

ОСНОВЫ ВЕБ-КОНСТРУИРОВАНИЯ

§ 1. Представление о веб-конструировании

1.1. Инструменты и методы разработки веб-сайтов

Большинство информационных ресурсов сети Интернет представлено в виде веб-страниц, которые объединяются в веб-сайты. Благодаря развитию телекоммуникационных технологий размещенная на веб-страницах информация доступна множеству людей. В отличие от информации на бумажных носителях, например книг, она может оперативно изменяться и практически мгновенно доставляться в любое место по запросу пользователя.

Для создания веб-страниц используется язык разметки гипертекстовых документов HTML (*HyperText Markup Language*).

В качестве инструментов веб-конструирования могут использоваться офисные программы, например входящие в состав Microsoft Office (Word, PowerPoint), которые не являются специальными средствами разработки веб-страниц. Лучших результатов можно достичь, используя специальные программы — веб-редакторы, например Microsoft FrontPage.

Выделяют две основные группы методов и соответствующих инструментов разработки веб-сайтов: *визуальные* и *ручные* (программные).

Визуальные методы позволяют производить все работы по созданию веб-страниц с высокой степенью автоматизации и не требуют знания языка разметки HTML. Они уменьшают трудоемкость и сроки разработки сайта. Суть визуальных методов отражена в принципе WYSIWYG (от англ. *What you see is what you get* — Что видишь, то и получаешь).

Разработано немало специальных инструментов — редакторов визуального конструирования. Наиболее известными являются Microsoft FrontPage, Adobe (Macromedia) Dreamweaver, NamoWebEditor и др. С помощью таких редакторов веб-страницы создаются (рисуются) в интерактивном режиме, при этом автоматически генерируется соответствующий **HTML-код**, который представляет собой набор команд языка разметки HTML. Отметим, что упомянутые редакторы позволяют выполнять все работы по созданию сайта без непосредственного подключения к сети Интернет, или, как говорят, в режиме *оффлайн* (*offline*). Затем созданный веб-сайт публикуется в сети Интернет, т. е. размещается на веб-сервере.



В последнее время для конструирования и сопровождения веб-сайтов используются системы управления их содержимым (контентом) — CMS (*Content Management System*), которые предоставляются специальными платными или бесплатными службами. Системы CMS представляют собой своего рода конструкторы. Они позволяют создавать сайт в режиме непосредственного подключения к сети, или *онлайн* (*online*), и сопровождать его в дальнейшем. При создании структуры сайта и разработке навигации по нему в системе CMS также не требуется знаний языка HTML.

Из бесплатных систем CMS наиболее популярны Joomla! (<http://joomla.ru/>) и Drupal (<http://www.drupal.ru>).

Конечно, трудно создать хороший сайт, не зная основ языка разметки HTML. Для работы с HTML-кодом могут использоваться специальные инструменты разработки, позволяющие набирать команды HTML **вручную**, например редакторы HotDog, Adobe HomeSite и др. Эти инструменты облегчают ввод и редактирование кода. Однако, вводить основные команды (**теги**) языка HTML можно даже в простейшем текстовом редакторе **Блокнот**, а просматривать результаты работы можно с помощью браузера.

Важную роль в выборе инструментов и методов веб-конструирования играет статичность или динамичность создаваемых страниц, а также наличие интерактивных элементов. *Статические* страницы отображаются браузером пользователя в том виде, в каком были созданы и размещены на веб-сервере. *Динамические* страницы генерируются по запросу пользователя — информация на них загружается серверными программами из баз данных. Такие базы данных обычно содержат информацию, которая требует постоянного обновления. Например, интернет-магазин пополняется сведениями о новых товарах и изменяющихся ценах.

Интерактивные (т. е. управляемые пользователем) элементы веб-страниц используются для ввода пароля, выбора товара, оценки его качества, ввода ответа при тестировании или голосовании, для формирования запроса на поиск информации в удаленной базе данных и т. п.



Кроме языка разметки гипертекстовых документов HTML, при создании веб-сайтов используют и специальные языки веб-программирования. Широкое применение получил язык сценариев Java Script. Написанные на нем конструкции, или *скрипты*, вставляются непосредственно на веб-страницы и интерпретируются браузером. Они используются для создания отдельных, как правило, интерактивных элементов веб-страниц, например динамических меню, часов, календарей, форм запросов, счетчиков посещений страниц, систем голосования и т. п. Для серверного программирования наиболее

часто используют языки PHP (от англ. *Hypertext Preprocessor* — препроцессор гипертекста), PERL (от англ. *Practical Extraction and Report Language* — практический язык для извлечения данных и составления отчетов).



1. В чем разница между визуальными и ручными методами веб-конструирования?
2. Какие инструменты могут использоваться при создании веб-сайтов?
3. Какие страницы называют статическими? Динамическими?

1.2. Проектирование сайта

Выделяют следующие основные этапы разработки веб-сайтов:

- определение тематики сайта, его целей и задач;
- проектирование структуры сайта, определение разделов и связей между страницами;
- разработка дизайна сайта, т. е. стиля оформления страниц;
- подготовка материалов (текстов и графики) для размещения на веб-страницах;
- конструирование страниц сайта (создание HTML-кода);
- размещение в сети (публикация) и тестирование сайта.

Рассмотрим на примере, как спроектировать веб-сайт кинотеатра.

Определим основную цель сайта: привлечение зрителей, и задачи: информирование о репертуаре кинотеатра, реклама фильмов.

Разработку проекта начнем с построения информационной модели сайта. Пусть для простоты наш первый сайт будет состоять из четырех веб-страниц. Структуру этого сайта для наглядности изобразим в виде двухуровневой схемы (рис. 1.1).

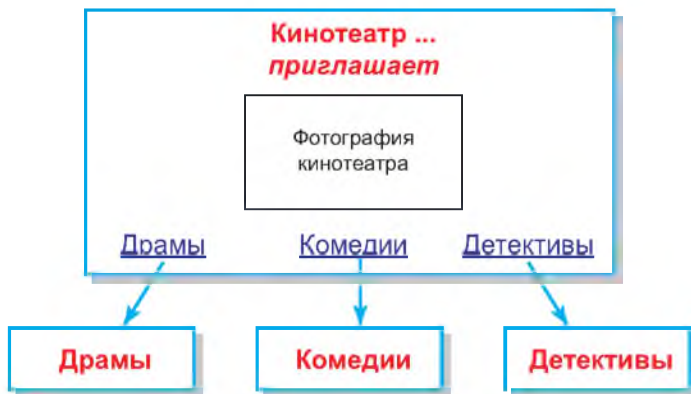


Рис. 1.1

На первом (верхнем) уровне схемы изобразим первую (главную) веб-страницу. На ней будут размещены: общая информация о кинотеатре (например, фотография, адрес) и гиперссылки.

На втором уровне схемы изобразим веб-страницы, посвященные фильмам определенных жанров, например «Драмы», «Комедии», «Детективы». Они имеют схожую структуру и будут отличаться лишь содержанием (рис. 1.2).

Название фильма	
Фрагмент фильма (фотография)	Жанр Режиссер Актеры Студия, год
Краткое содержание	

Рис. 1.2

Структура сайта в дальнейшем может быть усложнена добавлением страниц последующих уровней, например страниц, посвященных актерам, режиссерам, сценаристам, композиторам и т. п.

Теперь спроектируем дизайн сайта. Основными структурными элементами оформления веб-страниц являются текстовые блоки (основной текст, заголовки, списки, текстовые гиперссылки) и графические объекты: изображения (рисунки, фотографии, анимации), «обои» (фоновые рисунки), изображения-гиперссылки, разделительные линии. Для размещения элементов в заданных местах страниц будем использовать таблицы.

Сочетание свойств элементов веб-страницы определяет стиль ее оформления. **Стиль текста** задается совокупностью параметров формата: шрифт, размер, начертание; отступы, выравнивание, межсимвольный и междустрочный интервалы и др. **Стиль графических элементов** задается совокупностью параметров формы: размер, цвет, фактура материала, а также разнообразных эффектов: тень, блеск, прозрачность и т. п.

Для стилистически грамотного оформления документов удобно использовать готовые шаблоны — темы. Напомним, что **темой** называют специально разработанный набор элементов оформления и цветовых схем документа. С помощью темы можно задать стиль оформления как отдельной страницы, так и всего сайта. Темы позволяют задавать стили основного текста, заголовков, гиперссылок, списков; цвет фона; «обои»; цвет и толщину границ таблицы и т. п. Темы предлагаются во всех офисных программах, однако наибольшую пользу приносит их использование при создании презентаций и веб-сайтов.

Все страницы одного уровня будем оформлять в едином стиле.

Непосредственное конструирование страниц сайта начнем с подготовки всех его элементов (текстов, рисунков). Затем эти элементы и (или) ссылки на них будем вставлять в готовые шаблоны.

Файлы всех страниц веб-сайта будем сохранять в одной папке, например KINO. Это упростит размещение сайта на веб-сервере. В случае большого количества страниц или изображений их следует сохранять во вложенных папках, например KINO\FOTO\. Файлу главной веб-страницы обычно дают имя index или main с расширением .htm.

Только после размещения файлов в папках можно создавать гиперссылки, с помощью которых выполняется навигация по сайту — переходы со страницы на страницу. В нашем примере удобно сначала разработать страницы нижнего уровня, а затем оформить главную страницу и создать гиперссылки для вызова страниц нижнего уровня. Затем следует проверить работу ссылок в режиме оффлайн. Лишь после тщательной проверки и исправления ошибок созданный веб-сайт можно опубликовать, т. е. разместить на веб-сервере.



1. Какие этапы можно выделить при разработке веб-сайта?
2. Что понимают под стилем оформления веб-страницы?

Упражнение

Разработайте проект веб-сайта по одной из тем: Моя семья. Моя школа. Мои друзья. Мои любимые занятия. Моя Родина — Беларусь. Природа родного края. Интересные профессии.

§ 2. Использование офисных приложений для создания веб-страниц

2.1. Создание веб-страниц в редакторе MS Word

Любой документ MS Office можно сохранить в виде веб-страницы. Для этого достаточно выполнить команду **Файл → Сохранить как веб-страницу**. Однако далеко не всякий подготовленный для печати на бумаге документ будет хорошо выглядеть в браузере. Наилучшего результата можно достичь, если с самого начала готовить документ для размещения в сети Интернет.

Использование текстового редактора MS Word является одним из самых простых способов разработки веб-страниц. Рассмотрим на примере, как это делается. Начнем со страниц нижнего уровня, а затем оформим главную страницу и создадим гиперссылки.

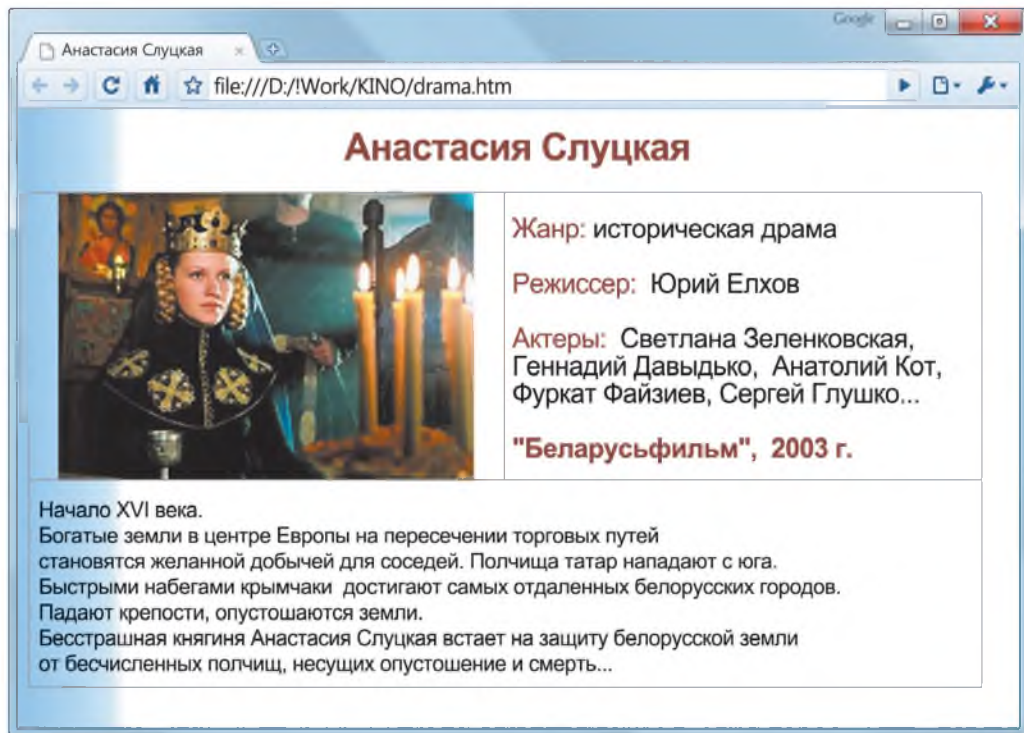


Рис. 1.3

Пример 1. Создать веб-страницу фильма «Анастасия Слуцкая» (рис. 1.3).

Прежде всего с помощью команды **Файл** → **Создать** выберем пункт **Веб-страница** (этот пункт можно выбрать и в области задач **Создание документа**). Затем, выполнив команду **Формат** → **Тема**, выберем для оформления страницы тему **Перетекание** (рис. 1.4).

Для размещения на странице фотографии и текстов используем таблицу из двух строк и двух столбцов. Объединим ячейки нижней строки. Вставим подготовленные заранее тексты и изображение из файлов. Оформим их в соответствии с рисунком 1.3.

Созданный документ сохраним как веб-страницу в папке KINO под именем drama.htm. Выберем тип сохраняемого файла **Веб-страница (*.htm; *.html)**. При этом все используемые в документе изображения будут помещены в отдельную папку с именем веб-страницы и расширением .files (в нашем примере — drama.files).

Правообладатель Народная асвета

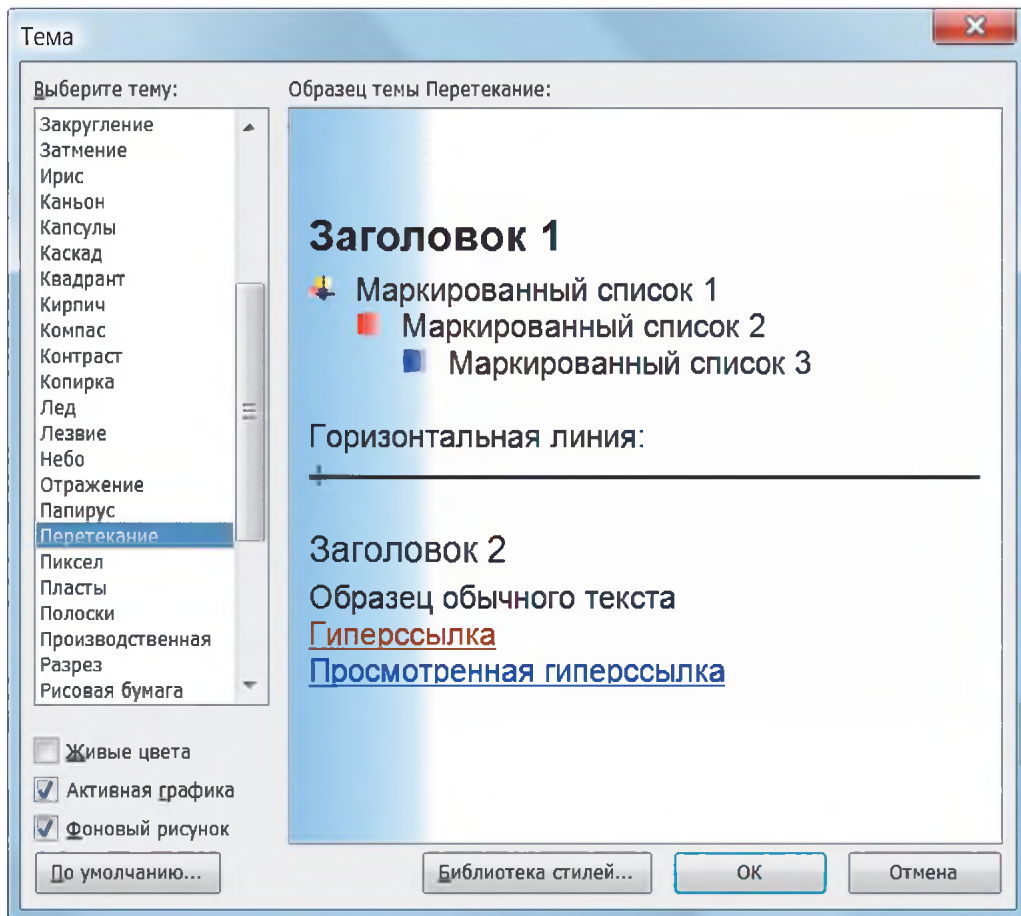


Рис. 1.4

Аналогичным способом создаются страницы других фильмов и жанров, например «Комедии» и «Детективы». Поскольку они имеют сходную структуру и отличаются лишь содержанием, то новые страницы можно получать редактированием уже созданных, заменяя изображения и тексты и сохраняя их под новыми именами.

Создадим веб-страницу фильма «В августе 44-го...» (рис. 1.5) на основе уже созданной нами страницы «Анастасия Слуцкая». Сохраним ее под именем avgust44.htm.

Пример 2. Создать главную страницу сайта кинотеатра «Беларусь» (рис. 1.6), содержащую гиперссылки на страницы фильмов.

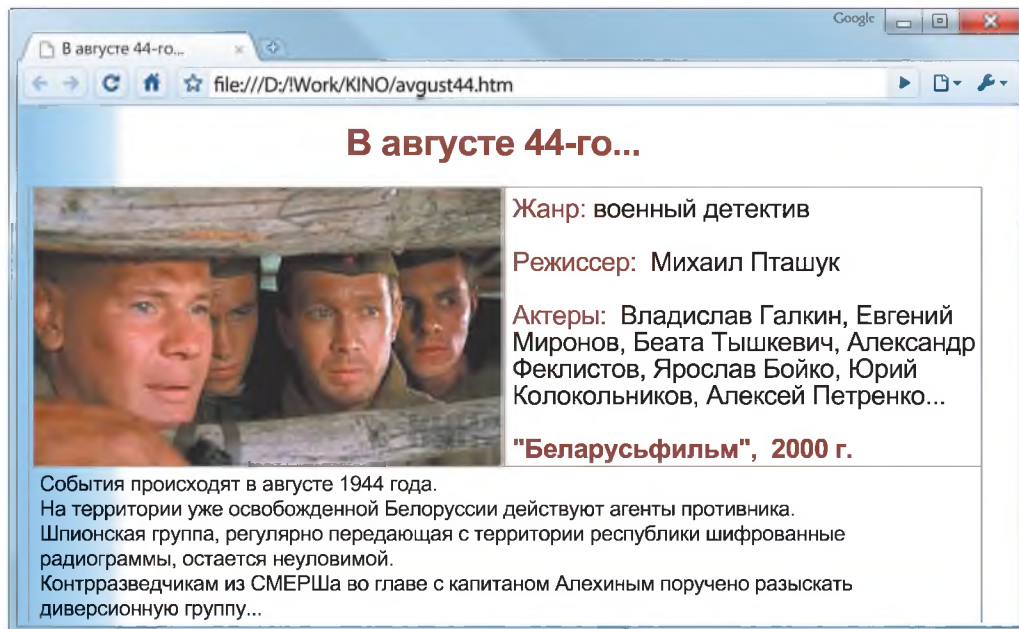


Рис. 1.5

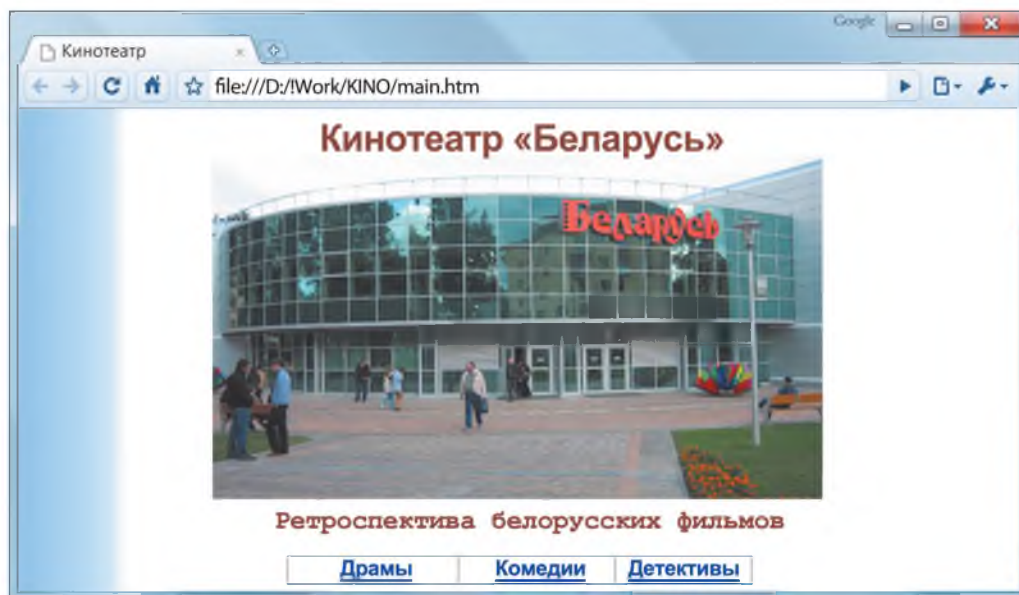



Рис. 1.6

Для главной страницы сайта используем тему **Перетекание**.

