

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МАЙКОПСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных систем в экономике и юриспруденции
Кафедра информационной безопасности и прикладной информатики

ИНФОРМАТИКА: ОСНОВЫ САЙТОСТРОЕНИЯ

Учебное пособие

Майкоп

2020 г.

УДК 002.6:004 (07)

ББК 73

И-74

Печатается по решению Научно-технического совета Федерального государственного бюджетного образовательного учреждения высшего образования «Майкопский государственный технологический университет»

Рецензенты:

Довгаль В.А., канд. техн. наук, доцент кафедры информационной безопасности и прикладной информатики Майкопского государственного технологического университета.

Панеш А.Х., канд. техн. наук, доцент кафедры прикладной математики, информационных технологий и информационной безопасности Адыгейского государственного университета.

Составители: кандидат экономических наук **Меретукова С.К.**

кандидат технических наук, доцент **Чундышко В.Ю.**

кандидат педагогических наук, доцент **Меретуков Ш.Т.**

кандидат философских наук **Козлова Н.Ш.**

И-74 Информатика: основы сайтостроения: учебное пособие / [составители: Меретукова С.К. и др.]; М-во науки и высш. образования РФ, ФГБОУ ВО «МГТУ», Фак. информац. систем в экономике и юриспруденции, Каф. информац. безопасности и приклад. информатики. - Майкоп: Изд. МГТУ, 2020. – 148 с.

ISBN 978-5-88941-153-6

Учебное пособие является частью учебно-методического комплекта по курсу информатики, включает материал по основам использования языка разметки HTML для создания сайтов, сопровождается большим количеством примеров. Кроме того, в учебном пособии предлагаются упражнения, задания для самостоятельного выполнения, тестовые задания разного уровня сложности.

Содержание пособия, его структура и логика изложения соответствуют требованиям, предъявляемым государственным образовательным стандартам высшего образования.

Предназначено для студентов всех форм обучения, изучающих информатику, основы сайтостроения, вычислительную технику и сети в отрасли, а также для преподавателей и специалистов любой предметной области для самостоятельного изучения основ сайтостроения.

ISBN 978-5-88941-153-6



9 785889 411536

УДК 002.6:004 (07)

ББК 73

© Меретукова С.К., Чундышко В.Ю.,
Меретуков Ш.Т., Козлова Н.Ш., 2020

Содержание

Введение	4
Глава 1. Структура html -документа	7
1.1 Структура простейшего html -документа	7
1.2 Форматирование текста	19
1.3 Выделение фрагментов текста	21
1.4 Задание размера символов.....	25
1.5 Гарнитура, цвет и размер шрифта	28
1.6 Задание цвета фона и текста web-страницы	31
1.7 Вставка горизонтальных линий и эффектов движения в текст страницы.....	34
1.8 Теги списков.....	36
Практическая часть.....	39
Тестовые задания для самоконтроля	45
Глава 2. Вставка объектов в html -документ	52
2.1 Размещение графики на web-страницах	52
2.2 Вставка таблиц в html-документ.....	66
2.3 Вставка гиперссылок в html-документ.....	88
2.3.1 Гиперссылки внутри документа.....	90
2.3.2 Гиперссылки на другие документы	94
2.3.3 Гиперссылки с помощью карты изображения.....	103
Тестовые задания для самоконтроля	108
Глава 3. Работа с фреймами	114
3.1 Синтаксис фреймов	115
Практическая часть.....	123
Тестовые задания для самоконтроля	130
Ответы к тестовым заданиям	137
Заключение	140
Глоссарий	141
Список использованной литературы	146

Введение

Сегодня необходимо констатировать тот факт, что в рамках высшего образования невозможно охватить весь круг вопросов, связанных с использованием ИКТ в различных сферах деятельности человека, изучить постоянно обновляющееся программное обеспечение и инновационные технологии. Темпы развития информатизации общества обгоняют учебные планы и образовательную программу. Выправить сложившееся положение способны дополнительные общеразвивающие программы, нацеленные на развитие информационных компетенций обучающихся. Они позволяют удовлетворить естественный интерес обучающихся к применению информационных технологий и способствуют лучшему усвоению материала предметов образовательного процесса. Таким образом, учебное пособие, в котором всесторонне рассматриваются возможности использования информационных технологий, наиболее востребованные на современном рынке труда: от создания простейших графических файлов до разработки персонального Интернет-ресурса, является весьма актуальным.

Учебное пособие «Информатика: основы сайтостроения» описывает комплексно и всесторонне различные аспекты создания Интернет-ресурса, от подготовки графических элементов и создания шаблонов до размещения уже готовых веб-страниц в сети Интернет, их оптимизации и SEO-продвижения в поисковых системах.

Учебное пособие «Информатика: основы сайтостроения» предназначено для преподавателей и других специалистов сферы образования, желающих научиться проектировать и создавать сайты и интересующихся проблемами компьютерного веб-дизайна. Пособие также будет полезно обучающимся разных уровней образования для ознакомления с особенностями языка HTML.

Язык HTML (Hyper Text Markup Language) – это независимый от платформ язык разметки текста. Документы, размеченные при

помощи этого языка, визуализируются браузерами в большинстве случаев одинаково. Исходный код представляет собой текст, между строк которого вставляются элементы разметки. Посетителю страницы эти элементы не видны, а виден результат их воздействия на документ.

Базовым понятием языка разметки гипертекста является – **тег** (tags – дескриптор, маркер). Тег всегда заключен между скобками < > и имеет следующий вид: <имя тега>. Теги бывают открывающими и закрывающими. Открывающий тег отличается от закрывающего тем, что перед закрывающим стоит косая черта "/". Набирать имена тегов можно как заглавными буквами, так и строчными. Совокупность открывающего (<тег>) и закрывающего (</тег>) дескрипторов называется **контейнером**.

<тег> содержимое контейнера </тег>

Для расширения возможности отдельных тегов и для более гибкого управления содержимым контейнеров в тегах используются атрибуты. Атрибуты указываются для открывающего тега. Синтаксис тега с атрибутами имеет следующий вид:

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

Если для тега не добавлен какой-либо допустимый атрибут, то браузер в этом случае будет подставлять значение, заданное по умолчанию.

Существует множество простых и сложных программ для написания страниц HTML. В своей основе все они делятся на две категории:

- программы, автоматизирующие вставку тегов, т.е. редактирующие исходный код;
- программы, работающие по методу WYSIWYG (What you see is what you get) – что видишь, то и получишь, когда WEB-мастер редактирует визуализированный конечный продукт, не видя исходный код.

Помимо специальных html-редакторов, не требующих особого знания синтаксиса языка, написание документа возможно

практически всеми стандартными текстовыми редакторами и процессорами, написанными для любых платформ. Вопрос состоит лишь в том, насколько профессионально должна выглядеть страница. Для создания динамичных, интерактивных страниц, включающих в себя массу нестандартных возможностей и имеющих заслуженно высокий рейтинг посещаемости, необходимы знания языка HTML и основ написания внедряемых сценариев скриптов. Для упрощения работы по написанию страниц можно использовать созданные заранее типовые шаблоны исходных кодов html-документов.

Глава 1. Структура html -документа

1.1 Структура простейшего html -документа

Документ HTML начинается открывающим тегом **<html>** и заканчивается закрывающим тегом **</html>**. Между данной парой контейнерных тегов располагаются две другие основные части html-документа: заголовок - в контейнер **<head>...</head>**; тело документа - в контейнер **<body>...</body>**. Таким образом, структура простого html-документа выглядит примерно так:

```
<html>
  <head> Заголовок документа </head>
  <body> Тело документа </body>
</html>
```

Между тегами **<head>** и **</head>** размещают заголовочные теги, которые выполняют две основные задачи:

- передать необходимую информацию браузеру;
- дать информацию поисковым системам о конкретном документе.

Содержимое тега **<head>** не отображается напрямую на веб-странице, за исключением тега **<title>**, устанавливающего заголовок окна веб-страницы. Внутри контейнера **<head>** допускается размещать следующие элементы: **<base>**, **<basefont>**, **<bgsound>**, **<link>**, **<meta>**, **<script>**, **<style>**, **<title>**. По мере продвижения более подробно будут рассмотрены все элементы контейнера **<head>**.

Парный тег **<title> </title>** - обязательный элемент заголовочной части html-документа, поэтому он должен присутствовать в любом html-документе, хотя бы формально. Синтаксис этого тега выглядит следующим образом:

```
<title> текст заголовка</title>
```

Тег <title> одинаково важен и для поисковых систем, и для посетителей. Поисковые системы при ранжировании результатов поиска придают большое значение ключевым словам, которые содержатся в теге <title>. Посетители видят его содержимое в виде ссылки на сайт в тех же результатах поиска (рис.1.1).

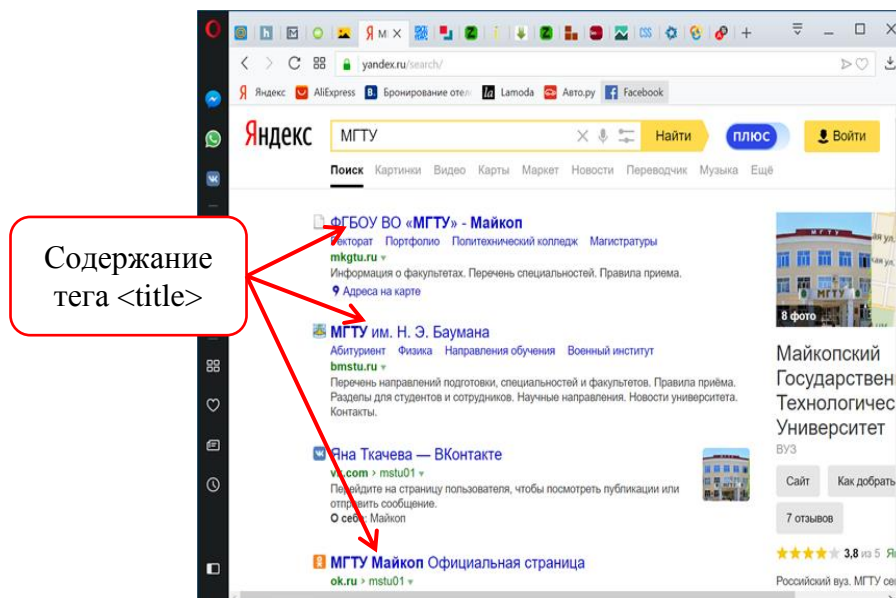


Рисунок 1.1 Содержимое контейнера <title>

Как было сказано выше, содержимое тега <title> отображается и в заголовке окна браузера.

Грамотно составить текст заголовка, заключенный между тегами <title> и </title> непросто, поскольку нужно учесть два нюанса:

- идеальный заголовок должен быть привлекательным для пользователей;
- содержать ключевые фразы для поисковых систем.

