечения читателей, так и для поисковой оптимизации.

**Menu settings:** Ссылку на новую страницу можно тут же включить в меню.

**Body:** Поле для текста заметки.

**Input format:** Drupal позволяет управлять вводом текста, разрешая одни теги

форматирования и ограничивая использование других. О тегах мы поговорим позднее. Форматы ввода можно изменить, и даже создавать новые форматы.

**Revision information:** Можно создавать несколько версий страниц сайта.

**Authoring information:** Здесь можно указать автора заметки.

Publishing options: Выбираем, как будет опубликована страница:

- Published — если не отметить этот пункт, страница сохранится в виде черновика и не будет доступна посетителям сайта.
- Promoted to front page — после выбора этого пункта опубликованная страница появится на главной странице.
- Sticky at top of lists — есть возможность поставить заметку на верх списка страниц.

## Язык HTML

HTML (HyperText Markup Language) — язык гипертекстовой разметки. Говоря простым языком, HTML — это язык, понятный веб-браузеру. Словарь HTML включает так называемые теги (tags) и атрибуты тегов. Теги и их атрибуты являются командами, указывающими браузеру, как следует отобразить веб-страницу на экране компьютера.

Большинство HTML-выражений имеет открывающий и закрывающий тег. Вот как будет выглядеть команда браузеру отобразить параграф с текстом «Параграф»:

```
<p>Параграф</p>
```

Открывающий тег дает команду браузеру начать новый параграф, закрывающий тег — показывает, где надо остановиться.

Теги можно комбинировать между собой, вкладывая одну конструкцию в другую следующим образом.

```
<p><em>Параграф</em></p>
```

То есть, вначале надо закрыть тег, который находится внутри, и уже потом, тег, который стоит снаружи. Вариант ниже неправильный:

```
<p><em>Параграф</p></em>
```

Основные теги разметки HTML-документа:

```
<html>
  <head>
    <title>Заголовок страницы</title>
  </head>
  <body>
    <h1></h1>
    <p>Текст страницы.</p>
    <p>Новый абзац с разрывом строки.<br />
    Продолжение абзаца.</p>
  </body>
</html>
```

Элементы <html>, <head>, <body> являются обязательными в каждом html-документе.

Содержание тега <head> является заголовком документа. Тут могут находиться такие полезные теги:

- <title>Имя страницы, отображаемое в заголовке браузера</title>;
- <meta name="description" content="Описание страницы">;
- <meta name="keywords" content="ключевые слова, содержащиеся на странице">;
- <meta name="URL" content="http://www.sample.com">, адрес основной страницы для индексации;
- <meta http-equiv="content-language" content="ru">, язык страницы;
- <meta http-equiv="content-type" content="text/html; charset=utf-8">, кодировка страницы;
- <meta http-equiv="refresh" content="5; URL=http://www.sample.com/">, команда перенаправить пользователя на указанную страницу через указанное в секундах время (полезно разместить страницу с подобным кодом в служебным папках, например, в папке с изображениями).

HTML-теги призваны отобразить структуру документа, а не стиль оформления его элементов. Несмотря на наличие тегов форматирования текста предпочтительно отказаться от них в пользу HTML- и CSS-стилей.

Определение структуры документа	Определение вида элемента (лучше отказаться от использования)
<strong>Акцентируемый текст</strong>	<bold>Выделение текста жирным шрифтом</bold>
<em>Акцентируемый текст</em>	<i>Выделение текста курсивом</i>
<del>Удаленный из первоначальной версии документа текст</del>	<strike>Перечеркивание текста</strike>
<blockquote>Цитата отдельным блоком.</blockquote>	
<cite>Цитата в тексте.</cite>	
<code>Компьютерный код</code>	
<kbd>Текст, вводимый пользователем</kbd>	
<samp>Пример в тексте</samp>	
<dfn>Определение</dfn>	
<pre>Введенный текст сохраняет исходное форматирование.</pre>	
<span>Встроенный элемент страницы.</span> Позволяет выделить часть информации внутри других тегов и установить для нее свой стиль.	
<div>Прозрачный контейнер, в который можно поместить часть содержимого страницы. Содержание контейнеров можно	

накладывая друг на друга, динамически скрывать одни и показывать другие контейнеры.</div>	
<!--Комментарий, не отображаемый браузером.-->	

При желании можно внутри тега указать, как браузеру отобразить данный элемент, например, <h1 style="color: red">.