Важный этап разработки веб-страницы кинотеатра — создание гиперссылок, открывающих страницы фильмов. Для размещения гиперссылок создадим таблицу. Для создания текстовых гиперссылок выполним следующие действия:

- Выделим текст гиперссылки, например слово «Драмы».
- С помощью команды **Вставка** → **Гиперссылка** или кнопки  на панели инструментов выберем пункт **Связать с файлом, веб-страницей**, выделим имя файла drama.htm и подтвердим выбор нажатием кнопки ОК.

Гиперссылки в виде изображений создаются аналогично.

Завершив создание главной страницы, не забудем сохранить ее под именем main.htm в той же папке, в которой ранее сохраняли страницы фильмов.

По окончании работы просмотрим созданные страницы в браузере. Проверим работу ссылок, переходя по ним на страницы фильмов и возвращаясь на главную страницу с помощью кнопки **Назад** браузера.



Повысить привлекательность веб-страницы можно, разместив на ней динамические и интерактивные элементы (анимации, аудио- и видеофрагменты, формы опросов). При этом следует помнить, что анимации часто отвлекают пользователей от восприятия текстовой информации, а большие размеры аудиовидеофайлов существенно замедляют загрузку страниц.



Пример 3. Разместить на главной странице сайта кинотеатра «Беларусь» бегущую строку.




С помощью команды **Вид** → **Панели инструментов** активизируем панель **Веб-компоненты** (рис. 1.7). Наждем на этой панели кнопку .

Рис. 1.7



В открывшемся окне **Бегущая строка** (рис. 1.8) наберем требуемый текст, например «**Предлагает!**». Сделаем нужные установки: выберем направление движения **Справа налево** и нажмем кнопку ОК.

Сохраним измененную страницу.

Редактор MS Word позволяет подключать к веб-странице видео- и звуковые файлы. Для этого необходимо нажатием значка **Звук**  или **Фильм**  на панели **Веб-компоненты** открыть соответствующее диалоговое окно, нажать кнопку **Обзор**, найти требуемый видео- или звуковой файл, установить парамет-

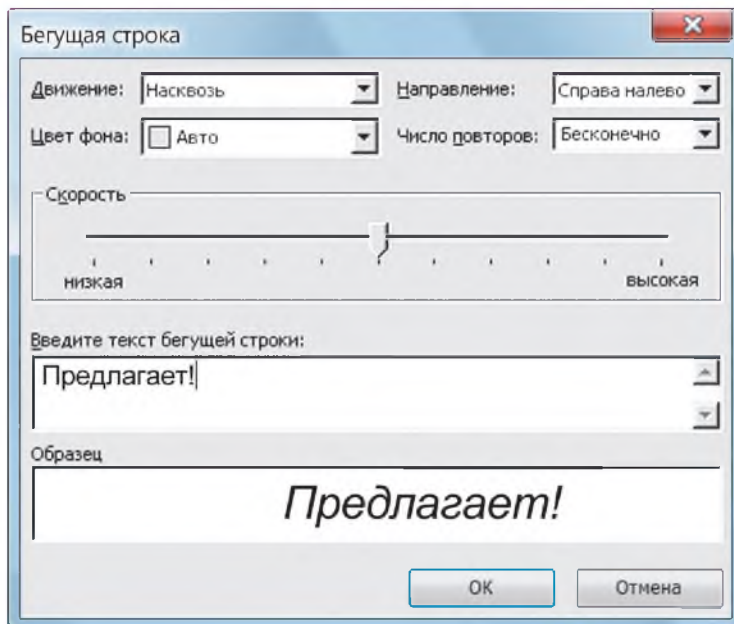


Рис. 1.8

ры проигрывания, например количество повторов, и подтвердить выбор нажатием кнопки ОК.



1. Как можно сохранить документ MS Word в виде веб-страницы?
2. С помощью каких действий можно создать гиперссылку?

Упражнение

Создайте фрагмент сайта по одной из тем: Моя Родина — Беларусь. Моя школа. Мои друзья. Мои любимые занятия. Моя семья (пример главной страницы представлен на рисунке). В качестве гиперссылок используйте картинки из коллекции клипов.



2.2. Сохранение презентаций PowerPoint в виде веб-страниц

Презентация PowerPoint, как правило, содержит несколько слайдов, каждый из которых может быть сохранен как отдельная веб-страница. Всю презентацию можно сохранить как сайт, структура которого будет соответствовать структуре презентации.

Пример 1. Создать фрагмент сайта на основе презентации «Функциональные блоки компьютера».

Откроем в редакторе PowerPoint презентацию «Функциональные блоки компьютера» (файл comp.ppt). Просмотрим ее структуру. Она состоит из 5 слайдов, которые содержат тексты и изображения (рис. 1.9).

Для сохранения презентации в виде веб-страницы в меню **Файл** выберем пункт **Сохранить как веб-страницу**. В появившемся окне **Сохранение документа** выберем тип сохраняемого файла **Веб-страница (*.htm, *.html)**, введем имя

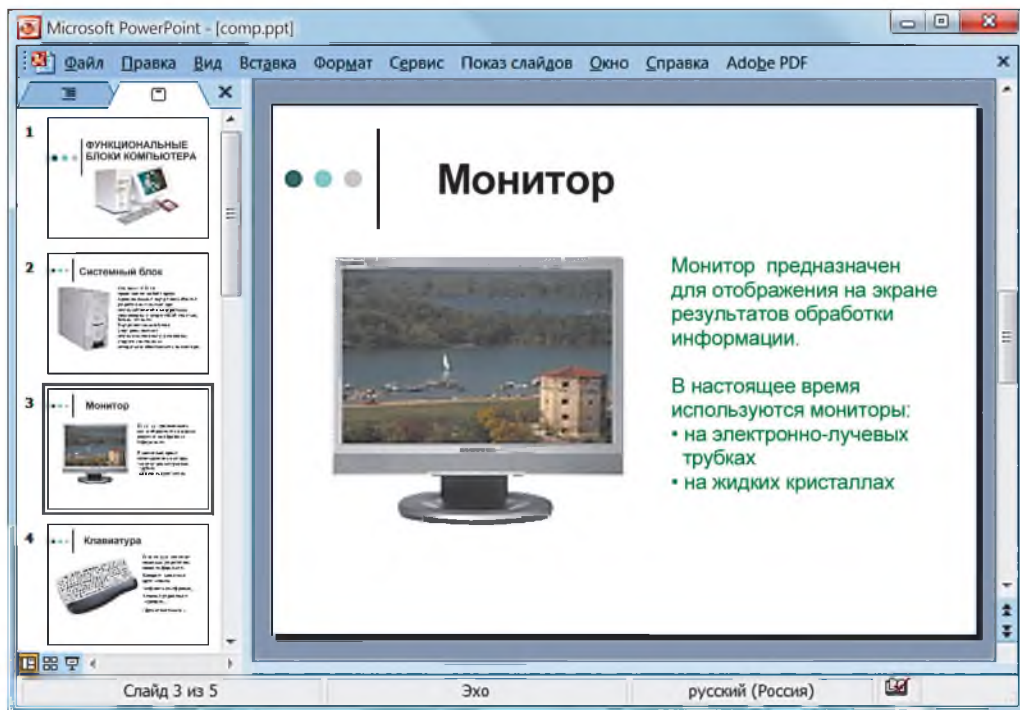


Рис. 1.9

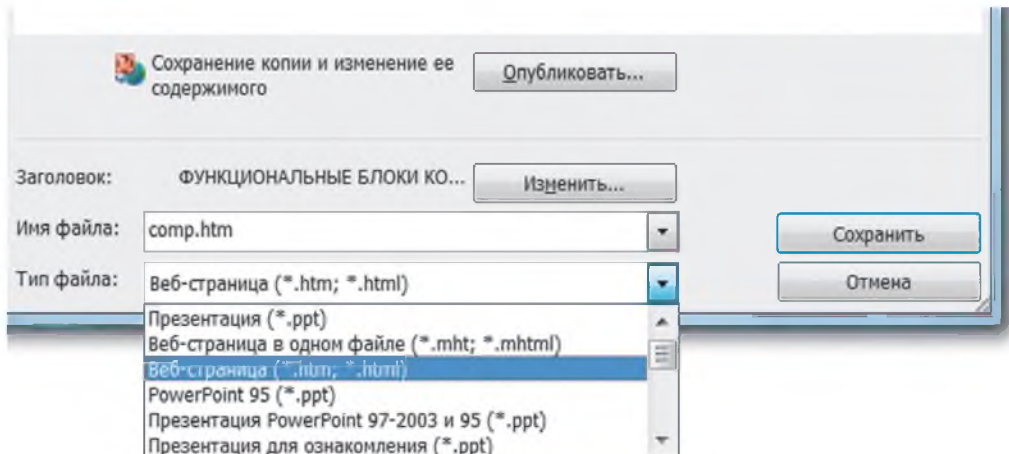


Рис. 1.10

файла или оставим прежнее (в нашем примере — comp.htm) и нажмем кнопку **Сохранить** (рис. 1.10).

При использовании предлагаемых по умолчанию настроек все слайды презентации сохранятся в папке comp.files в виде отдельных страниц, которые можно будет пролистывать в правой части окна браузера с помощью гиперссылок. Эти гиперссылки располагаются в левой части сохраненной страницы comp.htm (по умолчанию белого цвета на черном фоне).

Изменим эти настройки. С этой целью нажатием кнопки **Опубликовать** вызовем окно **Публикация веб-страницы** (рис. 1.11) и настроим параметры веб-документа. В этом окне можно выбрать публикуемые объекты (**Полная презентация** или **Слайды**), поддержку браузеров, а также изменить заголовки веб-страницы. По умолчанию как объект для публикации предлагается **Полная презентация**. В этом случае в виде веб-страниц будут сохранены все слайды.

Цветовые схемы, эффекты анимации и другие параметры настраиваются в окне **Параметры веб-документа** (рис. 1.12), которое вызывается кнопкой **Веб-параметры**.

На вкладке **Общие** выберем цветовую схему **Цвета презентации (выделение)**. В этом случае фон веб-страниц будет соответствовать фону слайдов презентации.

Установленный по умолчанию флажок **Добавить панель смены слайдов** следует снимать лишь в том случае, когда на слайдах предварительно установлены гиперссылки, указывающие переходы между слайдами.

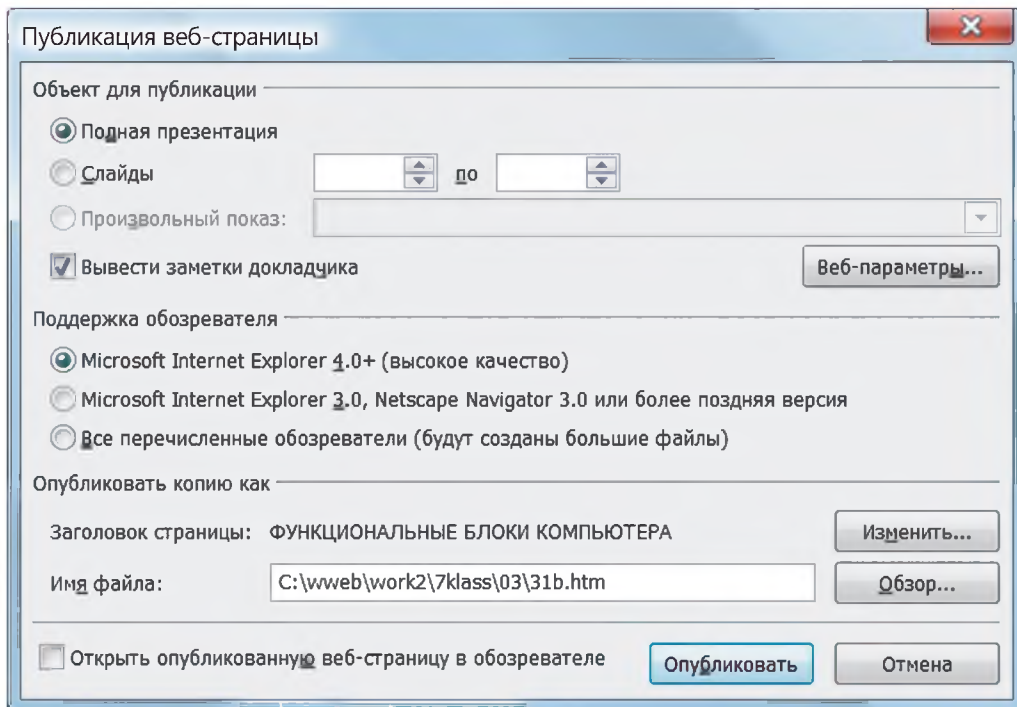


Рис. 1.11

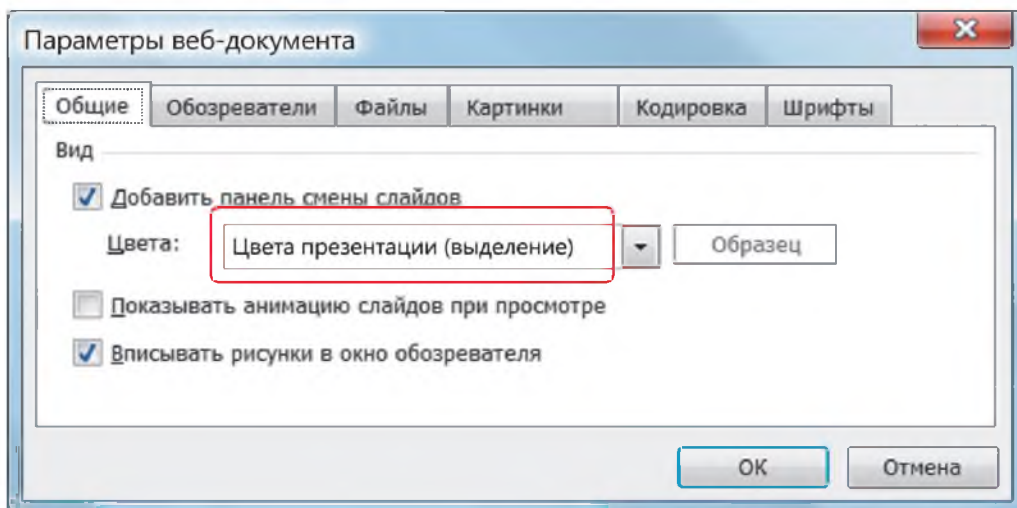


Рис. 1.12

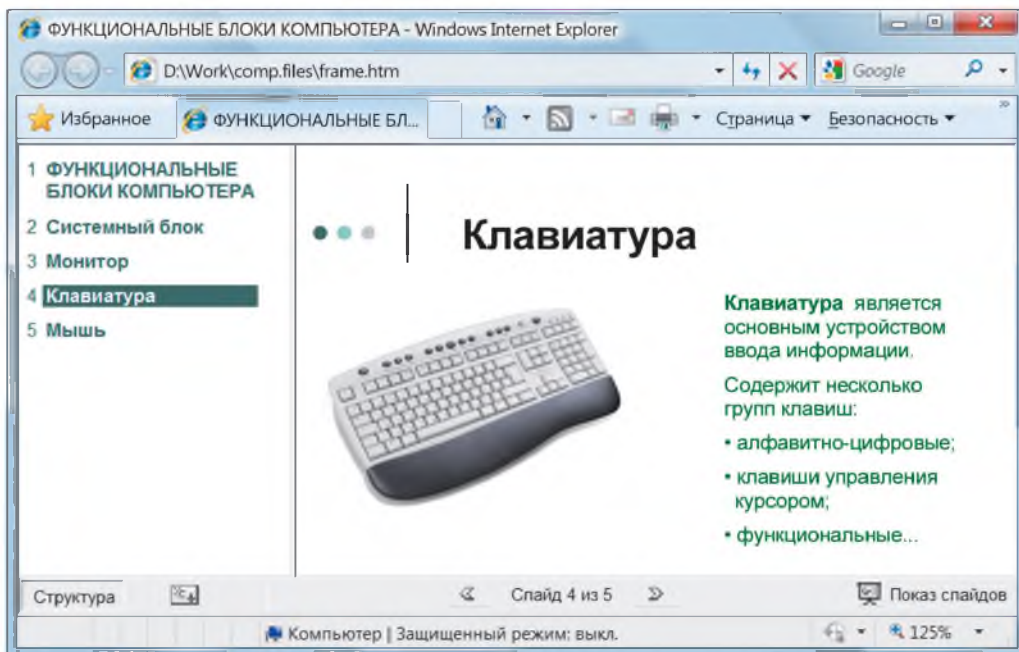


Рис. 1.13

Сохраним документ. Просмотрим сохраненные веб-страницы. Вид одной из них показан на рисунке 1.13.

В заключение отметим, что сохранение документов MS Office в виде веб-страниц является самым простым способом создания веб-сайтов и широко применяется в сфере образования для размещения в сети разнообразных материалов учебного назначения: сочинений, рефератов, докладов и презентаций. В профессиональном веб-конструировании эти инструменты и методы практически не используются из-за неоптимальности HTML-кода получаемых страниц, что затрудняет их редактирование и приводит к очень большим размерам файлов и медленной загрузке страниц.

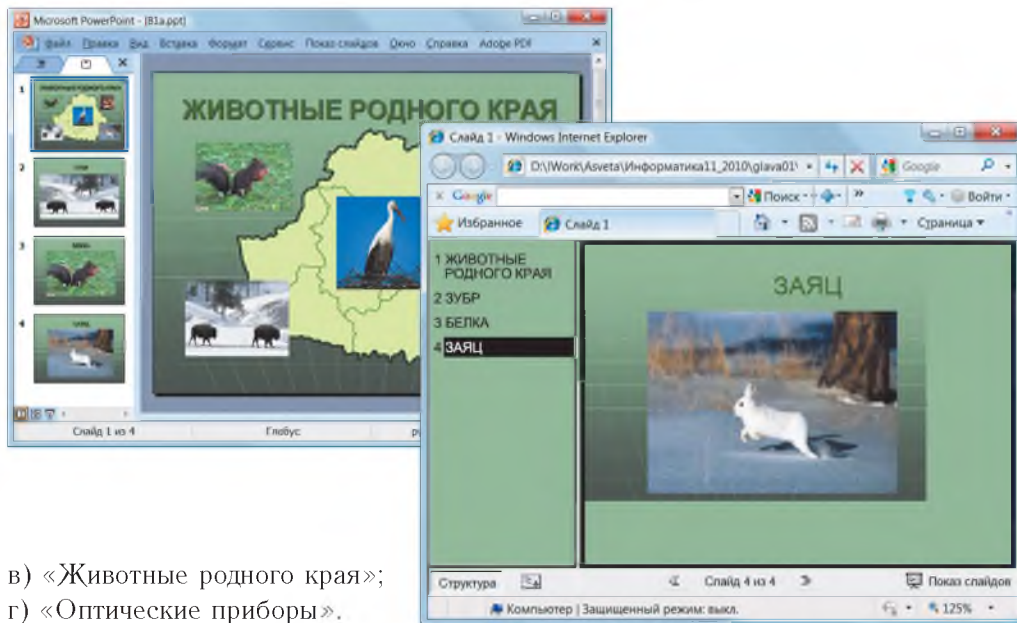


1. С помощью каких действий можно сохранить презентацию PowerPoint в виде веб-документа?
2. Какие параметры веб-страниц можно настраивать?