Поэтому при определении текста заголовка нужно учесть следующие требования:

- нельзя делать текст заголовка очень длинным, поскольку поисковыми системами это может быть воспринято как спам;
- для каждой страницы текст заголовка должен быть уникальным;

- текст заголовка должен отвечать содержанию страницы, в тексте заголовка следует указывать только те ключевые слова, на которые отвечает страница;
- порядок следования ключевых слов в тексте заголовка имеет значение, например, первое слово в тексте заголовка самое весомое для поисковой системы, следовательно, на первом месте необходимо употреблять самое важное ключевое слово на странице;
- нежелательно использовать бесполезные слова, например, не указывайте в тексте заголовка слова типа «главная страница». Это бесполезный мусор для пользователей.

Тег **<base>** служит для определения полного базового адреса текущего документа для гипертекстовых ссылок документа, заданных в неполной форме. Данный элемент можно использовать вне документа - в заголовке или в теле документа. При этом область действия базового адреса определяется от места размещения элемента base до следующего base.

Данный тег используется для html-документов, в которых используется относительный адрес и эти html-документы могут переноситься в другую папку или даже на другой компьютер без потери связи. Браузер ищет тег **<base>**, определяет полный адрес документа и корректно загружает его.

Мета-тег **"keywords"** (ключевые слова) поисковые системы используют для того, чтобы определить релевантность страницы тому или иному запросу. При формировании данного значения необходимо использовать только те слова, которые обязательно встречаются в самом html-документе. Использование тех слов, которых нет на странице, не рекомендуется. Ключевые слова нужно добавлять по одному, через запятую, в единственном числе. Например, не «ключевые слова», а «ключевой, слово». Рекомендованное количество слов в "keywords" – не более десяти. Кроме того выявлено, что разбивка этого значения на несколько строк влияет на оценку ссылки поисковыми машинами. Некоторые поисковые системы не индексируют сайты, в которых в значении

"keywords" повторяется одно и то же слово для увеличения позиции в списке результатов.

Html-код мета-тега "keywords" выглядит следующим образом:

```
<meta name="keywords" content="Перечень ключевых слов">
```

Мета-тег "**description**" (описание страницы) используется при создании краткого описания конкретной страницы сайта. Практически все поисковые системы учитывают его при индексации, а также при создании аннотации в выдаче по запросу. При отсутствии "description" поисковые системы выдают в аннотации первую строку документа или отрывок, содержащий ключевые слова. Отображается после ссылки при поиске страниц в поисковике, поэтому желательно не просто указывать краткое описание документа, но сделать его содержание привлекательным рекламным сообщением.

В случае правильного заполнения мета-тега "description" поисковые системы именно на его основе формируют **сниппет**¹ (англ. snippet – отрывок, фрагмент) страницы и выдается в виде краткого описания под ссылкой на странице (рис. 1.2).

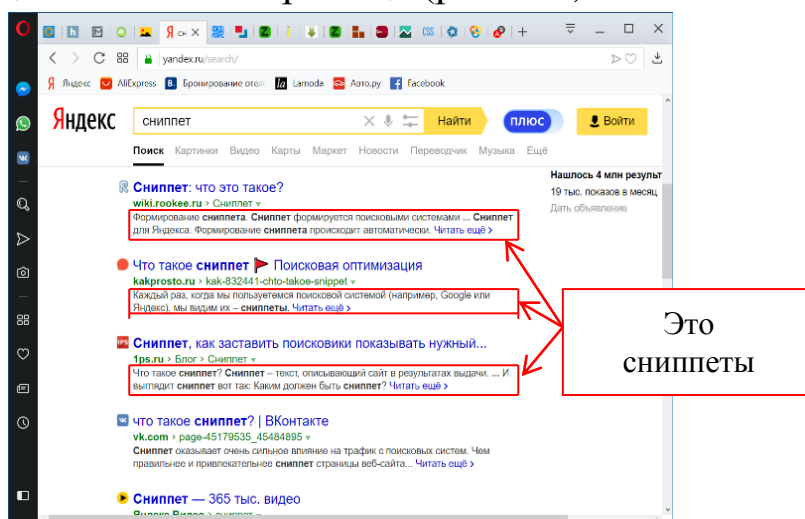


Рисунок 1.2 Сниппеты в поисковых системах

¹ Термином сниппет называют небольшие отрывки текста из найденной поисковой машиной страницы сайта, использующиеся в качестве описания ссылки в результатах поиска. Как правило, они содержат контекст, в котором встретилось ключевое слово в тексте на странице. В качестве сниппетов также может выводиться текст из мета-тега "Description". Просмотрев сниппет, можно приблизительно понять, соответствует ли страница именно вашему запросу, даже не открывая самой этой страницы.

Таким образом, правильный "description" обязательно должен содержать ключевое слово, короткое и точно описывающее то, о чем данная веб-страница. "Description" вместе с "title" образуют очень важную пару значений, от которых зависит то, перейдет пользователь из поисковой выдачи на веб-страницу или нет. Поэтому "description" и "title" нужно прописывать для каждой веб-страницы.

Html-код для создания мета-тега "description" будет следующим:

```
<meta name="description" content="содержимое мета-тега description">
```

Мета-тег "**content-type**" определяет тип документа и его кодировку. Этот мета-тег нужен для правильного отображения символов браузером.

Кодировок существует много. Большинство из них могут даже никогда не встретиться. Современные браузеры определяют тип кодировки автоматически, но для большей уверенности можно указать его и самостоятельно. При создании html-документа в операционной системе Windows, языком которой является русский, кодировка будет «windows-1251», а код для вставки мета-тега такой:

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

Мета-тег "**content-language**" применяется для указания языка документа, используется как поисковыми роботами, так и web-серверами. Html-код для вставки имеет следующий вид:

```
<meta http-equiv="content-language" content="ru">
```

Мета-тег "**refresh**" – необязательный мета-тега для указания времени (в секундах), через которое произойдет автоматическая перезагрузка документа или переход на другую html-страницу с заданным адресом (url). Например, чтобы производить автоматическую перезагрузку документа каждые 30 секунд, нужно вставить следующий мета-тег:

```
<meta http-equiv="refresh" content="30">
```

Чтобы перейти на страницу по адресу <http://www.alfatrans.ru/> спустя 5 секунд нужно записать так:

```
<meta http-equiv="refresh" content="5; http://www.alfatrans.ru/">
```

Мета-тег "**robots**" формирует информацию о гипертекстовых документах, которая поступает к роботам поисковых систем.

У "robots" могут быть следующие значения:

- index – страница должна быть проиндексирована;
- noindex – страница не индексируется;
- follow – гиперссылки на странице учитываются;
- nofollow – гиперссылки на странице не учитываются;
- all – включает значения index и follow, включен по умолчанию;
- none – включает значения noindex и nofollow.

Например, следующий html-код разрешает индексацию веб-страницы поисковыми машинами и запрещает переход по ссылкам:

```
<meta name="robots" content="index, nofollow">
```

Мета-теги "**author**" и "**copyright**" позволяют указать авторство и владельца контента на странице. Эти значения, как правило, не используются одновременно. Функция "author" и "copyright" – идентификация автора или принадлежности контента на странице. "Author" содержит имя автора веб-страницы, но в случае, если веб-сайт принадлежит какой-либо организации, целесообразнее использовать значение "copyright".

Контейнер `<body> ...</body>` предназначается для выделения той части документа, которая будет визуализирована для пользователя. Контейнер `<body> ...</body>` может иметь несколько атрибутов, которые будут рассмотрены ниже.

Исходный код html-документа сохраняется в текстовый файл с расширением htm или html.

Задание № 1.1. Структура html-документа.

По мере прохождения теоретического материала в практической

части вы должны продемонстрировать следующие возможности языка HTML:

- структурирование и оформление текста;
- использование маркированных и нумерованных списков;
- технология (табличная и блочная, фреймовая) верстка web-страниц;
- графические и текстовые ссылки;
- вставка графических объектов, видео и аудио;
- технология каскадных таблиц стилей CSS;
- формы с полями различных типов (списки, текстовые поля, переключатели и селекторы), а также с кнопками, позволяющими очистить форму и передать ее содержимое на сервер.

На прилагаемом к учебному пособию диске в папке «Учебные материалы» размещены текстовые документы и графические изображения, которые будут использованы при разработке web-страниц и выполнения практических заданий. Некоторые материалы для практических заданий можно брать из общедоступной универсальной интернет – энциклопедии Википедия (<http://www.wikipedia.org/>), сайтов автотранспортных предприятий, например сайта транспортной компании «РУСТРАНС» (<http://premier-log.ru>).

Для выполнения практических заданий необходимо иметь текстовый редактор, браузер для просмотра web-страниц и рабочую папку.

Выбор текстового редактора. Для создания html-документов можно использовать любой текстовый редактор, не вставляющий в документ специальные символы. В качестве таковой будет использоваться текстовый редактор Notepad++.

Notepad++ - это бесплатный аналог обычного блокнота Windows. В отличие от него эта программа намного удобней. Имеет подсветку синтаксиса, автозавершение кода, сворачивание блоков (удобно, если код на несколько страниц), вкладывающиеся открытые файлы

(удобно, если одновременно работать с несколькими html-документами). Все примеры будут выполнены именно в этом редакторе.

Выбор браузера. В качестве Интернет браузера можно использовать один из браузеров MS Internet Explorer, Mozilla Firefox, Opera. Примеры и задания будут демонстрироваться в браузере **Mozilla Firefox**².

Выбор рабочей папки. Для выполнения практических заданий понадобится рабочая папка, в которой нужно сохранять выполненные задания и копировать нужные файлы из папки «Учебные материалы», расположенная на прилагаемом к учебному пособию диске:

1. Выбор и создание рабочей папки. В корневом каталоге диска D: создайте папку **Textbook** (от англ. учебное пособие) и для сохранения выполненных заданий первой главы создайте в папке Textbook новую папку с названием **Chapter_1** (от англ. Глава_1).

Таким образом, рабочей папкой для заданий первой главы будет служить папка «D:\Textbook\Chapter_1». В случае необходимости, можно поместить рабочую папку на другом диске.

2. Создайте html-документ, который будет являться шаблоном для заданий практической части.

В Windows откройте программу Notepad++, установите тип кодировки UTF-8 (команда «Кодировки → Кодировка UTF-8») и наберите или скопируйте в окно редактора текст шаблона web-страниц (Листинг 1.1):

Листинг 1.1

Шаблон html-документа для web-страниц

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta http-equiv="content-type" content="text.html; charset=utf-8">
```

² Mozilla Firefox – свободный браузер на движке Gecko, разработкой и распространением которого занимается Mozilla Corporation. Второй по популярности браузер в мире и первый среди свободного ПО.