Короткий список CSS-атрибутов:

- color: цвет шрифта;
- font-family: гарнитура;
- font-size: размер текста;
- font-style: стиль текста (e.g.: oblique, italic);
- font-weight: начертание текста;
- text-align: выравнивание текста;
- text-decoration: оформление текста (e.g.: underline, line-through);
- list-style-type: стиль списка (e.g.: square, circle, disc, upper-roman, lower-roman).

Чтобы каждый раз не описывать стиль элемента внутри тега, рекомендуется создать правило во внешней таблице стилей, а для тега указать параметр class или id с именем соответствующего селектора, например, <h1 class="Red">.

Познакомиться с тегами и их атрибутами лучше всего на сайте [W3C](#).

## Язык XML

XML (eXtensible Markup Language, расширяемый язык разметки) служит для разметки структурированных документов, аналогичен языку HTML и позволяет определять свои собственные теги.

## Ссылки

Веб существует благодаря ссылкам. Синтаксис указания ссылок:

<a href="http://sample.com/contact">Текст ссылки</a>.

Ссылки могут быть **абсолютными** — с указанием используемого протокола и полного адреса страницы (как приведенный выше пример), и **относительными** — указывается только часть адреса (например, <a href="/contact">Текст ссылки</a>). Слеш (/) означает корневую папку, в которой «находятся» все создаваемые Drupal страницы. В случае сложной структуры надо указывать путь, включающий все «вышестоящие» директории, например, "/sites/default/files/image.jpg". Поскольку относительная ссылка указывается относительно текущей страницы сайта или корневой директории, она действует только в пределах одного сайта, чтобы создать внешнюю ссылку надо указывать абсолютный путь.

При желании можно организовать возможность **перехода внутри страницы**. Для этого часть текста помечается как якорь: <a name="имя\_якоря">Текст</a>. Ссылка ставится так: <a href="#имя\_якоря">Текст ссылки</a>. Имена якорей чувствительны к регистру букв.

Закладки можно использовать и для перехода на другие страницы, например <a href="http://sample.com/links.htm#имя\_якоря">так</a>.

По умолчанию, при переходе по ссылке новый документ открывается в текущем окне. С помощью параметра `target` можно выбрать **другие варианты открытия страницы**. Синтаксис таков: `<a href="/contact" target="_blank">Текст ссылки</a>`. Имеющиеся варианты:

`_self`: открыть документ в текущем окне браузера (значение по умолчанию);

`_blank`: открыть документ в новом окне;

`_parent`: открыть документ в родительском фрейме, если он есть, иначе действует, как `_self`;

`_top`: оменить все фреймы и загрузить страницу в полном окне, иначе действует, как `_self`.

Также бывает полезно указать **заголовок ссылки**, который будет отображаться при наведении курсора: `<a href="/contact" title="Форма для связи">Текст ссылки</a>`.

Чтобы указать поисковой системе не включать ссылку в оценку рейтинг страницы (link juice), следует использовать атрибут `nofollow`, например: `<a href="/contact" rel="nofollow">`.

Использование этого тега — избежать «перетекания» рейтинга ссылающейся страницы.

Для указания *адреса электронной почты* следует использовать ссылку следующего синтаксиса: `<a href="mailto:info@sample.com">Отправить письмо!</a>`. Клик на ссылку вызывает приложение электронной почты, установленное в системе пользователя по умолчанию, и открывает форму нового письма с указанным в ссылке адресом.