Упражнение

Создайте фрагмент сайта на основе готовой презентации:

- а) «Электрический ток»;
- б) «Пространственные фигуры»;



- в) «Животные родного края»;
г) «Оптические приборы».

§ 3. Основы языка разметки HTML

3.1. Создание HTML-документа в редакторе Блокнот

Веб-страница представляет собой текстовый документ, в котором расставлены команды языка HTML. Они интерпретируются браузером. Например, эти команды могут указывать, как должно отображаться содержание страницы на экране.

HTML-документ можно создавать в простейших текстовых редакторах, например в редакторе **Блокнот**, а затем сохранять в файле с расширением **.htm** или **.html**.

Разметка HTML-документа состоит в расстановке **тегов** — заключенных в угловые скобки команд языка HTML. Их можно набирать прописными или строчными латинскими буквами.

Краткое описание основных тегов приведено в Приложении 1.

Большинство тегов парные. Открывающим тегом описывается команда и начинается ее действие. Закрывающим тегом, который повторяет открывающий, но предваряется косой чертой / (слешем), это действие прекращается. Например, тег **** задает жирное начертание. Размеченный текст «Второе **слово**» выделено жирным шрифтом» браузером будет отображаться так:

Правосладелец Народная асвета

Второе **слово** выделено жирным шрифтом.

Рассмотрим структуру простейшего HTML-документа.

```
<html>
  <head>
    <title>Заголовок окна</title>
  </head>
  <body>
    Содержание документа
  </body>
</html>
```

HTML-документ начинается отрывающим тегом `<html>`, а заканчивается — закрывающим `</html>`. Внутри, как в контейнере, расположены два блока.

В блоке `<head>...</head>` (голова) размещается неотображаемая на странице служебная информация. Например, текст, заключенный между тегами `<title>...</title>`, отображается не на странице, а в заголовке окна браузера.

Вся отображаемая браузером информация: тексты, рисунки, видеофрагменты, анимации — размещается в блоке `<body>...</body>` (тело).

В HTML-документе можно также размещать неотображаемые браузером комментарии в угловых скобках с восклицательным знаком `<!--Комментарии-->`.

Пример 1. В редакторе **Блокнот** создать HTML-документ, содержащий текст:

Авторская страница
Это мой первый проект
Фамилия Имя Отчество

Сохранить созданный документ под именем `primer1.htm`. Просмотреть его в браузере.

Откроем редактор **Блокнот**. Наберем или скопируем готовый шаблон HTML-документа из файла `шаблон.htm`. Наполним его требуемым содержанием, т. е. введем заданный текст, как показано на рисунке 1.14.

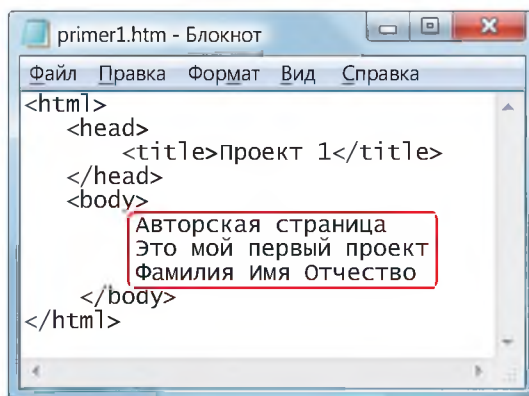


Рис. 1.14

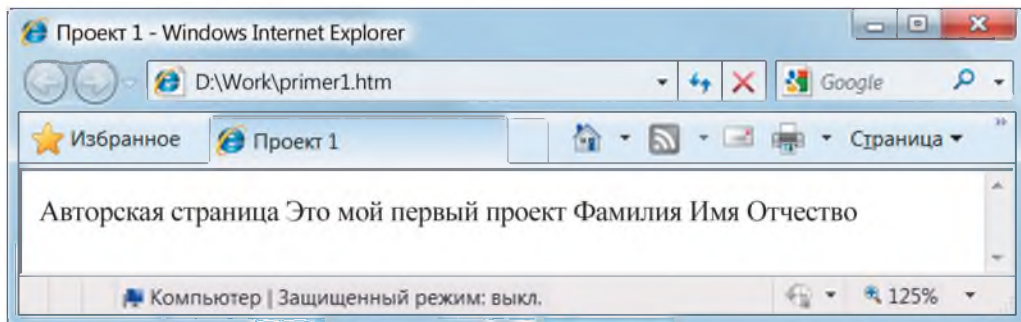


Рис. 1.15

Сохраним документ в формате HTML. Поскольку созданные в редакторе **Блокнот** документы по умолчанию сохраняются с расширением `.txt`, воспользуемся меню **Файл** → **Сохранить как**, в появившемся диалоговом окне выберем вариант **Все файлы**, введем имя и расширение: `primer1.htm`.

Откроем созданный документ в браузере (рис. 1.15). Обратим внимание, что текст отображается в одну строку, хотя набран в три строки с отступами. Переходы на новые строки, а также все пробелы более одного браузером игнорируются.

Итак, вид веб-страницы в браузере задается тегами HTML, но также зависит от типа браузера. В приведенных примерах используется браузер MS Explorer.

Для отображения текста в виде отдельных абзацев используют тег `<p>`. При просмотре в браузере абзацы отделяются друг от друга пустой строкой. Для принудительного перехода на новую строку без создания абзаца используют непарный тег `
`. Нередко между абзацами помещают разделительную линию, которая задается непарным тегом `<hr>`.

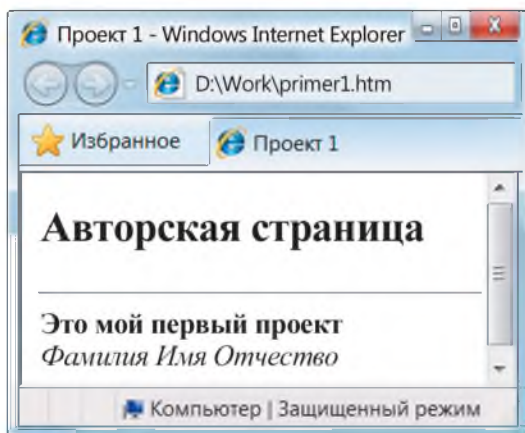


Рис. 1.16

За оформление текста отвечают теги форматирования (см. Приложение 1). Для отображения заголовков используются теги `<h1>...<h6>`. Заголовков уровня 1 — самый крупный, а уровня 6 — самый мелкий.

Начертание символов задается следующими тегами: `` — **полужирное**, `<i>` — *курсивное*, `<u>` — подчеркнутое.

Пример 2. Оформить созданную в примере 1 страницу в соответствии с рисунком 1.16.

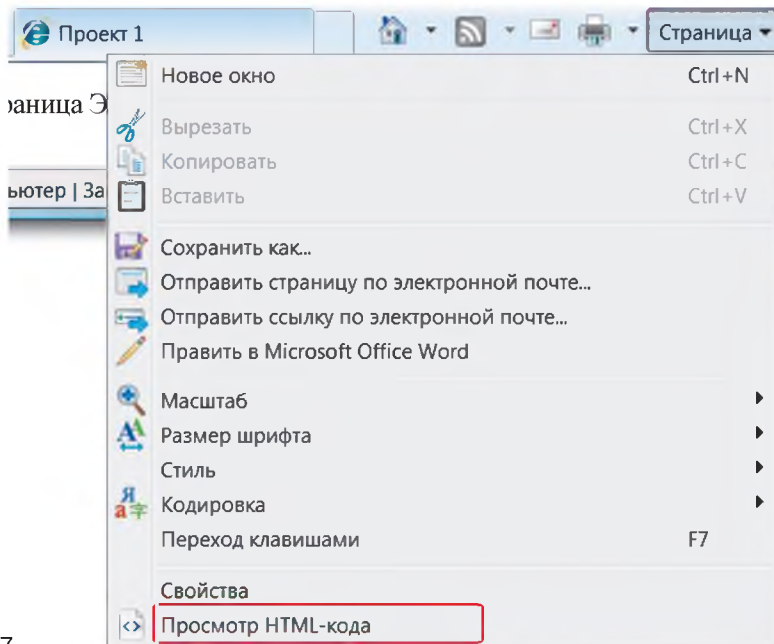


Рис. 1.17

Отредактируем HTML-документ в редакторе **Блокнот**. Его можно открыть непосредственно из браузера с помощью команды **Страница** (или **Вид**) → **Просмотр HTML-кода** (рис. 1.17).

Расставим теги:

```
<h2>Авторская страница</h2><hr>                <!--Заголовок уровня 2-->
<b>Это мой первый проект</b> <br>                <!--Жирный шрифт-->
<i>Фамилия Имя Отчество</i>                <!--Курсив-->
```

Сохраним документ в файле с именем avtor.htm. Для просмотра обновленной страницы нажмем кнопку **Обновить**  или клавишу F5.

Открывающие теги языка HTML могут содержать **атрибуты**, которыми задаются параметры разметки документа. Каждый атрибут имеет название (имя) и определенное значение, которое записывается в кавычках после знака «равно». Тег может иметь несколько атрибутов, которые перечисляют через пробелы, например:

```
<тег атрибут1="значение" атрибут2="значение"...>.
```

Порядок следования атрибутов в теге неважен. Если значение атрибута содержит только буквы английского алфавита, цифры или дефис и не содержит пробелов, кавычки можно опустить.

Вид всей веб-страницы задается атрибутами тега `<body>`, в то время как отдельные ее элементы, например заголовки, разделы, абзацы, таблицы, рисунки, могут иметь свое особенное оформление.

Цвет фона страницы задается атрибутом `bgcolor`, а цвет текста — атрибутом `text`. Значением этих атрибутов является цвет, который задается своим названием на английском языке, например `red` (красный), или его шестнадцатеричным кодом, например `#FF0000` (Приложение 2). По умолчанию цвет страницы белый, а цвет текста — черный.

Синий цвет страницы можно задать так: `<body bgcolor="blue">` или так: `<body bgcolor="#0000FF">`.

Заметим, что цвета, отображаемые разными браузерами, а также выводимые на печать, могут несколько отличаться от приведенных в Приложении 2.

Для выравнивания текста всей страницы, отдельного абзаца, раздела или заголовка используется атрибут `align`, который может принимать следующие значения: `center` — выравнивание по центру, `left` — по левому краю, `right` — по правому краю. Выравнивание текста по ширине использовать не рекомендуется, поскольку при уменьшении окна браузера между словами могут образовываться большие промежутки.

Для задания шрифта, цвета и размера символов текста используется тег ``. Шрифт задается атрибутом `face`, значением которого является название шрифта, например `Arial`. Цвет символов задается атрибутом `color`. Если шрифт не задан, то по умолчанию используется шрифт `Times` черного цвета.

Размер символов задается атрибутом `size` и может выражаться в условных единицах, которые могут принимать значения от 1 до 6. По умолчанию принято значение размера 3.

Размер символов может выражаться и в относительных единицах: числах по отношению к базовому размеру, например `size="+n"` или `size="-n"`. Так, для базового размера 3 атрибут `` задает размер 5, а атрибут `` — размер 1.

Пример 3. Создать веб-страницу в соответствии с рисунком 1.18.

Откроем в редакторе **Блокнот** файл `primer3.txt` с текстом объявления и введем HTML-код:

```
<html>
<head>
  <title>Объявление</title>
</head>
```

Правообладатель Народная асвета

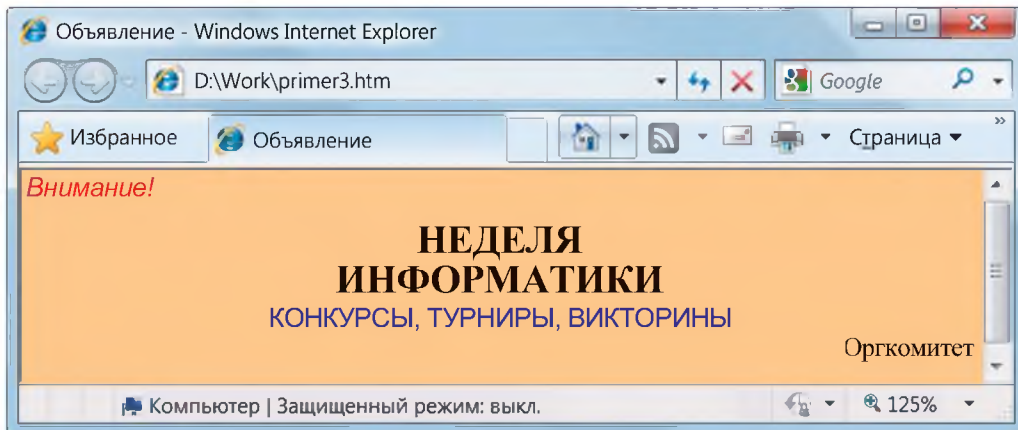


Рис. 1.18

```
<body bgcolor="gold">
  <font face="Arial" size="+2" color="red"><i>Внимание!
  </i></font>
  <h1 align="center">НЕДЕЛЯ<br>ИНФОРМАТИКИ</h1>
  <font face="Arial" size="+2" color="blue">
  КОНКУРСЫ, ТУРНИРЫ, ВИКТОРИНЫ</font>
  <p align="right">Оргкомитет</p>
</body>
</html>
```

Заметим, что допускается комбинирование и вложенность тегов. Так, для отображения слова «Внимание!» шрифтом Arial и курсивным начертанием в тег `<font...>` вложен тег `<i>`.

Сохраним документ в файле с именем Primer3.htm. Просмотрим его в браузере.



Фрагмент текста может отображаться в виде бегущей строки с помощью тега `<marquee>`. Направление движения указывается атрибутом `direction`. Например, движение слова «Внимание!» слева направо задается так:

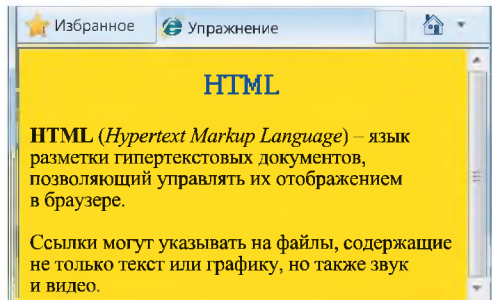
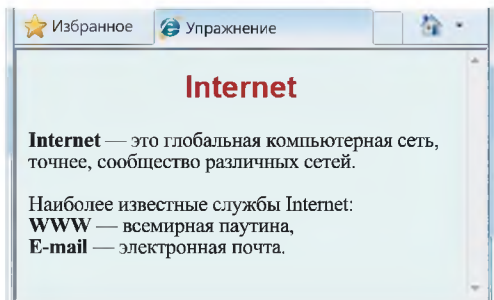
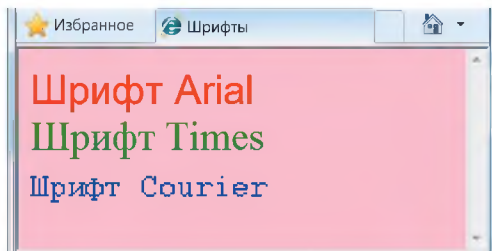
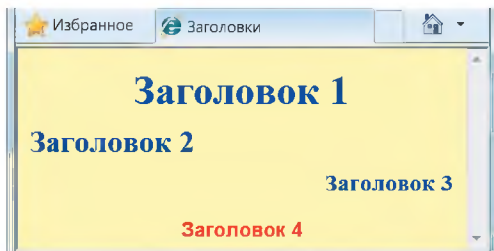
```
<marquee direction="right">Внимание!</marquee>
```



1. Каким тегом задается абзац? Переход на новую строку?
2. Какими тегами задается полужирное и курсивное начертание символов?
3. Как задать фон и цвет текста всей веб-страницы?
4. Какими тегами и атрибутами задается шрифт, размер и цвет символов?

Упражнение

Откройте предложенный учителем текстовый документ в редакторе **Блокнот**. Оформите и сохраните его в виде веб-страницы, изображенной на одном из рисунков. Для задания цвета фона воспользуйтесь Приложением 2.



3.2. Изображения на веб-страницах

Важную роль в оформлении веб-страниц играют графические объекты: фотографии, рисунки, фоновые изображения («обои»), разделительные линии. Они делают страницы более привлекательными, а во многих случаях являются и основными источниками информации.

Предназначенные для веб-страниц изображения можно создавать и редактировать в различных графических редакторах. Они могут быть получены с помощью цифрового фотоаппарата, сканера или взяты из сети Интернет. Наиболее предпочтительны графические форматы GIF, JPG и PNG, файлы которых имеют небольшие размеры, что позволяет ускорить процесс загрузки. В формате **JPG** сохраняют полутоновые изображения, например фотографии. Формат **GIF** поддерживает анимацию и прозрачный цвет.

Важно подчеркнуть, что все изображения, которые мы видим на веб-странице, хранятся в отдельных файлах, а в HTML-коде имеются лишь ссылки на них.

Для отображения рисунков предназначен непарный тег ``. Его неотъемлемым атрибутом является имя графического файла ``. Веб-страницы и файлы с изображениями могут храниться в разных папках, тогда в атрибуте `src` тега `` необходимо указывать путь. Указание путей обязательно для ссылок на все объекты, например на другие страницы, аудио- и видеофрагменты.

Тег `` может содержать и другие атрибуты, которые определяют способ отображения рисунка.

Высоту и ширину отображаемого на экране рисунка в пикселях можно задать значениями атрибутов `height` и `width` соответственно, а толщину рамки вокруг изображения — значением атрибута `border`. По умолчанию рамки нет, т. е. `border="0"`. Указание размеров позволяет увеличить или уменьшить видимое на экране изображение (при этом исходный рисунок и файл, в котором он хранится, остаются неизменными). Если размеры не заданы, то по мере загрузки рисунка может изменяться расположение текста и других объектов на экране. Чтобы этого не происходило, полезно всегда указывать эти атрибуты, даже если изображение не масштабируется.

Значения атрибута `align` задают выравнивание текста относительно рисунка (`top` — по верхнему краю; `middle` — по середине; `bottom` — по нижнему краю) или способ обтекания рисунка текстом (`left` — рисунок слева от текста; `right` — рисунок справа от текста).

С помощью атрибута `alt` можно задать текст сообщения, которое будет выводиться вместо рисунка, если он не найден или если отключен его показ в браузере. Кроме того, этот текст появляется в виде подсказки при подведении курсора мыши к рисунку.

В качестве фона страницы можно использовать изображение из файла. Фоновый рисунок («обои») задается с помощью атрибута `background` в теге `<body>`. Чтобы повторяющийся фоновый рисунок заполнял страницу без стыков, его верхняя и нижняя, левая и правая стороны не должны отличаться. Задать фоновый рисунок и одновременно закрасить его некоторым цветом нельзя.

Пример 1. Разместить на веб-странице изображения из файлов в соответствии с рисунком 1.19.

В редакторе **Блокнот** откроем файл `graf.txt` с шаблоном страницы.

В теге `<body>` укажем рисунок фона `kletka1.gif`. Расставим теги форматирования текста. Вставим теги для отображения рисунка волка с прозрачным фоном из файла `volk.gif` (без атрибутов); анимации из файла `tv.gif` (граница толщиной 2 пикселя); двух фотографий аиста из файла `aist.jpg` разных размеров.