```
<meta name="robots" content="noyasa">
<meta name="keywords" content="Перечислите ключевые слова">
<meta name="description" content="Краткое описание страницы">
<title>Название страницы web-сайта</title>
</head>
<body>
    Отображаемая часть html-документа
</body>
</html>
```

Сохраните документ в своей рабочей папке «D:\Textbook\Chapter_1» под именем **temp.html**.

Замечания. 1. Из всех вышеперечисленных к использованию на каждой странице сайта мета-теги и их атрибуты, выберите только некоторые наиболее часто используемые. По мере прохождения будут добавляться новые теги в заголовочной части шаблона.

2. Практические примеры и задания будут строиться на основе шаблона temp.html.

3. Просмотрите созданную web-страницу с помощью браузера Firefox (рис. 1.3). Убедитесь, что в заголовке окна в верхней строке браузера отразился текст «Название страницы web-сайта», а текст, размещенный в контейнере `<body> </body>` отображается в окне браузера.

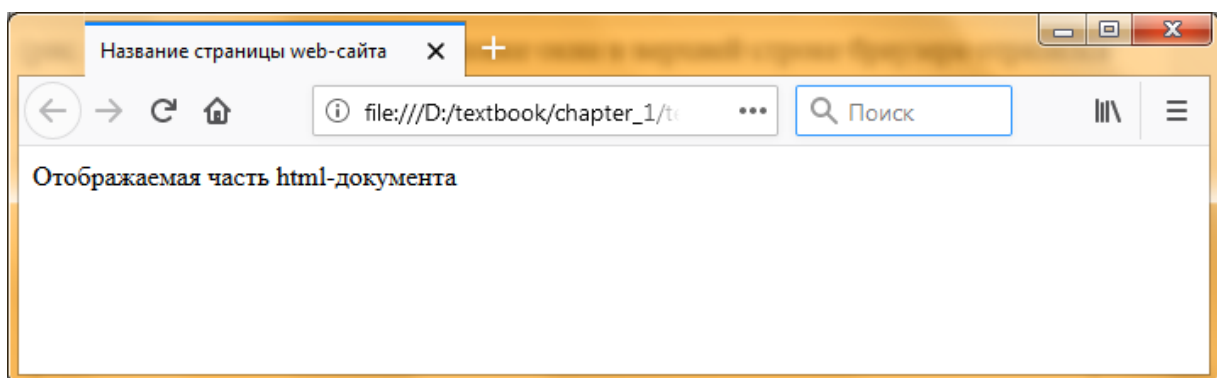


Рисунок 1.3 Результат просмотра страницы temp.html в браузере Firefox

Настройка кодировки в браузере Mozilla Firefox. Иногда случается так, что текст на сайте отображается каракулями (крякозябрами). Причиной этого недоразумения является неправильная кодировка, используемая в браузере. Такие сбои происходят редко, главным образом это вина разработчиков, которые выпуская обновление программы, не учитывают различные факторы, которые накладываются друг на друга и приводят к сбоям. Такое случается и по вине самого браузера на стороне пользователя.

Программа выставляет кодировку самостоятельно, но, если нужно указать принудительно тип кодировки, следует зайти в настройках программы и сделать это вручную. Существует достаточно большое количество кодировок, все они имеют свое собственное имя, но для русскоязычной аудитории рекомендуется использование кодировки под именем «Windows 1251».

Если у вас некорректно отображается содержимое страницы, попробуйте изменить настройки кодировки используемого вами браузера. Для этого укажите, чтобы браузер проводил автоматически определение кодировки. На рисунках 1.4 – 1.6 отображаются примеры изменения настройки кодировки в браузере Mozilla Firefox:

1. В верхнем левом углу нажмите функциональную кнопку «Firefox».
2. В открывшемся окне нажмите на меню «Настройки».

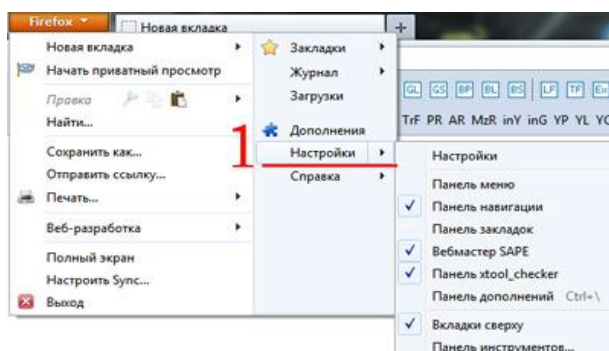


Рисунок 1.4 Выбор меню «Настройки»

3. В окне настроек нужно перейти на вкладку «Содержимое».
4. В этом окне выберите кнопку «Дополнительно» и нажмите ее.

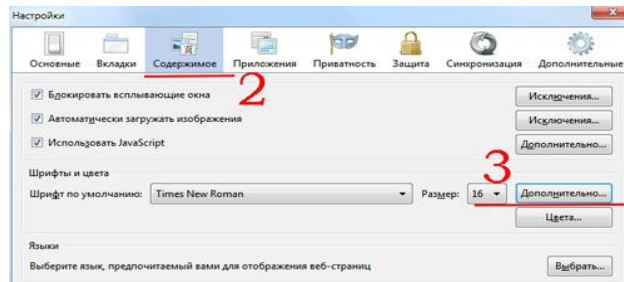


Рисунок 1.5 Выбор команды «Дополнительно»

5. В нижней части открывшегося окна расположен чек-бокс, где отображена действующая на данный момент кодировка, нажмите и выберите кодировку «Windows 1251».

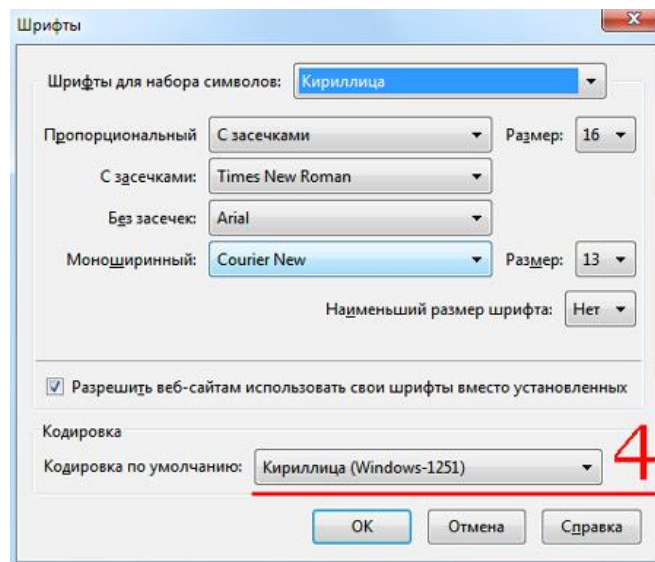


Рисунок 1.6 Выбор кодировки по умолчанию «Windows 1251»

При создании собственного сайта в заголовочной части страницы следует обязательно указывать кодировку символов для текущего документа.

4. Используя соответствующие, мета-теги, выполните самостоятельно следующие задания:

– сохраните файл temp.html в своей рабочей папке под новым именем **task-11.html**;

– задайте ключевые слова: "Мета-теги, HTML, CSS, JavaScript";

- определите следующий текст как описание страницы "Мета-теги могут содержать описание html-страницы, ключевые слова к ней, информацию об авторе, управляющие команды для браузера и поисковых роботов, а также прочую служебную информацию";
- разрешите индексацию web-страницы поисковыми машинами, а переход по ссылкам запретите;
- установите автоматическую перезагрузку текущей веб-страницы через 10 секунд;
- сделайте так, чтобы через 5 секунд после загрузки страницы, браузер перешел на страницу temp.html;
- определите следующий текст "Мета-теги HTML: описание (description), ключевые слова (keywords), заголовок (title) и другие мета-теги" заголовком страницы;
- определите следующий текст "Мета-теги (англ. meta tags) – предназначены для предоставления структурированных мета-данных о веб-странице" как описание web-страницы;
- посмотрите документ в браузере Firefox (рис. 1.7).

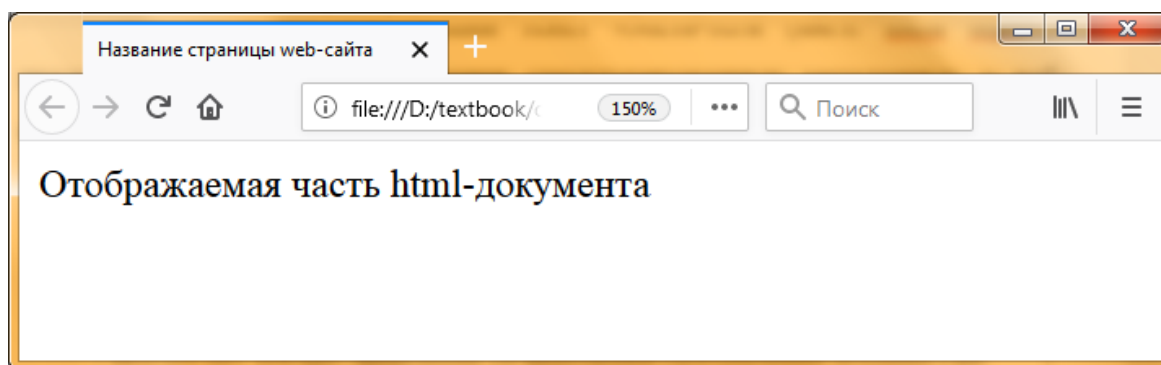


Рисунок 1.7 Результат просмотра страницы task-11.html в браузере Firefox после внесенных изменений

Обратите внимание на то, что через 5 секунд после загрузки страницы task-11.html браузер должен перезагрузить страницу temp.html.

1.2 Форматирование текста

Место переноса в строке определяется только программой просмотра. Символы перевода строки оригинального документа игнорируются, в результате чего текст, отлично смотревшийся в окне редактора, может превратиться в сплошной неудобочитаемый текст в окне программы просмотра.