Можно добавить в ссылку дополнительные **переменные**:

```
<a href="mailto:info@sample.com?
subject=Тема_письма&body=Содержание_письма_со_ссылкой_на_сайт_http://vasilevsky.net&c
s=Адрес, куда направляется копия письма">
```

В теме письма пробелы стоит заменить либо нижним подчеркиванием, либо комбинацией `%20`. Для перехода на новую строку в тексте письма используйте комбинацию `%0A`. Помимо копии (параметр — `cc`) можно отправить скрытую копию (параметр — `bcc`).

Можно даже указать имя получателя: `<a`

```
href="mailto:Имя_Вебмастера_&lt;info@sample.com&gt;">Отправить письмо!</a>
```

## Списки

Существует три типа списков: нумерованный список (создается с помощью тега `<ul>`), нумерованный список (`<ol>`) и список определений (`<dl>`). Структура списка:

```
<ul>
```

```
<lh>Заголовок списка</lh>
```

```
<li>Пункт списка,</li>
```

```
<li>Пункт списка.</li>
```

```
</ul>
```

С помощью параметр `type` можно определить тип нумерации нумерованного или оформление нумерованного списка: `type="A/a/I/i/disk/square/circle"`. Также можно определить цифру, с которой следует начать нумерацию: `type="n"`.

Список определений создается так:

```
<dl>
```

```
<dt>Термин</dt>
```

```
<dd>Определение</dd>
```

```
</dl>
```

Списки могут вкладываться друг в друга.

## Таблицы

Структура таблицы (таблица из двух столбцов и двух рядов):

```
<table>
<caption>Название таблицы</caption>
<tr>
<th>Заголовок таблицы</th>
<th>Заголовок</th>
</tr>
<tr>
<td>Данные</td>
<td>Данные</td>
</tr>
</table>
```

Размеры всей таблицы и ее элементов можно установить с помощью параметров `width=" "` и `height=" "`.

Для оформления табличной сетки можно воспользоваться параметрами `frame` и `rules`, например: `<table frame="box" rules="none">`. Возможные значения `frame`: `void` (без рамки), `box` (с рамкой). Возможные значения `rules`: `none` (без сетки ячеек), `groups` (с линиями между группами), `rows` (с линиями между строками).

Параметр `align` (как и `valign`) внутри тега `<table>` определяет выравнивание таблицы на странице, внутри тегов `<tr>` и `<td>` — выравнивание внутри строки или ячейки.

Параметр `char` задает символ-разделитель при выравнивании по символу (`align="char"`). Параметр `charoff` определяет в процентах или пикселях величину отступа относительно начала ячейки. Пример: `<tr align="char" char="," charoff="5%">`

Параметр `cellpadding` задает свободное пространство между содержимым ячейки и ее границами, параметр `cellspacing` — пространство между ячейками (по умолчанию равняется 2).

Цвет сетки указывается с помощью параметра `border-color`, фона — `bgcolor`.

Объединить колонки можно с помощью атрибута `colspan=" "`, например, `<tr><td colspan="3">Заголовок</td>`. Строки объединяются с помощью атрибута `rowspan=" "`, например, `<tr><td rowspan="2">`.

## Формы

Формы состоят из двух частей: HTML-кода с описание формы и сценария, который обрабатывает введенные в форму данные. Для обработки данных могут быть использованы CGI-сценарии (Common Gateway Interface), которые устанавливаются на сервере веб-мастера. Соответственно, сервер должен поддерживать сценарии CGI.

В коде формы указывается адрес сценария обработки данных (например, `action="http://samle.com/subscribe/index.php"`). Возможны два метода обработки:

- POST — позволяет отправить введенные данные в теле сообщения;

- GET — дает возможность отослать информацию на сервер, добавив данные в конец адреса URL; не подходит для передачи данных больших форм, так как длина URL ограничена 8192 символами; не подходит для передачи секретной информации.

## Стили и форматирование текста

Непосредственно в коде веб-страницы можно указать, как должен выглядеть тот или иной элемент, однако более эффективный способ заключается в отделении содержания от оформления (метод, также существующий в текстовых редакторах). Это позволит без редактирования каждой отдельной страницы вносить изменение в оформление всего сайта. Для такого оформления страниц используются каскадные таблицы стилей (Cascading Style Sheets, CSS). Изменения в CSS-файле мгновенно отображаются во всех связанных веб-страницах.