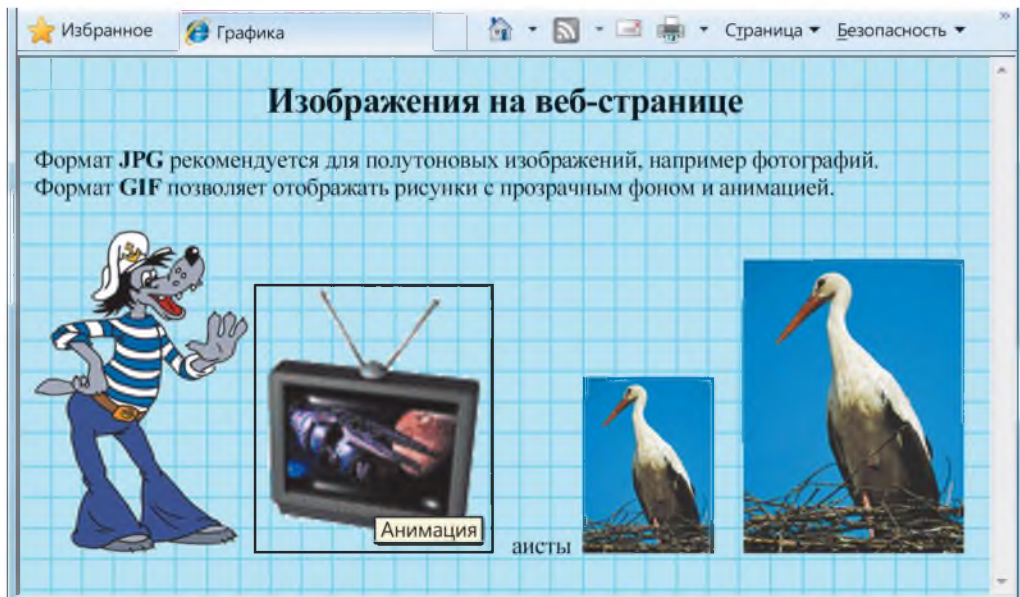


Рис. 1.19

HTML-код веб-страницы может иметь следующий вид:

```
<html>
<head><title>Графика</title></head>
  <body background="kletka1.gif">
    <h2 align="center">Изображения на веб-странице</h2>
    Формат <b>JPG</b> рекомендуется для полутоновых изобра-
    жений, например фотографий.<br>
    Формат <b>GIF</b> позволяет отображать рисунки с проз-
    рачным фоном и анимацией.<p>
    
    
    аисты 
    
  </body>
</html>
```

Сохраним HTML-документ в файле с именем graf.htm. Просмотрим его в браузере. При наведении курсора на область рисунка tv.gif появляется подсказка «Анимация».

Правообладатель Народная асвета

Чтобы отображать текст на экране в том же виде, что и в редакторе **Блокнот**, используют тег `<pre>`. Это позволяет упростить форматирование, например не вставлять теги `
` в конце каждой строки.

Пример 2. Создать веб-страницу «Мой родны кут» (рис. 1.20).



Рис. 1.20

В редакторе **Блокнот** откроем файл `kut.txt` с текстом отрывка из поэмы Якуба Коласа «Новая зямля». Добавим теги начала и завершения веб-страницы. Расставим теги форматирования. Заключим текст стихотворения в теги `<pre>...</pre>`.

Ниже текста вставим фотографию музея Якуба Коласа из файла `muzkolas.jpg`. В качестве фона используем изображение листьев из файла `bg02.jpg`.

HTML-код этой веб-страницы может иметь следующий вид:

```
<html>
  <head><title>Мой родны кут</title>
</head>
  <body background="bg02.jpg">
    <h2 align="center">Мой родны кут</h2>
    <i>Якуб Колас. 3 паэмы "Новая зямля"</i>
```

```
<pre>Мой родны кут, як ты мне мілы,  
Забыць цябе не маю сілы!  
Не раз, утомлены дарогай,  
Жыццём вясны мае убогай,  
К табе я ў думках залятаю,  
І там душою спачываю.</pre>  
  
</body>  
</html>
```

Сохраним HTML-документ в файле с именем kut.htm.

От удачного выбора цвета фона или фонового рисунка, шрифта и цвета текста, а также других элементов оформления страницы зависит восприятие ее содержания. В Интернете свободно распространяется громадное количество «обоев» и готовых шаблонов веб-страниц разного назначения (например, на сайте «Азбука дизайнера»). Создание собственных веб-страниц на их основе сводится в основном к вводу текста и ссылок на нужные изображения.



Весьма популярны, например, шаблоны поздравительных открыток и рекламных страниц. Такой шаблон несложно разработать самостоятельно.



Пример 3. Создать веб-страницу — шаблон поздравительной открытки (рис. 1.21).

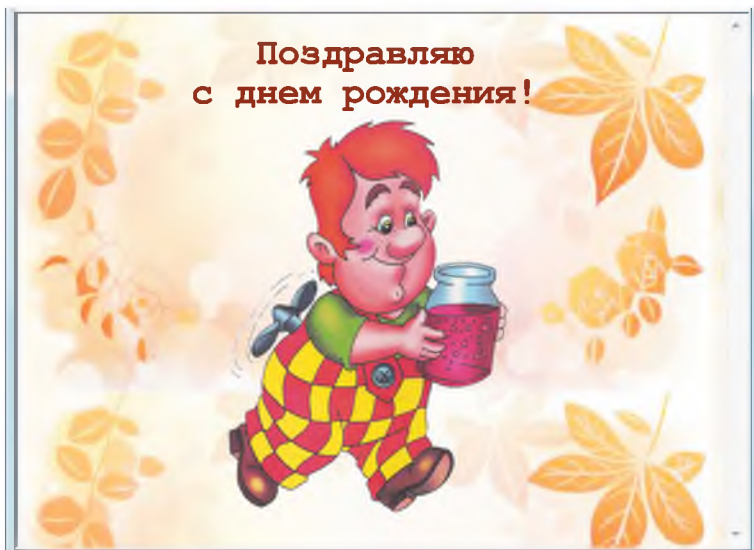


Рис. 1.21

HTML-код веб-страницы может иметь следующий вид:

```
<html>
  <head><title>Открытка</title>
</head>
  <body background="bg003b.jpg" text=maroon>
    <font face="courier">
      <h1 align="center">Поздравляю<br>с днем рождения!<br>
      </h1>
    </body>
</html>
```

На основе этого шаблона можно создавать различные открытки, изменяя текст поздравления, «обои» и рисунок.



1. Какой тег отображает рисунок на веб-странице?
2. Как задаются размеры изображения?
3. Какой тег задает фоновый рисунок на веб-странице?

Упражнения

1. Разместите на одной странице несколько изображений:

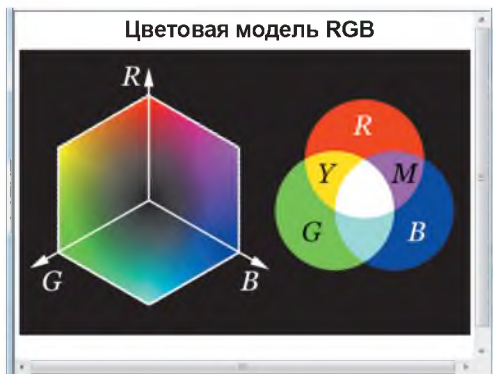
а) разного размера (например, 150×120, 100×80);



б) в рамках разной толщины (например, 0, 1, 3, 8).



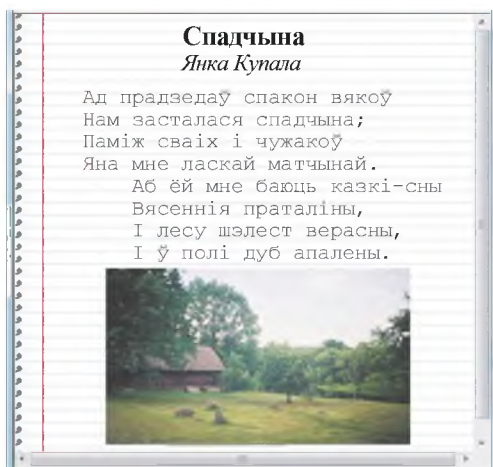
Правообладатель Народная асвета



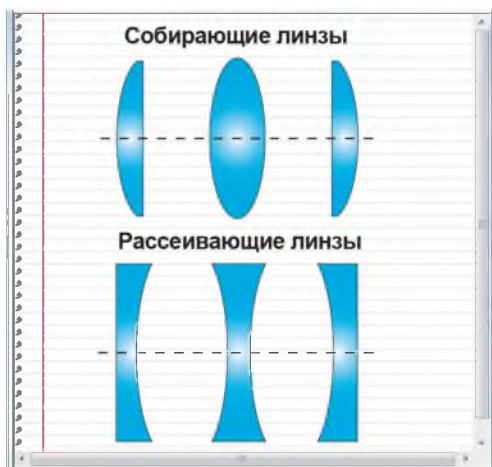
а



б



в



г

2. Откройте предложенный учителем текстовый документ в редакторе **Блокнот**. Оформите и сохраните его в виде веб-страницы, изображенной на одном из рисунков.

3.3. Создание гиперссылок

Переходы со страницы на страницу веб-сайта выполняются с помощью гиперссылок. За организацию ссылок на языке HTML отвечает тег `<a>` с обязательным атрибутом `href`, значением которого является адрес (URL) ресурса. Например, HTML-код текстовой гиперссылки на страницу «Графика», которая сохранена в файле `graf.htm` в той же папке, что и исходная страница, имеет вид:

`Графика`, т. е. адресом ресурса является имя файла.

Обычно близкие по тематике страницы сайта сохраняют в соответствующих папках. Пусть, например, на диске D: в папке `кино` хранят главную страницу `main.htm`, во вложенной папке `dramy` — страницу фильма `drama2.htm`, а в папке `multy` — страницы `karlson.htm` и `ну погоди.htm` (рис. 1.22). В этом случае при создании гиперссылок необходимо указывать пути к файлам. Возможны два варианта:

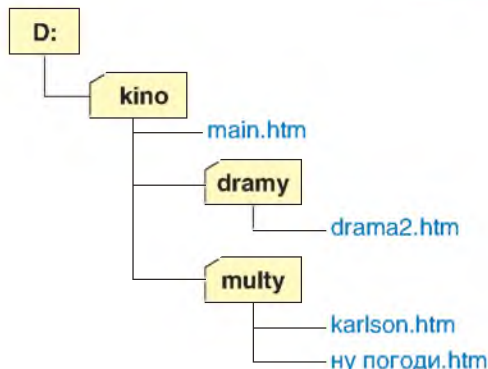


Рис. 1.22

- записать *полный путь* от корня диска к искомой странице;
- указать *относительный адрес* (путь от исходной страницы к искомой). Например, ссылка со страницы `main.htm` на страницу `karlson.htm` будет такой:

```
<a href="multy/karlson.htm">Карлсон</a>.
```

Ссылка со страницы `drama2.htm` (папка `dramy`) на страницу `karlson.htm` (папка `multy`) будет иметь вид:

```
<a href="../../multy/karlson.htm">Карлсон</a>.
```

Символы `../` (две точки и слеш) обозначают возврат в папку `кино` из вложенной папки `dramy`.

При переносе папки `кино` со всеми вложениями в другое место, в том числе на другой диск или компьютер, относительные адреса не изменяются!

Гиперссылкой может служить не только текст, но и рисунок. Для создания такой гиперссылки между парой тегов `<a href >...` нужно вставить тег, указывающий на файл изображения, например:

```
<a href="graf.htm"></a>.
```

Возможно совместное использование текста и графики в одной ссылке, например:

```
<a href="graf.htm">Графика</a>.
```

По умолчанию вызванная гиперссылкой страница открывается в том же окне, что и исходная, замещая ее. Назад можно вернуться с помощью соответствующей кнопки на панели инструментов браузера. Для просмотра вызванной страницы в новом окне необходимо указать атрибут `target="_blank"`, например:

```
<a href="graf.htm" target="blank">Графика</a>.
```

Гиперссылка может указывать не только на веб-страницу, но и на любой размещенный в сети по известному адресу информационный ресурс: рисунок, текст, презентацию, аудиовидеофрагмент. Конечно, на компьютере пользователя должно быть установлено соответствующее программное обеспечение для его отображения.

Пусть, например, реферат по астрономии luna.doc размещен в папке astro на сайте www.referaty.by. Гиперссылка для его просмотра или скачивания может быть задана тегами:

```
<a href="www.referaty.by/astro/luna.doc">Луна</a>.
```

Еще раз подчеркнем, что расставлять гиперссылки следует только после создания ресурсов, на которые они указывают, и размещения их файлов в заданных папках.

Пример 1. Создать главную страницу веб-сайта «Мои работы» (рис. 1.23). Разместить на ней гиперссылки на страницы «Графика», «Об авторе» (открываются в том же окне), а также на рисунок (открывается в новом окне).

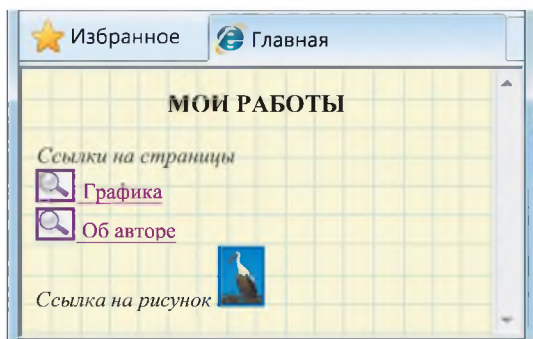


Рис. 1.23

Пусть фрагмент сайта будет размещен в папке PRO. Скопируем в эту папку созданные ранее файлы веб-страниц «Графика» graf.htm и «Об авторе» avtor.htm вместе со всеми используемыми рисунками.

Оформим главную страницу в соответствии с рисунком 1.23. Расставим гиперссылки. HTML-код этой веб-страницы может иметь следующий вид:

```
<html>
<head><title>Главная</title></head>
<body background="kletka2.gif">
  <h3 align="center">МОИ РАБОТЫ</h3>
  <i>Ссылки на страницы</i><br>
  <a href="graf.htm">Графика</a><br>
  <a href="avtor.htm">Об авторе</a><br>
  <i>Ссылка на рисунок</i>
  <a href="aist.jpg" target="_blank">
    </a>
</body>
</html>
```

Сохраним HTML-документ в файле с именем index.htm.

Проверим работу гиперссылок. Страницы «Графика» и «Об авторе» открываются в текущем окне после щелчка мышью по тексту ссылки или картинке с символическим изображением лупы (файл icon.gif). Рисунок (файл aist.jpg) открывается в новом окне после щелчка мышью по его уменьшенному изображению (из того же файла aist.jpg).

Заметим, что для ускорения загрузки страницы с большим количеством графических ссылок следует использовать миниатюрные изображения из отдельных файлов.

В рассмотренном примере все файлы хранятся в одной папке. В случае большого количества страниц или изображений их следует размещать во вложенных папках и указывать пути к ним.



Гиперссылки могут указывать не только на отдельные документы, но и на заданные места в документе. Ссылки внутри длинных страниц повышают удобство их просмотра. Они создаются в два этапа:

1) сначала в месте, куда нужно перейти по ссылке, устанавливается закладка, например:

```
<a name="Закладка1">...</a>;
```

2) затем создается ссылка на эту закладку:

```
<a href="#Закладка1"> текст гиперссылки </a>.
```

Для ссылки на адрес электронной почты его нужно указать в качестве URL, например

```
<a href="mailto:sova@les.by">письмо сове</a>.
```

Как уже отмечалось, в блоке <head> могут размещаться теги, которые содержат служебную информацию для браузера и поисковых систем. Они называются **мета-тегами**. Большинство мета-тегов являются необязательными. Однако некоторые из них весьма полезны.

Так, браузер будет отображать текст страницы на национальном языке при указании таблицы кодировки в мета-теге charset. Например, для кириллицы мета-тег будет таким:

```
<meta http-equiv="content-type" content="text/html;  
charset=Windows-1251">
```

Для хранения информации об авторе используют мета-тег

```
<meta name="author" content="Алесь Палескі">.
```



По словам, перечисленным через запятую или пробел в мета-теге "keywords" (ключевые слова), поисковые системы индексируют веб-страницы, т. е. помещают информацию о них в свою базу данных, чтобы потом оперативно выдавать по запросам пользователей. Длина строки ключевых слов не должна превышать 800 символов. Слова в ней не должны повторяться, например:

```
<meta name="keywords" content="Образование Мультимедиа  
Multimedia Графика Видео Аудио">.
```



1. С помощью какого тега создается гиперссылка?
2. Как в качестве гиперссылки использовать изображение?

Упражнение

Создайте фрагмент веб-сайта из 3—4 веб-страниц по одной из тем: Дом, в котором я живу. Мой класс. Моя школа. Любимые стихи (песни, книги, фильмы). Любимые поэты (писатели, художники, артисты). Любимые занятия (пример главной страницы представлен на рисунке).



§ 4. Подготовка изображений для Интернета

При создании веб-сайта приходится заботиться о том, чтобы его страницы загружались как можно быстрее. Это особенно актуально, если на страницах сайта необходимо разместить много изображений. Поэтому так важен грамотный выбор формата их хранения.

Как Вы знаете, нарисованные в графическом редакторе Paint картинки по умолчанию сохраняются в формате BMP (*Bit Map Picture* — битовая карта изображения). В этом формате изображения хранятся в файлах в упакованном виде. Информационный объем изображения в байтах равен произведению его ширины **W** (width) и высоты **H** (height) в пикселях и глубины цвета **C** в битах на 1 пиксель (**W·H·C**). Так, файл с небольшой картинкой размером 100×100 пикселей при глубине цвета 24 бит/пиксель будет иметь размер около 29 Кбайт. Объем файла с изображением 800×600 пикселей будет иметь размер около 1,4 Мбайт, а время его загрузки при скорости модема 56 кбит/с составит более 200 с, что совершенно неприемлемо.

В большинстве графических форматов изображения хранятся в файлах в упакованном виде. Существуют различные способы уменьшения размера файла изображения с минимальными потерями качества. Этот процесс называют **оптимизацией**.

При большом разнообразии графических форматов для размещения на сайтах в основном используются изображения форматов GIF, JPG и PNG.

Выбор формата хранения изображения во многом зависит от свойств самого изображения. Так, формат **GIF** используют для изображений с четкими линиями, однородными заливками, текстом, например чертежей, карт. Такие изображения называют *штриховыми*. Этот формат поддерживает палитру оттенков, состоящую не более чем из 256 цветов, и позволяет задавать прозрачный фон. Оптимизация заключается в уменьшении количества выбранных цветов. Важным достоинством формата GIF является возможность анимации. Поэтому этот формат широко используется для размещения на веб-страницах простых графических элементов: символов, формул, логотипов, рекламных роликов.

Формат **JPG** позволяет передавать богатую палитру цветов и лучше всего подходит для изображений с плавным переходом тонов без четких линий и контуров, например портретов, пейзажей. Такие изображения называют *полутонными*. Оптимизация заключается в замене областей с небольшим градиентом цвета на однотонные. Степень сжатия характеризуют показателем качества от 0 до 100. Часто по умолчанию его принимают равным 65, что соответствует сжатию исходного упакованного изображения в 6—10 раз. Сохранение изображений в формате JPG предусмотрено практически во всех современных цифровых фотоаппаратах.

При подготовке изображений для Интернета выработаны определенные рекомендации. Размеры сохраняемых в файлах изображений должны соответствовать размерам изображений, отображаемых на веб-страницах, несмотря на то, что их ширину и высоту можно задавать в теге ``. Например, для мониторов 17" с разрешением 1280×768 пикселей ширина изображения на странице чаще всего не превышает 600—800 пикселей, а высота — 400—600. Если изображение обтекает текстом, то его размер должен находиться в пределах 200—400 пикселей.

Процесс оптимизации изображения сводится к двум основным операциям:

- приведение изображения к требуемому размеру;
- сжатие изображения с сохранением оптимального качества.

Пример 1. Фотография размером 2560×1920 пикселей сохранена в файле объемом 2,4 Мбайт. Ее ширину и высоту уменьшили в 4 раза. Определить объем файла и размеры изображения.

Размер изображения станет равным 640×480 пикселей, при этом объем файла уменьшится в $4 \cdot 4 = 16$ раз и составит 0,15 Мбайт. В 16 раз уменьшится и время загрузки этого изображения!

Заметим, что изменять размеры изображений и сохранять их в форматах GIF или JPG можно практически в любом растровом графическом редакторе, в том числе в Paint. Для сжатия изображений требуется более сложный редактор.

Рассмотрим оптимизацию изображений в свободно распространяемом графическом редакторе Paint.Net (<http://paintnet.ru>).

Пример 2. Фотография размером 1200×1200 пикселей сохранена в файле goza1200.jpg объемом 745 Кбайт. Оптимизировать изображение так, чтобы на веб-странице его размер составил 300×300 пикселей.

Откроем в редакторе Paint.Net изображение из файла goza1200.jpg (рис. 1.24).

Прежде всего, уменьшим размер изображения. Для этого из меню **Изображение** вызовем окно **Изменение размера** и установим значение **Ширина: 300** пикселей (рис. 1.25). При включенном флажке **Сохранять пропорции** в такой же пропорции изменится и высота.

Сохраним уменьшенное изображение в формате JPG. Для этого с помощью меню **Файл** вызовем диалоговое окно **Сохранить как**, выберем тип файла JPEG и введем новое имя файла, например goza300.jpg. После нажатия кнопки **Сохранить** появится диалоговое окно настройки качества изображения **Параметры сохранения**. По умолчанию показатель **Качество** нашего изображения равен 100. Размер файла составляет примерно 83 Кбайт.