Существуют специальные команды, выполняющие перевод строки и задающие начало нового абзаца. Также имеется команда, которая запрещает программе браузера каким-либо образом изменять форматирование текста и позволяет точно воспроизвести на экране заданный фрагмент текстового файла.

Тег перевода строки **
** отделяет строку от последующего текста или графики. Тег абзаца **<p>** тоже отделяет строку, но еще добавляет пустую строку, которая зрительно выделяет абзац.

Тег **<p>** (англ. paragraph – параграф) – это абзац. Синтаксис этого тега выглядит следующим образом:

<p>...</p>

Тег абзац не может содержать в себе другие теги абзаца. Следующая конструкция будет не верна:

<p> <p>текст</p> </p>

Для горизонтального выравнивания содержимого абзаца в теге **<p>** применяется атрибут **align**. Тег **<p>** с атрибутом **align** имеет следующий вид:

<p align="значение">...</p>

Значением атрибута является одно из зарезервированных слов:

- **center** – выравнивание каждой строки по центру параграфа;
- **left** – значение по умолчанию, выравнивание слева, каждая строка вплотную прижимается к левой стороне абзаца;
- **right** – выравнивание справа, каждая строка прижимается к правой стороне абзаца;
- **justify** – выравнивание по ширине, каждая строка равномерно распределяется по ширине, в случае необходимости

между элементами или словами добавляются пробелы. Первый и последний элементы строки всегда прижаты к соответствующим сторонам.

Задание № 1.2.Использование тегов перевода строки и абзаца.

1. Откройте шаблон html-документа temp.html. В его заголовочной части произведите следующие изменения:

- в теге <title> измените текст заголовка на "Теги перевода строки и абзаца";
- в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы.

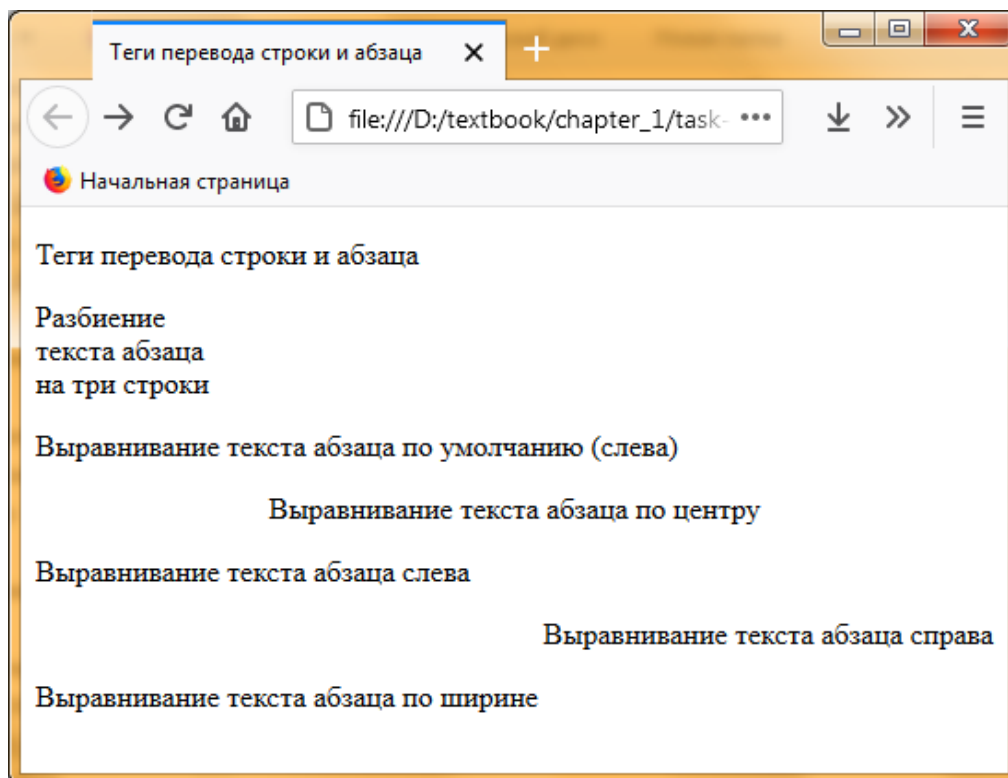


Рисунок 1.8 Результат просмотра html-документа task-12.html

2. В тело документа добавьте нижеприведенный фрагмент html-кода:

<p> Теги перевода строки и абзаца </p>

<p>Разбиение
текста абзаца
 на три строки</p>

<p>Выравнивание текста абзаца по умолчанию (слева)</p>

<p align="center">Выравнивание текста абзаца по центру</p>
 <p align="left">Выравнивание текста абзаца слева</p>
 <p align="right">Выравнивание текста абзаца справа</p>
 <p align="justify">Выравнивание текста абзаца по ширине</p>

3. Сохраните шаблон html-документа под именем **task-12.html** в своей рабочей папке. Просмотрите документ в браузере (рис. 1.8). Самостоятельно поэкспериментируйте с гарнитурой, с цветом и с размером шрифта текста документа.

1.3 Выделение фрагментов текста

Для выделения текстовой информации в html-документе можно использовать различные стили шрифтов. Основные стили шрифтов, поддерживаемых большинством браузеров, заданы в таблице 1.1.

Таблица 1.1

Основные стили шрифтов

Стиль	Элемент или тег	Результат
Bold	 Текст жирный 	Текст жирный
Italic	<I> Текст наклонный </I>	<i>Текст наклонный</i>
Mono spaced	<TT> Текст фиксированный </TT>	Текст фиксированный
Big	Этот текст <BIG> большой </BIG>	Этот текст большой
Small	Этот текст <SMALL> маленький </SMALL>	Этот текст маленький
Sub	Этот текст _{подстрочник}	Этот текст <small>подстрочник</small>
Sup	Этот текст ^{надстрочник}	Этот текст ^{надстрочник}

Возможно использование комбинированных выделений:

Транспортная компания АЛЬФА-ТРАНС

Но при этом необходимо помнить следующее правило записи комбинированных тегов:

- `<тег-1> <тег-2> ... </тег-2> </тег-1>` – правильная запись;
- `<тег-1> <тег-2> ... </тег-1> </тег-2>` – ошибочная запись.

Тег `<pre>` – весь текст, заключенный в контейнере, будет визуализирован браузером точно так, как он визуализирован в исходном коде документа. Кроме того, текст выводится моноширинным шрифтом, что значительно упрощает задачу форматирования текста в колонки. Элемент `<pre>` поддерживается не всеми браузерами, он может иметь атрибут **`width="число символов"`**, который задает ширину отводимого пространства под текст в символах. Внутри тега `<pre>` допустимо использование других элементов для форматирования текста, например: ``, `<i>`.

Синтаксис: `<pre width="число символов">...текст.. </pre>`

Элемент `<pre>` может быть использован:

- для печати стихов, в которых взаимное расположение строк задал сам автор;
- для отображения отформатированного фрагмента программного кода какого-нибудь языка программирования.

Пример 1.

`<p align="center">Фрагмент стихотворения Исхака Машбаша</p>`

`<p align="center"> Черкеска </p>`

`<pre> <p>`

Не зря джигит тобой гордится,
Черкеска белая, как снег,
Когда он, легкий, словно птица,
На скачках обгоняет всех.

`</pre> <p>`

`<p>Исхак Машбаш</p>`

Тег **<blockquote>** – позволяет выделить объемный текст-цитату из общего текста. Синтаксис этого тега следующий:

<blockquote> текст </blockquote>

По умолчанию до и после элемента будет вставлена пустая строка. Также будет добавлен отступ для отделения, содержащегося в элементе текста от остального текста документа.

Пример 2.