CSS определяет форматирование текста либо в специфическом теге (например, `<p>`, `<em>`, `<blockquote>`) или специфическом классе, который создает веб-мастер.

Есть возможность задать стиль для отдельной страницы (указав стиль в заголовке страницы) и даже для отдельного элемента страницы (в соответствующем теге).

Важно понимать, как браузер пользователя интерпретирует код страницы. Приоритет имеет «более близкая» к элементу информация. Например, стиль ячейки имеет приоритет над стилем строки, а стиль строки — над стилем таблицы. В общем случае, стиль абзаца имеет приоритет над стилем страницы, а стиль страницы — над стилем всего сайта. CSS-стиль, примененный с атрибутом `class`, блокирует атрибут HTML-стиля. Если к одному элементу страницы применяются два стиля, которые не противоречат друг другу, будут использованы оба стиля.

Для использования CSS надо создать файл стилей с набором правил (с расширением `.css`) и дать на него ссылку в заголовке страниц, на которые должно распространяться его действие.

Структура файла:

```
body {
  padding: 5em 0 0 3em;
  font-family: tahoma, verdana, arial, helvetica;
  border-top: 10px solid gray;
}
```

Вначале указывается элемент страницы, затем в фигурных скобках через точку с запятой задаются необходимые параметры.

В одном правиле можно указывать сразу несколько элементов, например: `h1,h2,h2 {color:red;}`.

Внутри тега стиль определяется так: `<h1 style="color:red;">`.

В заголовке отдельной страницы стиль прописывается так:

```
<style type="text/css">
h1 {color:red;}
</style>
```

Ссылка на внешний файл с таблицей стилей оформляется в заголовке страницы так:

```
<link rel="stylesheet type="text/css" href="style.css">
```

Короткий список свойств основных элементов

- Фон: background-color.
- Шрифт: font-size, font-family, font-style (normal/italic), font-weight (normal/bold), font-color, font (позволяет определить все вышеперечисленные свойства).
- Текст: line-height (отступ сверху, normal/n/%), text-indent (отступ текста), text-decoration (none/underline/overline/line-through), text-align (left/right/center/justify), text-transform (none/capitalize/uppercase/lowercase).
- Блоки: margin-top (auto/n/%), margin-right, margin-bottom, margin-left, padding-top (n/%), padding-right, padding-bottom, padding-left, padding, border-top-width (n/thin/medium/thick), border-right-width, border-bottom-width, border-left-width, border-width, border-color, border-style (none/dotted/dashed/solid/double/groove/ridge/inset/outset), border-top, border-right, border-left, border-bottom, border, width (n/%), height, float (left/right/none), clear (расположение остальных элементов, left/right/both/none).
- Списки: display (none/block/inline/list-item), white-space (normal, pre, nowrap), list-style-type (disc/circle/square/decimal/lower-roman/upper-roman/lower-alpha/upper-alpha/none), list-style-image (url/none), list-style-position (inside/outside), list-style.

## Использование цветов

В HTML и CSS для обозначения цветов используются шестнадцатеричные значения в формате #xxxxxx или названия цветов. В первом случае три пары чисел указывают уровень красного, зеленого и синего цвета (RGB).

Консорциум W3C установил список из 16 названий цветов: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow. Поскольку браузеры могут по-разному отображать одно и то же наименование, лучше использовать шестнадцатеричный формат.

Когда цветопередача мониторов была ограничена 256 цветами, в качестве стандарта был утвержден список из 216 «безопасных цветов» (Web Safe Colors). В настоящее время список потерял актуальность.

## Тег <iframe>

Тег <iframe> позволяет вставить в одну страницу содержание другой страницы. Полезная функция, если надо включить сервис, находящийся на другом сайте, например, гостевую книгу или блог. Пример кода:

```
<iframe src="http://www.narod.ru/guestbook/?owner=8519087" width="100%" height="400"
frameborder="0" align="top" scrolling="auto"></iframe>
```

## Фреймы

Фреймы — способ составить веб-страницу из нескольких независимых страниц. Раньше было популярно навигационное меню делать с помощью фрейма. Большого смысла в этом нет и лучше воздержаться от использования фреймов.