Продолжим оптимизацию путем сжатия изображения с сохранением приемлемого качества. С помощью движка будем уменьшать показатель **Качество** до появления видимых на глаз искажений на оптимизированном изображении. Так, при по-

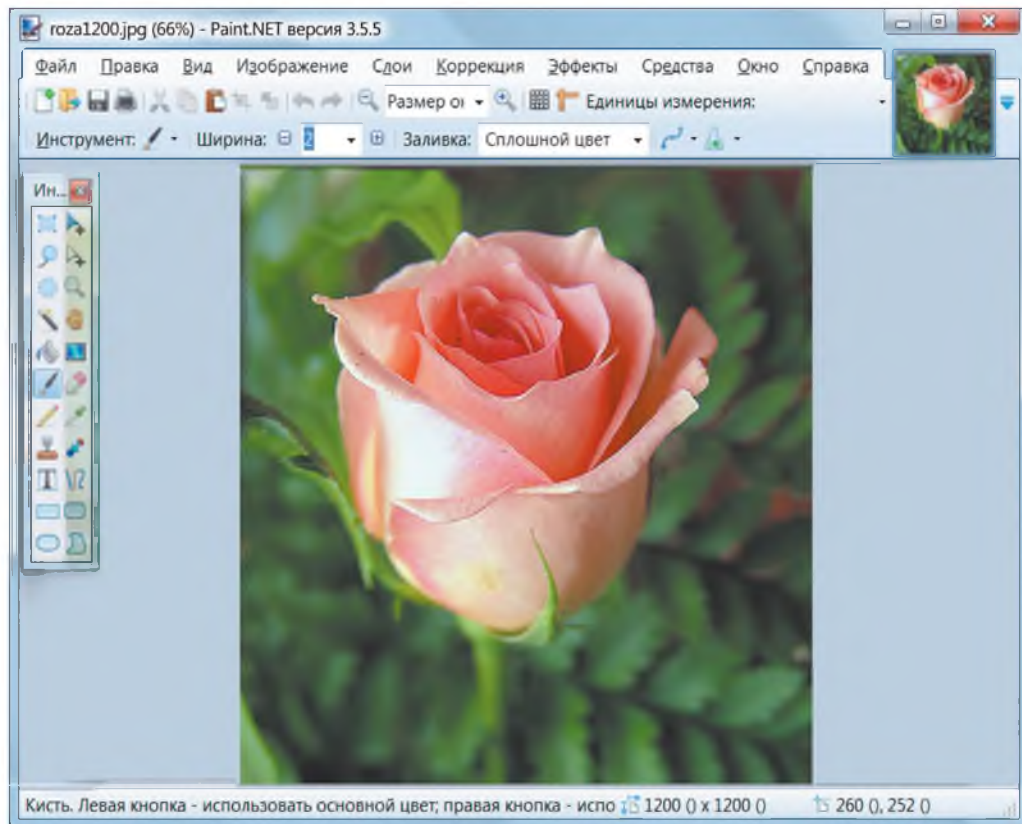


Рис. 1.24

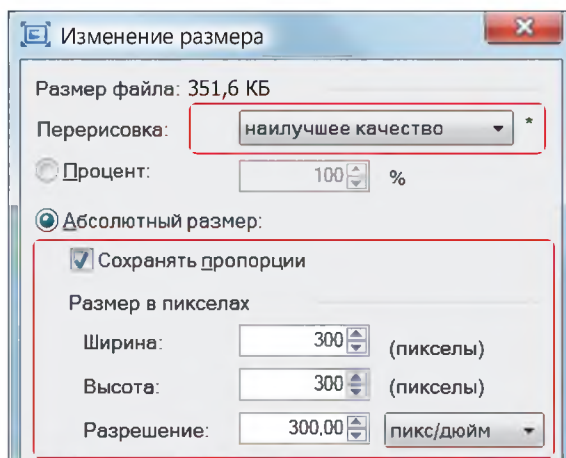


Рис. 1.25

казателе 50 видимых изменений еще нет, а размер файла при этом составляет около 19 Кбайт (рис. 1.26). Остановимся на этом значении, нажав кнопку ОК. В итоге качество изображения ухудшилось незначительно, а размер файла с оптимизированным изображением стал почти в 40 раз меньше исходного.



Гораздо большие возможности оптимизации изображений предоставляет графический редактор Adobe Photoshop.

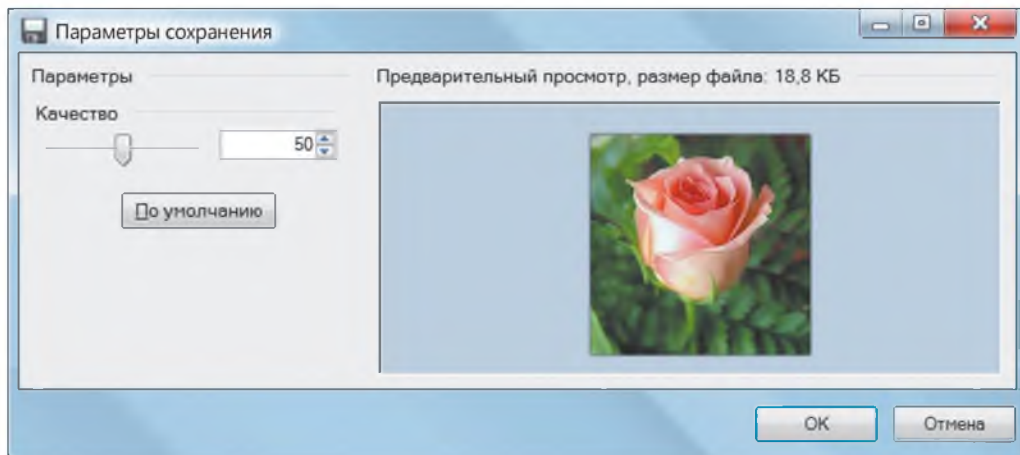


Рис. 1.26

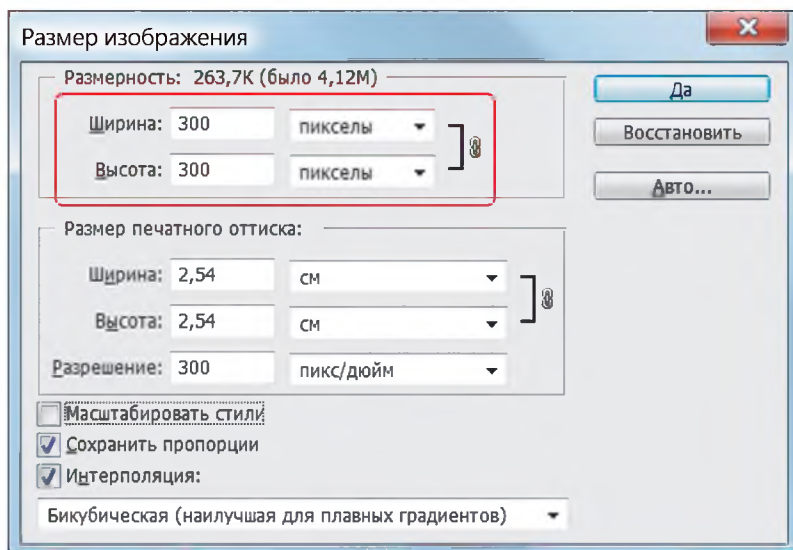


Рис. 1.27



Пример 3. Оптимизировать изображение из файла roza1200.jpg так, чтобы на веб-странице его размер составил 300×300 пикселей.

Откроем в редакторе Adobe Photoshop изображение из файла roza1200.jpg.

Для изменения размера изображения из меню **Изображение (Image)** вызовем окно **Размер изображения (Image Size)** и установим значение **Ширина (Width)** 300 пикселей (рис. 1.27). При включенном флажке **Сохранить пропорции** в такой же пропорции изменится и высота.

Из меню **Файл (File)** вызовем окно **Сохранить для веб и устройств (Save for Web)** и с помощью вкладки **4 варианта** получим четыре изображения: исходное (слева вверху) и оптимизированные с качеством 65 % (справа вверху), 30 % и 10 % (внизу) (рис. 1.28).

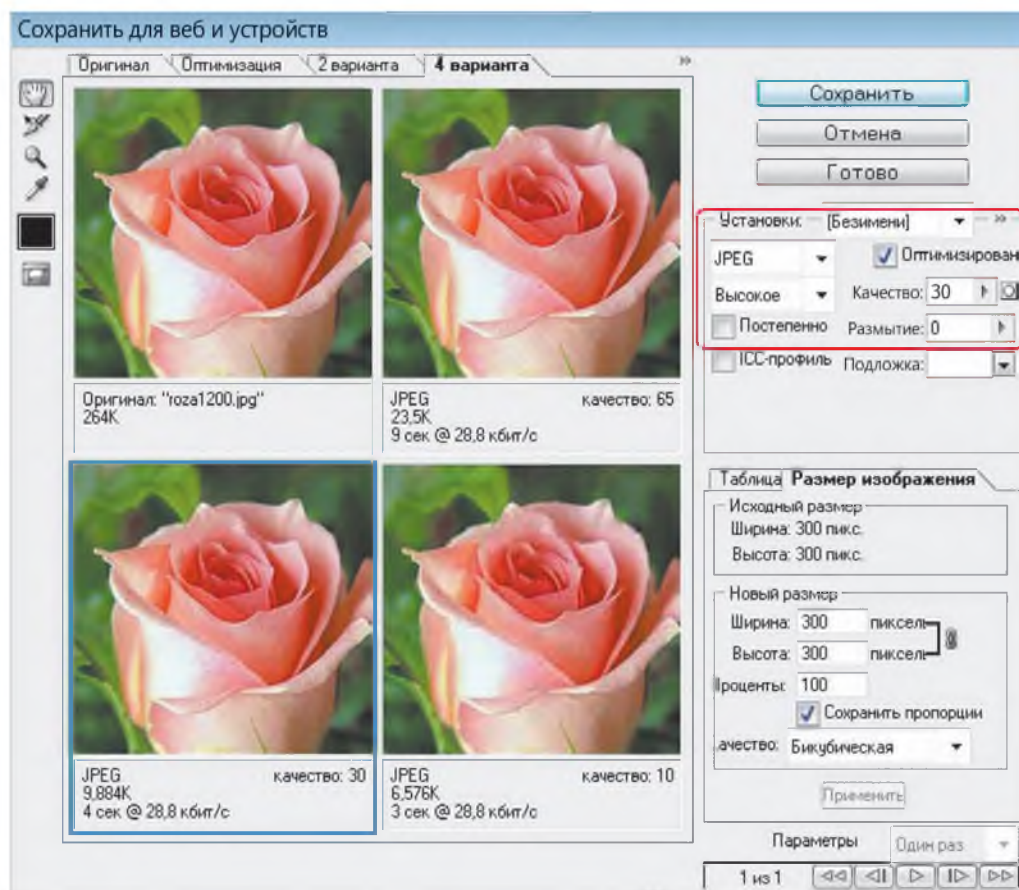


Рис. 1.28

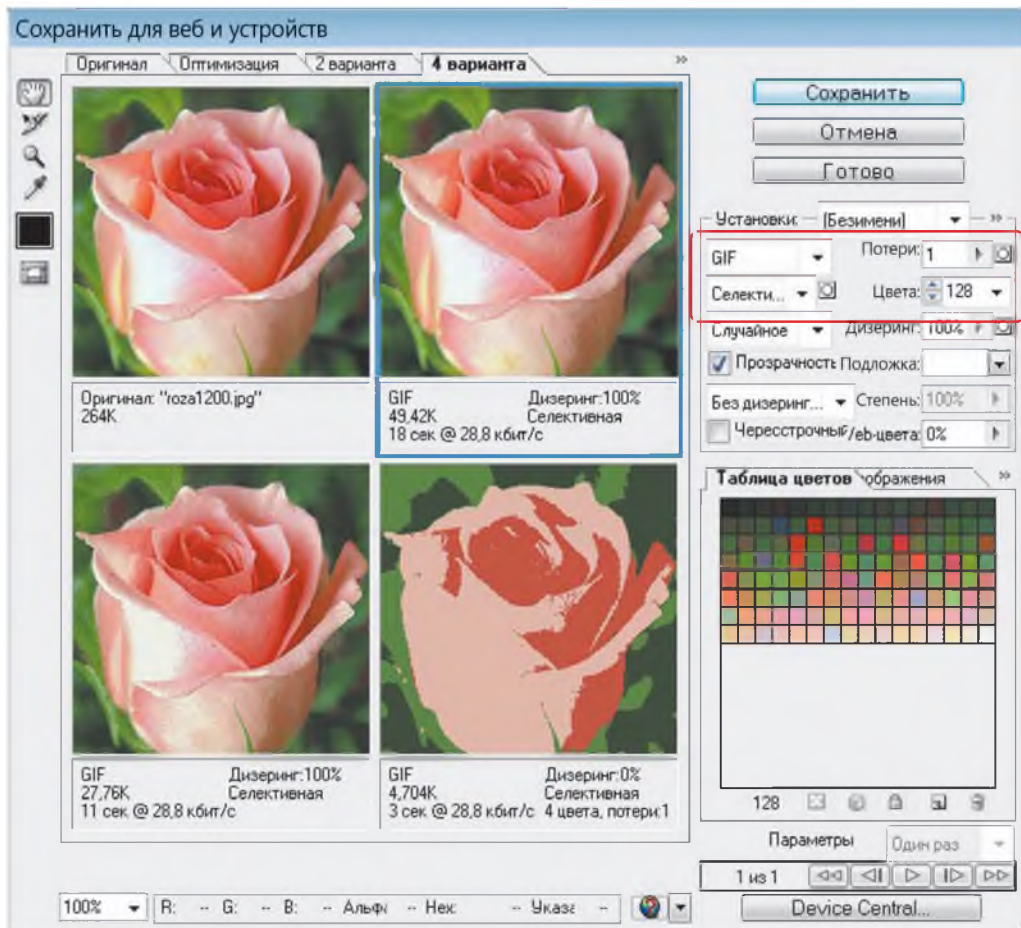


Рис. 1.29

Будем уменьшать показатель **Качество** до появления видимых изменений на оптимизированном изображении. Это можно сделать грубо, выбрав в поле **Установки (Setting)** один из показателей качества (**Максимальное, Высокое, Среднее, Низкое**), и точно, установив подходящее числовое значение. Остановимся, например, на значении 30 и нажмем кнопку **Сохранить (Save)**.

Объем оптимизированного файла не превышает 10 Кбайт, т. е. уменьшился почти в 75 раз. Не забудем ввести новое имя файла, например roza300-30.jpg.

Аналогичным способом оптимизируют изображения в формате GIF. В этом случае уменьшают количество цветов от 256 до значения, при котором появляют-

ся видимые искажения. На рисунке 1.29 показаны четыре изображения: исходное (слева вверху) и оптимизированные с количеством цветов 128 (справа вверху), 32 (слева внизу, потери качества невелики) и 4 (справа внизу, качество неприемлемо).



1. Почему необходимо уменьшать объемы файлов с изображениями для веб-страниц?
2. В каких форматах рекомендуется хранить изображения для веб-страниц?
3. В чем заключается оптимизация изображений для Интернета?

Упражнения

1. Оптимизируйте изображения, сохраненные в предложенных учителем файлах.
2. С помощью клавиши Print Screen поместите в буфер обмена изображение на экране, например Рабочий стол. Откройте редактор Paint, с помощью комбинации клавиш Ctrl + V вставьте изображение из буфера и сохраните под именем ris.bmp. Сохраните это же изображение в форматах JPG и GIF. Сравните объемы файлов.

§ 5. Веб-конструирование в редакторе FrontPage

5.1. Основные элементы интерфейса

Вы уже научились создавать веб-страницы в программах MS Office, а также с использованием языка разметки HTML в простейшем текстовом редакторе **Блокнот**. Рассмотрим теперь возможности специального веб-редактора Microsoft FrontPage, который предназначен для разработки веб-сайтов и относится к программным средствам визуального веб-конструирования. При работе с веб-редактором FrontPage можно обойтись без знания языка разметки гипертекстовых документов HTML. Веб-страница просто конструируется на экране и сохраняется в формате HTML. Выполняемые при этом действия по оформлению веб-документа напоминают работу в текстовом редакторе MS Word.

Интерфейс редактора FrontPage достаточно прост, чтобы пользователь смог быстро освоить основные приемы работы. После запуска FrontPage открывается окно, основные элементы которого представлены на рисунке 1.30.

Меню и панели инструментов **Стандартная**, **Форматирование**, **Рисование** по своим возможностям и приемам использования напоминают аналогичные панели текстового редактора MS Word.

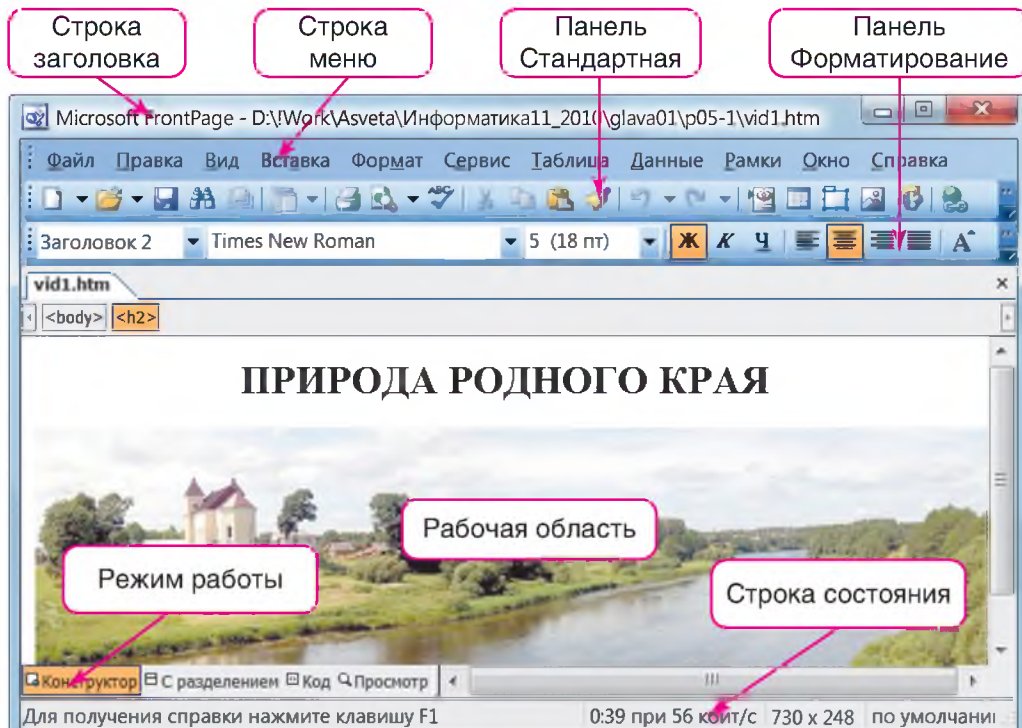


Рис. 1.30

Вид **Рабочей области** зависит от выбранного режима работы.

В режиме **Конструктор** веб-страница строится из текстовых блоков и графических объектов. При этом автоматически генерируется ее HTML-код, который можно просматривать и редактировать в режиме **Код**.

Режим **С разделением** (рис. 1.31) является комбинацией этих двух режимов.

Режим **Просмотр** позволяет просматривать созданные страницы.

С помощью меню **Вид** можно выбрать и другие полезные режимы работы.

Режим **Страница** предназначен для создания и редактирования веб-страниц.

В режиме **Папки** просматривается структура папок сайта, выполняются файловые операции.

Режим **Переходы** отображает структуру связей между страницами и позволяет ее изменять.

Режим **Гиперссылки** наглядно отображает систему ссылок, обеспечивает их проверку и редактирование.