## Специальные символы

В HTML существует набор специальных символов. Специальные символы можно указать либо с помощью имени (например, &mdash;), либо с помощью числового значения

(например, &#8212;).

#### Теги специальных символов

«	&laquo;
»	&raquo;
©	&copy;
±	&plusmn;
-	&ndash;
—	&mdash;
Неразрывный пробел	&nbsp;

## Типографы

Упростить себе работу по верстке HTML кода можно с помощью специальных сервисов:

- [Типограф Студии Артемия Лебедева](#)
- [Типограф Кирилла Панфилова Devangari 3](#)

Типограф существенно облегчает жизнь веб-дизайнеру и делает за него массу полезной работы: поставит вместо знаков дюйма нормальные кавычки («елочки» и „лапки“), удалит лишние пробелы, в нужных местах поменяет дефисы на тире, неразрывным пробелом «привяжет» все короткие союзы и предлоги к следующим за ними словам и т.д. Результат — корректная экранная типографика.

## LJ-tegs

Если вы решите дублировать информацию со своего сайта в дневнике LiveJournal, вам будет полезно знать несколько «фирменных» тегов, используемых этим сервисом.

`<lj-cut text="текст ссылки"> </lj-cut>` - информация, которая находится между тегами, скрывается на главной странице, по умолчанию ссылкой является надпись Read more, однако текст ссылки можно отредактировать, давая описание содержания, на которое она ведет. Если скрывается все продолжение до конца, то закрывающий тег можно не ставить.

Данный тег позволяет улучшить читаемость блога: на главную страниц выводится краткий анонс со ссылкой на статью. Это особенно полезно, когда ваш блог выводится в ленте ваших друзей. Этот тег также полезен для убирания с основной страницы блога больших изображений.

`<lj user="имя пользователя в LJ">` - с помощью этого тега можно дать ссылку на другого пользователя или сообщество LiveJournal.

## Использование мультимедиа

Использовать у себя на сайте изображения, видео и аудио записи — хорошая идея. Проблема в том, что пользователь не любит ждать, и чем быстрее загружается страница, тем лучше. Соответственно, прежде, чем помещать графический или мультимедийный файл себе на страницу, его необходимо оптимизировать.

## Оптимизация графических файлов

Подготовка изображения для загрузки на сервер состоит из двух шагов:

1. Уменьшение размера (например, до 500-600 пикселей в ширину);
2. Сохранение изображения в нужном формате и с необходимым коэффициентом сжатия.

На веб-страницах используются три формата графических файлов: JPEG, PNG и GIF.

Формат графических файлов JPEG является одним из самых распространенных способов сжатия фотоизображений, так как позволяет существенно уменьшить объем файла без заметной для человеческого глаза потери качества. При сохранение jpeg-файла в программе редактирования графики (например, [IrfanView](#) или [GIMP](#)) можно выбрать степень сжатия от 1 до 100 условных единиц. Чем больше число, тем лучше качество изображение и тем больше размер файла. Степень сжатия определяется характером изображения, как правило для веба выбирают коэффициент от 30% до 60%. Сохранение файла с прогрессивным jpeg-кодированием (опция Progressive) позволит выводить изображение на экран пользователя по мере загрузки файла.

Еще меньший размер можно получить, используя формат GIF и PNG. Преимуществами gif-файлов также является возможности определять один из цветов рисунка как прозрачный и создавать анимированные изображения. Палитра из 256 цветов (против 16 777 216 у jpeg) ограничивает применение формата gif «рисованными» изображениями. Формат PNG был создан как альтернатива проприетарному формату GIF.

Вывод изображения осуществляется командой: ``

Если указать ширину и высоту изображения (в пикселях), браузер сразу зарезервирует под изображение место на странице, иначе в процессе загрузки текст на странице будет дергаться.

Изображение можно сделать ссылкой: `<a href="http://sample.com/"></a>`

Для использования на сайте мультипликации можно использовать технологию Flash. Сжатый и оптимизированный для просмотра в браузере Flash-файл имеет разрешение .swf. У пользователя для просмотра flash-анимации должен стоять Flash Player.

## Оптимизация видео-файлов

Краткий ликбез, необходимый для оптимального использования видео у себя на сайте.

**Стандарт** — это спецификация (описание) алгоритма кодирования (сжатия) данных.

Популярные стандарты:

- Стандарты MPEG (Moving Picture Expert Group) — отличаются высокой скоростью компрессии/декомпрессии видео и звука и высокой степенью сжатия без заметной потери качества изображения. Стандарт **MPEG-1** обеспечивает «картинку» аналогичную VHS-видео. **MPEG-2** (DVD-Video) по качеству соизмерим с телевизионным изображением. **MPEG-3** разрабатывался для систем телевидения высокой четкости (HDTV) и сейчас является частью стандарта MPEG-2. **MPEG-4** используется в разработке интерактивного мультимедиа, графических приложений и цифрового телевидения.
- **H.264** (также известен как MPEG-4 AVC (Advanced Video Coding) — принят в

качестве стандарта для сжатия видео высокой чёткости (HD, HDTV), распространяемого на оптических носителях нового поколения: Blu-ray и HD DVD. Сжатие H.264 обеспечивает отличное качество изображения и небольшой объём файла, но предъявляет более высокие требования к оборудованию для кодирования и воспроизведения видеофайлов.

- **WMV** (Windows Media Video) — формат сжатия видео от Microsoft; для записи, как правило, используются контейнер Windows Media (\*.wmv).
- **Real Media** — формат позволяет менять качество видео-потока в зависимости от качества канала связи, используется для живой телевизионной трансляции в Интернете.
- Другие форматы: Apple Quicktime, VCD (Video CD), Super Video CD, ASF (Active Streaming Format).

**Кодек** — программная реализация алгоритма кодирования. Популярные кодеки:

- DivX — самый распространённый проприетарный кодек стандарта MPEG-4;
- Xvid — открытый кодек, основанный на одной из версий кодека DivX;
- x264 — кодек для сжатия в стандарте H.264.

Строго говоря за декодирование отвечает декодер (например, ffDShow), хотя под кодеком часто понимают и кодер, и декодер. Один из самых популярных пакетов декодеров — [K-Lite Codec Pack](#). Если в системе не установлен необходимый декодер, то видеозапись не будет воспроизведена. Определить, какой именно декодер отсутствует, можно либо с помощью проигрывателя, либо с помощью специальной программы, например, AVIcodec или GSpot.

**Контейнер** - формат файла, в котором сохраняется видеоряд, звуковая дорожка/дорожки, субтитры и служебная информация. Популярные контейнеры:

- **AVI** (Audio and Video Interleaved) — стандартный медиаконтейнер для Microsoft Windows; позволяет использовать различные кодеки, обычно DivX и XviD.
- **MP4** — медиаконтейнер, поддерживающий аудио и видео из группы MPEG-4; официальный стандарт контейнера для видео H.264; может содержать звук только в формате AAC.
- **Matroska** — открытый и гибкий формат контейнера; расширения файлов: \*.mkv — для видео с субтитрами и звуком, \*.mka — для аудиофайлов, \*.mks — для субтитров.
- **Flash Video (FLV)** — медиаконтейнер для передачи видео через Интернет; используется популярными сервисами видеохостинга YouTube, Google Video, RuTube и другими.
- **Ogg** — медиаконтейнер для аудиокодека Vorbis и видеокодека Theora.
- **Windows Media** — формат контейнера для видеозаписей, разработанный Microsoft; расширения: \*.asf, \*.wmv, \*.wm.
- Другие виды контейнеров: Ogg Media (\*.ogm), RealMedia (\*.rm, \*.rv, \*.ram), QuickTime (\*.mov, \*.qt), VOB (\*.vob), DivX Media (\*.divx).