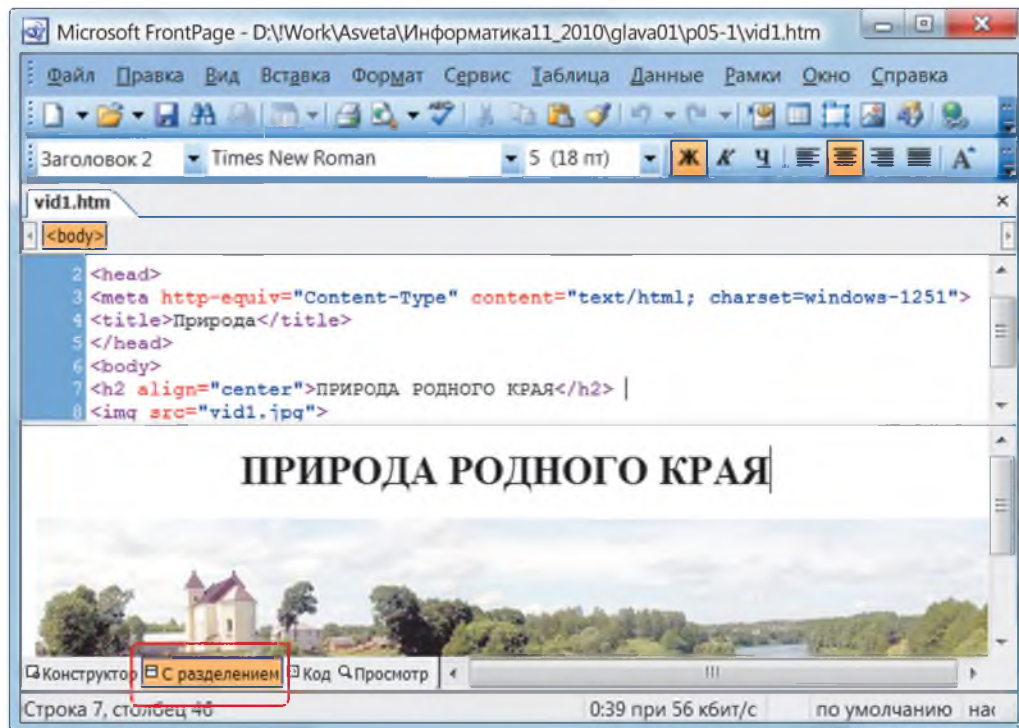


Рис. 1.31

5.2. Работа в редакторе FrontPage

Рассмотрим основные приемы работы в редакторе FrontPage на примере разработки веб-страницы «Озера Беларуси». Она будет содержать текст и фотографии, которые следует подготовить заранее.

Начнем с создания и оформления текстового документа.

Пример 1. Создать веб-страницу «Озера Беларуси» (рис. 1.32).

Запустим редактор FrontPage. Откроется пустая страница с установленными по умолчанию параметрами.

Перед началом работы полезно проверить настройки редактора, и, прежде всего, кодировку. Для этого с помощью команды **Файл → Свойства** откроем окно **Свойства страницы** и на вкладке **Язык** установим **кириллица** (рис. 1.33).

Будем работать в режиме **Конструктор**. Однако при необходимости можем просматривать и редактировать автоматически генерируемый HTML-код в режимах **Код** или **С разделением**.

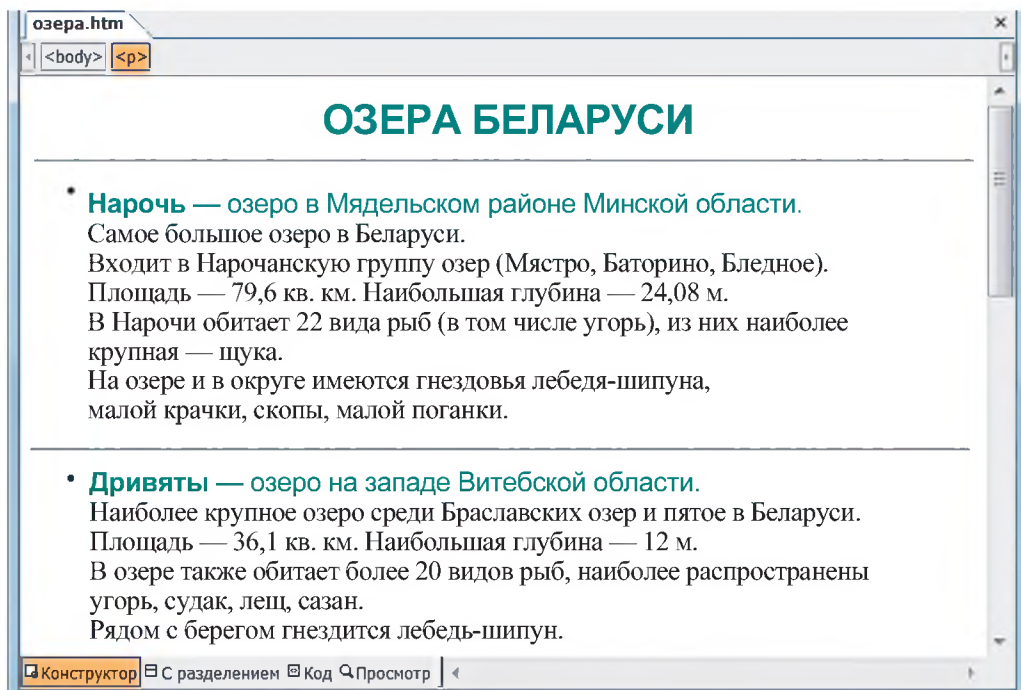


Рис. 1.32

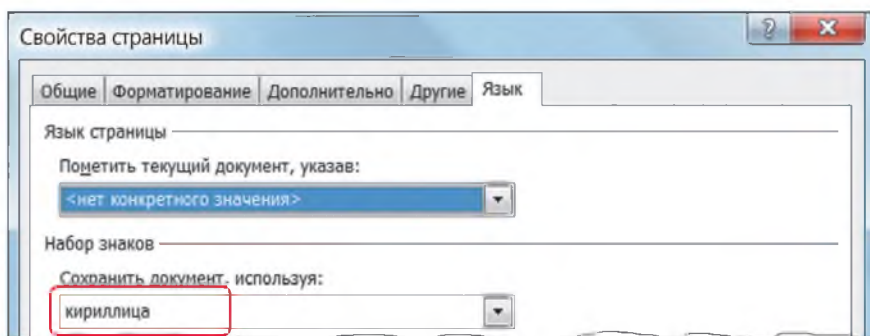


Рис. 1.33

Начнем с размещения и оформления текста, который был предварительно сохранен в файле `озера.txt`. Откроем его в редакторе **Блокнот**, скопируем нужные фрагменты и с помощью буфера обмена вставим на создаваемую страницу. Текст будет оформлен принятым по умолчанию стилем: шрифт Times, размер 3 (12 пт), выравнивание по левому краю (рис. 1.34).

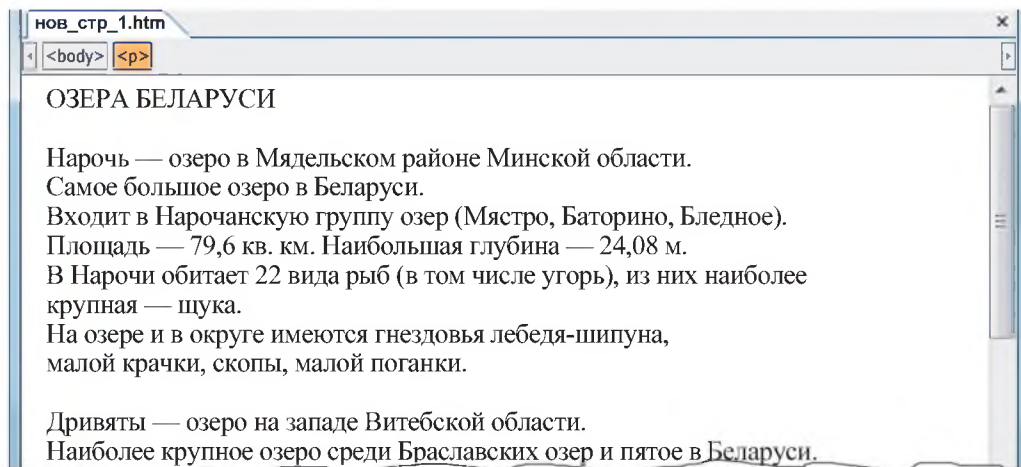


Рис. 1.34

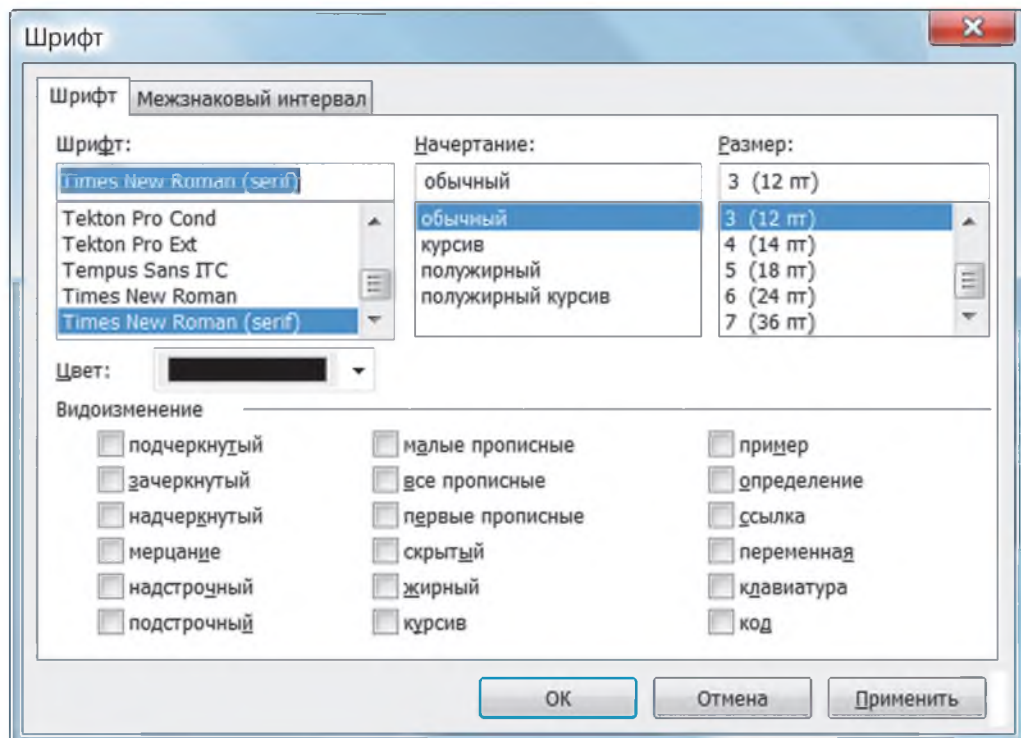


Рис. 1.35

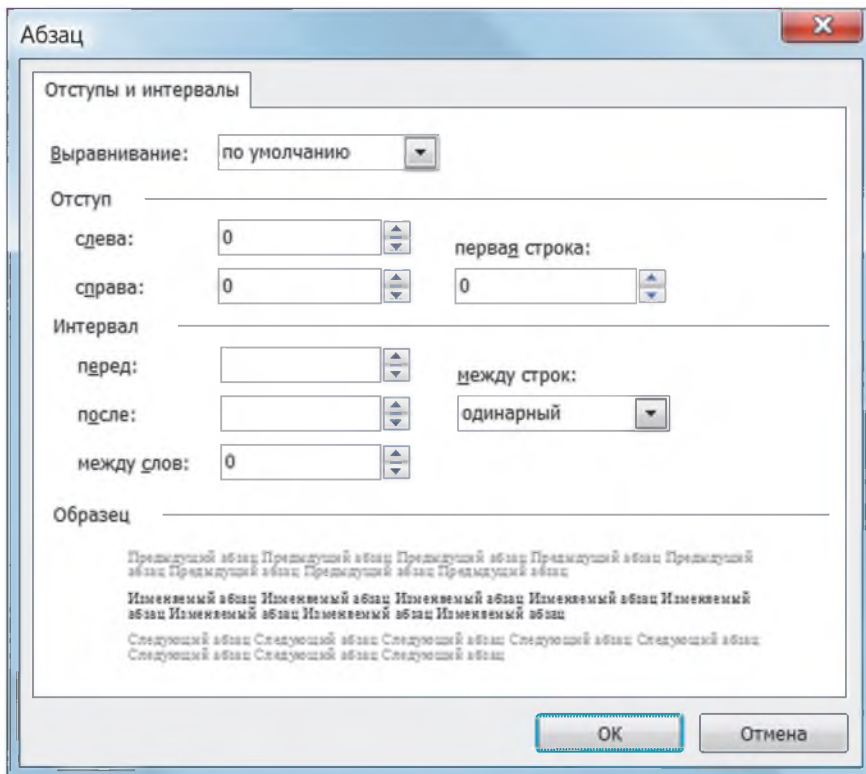


Рис. 1.36

Важно помнить, что отображение веб-страницы на экране зависит от разрешения монитора и настройки браузера. Поэтому размер шрифта принято задавать в условных единицах от 1 до 7. Если размер шрифта не указан, то по умолчанию он принимается равным 3, что при установке в браузере значения **Размер шрифта** → **средний** соответствует 12 пунктам.

Для форматирования текста используем панель инструментов **Форматирования**, а также окна **Шрифт** (рис. 1.35) и **Абзац** (рис. 1.36) из меню **Формат**.

Установим параметры заголовка: шрифт Arial, размер 5 (18 пт), цвет бирюзовый, выравнивание по центру. Параметры основного текста оставим без изменений, лишь выделим первые строки каждого абзаца (шрифт Arial, цвет бирюзовый). Названия озер выделим полужирным шрифтом.

Повышению удобочитаемости текста и привлекательности страниц способствует использование списков, а также разбиение больших страниц на отдельные блоки с помощью горизонтальных разделительных линий.

Маркированные и нумерованные списки создаются с помощью панели инст-

рументов **Форматирование**, а также диалогового окна **Список**, которое открывается из меню **Формат**. Для оформления веб-страницы «Озера Беларуси» используем графические маркеры.

С помощью команды **Вставка** → **Горизонтальная линия** поместим на страницу горизонтальные разделительные линии. В окне **Свойства горизонтальной линии** зададим параметры линии в соответствии с рисунком 1.37.

Теперь страница будет выглядеть так, как на рисунке 1.32. Сохраним документ, заменив предлагаемое по умолчанию имя нов_стр_1.htm на более подходящее озера.htm.

Займемся теперь размещением на созданной странице изображений. Подчеркнем, что все изображения, которые мы видим на веб-страницах, хранятся в отдельных файлах, а на самой странице имеются лишь ссылки на соответствующие файлы.

Пример 2. Разместить на веб-странице «Озера Беларуси» изображения в соответствии с рисунком 1.38.

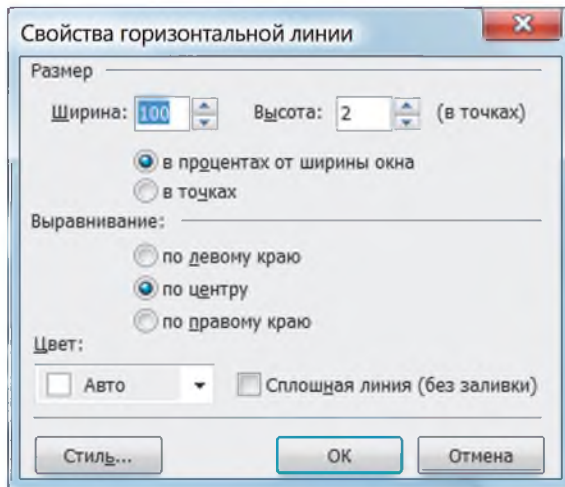



Рис. 1.37



Рис. 1.38

Установим курсор в то место текста, к которому будет привязан рисунок, например перед началом первой строки. С помощью команды **Вставка** → **Рисунок** или кнопки  на панели инструментов (рис. 1.39) вызовем окно **Рисунок** и выберем требуемый файл с изображением, например parash.jpg.

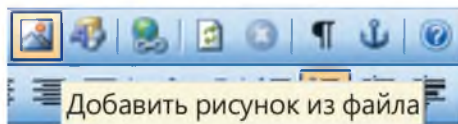



Рис. 1.39

Как и в документах MS Word, изображения могут располагаться по отношению к тексту различными способами. По умолчанию рисунок отображается в выбранном месте строки без обтекания текстом (рис. 1.40, *а*). При этом можно выбрать способ выравнивания.

На рисунке 1.40, *а* текст и изображение выровнены по левому краю. Рисунок 1.40, *б* иллюстрирует обтекание изображения слева. (На это указывает непечатаемый символ стрелка влево  в точке привязки. При необходимости эту точку можно перетаскивать с помощью мыши.)

- 

Нарочь — озеро
Самое большое озеро в Беларуси.
Входит в Нарочанскую группу озер

• **Нарочь — озеро в Мядельском районе**



☑Самое большое озеро в
Входит в Нарочанскую г
Площадь — 79,6 кв. км.
В Нарочи обитает 22 вид
крупная — щука.
На озере и в округе имеются гнездовья лебе
малой крачки, скопы, малой поганки.

Рис. 1.40

*а**б*

Параметры изображения настраиваются в диалоговом окне **Свойства рисунка** (рис. 1.41). Вызовем это окно двойным щелчком мыши по выбранному рисунку или с помощью контекстного меню. Зададим **Обтекание: справа, Выравнивание: по правому краю, Толщина границы: 0** и другие параметры в соответствии с рисунком 1.41. Здесь же можно установить размер изображения, но проще подобрать размер рисунка, выделив его и перетаскив мышью появившиеся маркеры, как в любом редакторе.

Аналогичным способом вставим второй рисунок из файла braslav.jpg.

В редакторе FrontPage предусмотрена возможность простейшей обработки изображений (коррекция яркости и контрастности, повороты, обрезка и т. д.). С помощью меню **Вид** вызовем панель инструментов **Рисунки**. Увеличим яр-

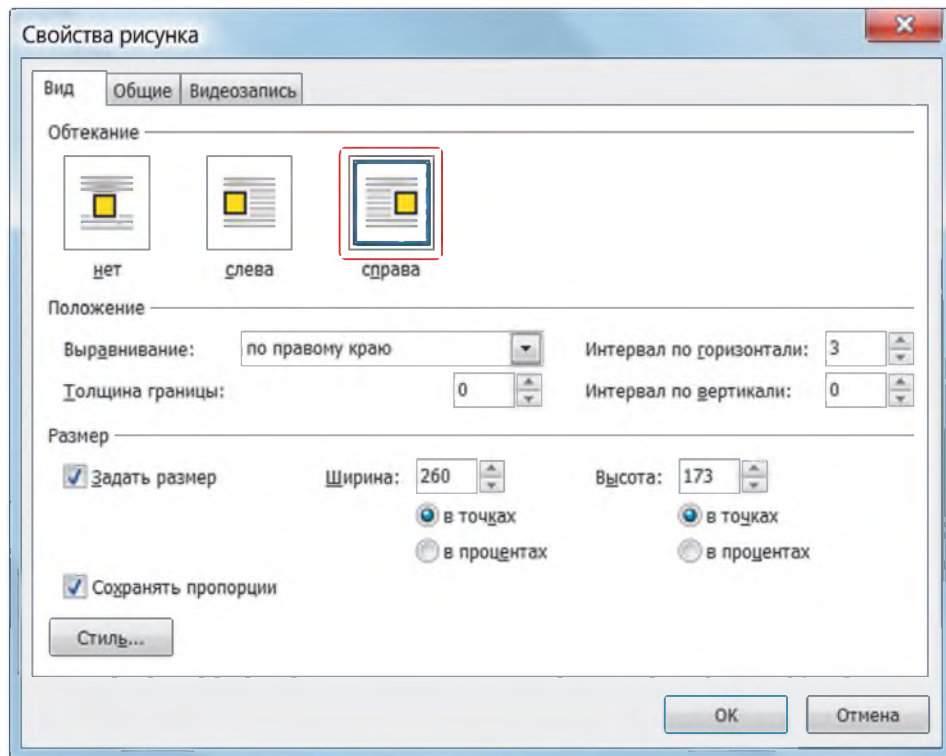




Рис. 1.41

кость и уменьшим контрастность изображений, нажав соответственно кнопки  и .

Страница примет требуемый вид (см. рис. 1.38). Сохраним ее под именем `озера2.htm` и просмотрим в браузере.

Для привлечения внимания к информации можно создать бегущую строку.



Пример 3. На веб-странице «Озера Беларуси» добавить бегущую строку «Приглашаем в лодочный поход».

Установим курсор под нижней горизонтальной линией. Выполним последовательность действий: **Вставка** → **Веб-компонент** → **Бегущая строка**. В окне **Свойства бегущей строки** введем текст «Приглашаем в лодочный поход». Зададим ее параметры в соответствии с рисунком 1.42. Шрифт, размер и начертание текста устанавливается с помощью меню **Шрифт**, которое вызывается кнопкой **Стиль**.