**Битрейт** — показатель, который характеризует объём информации в единицу времени, как правило в каждой секунде записи. Чем меньше битрейт, тем «легче» файл, и тем хуже качество изображения и звука.

## Оптимизация аудио-файлов

Наиболее популярный формат для сжатия аудиоинформации — это, безусловно, **MP3** (полное название — MPEG-Audio Layer-3). Форматы WMA (Windows Media Audio) не рассматриваю, поскольку не на каждом устройстве есть возможность проигрывать этот формат.

**Vorbis** — свободно распространяемый формат для сжатия звука; превосходит MP3 по ряду параметров; к сожалению, менее распространен. Формат хранения и соответствующий кодек называется OGG.

Также распространены следующие форматы:

- **MIDI** (Musical Instrument Digital Interface) — используется для инструментальной музыки, имеют малый формат и невысокое качество звука, поддерживаются браузерами без дополнительных плагинов.
- **WAV** (Waveform Extension) — файлы отличаются удовлетворительных звучанием и большим размером, поддерживаются браузерами без дополнительных плагинов.
- **AIF** (Audio Interchange File Format) — по качеству звука и размеру файла аналогичны wav-файлам, не требуют дополнительного программного обеспечения.
- **RA, RAM, RPM, Real Audio** — обеспечивает более высокую степень сжатия и меньший размер файла, чем MP3, но менее качественное звучание, позволяет потоковое воспроизведение звука, требует от пользователя установки дополнительного программного обеспечения.

Сжать аудиофайл в MP3 с нужным битрейтом можно с помощью программ [CDex](#) и [EAC](#). Лучший кодек для создания mp3-файлов — это открытый [Lame](#) (встроен в CDex).

Рекомендуемые настройки:

- Version — MPEG-2;
- Bitrate Min, Max — для записи голоса оптимальный параметр: 128 Kbps;
- Mode — Stereo (на низких битрейтах можно попробовать J-stereo или Forced stereo);
- Quality — Normal (q=5) для низких битрейтов (меньше 160 Кбит/с), High (q=2) для высоких битрейтов, Voice для голосовых записей.
- VBR Method и VBR Quality — настройка алгоритма кодирования с переменным битрейтом. Рекомендованное значение VBR Quality — VBR 4 (наилучшее качество обеспечивает значение VBR 0).

Экспериментируя с различными значениями можно достичь оптимального соотношения размер файла и качества звучания. Речь можно записывать с переменным и максимальным битрейтом в 128 Кбит/с.

Для загрузки и отображения видеофайлов, flash-роликов и звуковых файлов, которые браузер не понимает по умолчанию, используется тег <object>. С помощью этого тега можно подключить к браузеру специальный модуль (плагин) или запустить соответствующую формату файла программу.

Для совместимости со старыми версиями браузеров и для загрузки внешних данных рекомендуется внутрь контейнера <object> поместить контейнер <embed>:

```
<object><embed src="samle.swf" width="400" height="300" type="application/x-shockwave-flash" pluginspage="http://get.adobe.com/ru/flashplayer/"></embed></object>
```

## Скрипты и сценарии

Расширить интерактивные возможности сайта можно с помощью дополнительных модулей Drupal, а можно с помощью сторонних приложений. Самый простой способ — использование сценариев, которые в виде кода включаются в страницы сайта. Сценарии могут выполняться на сервере, а могут непосредственно на компьютере посетителя сайта. Последние выполняют простые задачи и чаще всего пишутся на языке JavaScript.

Сценарий JavaScript отслеживает определенные действия пользователя (например, клик мышкой) или события (например, загрузка страницы) и в нужный момент выполняет определенные действия (например, выводится информация).