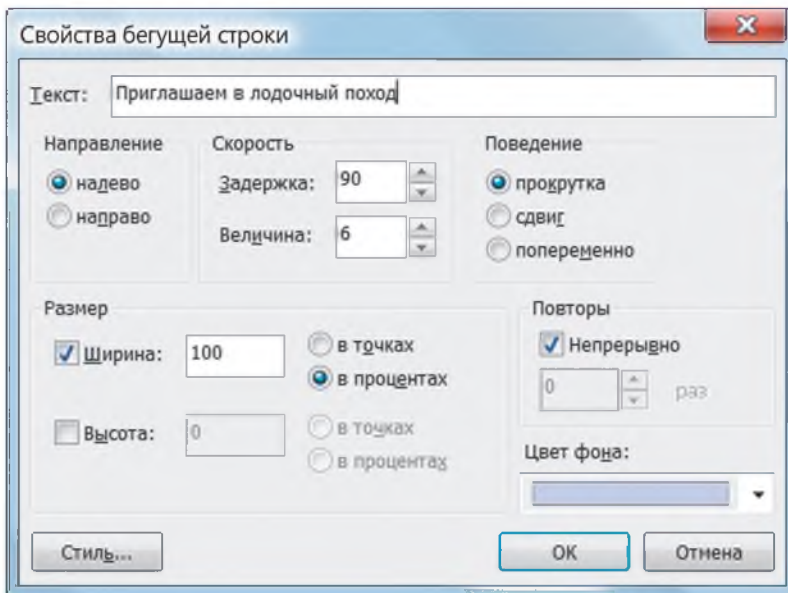


Рис. 1.42

В результате в нижней части страницы появится светло-голубая полоса, по которой справа налево будет перемещаться текст

Приглашаем в лодочный поход



1. Каково назначение редактора FrontPage?
2. Какие возможности форматирования текста имеет FrontPage?
3. Как задается размер шрифта на веб-страницах?

Упражнение

Создайте одну из веб-страниц: Реки Беларуси, Заповедники, Парки.

Используйте фотографии и тексты из указанных учителем файлов. Оформите страницу в соответствии с образцом на рисунке 1.38.

5.3. Использование таблиц

Размещать текст в несколько колонок, а также располагать рисунки и текст в требуемых местах веб-страниц удобно с помощью таблиц.

Для вставки таблицы проще всего использовать стандартную панель инструментов, однако наиболее полно возможности работы с таблицами представлены в меню **Таблица**. Например, в диалоговых окнах **Вставка таблицы** и **Свойства**

таблицы можно указать не только количество строк и столбцов таблицы, но и способ выравнивания, цвет фона ячеек, цвет и толщину границ, интервал между ячейками и другие параметры.

Заметим, что изменить параметры можно и после создания таблицы. В этом случае удобно пользоваться панелью **Таблицы** (рис. 1.43), которая вызывается с помощью команды **Вид → Панели инструментов**.

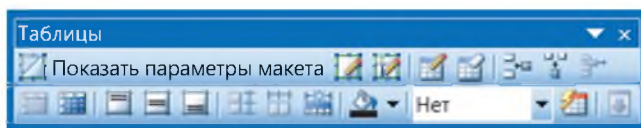



Рис. 1.43

Рассмотрим на примере, как используются таблицы для размещения объектов на веб-страницах.

Пример 1. Создать веб-страницу «Города Беларуси», на которой изображена карта Беларуси с гербами городов и указаны даты их основания (рис. 1.44).

Откроем редактор MS FrontPage. С помощью кнопки  на панели инструментов или меню **Таблица** вставим таблицу из четырех строк и трех столбцов.

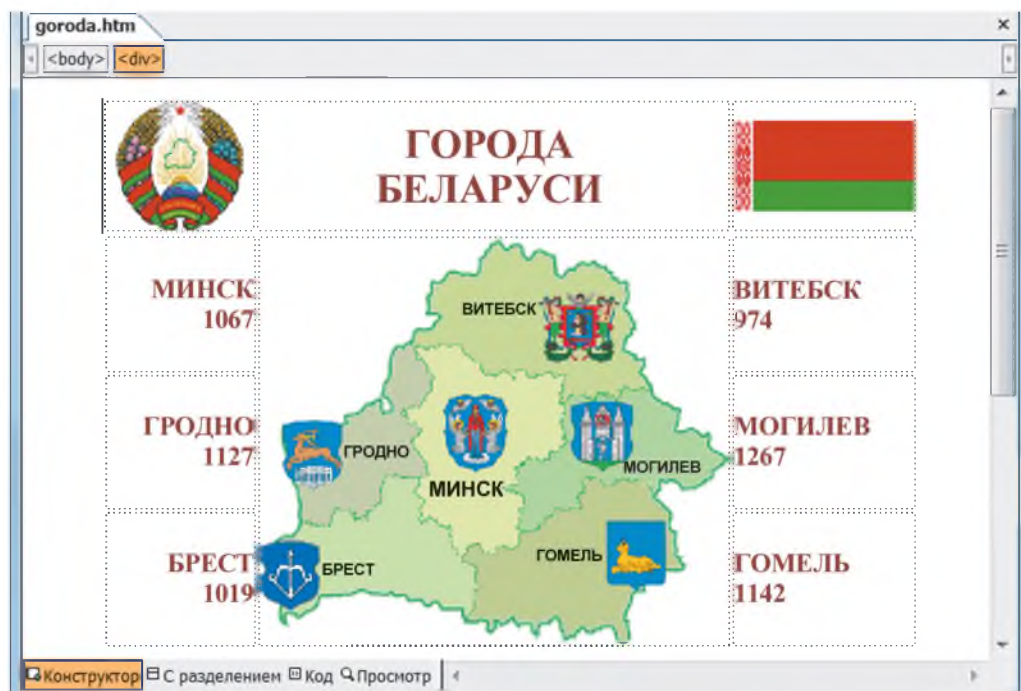


Рис. 1.44

В средней ячейке первой строки наберем заголовок «Города Беларуси». Оставшиеся ячейки второго столбца объединим. Для этого выделим объединяемые ячейки и выполним команду **Таблица → Объединить ячейки**. В полученную ячейку вставим изображение карты Беларуси из файла belarus2.jpg.

В ячейки 1 и 3 первой строки вставим изображения герба и флага Беларуси. В остальные ячейки введем названия городов и даты их основания. Отформатируем надписи и установим размеры ячеек в соответствии с рисунком 1.44. Проще всего изменять ширину столбцов, перетаскивая их границы с помощью мыши.

Сделаем невидимыми границы ячеек. Выделим таблицу и вызовем диалоговое окно **Свойства таблицы** (рис. 1.45). В поле **Положение** установим **Выравнивание: по центру**, и в поле **Границы — Размер: 0**. Такие границы в редакторе

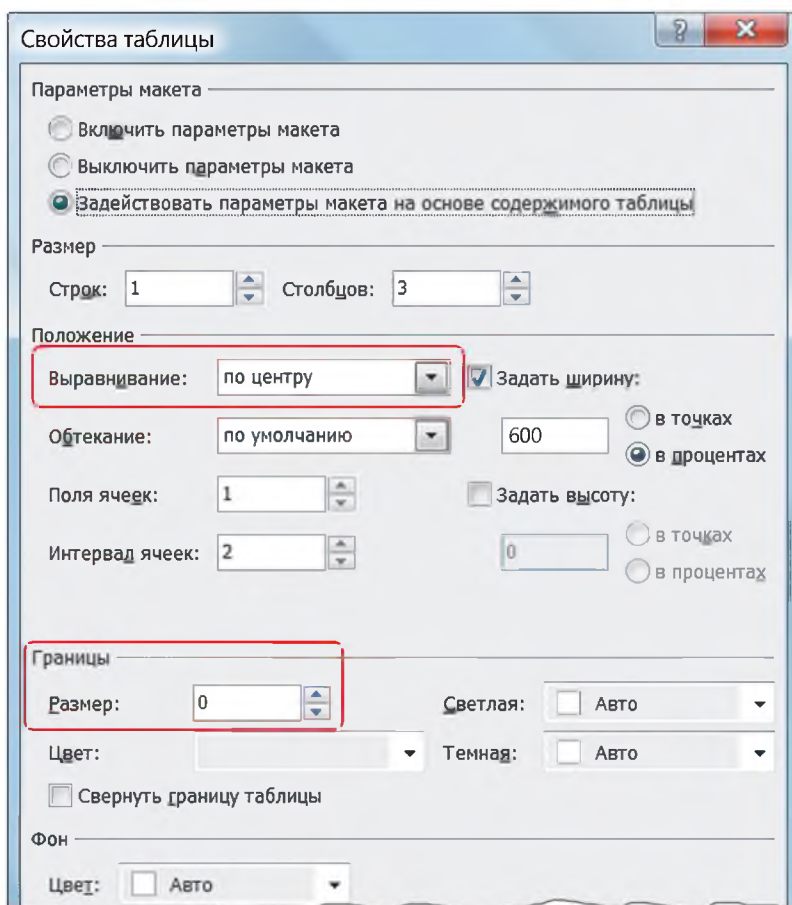


Рис. 1.45

FrontPage отмечаютя пунктирными линиями, а при просмотре веб-страницы в браузере не видны.

Теперь страница будет выглядеть так, как на рисунке 1.44. Сохраним ее под именем goroda.htm и просмотрим в браузере.



Для построения таблицы в языке HTML используются теги `<table>` — таблица; `<tr>` — строка таблицы; `<td>` — ячейка таблицы.

В этих тегах можно использовать атрибуты `width` — ширина таблицы или ячейки, `bgcolor` — цвет фона таблицы или ячеек, `border` — толщина рамки и `bordercolor` — цвет рамки и другие (см. Приложение 1).

Ширина таблицы может быть задана в пикселях или в процентах по отношению к ширине страницы, например `width=450` или `width=40%`.

По умолчанию при отображении таблицы создается эффект выпуклости границы. Если задать атрибуты `border=1` и `bordercolor=green`, то эффект выпуклости исчезнет и таблица будет ограничена двойной рамкой зеленого цвета. Атрибут `cellspacing=0` задает одинарную рамку.



Пример 2. В окне кода редактора FrontPage создать таблицу из четырех столбцов и одной строки шириной 600 пикселей с рамкой зеленого цвета толщиной 1 пиксель. Разместить в ячейках изображения (рис. 1.46).

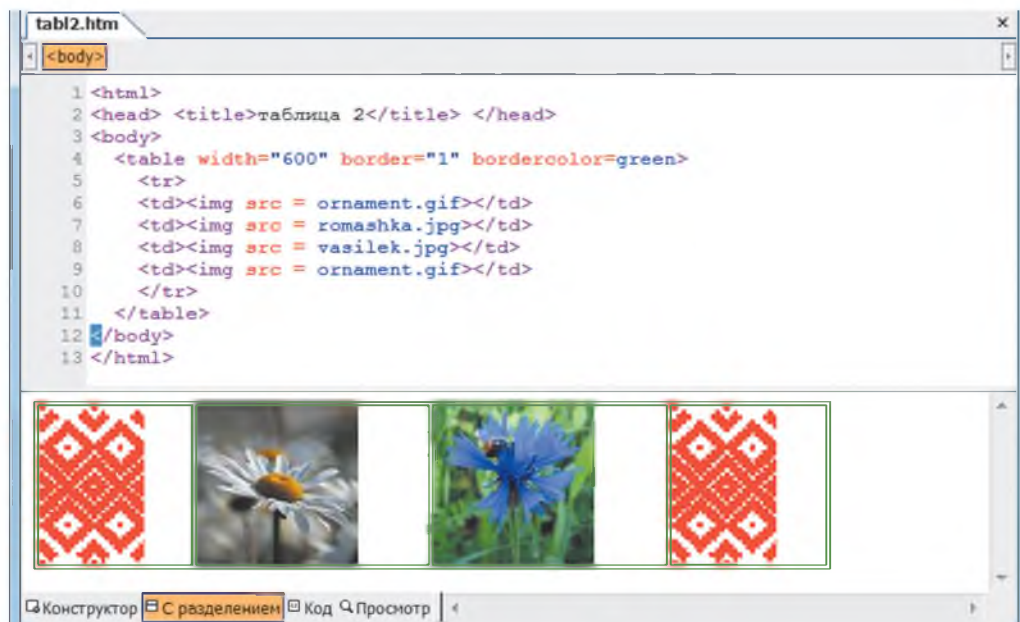


Рис. 1.46

Откроем редактор MS FrontPage. Перейдем в режим **С разделением**. В верхнем окне наберем HTML-код. В нижнем окне будет отображен результат (см. рис. 1.46). Сохраним документ под именем `tabl2.htm`. Просмотрим веб-страницу в браузере.



1. Для чего используются таблицы на веб-страницах?
2. Как вставить таблицу в редакторе FrontPage?

Упражнения

1. Создайте веб-страницу «Разложение белого света в спектр».

Для этого используйте таблицу из четырех столбцов и семи строк. В объединенные ячейки левого и правого столбцов вставьте изображения призмы и радуги из файлов `prizma.jpg` и `raduga.jpg`. В ячейки второго столбца поместите названия цветов и залейте их соответствующим фоном. В ячейках третьего столбца расположите мнемоническое правило следования цветов спектра.

	красный	каждый	
	оранжевый	охотник	
	желтый	желает	
	зеленый	знать	
	голубой	где	
	синий	сидит	
	фиолетовый	фазан	

2. Создайте веб-страницу по одной из тем: Природа родного края. Растения. Животные. Птицы. Рыбы. Дары леса. Грибы. Ягоды. Наш сад. Овощи. Фрукты. Цветы.

Тексты и изображения из файлов разместите в таблице из 4—6 ячеек. В качестве образца оформления используйте веб-страницы из примеров 1 и 2.

5.4. Разработка веб-сайта

Рассмотрим особенности разработки веб-сайта в редакторе FrontPage на примере сайта «Песняры беларускай зямлі».

Этап проектирования сайта не отличается от описанного ранее. Создадим сайт из четырех страниц (главной и трех персональных: «Якуб Колас», «Янка Купала» и «Максім Багдановіч»). Все страницы сайта будем сохранять в файлах с соответствующими именами в одной папке, например `pisnen`. Структуру сайта изобразим в виде двухуровневой схемы (рис. 1.47).

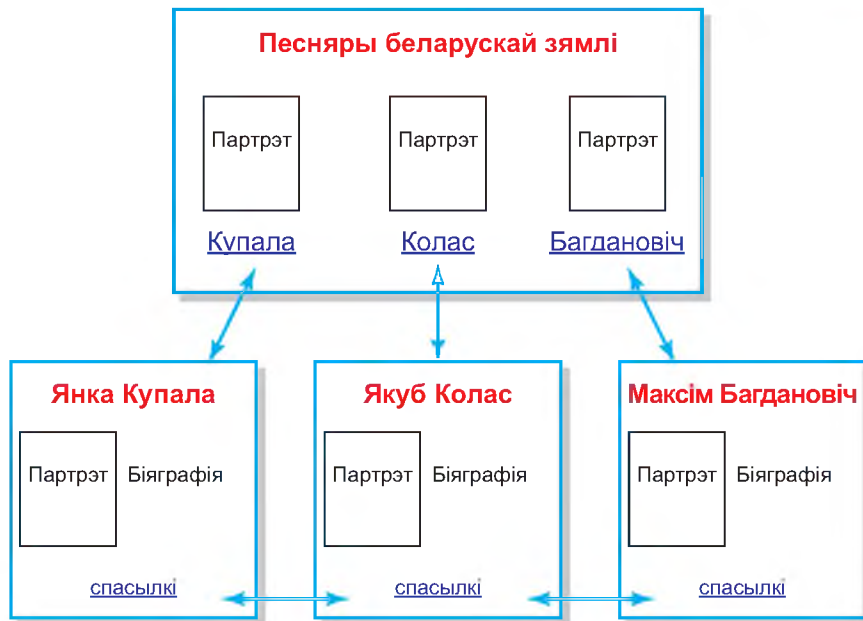


Рис. 1.47

На главной странице разместим заголовок «Песняры беларускай зямлі» и гиперссылки, позволяющие открывать страницы второго уровня. В качестве гиперссылок будем использовать не только текст (например, фамилии), но и изображения (например, небольшие портреты).

Каждая персональная страница будет содержать текстовую (биографические сведения) и графическую информацию (портреты, фотографии памятных мест). Внизу расположим гиперссылку, обеспечивающую возврат на главную страницу сайта, и две ссылки для перехода с одной персональной страницы на другую.

Заранее подготовим все изображения. Поместим их во вложенную папку pismen/Ris/.

Непосредственное конструирование начнем с персональных страниц, которые оформим в едином стиле.

Напомним, что для облегчения подбора элементов оформления можно использовать готовый шаблон оформления. В редакторе FrontPage предлагается несколько шаблонов — **Тем**. Тему можно применять к отдельным страницам, а также ко всему сайту. В последнем случае при создании каждой новой страницы тема устанавливается автоматически.

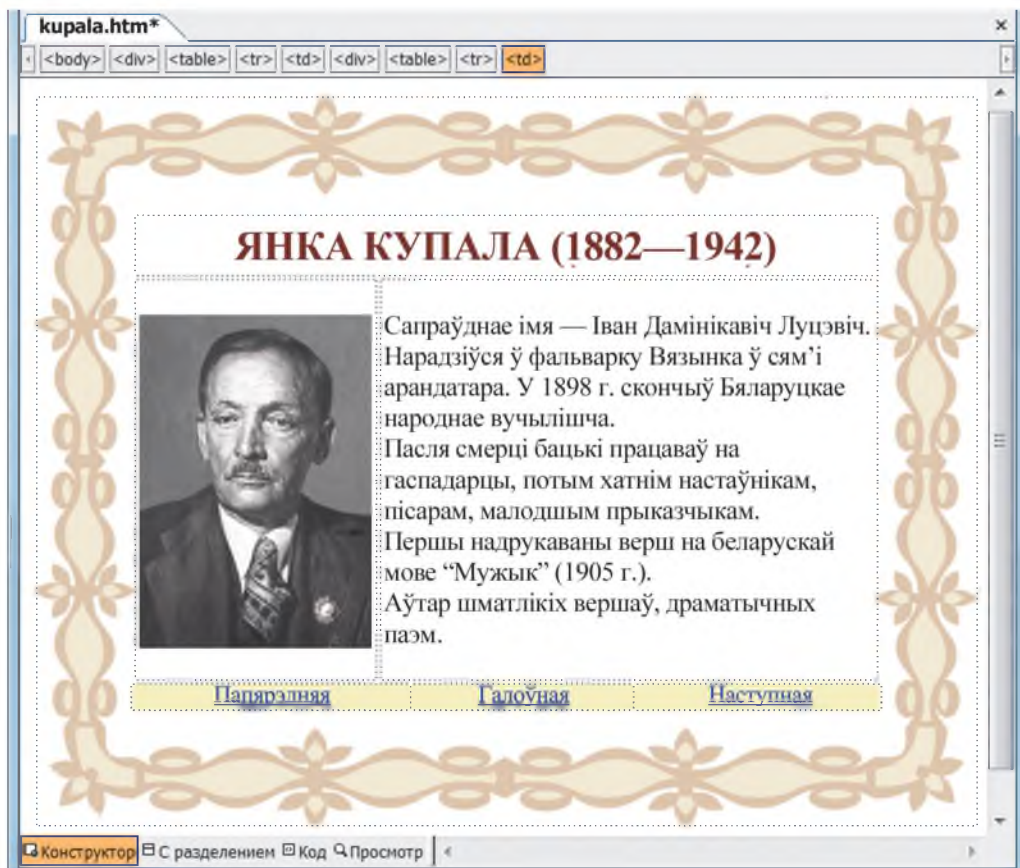


Рис. 1.48

Пример 1. Создать веб-страницу «Янка Купала» (рис. 1.48).

Запустим редактор FrontPage. Откроется пустая страница с установленными по умолчанию параметрами. Проверим кодировку (**Windows-1251 — Кириллица**).