Код JavaScript располагается либо в заголовке, либо в теле страницы между тегами `<script language="JavaScript">` и `</script>`. Если сценарий является самостоятельным .js-файлом, то ссылка на него оформляется так:

```
<script language="JavaScript src=имя_файла.js">  
</script>
```

## Google для веб-мастера

Google предоставляет веб-мастеру массу полезных возможностей и инструментов. Начнем с основного сервиса Google — поиска. Используя определенные команды в поисковом запросе, можно сузить поле поиска и получить более точные (и интересные) результаты. Например, команда **intitle:** ограничит поле поиска заголовками веб-страниц (содержание тега `<title>`). Параметры запроса надо вводить без пробела после команды, например *intitle:веб дизайн*. Другие команды:

**inurl:** поиск слов запроса в адресах веб-страниц, например, *inurl:vasilevsky*;

**intext:** поиск в тексте веб-страницы без учета содержания заголовков страниц и ссылок;

**site:** поиск в конкретном домене или поддомене, например, *хокку site:vasilevsky.net*;

**link:** поиск страниц, ссылающихся на заданный сайт, например, *link:vasilevsky.net*;

**related:** поиск похожих (по мнению Google) на указанную страниц, например, *related:vasilevsky.net*;

**cache:** поиск копии удаленных страниц в кэше Google;

**filetype:** поиск файла с заданным разрешением, например, *хокку filetype:pdf*;

## Обновление Drupal

Время от времени выходит обновленная версия Drupal. Обновленная версия может включать новые функции и исправление ошибок. В первую очередь, обновление важно с точки зрения безопасности системы. Поэтому рекомендуется включить модуль **Update status**, который регулярно проверяет актуальность установленной системы, сторонних модулей и тем и, в случае наличия новых версий, отправляет соответствующее сообщение на e-mail.

Какова процедура обновления системы:

Для начала стоит убедиться, что обновленная версия соответствует текущим возможностям хостинга (<http://drupal.org/requirements>) и совместима со сторонними модулями и темами.

Если все в порядке, скачиваем архив с системой себе на компьютер и готовимся к обновлению.

1. В первую очередь необходимо сохранить резервную копию

- базу данных;
  - содержание папки **sites**: файл конфигурации, сторонние модули и темы, а также пользовательские файлы;
  - файлы **.htaccess** и **robots.txt**, если в них вносились изменения.
2. Зайдите в систему под именем пользователя-администратора (первый пользователь системы: ID 1). Не выходите из системы до окончания процедуры обновления!
  3. Переведите сайт в оффлайн на странице <http://sample.net/admin/settings/site-maintenance>.
  4. Если используете стороннюю тему, переключитесь на одну из основных тем: Garland или Bluemarine.
  5. Отключите все сторонние модули.
  6. Удалите все файлы и папки текущей установки Drupal.
  7. Распакуйте архив с обновленной системой и скопируйте содержание папки на сервер.
  8. Перенесите на сервер файлы, сохраненные на первом шаге. Если в файлы **.htaccess** и **robots.txt** вносились изменения, воссоздайте эти изменения в новых версиях файлов.
  9. Убедитесь, что в новом файле конфигурации содержится корректная информация.
  10. Откройте страницу <http://sample.net/update.php>, вам будет предложено запустить процедуру обновления системы — запускайте.
  11. Если все прошло успешно, система предложит перейти на главную страницу или в панель управления. Теперь можно включить сторонние модули и предпочитаемую тему оформления. Если вы видите сообщения об ошибках, читайте файл UPGRADE.txt или раздел <http://drupal.org/upgrade>.
  12. Включите сторонние модули и еще раз запустите программу update.php для обновления таблиц базы данных.
  13. Включите предпочитаемую тему оформления.
  14. Возвращайтесь в online: <http://sample.net/admin/settings/site-maintenance>.

## Обновление дополнительно установленных модулей

Процесс обновления конкретного модуля может иметь свои особенности, которые, как правило, описываются в файлах README.txt и INSTALL.txt. В общем случае процедура выглядит следующим образом:

1. Перевести сайт в off-line.
2. Отключить модуль.
3. Удалить папку модуля и загрузить на сервер обновленную версию.
4. Активировать модуль.
5. Запустить файл /update.php.
6. Проверить права доступа и настройки модуля.
7. Перевести сайт в on-line.

## Список литературы

Создание и раскрутка сайтов в Интернете. Алекс Экслер. — М.: НТ Пресс, 2007.