Цвет фона или фоновый рисунок всей страницы задаются в диалоговом окне **Свойства страницы** на вкладке **Форматирование**, которая вызывается с помощью меню **Формат → Фон**. Мы создадим свой шаблон оформления страниц на основе изображения рамки. Используем его в качестве фона таблицы. Вставим таблицу из одной ячейки. В диалоговом окне **Свойства таблицы** зададим ее ширину (720 пикселей) и высоту (560 пикселей) в соответствии с размерами рамки, установим флажок **Использовать фоновый рисунок** и укажем имя файла `ramka.gif`. В эту таблицу вставим еще три таблицы шириной 80 %: из одной

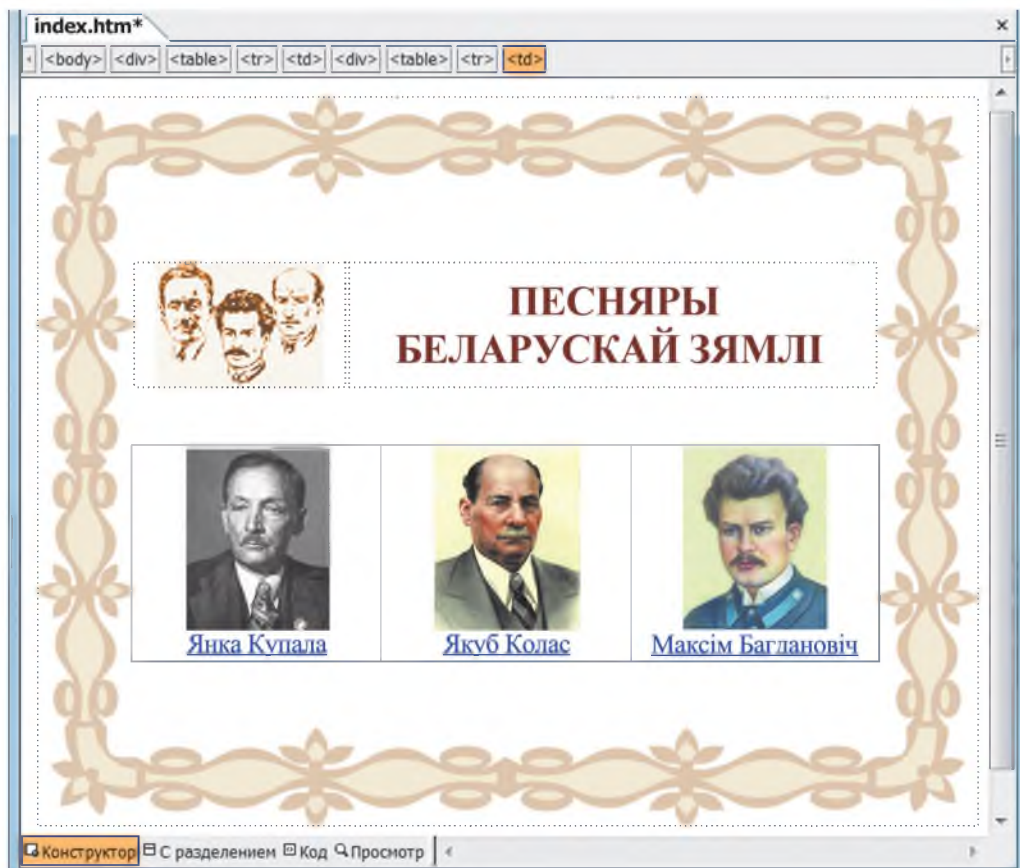


Рис. 1.49

ячейки — для размещения заголовка, из двух — для рисунка и биографии, и из трех — для текстовых гиперссылок.

Оформим страницу в соответствии с рисунком 1.48: скопируем заранее подготовленный текст из файла `biograf.txt`, вставим рисунок из файла `kupala.jpg`. Заметим, что можно использовать и одну таблицу с объединенными ячейками, но в ней труднее размещать объекты в требуемых местах. Сохраним страницу под именем `Kupala.htm` в папке `pismen`.


Аналогично оформим персональные страницы «Якуб Колас» (файл `Kolas.htm`) и «Максім Багдановіч» (файл `Bagdanovich.htm`). Главную страницу (файл `index.htm`) оформим в соответствии с рисунком 1.49.

Напомним, что сайт представляет собой совокупность веб-страниц, связанных гиперссылками. Для создания гиперссылки требуется выделить текст или ри-

сунок и задать адрес перехода (адрес веб-страницы, имя файла или документа) по данной ссылке. В качестве гиперссылок могут выступать слова, группы слов, изображения.

Текстовые ссылки обычно выделяются цветом и подчеркиванием, а графические — рамкой. Благодаря этому они отличаются от обычного текста и других элементов страницы. По умолчанию ссылки на непросмотренные страницы отображаются синим цветом, а на просмотренные — фиолетовым.

Пример 2. Создать текстовые и графические гиперссылки между страницами веб-сайта «Песняры беларускай зямлі».

Для создания гиперссылки выделим портрет Янки Купалы и подпись. С помощью кнопки **Добавление гиперссылки**  на стандартной панели инструментов или команды **Вставка** → **Гиперссылка** вызовем диалоговое окно, в котором укажем адрес перехода Kupala.htm (рис. 1.50). Аналогичным образом создадим все ссылки в соответствии со схемой навигации по сайту (см. рис. 1.47).

Вызванная щелчком мыши по ссылке страница по умолчанию открывается в текущем окне браузера. Можно задать ее открытие в новом окне браузера. Для этого в диалоговом окне **Добавление гиперссылки** щелкают по кнопке **Выбор рамки** и в открывшемся окне **Конечная рамка** выбирают вариант **Новое окно**.

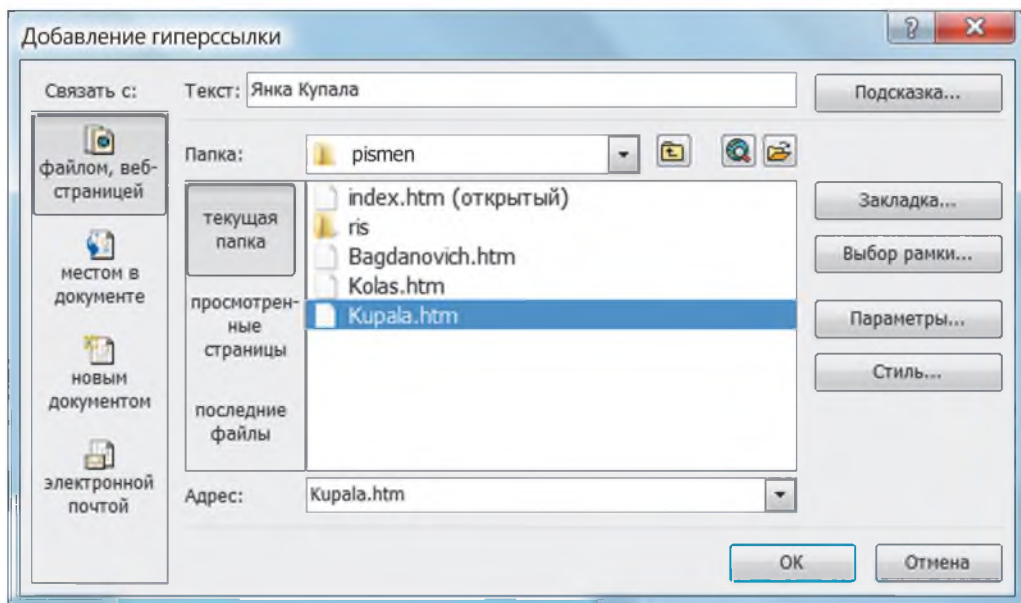


Рис. 1.50

Заметим, что в редакторе FrontPage создание гиперссылок выполняется в режиме работы **Конструктор**, а проверить работу гиперссылок можно в режиме **Просмотр**.

Ссылки могут обеспечивать переход к веб-страницам или иным документам, например рисункам, презентациям, видеофрагментам, расположенным как на данном сайте (*внутренние ссылки*), так и на других сайтах (*внешние ссылки*). Переходы внутри документа выполняют *внутристраничные* гиперссылки. Они облегчают навигацию по странице, например быстрый переход из конца страницы в начало.



Перед созданием внутристраничных ссылок нужно сначала расставить **закладки**. Для этого курсор мыши устанавливают в том месте страницы, куда предполагается переход по ссылке. Затем с помощью команды **Вставка → Закладка** вызывают диалоговое окно **Закладка**, в котором вводят имя закладки, например **начало** (рис. 1.51). Далее в окне **Добавление ссылки** выбирают вариант **Связать с местом в документе** и указывают требуемую закладку (рис. 1.52).

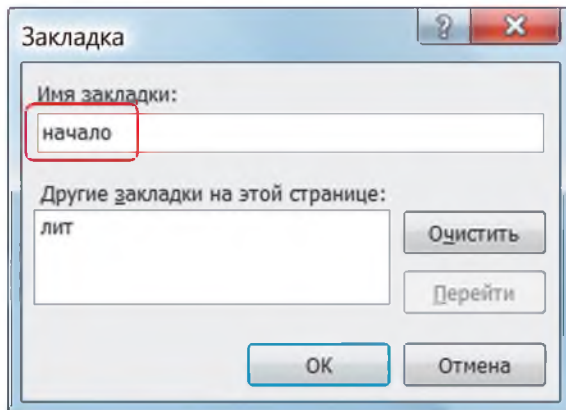


Рис. 1.51

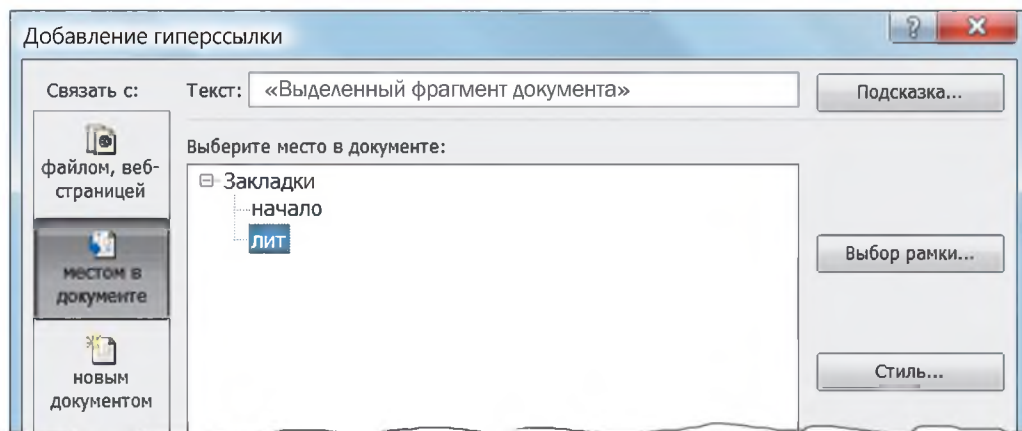


Рис. 1.52



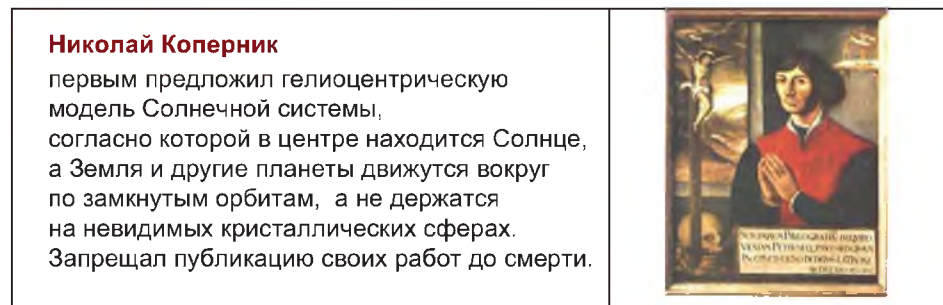
В качестве гиперссылок на веб-страницах нередко используют кнопки, которые при наведении на них курсора могут изменять свои цвет и форму. В редакторе FrontPage предусмотрена возможность размещения таких кнопок. Для этого в меню **Вставка** → **Веб-компонент** выбирают компонент **Интерактивная кнопка**. Затем в появившемся диалоговом окне **Меняющиеся кнопки** выбирают кнопку нужной формы и задают ее параметры и эффекты.



1. Какие элементы могут выступать в качестве гиперссылок?
2. На какие объекты может указывать гиперссылка?

Упражнение

1. Создайте фрагмент веб-сайта «Галерея ученых» (писателей, спортсменов, музыкантов и т. п.), оформив его по приведенным образцам. Гиперссылки на главной странице представьте в виде миниатюр с подписями внизу.



5.5. Публикация сайта

Публикацией сайта называют его размещение на сервере или локальном компьютере с возможностью вызова из сети (глобальной или локальной).

Процесс публикации сайта заключается в переносе файлов сайта на сервер. Адрес перенесенного сайта может быть, например, таким: www.poets.webserver.by.

Для публикации сайта, подготовленного в редакторе FrontPage, необходимо с помощью команды **Файл → Опубликовать узел** вызвать диалоговое окно **Свойства удаленного веб-узла**.

Для размещения созданного сайта на сервере в строке **Расположение удаленного веб-узла** следует указать адрес ftp-узла, который для всех пользователей является одинаковым, и нажать кнопку ОК.

В появившемся окне **Требуется имя и пароль** вводят имя и пароль, которые понадобятся в дальнейшем для обновления, добавления или удаления размещенных на сайте материалов.

Нажатие кнопки ОК открывает окно, на левой панели которого (**Локальный веб-узел**) (рис. 1.53) отображаются файлы и папки, содержащие страницы и ри-

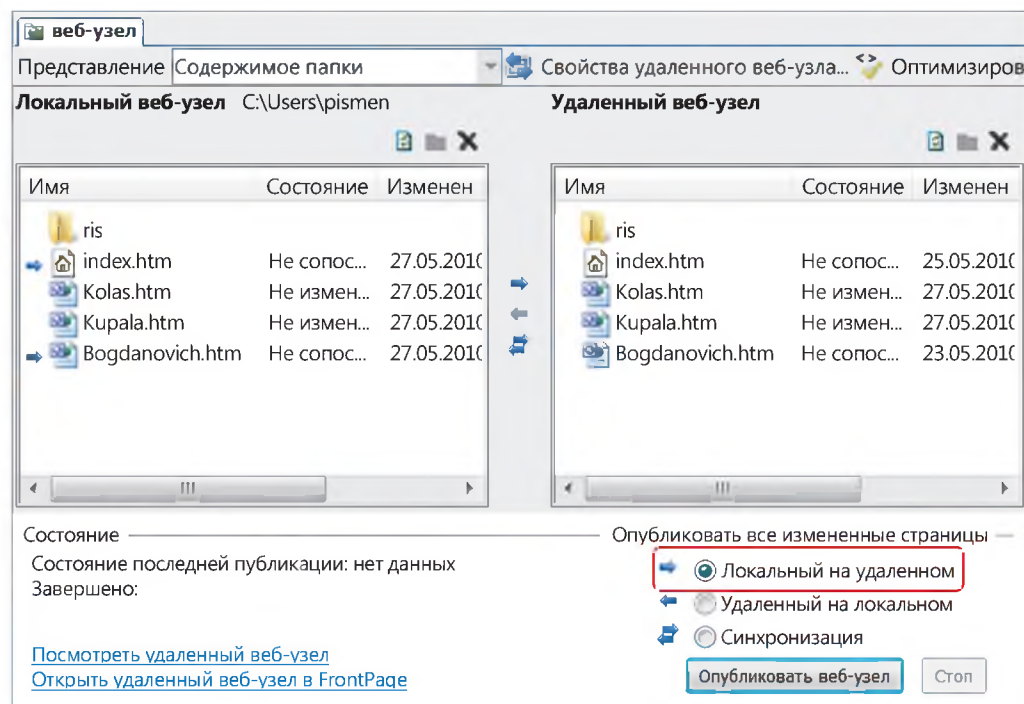


Рис. 1.53

сунки созданного фрагмента веб-сайта. На правой панели этого окна (**Удаленный веб-узел**) отображаются файлы и папки, уже размещенные на web-сервере, например страницы сайта «Песняры беларускай зямлі» (см. рис. 1.53).

Если публикация выполняется впервые, правая панель пуста. Выберем требуемый режим, например **Все файлы локального веб-узла копировать на сервер**, и нажмем кнопку **Опубликовать веб-узел**.

Как только копирование всех файлов завершится, пользователь сможет просмотреть размещенный в сети сайт. Для этого в строке браузера вводится адрес, например www.poets.webserver.by.



1. Что понимают под публикацией сайта?
2. Какие действия надо выполнить для публикации сайта, подготовленного в редакторе FrontPage?

§ 6. Создание фрагментов сайтов по различным предметным областям

Изучив этапы разработки веб-сайтов и основные приемы работы в редакторе Front Page, мы можем создавать сайты, посвященные различным предметным областям.

Пример. Создать фрагмент веб-сайта «Замки Беларуси» (рис. 1.54, 1.55).



Рис. 1.54



Рис. 1.55

Создадим фрагмент сайта из пяти страниц: главной страницы с маленькими картинками и четырех страниц, посвященных замкам. Все страницы будем сохранять в файлах с соответствующими именами в одной папке, например `zamki`. Заранее подготовленные изображения поместим во вложенную папку `zamki /ris/`.

Оформим страницы в одном стиле в соответствии с рисунком 1.55. Вставим таблицу с одной ячейкой. Установим ширину 640 и высоту 480 пикселей. В качестве фона таблицы используем изображение свитка из файла `svitok.gif`.

Вставим еще одну таблицу из одного столбца и четырех строк. В первой строке разместим заголовок. Во вторую строку вставим рисунок, например из файла `nesvizh.jpg`, в третью — поместим текст из файла `zamki.txt`. В четвертой строке разместим разделительную линию, тексты ссылок и изображения стрелок из файлов или **Коллекции клипов**.

Главную страницу оформим в соответствии с рисунком 1.54. В таблице из двух столбцов и двух строк разместим используемые в качестве гиперссылок картинки с подписями. Создадим гиперссылки для переходов со страницы на страницу.

Просмотрим страницы в браузере. Проверим работу гиперссылок.

Упражнение

Создайте фрагмент сайта (не менее 3—4 страниц, содержащих текст и изображения) по одной из тем учебных предметов.

Беларуская літаратура. Жыццё Ефрасінні Полацкай. Жыццё і творчасць Кірылы Тураўскага. Францыск Скарына — першадрукар і асветнік. Мікола Гусоўскі. Песня пра зубра. Адам Міцкевіч. Францішак Багушэвіч. Кандрат Крапіва. Васіль Быкаў. Беларуская паэзія XX стагоддзя.

Русская литература. А. С. Пушкин. М. Ю. Лермонтов. Н. В. Гоголь. Н. А. Некрасов. А. Н. Островский. И. С. Тургенев. Ф. М. Достоевский. Л. Н. Толстой. А. П. Чехов. М. Горький. Серебряный век русской культуры. Русская поэзия XX века.

Всемирная история. Первая мировая война. Октябрьская революция. Образование СССР. Начало Великой Отечественной войны. Блокада Ленинграда. Сталинградская битва. Курская дуга. Партизанское движение. Операция «Багратион». Капитуляция Германии. Основание ООН. Распад СССР. Образование СНГ.

Гісторыя Беларусі. Грунвальдская бітва. Першая сусветная вайна на беларускіх землях. Гады грамадзянскай вайны. Вялікая Айчынная вайна. Партызанскі рух. Вызваленне Беларусі. Помнікі гісторыі. Стварэнне Рэспублікі Беларусь.

География. Великие географические открытия. Страны и континенты (Природные условия. Погода и климат. Население. Промышленность. Сельское хозяйство). Геоэкологические проблемы. Изменение климата. Загрязнение окружающей среды. Охрана природы.

Биология. История открытия клетки. Строение клетки. Обмен веществ. Фотосинтез. Экосистема. Происхождение жизни. Биологическая эволюция. Происхождение и эволюция человека. Биосфера — живая оболочка планеты. Человек и окружающая среда. Влияние различных факторов среды на человека.

Математика. Графики функций. Тригонометрические функции. Показательная и логарифмическая функции. Правильные многоугольники. Пространственные фигуры. Многогранники и их изображения. Тела вращения.

Физика. Электромагнитная индукция. Гармонические колебания. Маятники. Волны. Звук. Электромагнитные колебания. Производство и передача электрической энергии. Электромагнитные волны. Шкала электромагнитных волн. Интерференция. Дифракция. Законы преломления. Оптические приборы.

Химия. Периодическая система химических элементов Д. И. Менделеева. Атомы и молекулы. Неорганические соединения. Кислород. Сера. Азот. Фосфор. Углерод. Кремний. Металлы. Химическое строение органических соединений. Углеводороды. Переработка нефти. Синтетические моющие средства. Углеводы. Азотсодержащие органические соединения. Аминокислоты. Синтетические высокомолекулярные соединения. Применение полимеров. Белки.