```
<p>Фрагмент стихотворения великого русского поэта<p>
```

```
<p align="center"> <b>Кавказ<b> <p>
```

```
<blockquote> <p>
```

```
Кавказ подо мною. Один в вышине <br>
```

```
Стою над снегами у края стремнины; <br>
```

```
Орел, с отдаленной поднявшись вершины, <br>
```

```
Парит неподвижно со мной наравне. <br>
```

```
Отселе я вижу потоков рожденье <br>
```

```
И первое грозных обвалов движенье. <p>
```

```
</blockquote>
```

```
<p>А.С. Пушкин<p>
```

Задание № 1.3.Выделение фрагмента текста.

1. Откройте шаблон html-документа temp.html. В его заголовочной части произведите следующие изменения:

– в теге <title> измените текст заголовка на "Теги выделения текста на web-странице";

– в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы.

2. Чтобы посмотреть, как выглядят различные типы выделения в окне браузера, в тело документа добавьте или скопируйте нижеприведенный фрагмент html-кода:

`<p align="center"> Способы выделения фрагментов текста </p>`

`<p align="center"> Текст жирный </p>`

`<p align="center"> <i> Текст наклонный </i> </p>`

`<p align="center"> <tt> Текст фиксированный </tt></p>`

`<p align="center"> Этот текст <big> большой </big> </p>`

`<p align="center"> Этот текст <small> маленький </small> </p>`

`<p align="center"> Этот текст _{подстрочник} </p>`

`<p align="center"> Этот текст ^{надстрочник} </p>`

3. Сохраните шаблон html-документа под именем **task-13_a.html** в своей рабочей папке. Просмотрите документ в браузере (рис. 1.9).

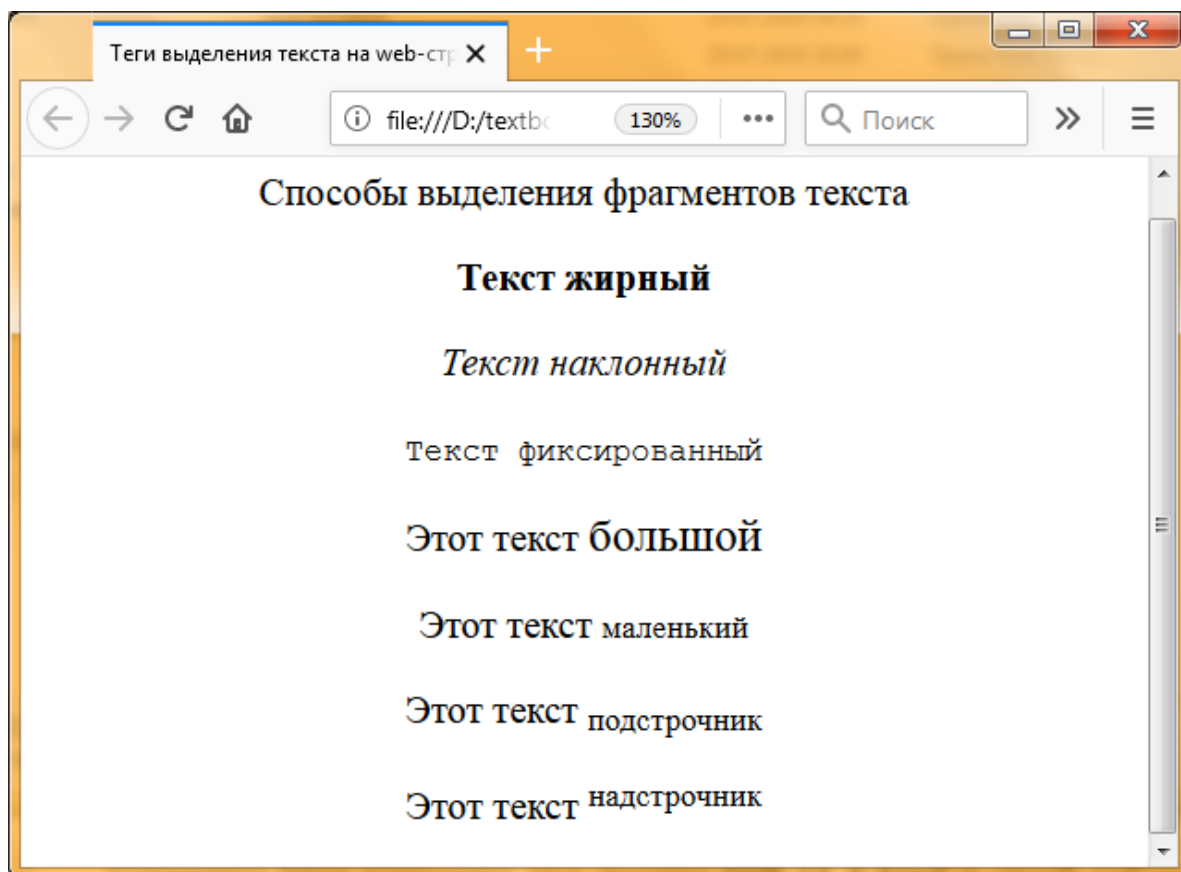


Рисунок 1.9 Результат просмотра html-документа task-13_a.html в браузере

4. Добавьте в документ task-13_a.html ниже следующий html – код:

`<p align=center>Способы оформления формул</p>`

`<p> (a + b)²=a²+2ab+b²</p>`

`<p> CO₂ – углекислый газ</p>`

Сохраните html-документ под именем **task-13_b.html** в своей рабочей папке. Просмотрите документ в браузере (рис. 1.10).

5. Добавьте в документ task-13_b.html фрагменты кода из примера 1, 2 и сохраните html-документ под именем **task-13.html** в своей рабочей папке. Просмотрите документ в браузере (рис. 1.11).

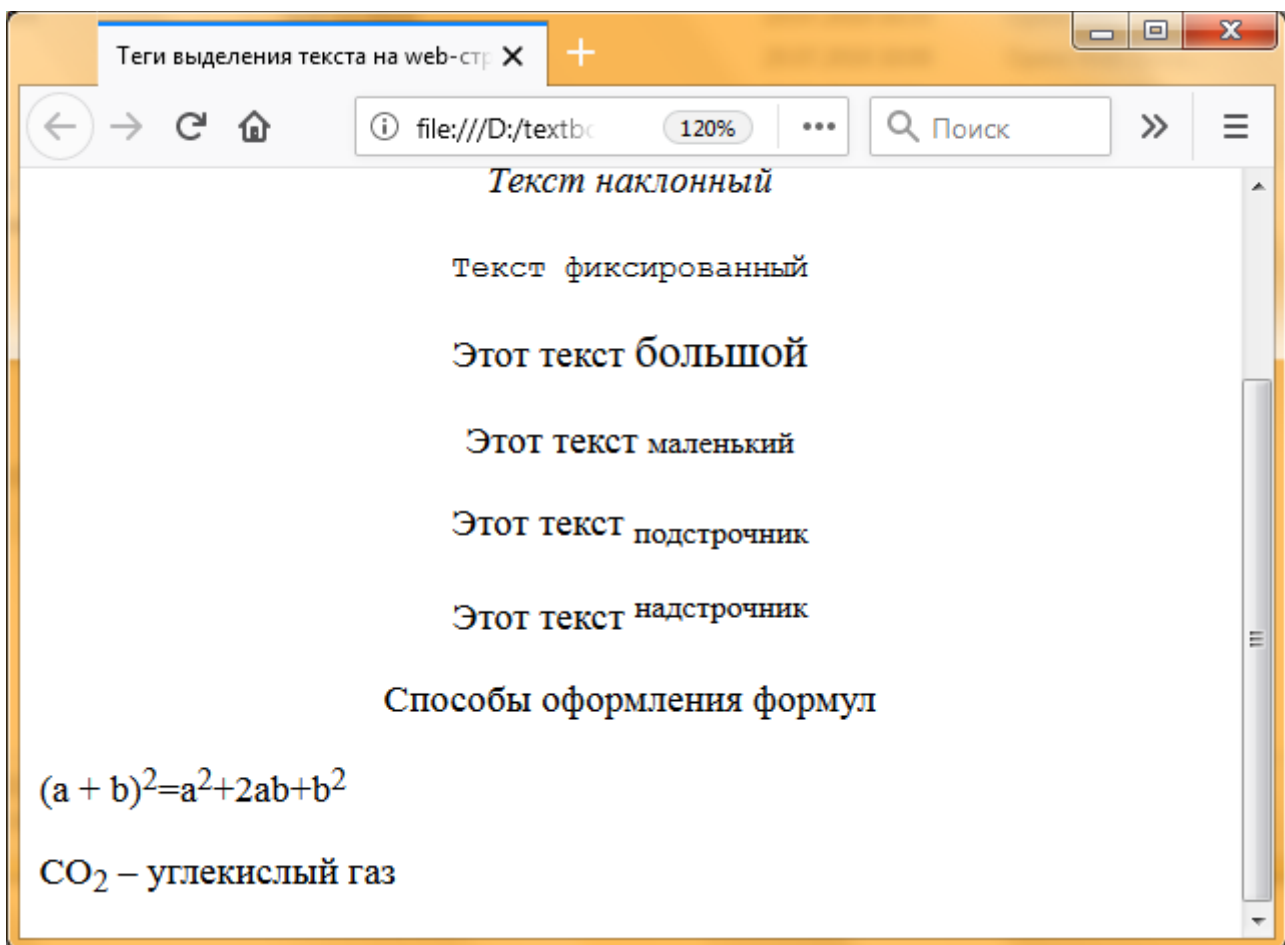


Рисунок 1.10 Результат просмотра html-документа task-13_b.html
в браузере

1.4 Задание размера символов

Существуют два способа управления размером текста, изображаемого браузером:

- использование стилей заголовка;
- задание размера шрифта основного документа или размера текущего шрифта.

В web-документе можно использовать шесть тегов заголовков (от <h1> до <h6>). h1 – самый большой, а h6 – соответственно, самый маленький. Каждому тегу соответствует конкретный стиль, заданный параметрами настройки браузера.

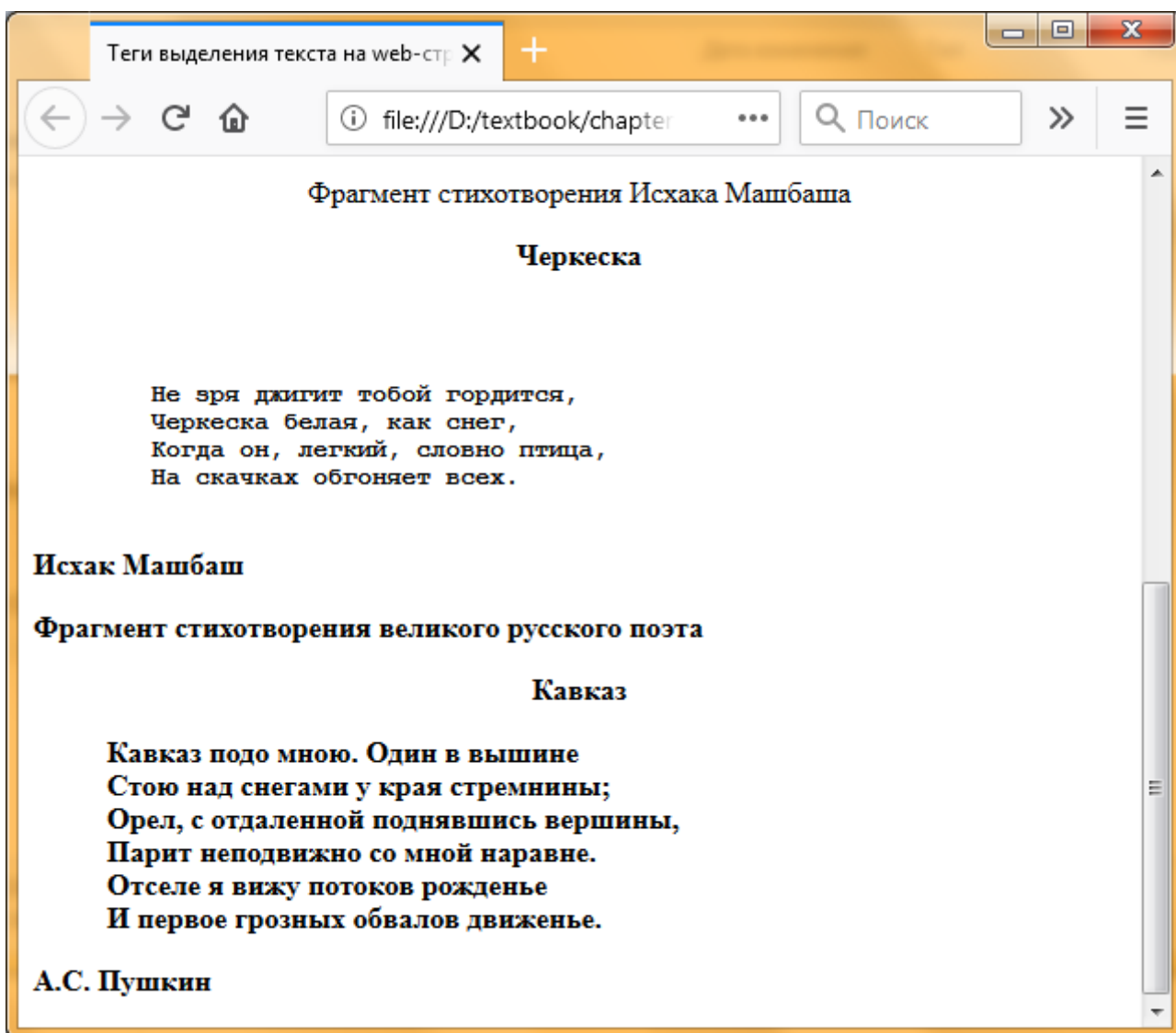


Рисунок 1.11 Результат просмотра фрагмента страницы task-13.html в браузере Firefox

Задание № 1.4. Использование стилей заголовка.

1. Откройте шаблон html-документа temp.html. В его заголовочной части произведите следующие изменения:

– в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;

– в теге <title> измените текст заголовка на "Теги заголовков на web-странице".

2. В тело документа добавьте нижеприведенный фрагмент html-кода:

```
<p align="center"> <b> Стили заголовка </b></p>
```

```
<p> <h1> Заголовок первого уровня</h1></p>
```

```
<p> <h2> Заголовок второго уровня </h2></p>
```

```
<p> <h3> Заголовок третьего уровня </h3></p>
```

```
<p> <h4> Заголовок четвертого уровня </h4></p>
```

```
<p> <h5> Заголовок пятого уровня </h5></p>
```

```
<p> <h6> Заголовок шестого уровня </h6></p>
```

3. Сохраните шаблон html-документа под именем **task-14.html** в своей рабочей папке. Просмотрите документ в браузере Firefox (рис. 1.12).

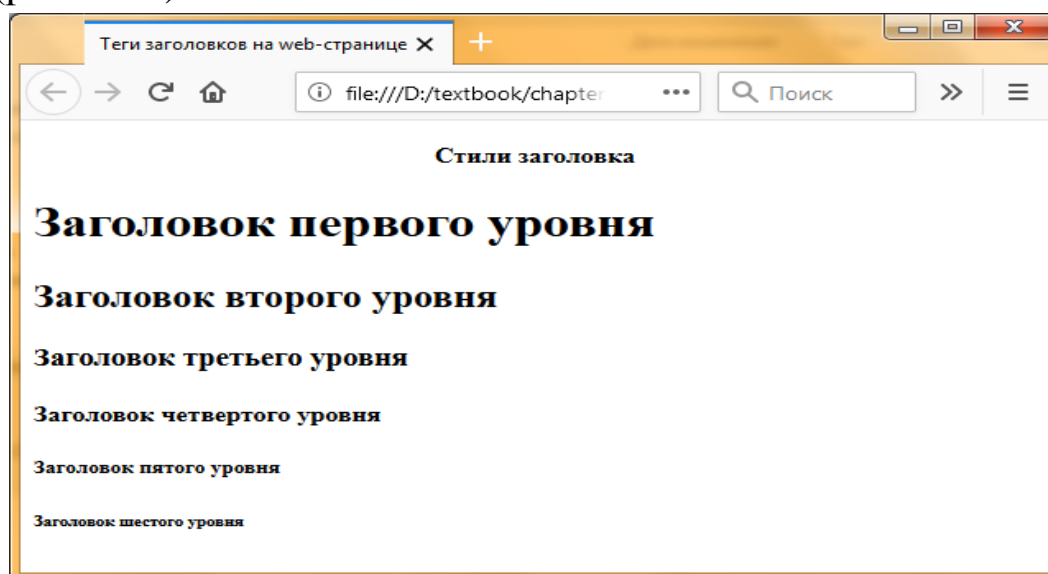


Рисунок 1.12 Результат просмотра html-документа task-14.html в браузере

1.5 Гарнитура, цвет и размер шрифта

Для управления гарнитурой, цветом и размером текста используется тег ``. Для этих целей предназначены соответствующие атрибуты:

– **face** – задает тип (гарнитура) шрифта, использующийся для отображения текста. Если заданный шрифт отсутствует на компьютере, то браузер заменит его наиболее близким. Можно задавать несколько шрифтов:

```
<font face="Comic Sans MS, Arial">Шрифт Comic Sans MS</font>
```

В этом случае браузер будет использовать первый заданный шрифт Comic Sans MS. В случае если он не установлен у пользователя, то браузер будет искать второй шрифт Arial. Шрифты Arial и Times New Roman являются стандартными шрифтами Windows, поэтому рекомендуется использовать по возможности именно эти шрифты на страницах.

– **color** – задает цвет текста шестнадцатеричным числом (`#FF00FF`) либо именем. При задании цвета шестнадцатеричным числом можно представить этот цвет разложенным на три составляющие: красный (R – Red), зеленый (G – Green) и синий (B – Blue), каждый из которых имеет значение от 0 до FF. В этом случае мы имеем дело с так называемым форматом RGB. В таблице 1.2 приведены записи 16 стандартных цветов в различных форматах (по названию, шестнадцатеричном и RGB), которые можно записать в значении свойства color.

Эти цвета поддерживают все браузеры. Кроме них есть еще порядка 130 дополнительных ключевых слов для различных оттенков цветов. Полную таблицу этих цветов можно посмотреть в справочниках.

Использование таких ключевых слов допустимо, но есть риск, что какое-нибудь слово не воспримется браузером. Поэтому рекомендуется вместо ключевых слов записывать шестнадцатеричный код цвета.

– **size** – задает значение размера шрифта, выраженный в

условных величинах от 1 до 7.

Можно комбинировать несколько атрибутов в одном теге .

Задание № 1.5. Установка гарнитуры, цвета и размера шрифта.

1. Откройте шаблон html-документа temp.html. В его заголовочной части произведите следующие изменения:

– в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;

Таблица 1.2

Запись 16 стандартных цветов в различных форматах

Ключевое слово цвета	Русское название	HEX-код	RGB
red	Красный	#FF0000	255, 0, 0
maroon	Коричневый	#800000	128, 0, 0
yellow	Желтый	#FFFF00	255, 255, 0
olive	Оливковый	#808000	128, 128, 0
lime	Лайм	#00FF00	0, 255, 0
green	Зеленый	#008000	0, 128, 0
aqua	Сине-зеленый	#00FFFF	0, 255, 255
teal	Сине-зеленый	#008080	0, 128, 128
blue	Синий	#0000FF	0, 0, 255
navy	Темно-синий	#000080	0, 0, 128
fuchsia	Фуксия	#FF00FF	255, 0, 255
purple	Фиолетовый	#800080	128, 0, 128
white	Белый	#FFFFFF	255, 255, 255
silver	Светло-серый	#C0C0C0	192, 192, 192
gray	Серый	#808080	128, 128, 128
black	Черный	#000000	0, 0, 0

– в теге <title> измените текст заголовка на "Гарнитура, цвет и размер шрифта на web-странице".

2. В тело документа добавьте нижеприведенный фрагмент html-кода:

```
<h1><b>Задание гарнитуры, цвета и размера шрифта</h1></b>
```

```
<font face="Arial" color="blue" size="2">
```

```
<p>Шрифт-Arial. Цвет текста-синий, формат-словами.
```

```
Размер=2</p>
```

```
</font>
```

```
<font face="Times New Roman" color="#0095B6" size="3">
```

```
<p>Шрифт-Times New Roman. Цвет текста "Вода пляжа Бонди"  
в 16-тиричном формате. Размер=3</p>
```

```
</font>
```

```
<font face="Comic Sans MS, Times New Roman, Arial"  
color="RGB(255,0,255)" size="5">
```

```
<p>Шрифт- Comic Sans MS, если его нет, то Times New Roman  
или Arial. Цвет текста "Фуксия" в формате RGB. Размер=5</p>
```

```
</font>
```

3. Сохраните шаблон html-документа под именем **task-15.html** в своей рабочей папке. Просмотрите документ в браузере (рис. 1.13). Самостоятельно поэкспериментируйте с гарнитурой, с цветом и с размером шрифта текста документа.

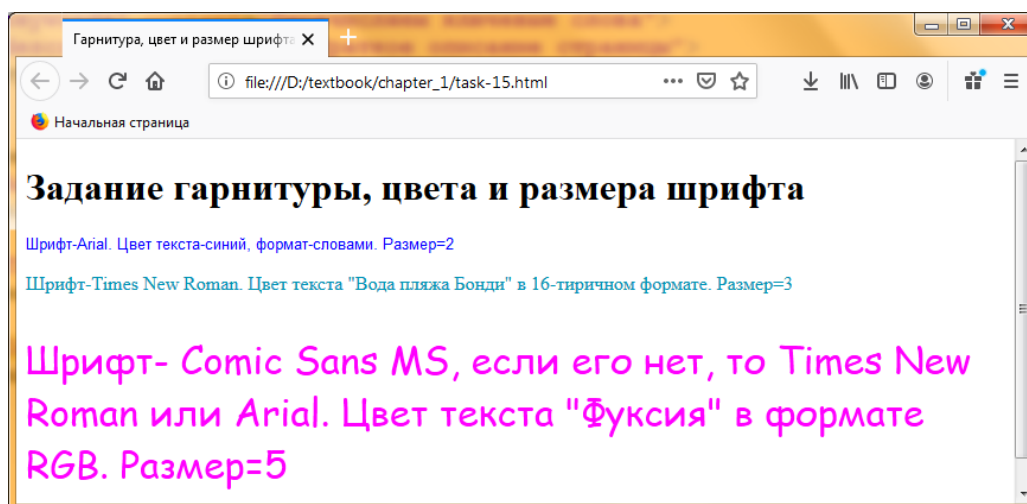


Рисунок 1.13 Результат просмотра страницы task-15.html в браузере

1.6 Задание цвета фона и текста web страницы

При изображении фона и текста браузеры используют цвета, установленные по умолчанию, и они заданы параметрами настройки браузера. Если вы хотите задать другие цвета, то это надо делать в начале файла html-документа в теге <body>. Для этого в начальном теге <body> используются атрибуты:

– **background** – задает графическое изображение, которое как черепица заполнит фон документа. Синтаксис этого атрибута задается следующим образом:

background="имя графического файла"

В качестве фонового изображения обычно выбирается маленькое не слишком яркое изображение, чтобы создать интересный, но не навязчивый тип фона. Кроме того, маленькая простая картинка загружается из сети гораздо быстрее, чем сложное полноэкранное изображение. Браузер автоматически повторяет (в виде мозаики) картинку по вертикали и по горизонтали, заполняя все окно.

– **bgcolor** – атрибут задает цвет фона документа при помощи шестнадцатеричных значений интенсивности цветов RGB или при помощи строчного литерала, соответствующего названию цвета. Синтаксис этого атрибута задается следующим образом:

bgcolor="цвет"

– **text** – задает используемый по умолчанию цвет текста, который не является гиперссылкой. По умолчанию такой текст будет черным. Синтаксис этого атрибута выглядит следующим образом:

text="цвет"

Замечание. В таблице 1.2 приведены записи цвета в формате RGB и соответствующие им названия.

Задание № 1.6. Установка цвета фона и текста.

1. Откройте шаблон html-документа temp.html. В его заголовочной части произведите следующие изменения:

– в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;

– в теге <title> измените текст заголовка на "Установка цвета фона и цвета текста по умолчанию".

2. В тег <body> добавьте значения для атрибутов background и text:

```
<body bgcolor="Aqua" text="black">
```

```
<p>Черный текст на сине-зеленом фоне</p>
```

```
</body>
```

3. Сохраните шаблон html-документа под именем **task-16_a.html** в своей рабочей папке. Просмотрите документ в браузере (рис. 1.14).

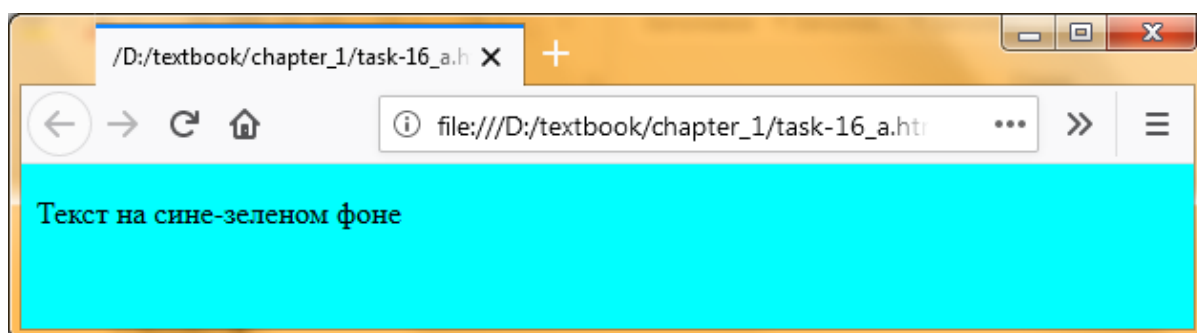


Рисунок 1.14 Результат просмотра страницы task-16_a.html в браузере

4. В html-документ task-16_a.html измените параметры тега <body> и текст документа следующим образом:

```
<body background="fon_01.jpg" text="red">
```

```
<p>Красный текст на фоне рисунка fon_01.jpg</p>
```

```
</body>
```

Замечание. 1. На прилагаемом диске в папке «Учебные материалы\Глава_1\Фоновые рисунки» размещен набор файлов графических изображений для фона.

2. В данном примере файл изображения fon_01.jpg должен быть размещен в каталоге, где находится файл task-16_a.html.

Поэтому скопируйте файл `fon_01.jpg` с прилагаемого диска в свою рабочую папку.

5. Сохраните html-документ `task-16_a.html` под именем **task-16_b.html**. Просмотрите документ в браузере (рис. 1.15).

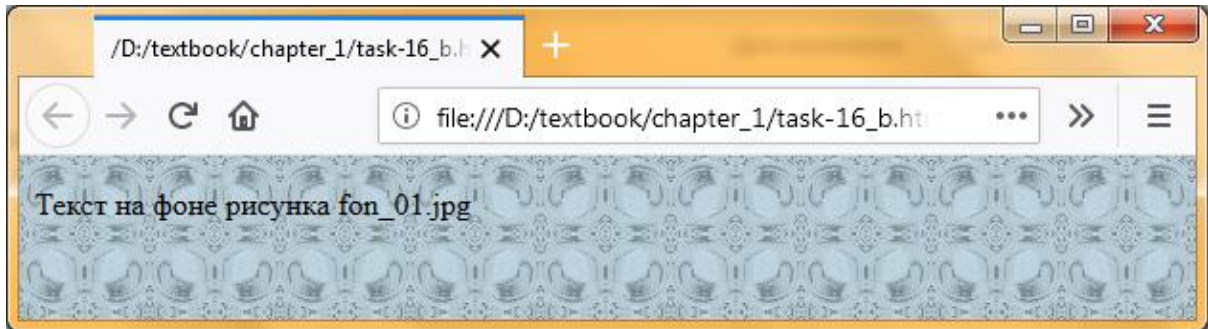


Рисунок 1.15 Результат просмотра страницы `task-16_b.html` в браузере

6. В html-документ `task-16_b.html` добавьте следующий фрагмент html-кода:

```
<font color="blue">
```

```
<p>Переопределение красного текста на синий</p>
```

```
</font>
```

```
<p>Снова красный текст на фоне рисунка fon_01.jpg</p>
```

7. Сохраните документ под именем **task-16_c.html** и просмотрите его в браузере (рис 1.16). Сделайте выводы.

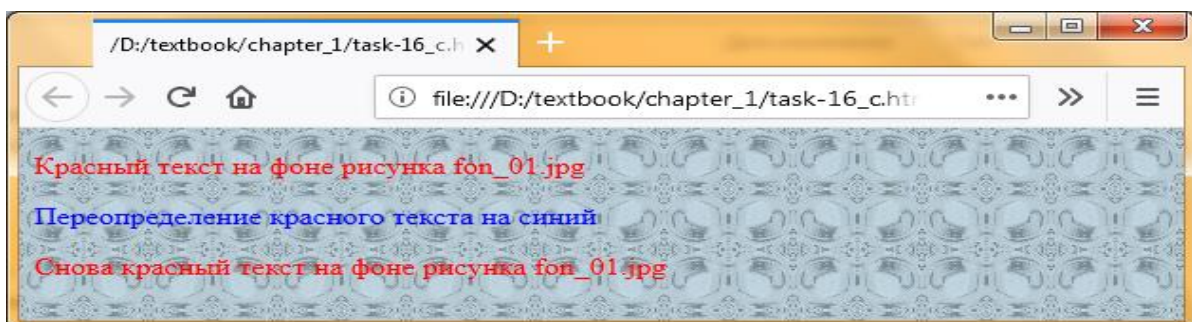


Рисунок 1.16 Результат просмотра страницы `task-16_c.html` в браузере

1.7 Вставка горизонтальных линий и эффектов движения в текст страницы

Метка `<hr>` описывает горизонтальную линию. Метка может дополнительно включать атрибуты **size** (определяет толщину линии в пикселях) и/или **width** (определяет размах линии в процентах от ширины экрана).

Любое движение текста в строке определяется двойным тегом `<MARQUEE>`, который имеет несколько параметров, заданные в таблице 1.3.

Таблица 1.3

Параметры тега `<MARQUEE>`

Параметр	Описание	Значения
direction	Направление движения	right - справа-налево
		left - слева-направо (по умолчанию)
behavior	Цикличность движения	scrol - движение по непрерывному циклу
		slide - движение до края и там остановка
		alternate - движение до края и возврат к другому краю
bgcolor	Цвет фона строки	все известные цвета
width	Ширина строки	в пикселях или процентах
height	Высота строки	в пикселях или процентах
font size	Размер текста в строке	от 1 ...7

Задание № 1.7. Пример использования горизонтальных линий и эффектов движения текста.

1. Откройте шаблон html-документа temp.html. В его заголовочной части произведите следующие изменения:

– в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;

– в теге `<title>` измените текст заголовка на "Горизонтальные линии и эффекты движения текста".

2. В тело документа добавьте нижеприведенный фрагмент html-кода:

```
<h1>Коллекция горизонтальных линий</h1>
```

```
<hr size="2" width="100%"><br>
```

```
<hr size="4" width="50%"><br>
```

```
<hr size="8" width="25%"><br>
```

```
<hr size="16" width="12%"><br>
```

Сохраните шаблон html-документа под именем **task-17_a.html** в своей рабочей папке. Просмотрите документ в браузере Firefox (рис. 1.17).

3. Добавьте в документ task-17_a.html фрагмент, демонстрирующий эффект движения текста на странице html-документа:

```
<marquee>Я бегущая строка</marquee>
```

```
<p><font size="5" color="red"><marquee>Я бегущая строка</marquee></font></p>
```

```
<p><font size="7"><marquee direction= right>Я бегущая строка</marquee></font></p>
```

```
<p><font size="5" color="green"><marquee>Я бегущая строка</marquee></font></p>
```

4. Сохраните документ task-17_a.html под новым именем **task-17_b.html**. Просмотрите документ в браузере Firefox. Сделайте ВЫВОДЫ.

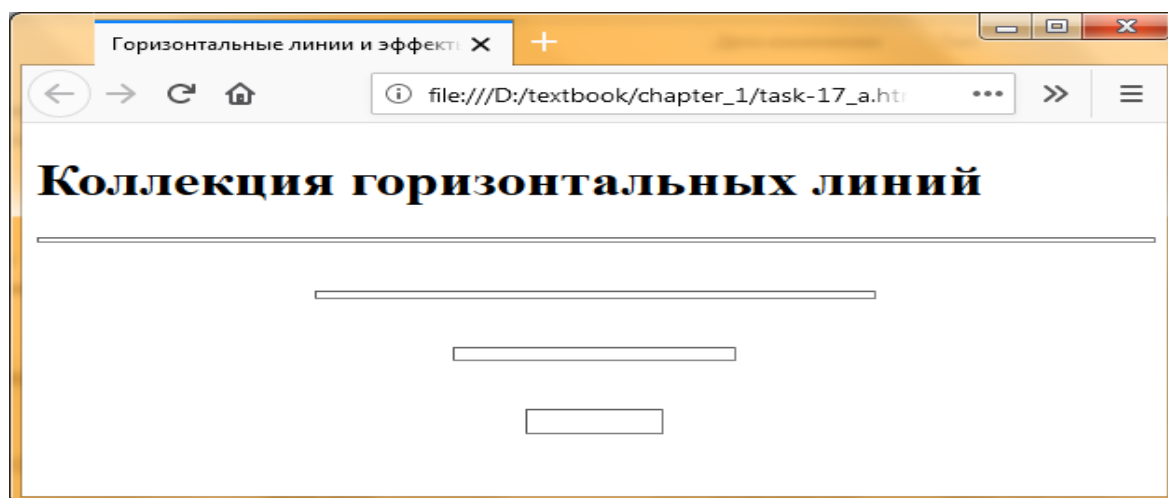


Рисунок 1.17 Результат просмотра страницы task-17_a.html в браузере

5. Внесите изменения в текст документа так, чтобы движение было по непрерывному циклу, и от края до края и назад. Задайте для фона движущейся строки размер и цвет. Сохраните измененный документ под именем **task-17_c.html**. Просмотрите документ в браузере Firefox. Сделайте выводы.

1.8 Теги списков

Существует три основных вида списков в HTML-документе:

- нумерованный;
- маркированный;
- список описаний.

Можно создавать вложенные списки, используя различные теги списков или повторяя одни внутри других. Для этого необходимо разместить одну пару тегов (стартовый и завершающий) внутри другой.

В нумерованном списке браузер автоматически вставляет номера элементов по порядку. Это означает, что если вы удалите один или несколько элементов нумерованного списка, то остальные номера автоматически будут пересчитаны.

Нумерованный список начинается стартовым тегом `` и завершается тегом ``. Каждый элемент списка начинается с тега ``. Например:

```
<ol>
  <li> Программирование
  <li> Алгоритмизация
  <li> Проектирование
</ol>
```

Тег `` может иметь параметры:

TYPE=A|a|I|i|1 – определяет вид счетчика:

- A – большие латинские буквы (A, B, C ...)
- a – маленькие латинские буквы (a, b, c ...)
- I – большие римские цифры (I, II, III ...)

- i – маленькие римские цифры (i, ii, iii...)
- 1 – обычные цифры (1, 2, 3...)

START = "n" – определяет число, с которого начинается отсчет.

Маркированный список начинается стартовым тегом **** и завершается тегом ****. Каждый элемент списка начинается с тега ****. Например:

```
<ul>
  <li> Программирование
  <li> Алгоритмизация
  <li> Проектирование
</ul>
```

Тег **** может иметь параметр, определяющий внешний вид маркера:

TYPE=disc|circle|square>

disc – заполненный кружок;

circle – незаполненный кружок;

square – заполненный квадратик.

Список определений служит для создания списков типа «термин» -«описание». Он начинается с тега **<dl>** и завершается тегом **</dl>**. Каждый термин начинается тегом **<dt>**, а описание – тегом **<dd>**.

Задание 1.8. Нумерованные, маркированные списки и список определений.

1. Откройте шаблон html-документа temp.html. В его заголовочной части произведите следующие изменения:

- в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;

- в теге <title> измените текст заголовка на "Нумерованные, маркированные списки и список определений".

2. В тело документа добавьте нижеприведенный фрагмент html-кода:

```
<p><h1>Нумерованный список </h1></p>
```

```
<ol>
```

```
  <li> Программирование
```

```
  <li> Алгоритмизация
```

```
  <li> Проектирование
```

```
</ol>
```

```
<p><h1>Маркированный список </h1></p>
```

```
<ul>
```

```
  <li> Программирование
```

```
  <li> Алгоритмизация
```

```
  <li> Проектирование
```

```
</ul>
```

```
<p><h1>Списков определений </h1></p>
```

```
<dl>
```

```
  <dt> <b> Отдел маркетинга </b>
```

```
    <dd> Данный отдел занимается продвижением продуктов
```

и услуг

```
  <dt> <b> Финансовый отдел </b>
```

```
    <dd> Данный отдел занимается всеми финансовыми
```

операциями

```
  <dt> <b> Отдел кадров </b>
```

```
    <dd> Данный отдел занимается учетом и набором новых
```

```
    сотрудников фирмы, распределением отпусков,
```

```
    отслеживанием больничных листов и т.д.
```

```
</dl>
```

3. Сохраните шаблон html-документа под именем **task-18.html** в своей рабочей папке. Просмотрите документ в браузере (рис. 1.18).

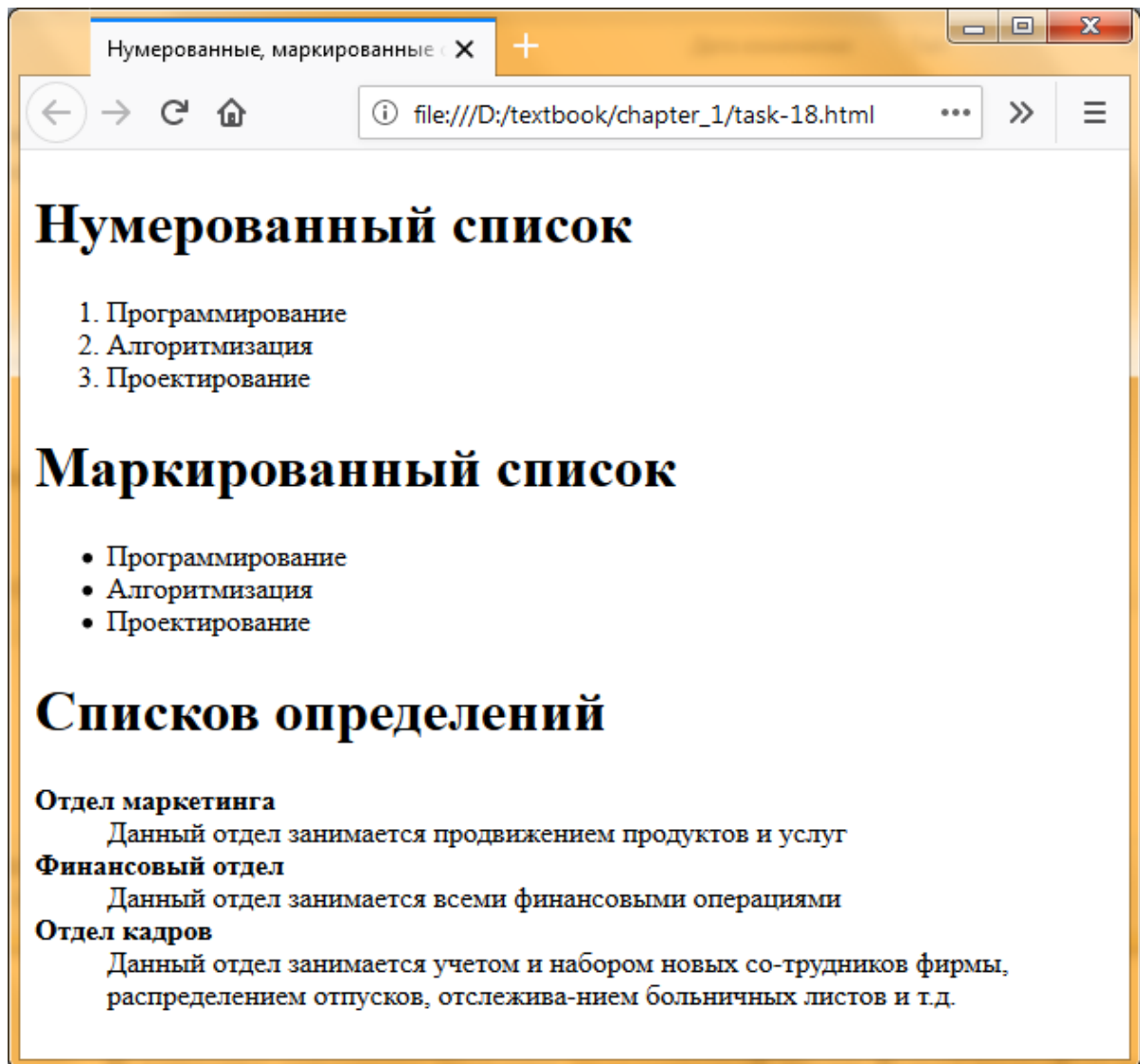


Рисунок 1.18 Результат просмотра страницы task-18.html в браузере

Практическая часть

В качестве сквозного примера практической части будет разрабатываться небольшой Web-сайт транспортной организации по перевозке грузов под условным названием «Альфа-Транс». Сайт должен состоять из нескольких связанных между собой статических html-страниц.

На сайтах различных фирм, организаций почти всегда имеется страница, содержащая краткую информацию об организации, роде ее деятельности. Создание сайта начинается с такой страницы.

Документ, содержащий информацию о транспортной организации «Альфа-Транс», хранится в файле «O нас.txt» папки «Учебные материалы\Глава_1». Материалы из этого документа будут использоваться в дальнейшем.

Для начала в папке «D:\Textbook\Chapter_1» создайте папку, в которой будете сохранять разрабатываемые web-страницы. В дальнейшем с русским языком в названии папок и файлов могут возникнуть проблемы, поэтому имя папки нужно писать в англоязычной раскладке клавиатуры. Например, в качестве папки для сайта можно определить папку с именем «AlfaTrans.ru» – по названию транспортной компании. Далее выполните следующие действия:

1. В папке «D:\Textbook\Chapter_1» создайте папку с именем «AlfaTrans.ru».

2. В программе Notepad++ откройте шаблон temp.html, размещенный в папке «D:\Textbook\Chapter_1» и в заголовочной части произведите следующие изменения:

- в мета-теге "keywords" запишите ключевые слова в соответствии с содержанием web-страницы: "грузоперевозки, перевозки, транспортная компания, альфа-транс, доставка, России, Адыгея, Майкоп";

- в мета-теге "description" запишите описание страницы в соответствии с содержанием web-страницы: "Транспортная компания «АЛЬФА-ТРАНС» осуществляет перевозки в большое количество городов России, по р. Адыгея, у нас в реестре более 2000 населенных пунктов".

- в теге <title> измените текст заголовка на "Транспортная компания АЛЬФА-ТРАНС – выгодные грузоперевозки";

3. В тело документа добавьте или скопируйте нижеприведенный фрагмент html-кода (Листинг 1.2):

html-документа об организации

О нас

ТРАНСПОРТНАЯ КОМПАНИЯ «АЛЬФА-ТРАНС»

Вопрос быстрой доставки грузов по территории страны всегда остается актуальным для физических и юридических лиц. Сегодня наиболее оптимальным вариантом считаются перевозки грузов автомобильным транспортом, поскольку карта автомобильных дорог обширна и разнообразна, позволяет достигать даже наиболее отдаленных уголков России. Стоимость услуги является оптимальной, а современные технологии в сфере логистики обеспечивают целостность и сохранность грузов на протяжении всего времени следования, не зависимо от расстояния.

«АЛЬФА-ТРАНС» - крупная транспортно-экспедиционная компания. На рынке с 2000 года. Оказываем услуги перевозки грузов весом от 800 кг по населенным пунктам республики Адыгея и по регионам всей России автотранспортом любой грузоподъемности.

В работе мы используем как собственный автопарк, так и привлеченный транспорт, что позволяет охватывать все направления грузовых перевозок, в том числе сложные и непопулярные.

Мы хотим, чтобы сайт сделал более удобным и мобильным общение между нами и нашими постоянными клиентами.

4. Сохраните файл под именем **about_1.html** в папке «AlfaTrans.ru». Обратите внимание, что расширение у файла должно быть именно html.

5. Просмотрите созданную Web-страницу с помощью браузера Firefox (рис.1.19). Убедитесь, что название Web-страницы "Транспортная компания «Альфа-Транс»" отразилось в верхней строке браузера.

Обратите внимание, что символы перевода строки документа игнорируются браузером, и в окне просмотра отображается сплошной неудобочитаемый текст. Изменяя размеры окна просмотра, обратите внимание на то, что текст форматируется в зависимости от размеров окна.

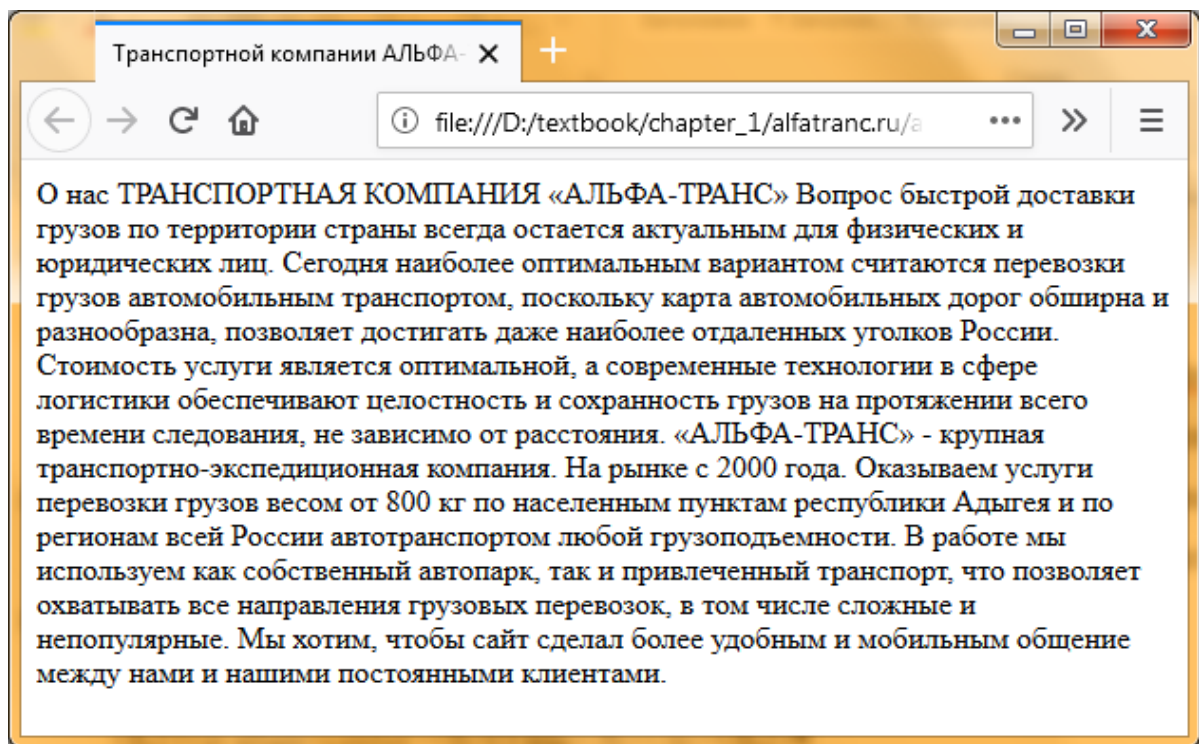


Рисунок 1.19 Результат просмотра страницы about_1.html
в браузере Firefox

6. Внесите следующие изменения в файл about_1.html:

```
<h2><p align="center"><b>О нас</b></p></h2>
```

```
<h2><p align="center"><b>ТРАНСПОРТНАЯ КОМПАНИЯ  
«АЛЬФА-ТРАНС» </b></p></h2>
```

```
<h3><p align="justify"> Вопрос быстрой доставки грузов по  
территории страны всегда остается актуальным для физических и  
юридических лиц. <br>
```

```
Сегодня наиболее оптимальным вариантом считаются перевозки  
грузов автомобильным транспортом, поскольку карта автомобильных  
дорог обширна и разнообразна, позволяет достигать даже наиболее  
отдаленных уголков России.<br>
```

```
Стоимость услуги является оптимальной, а современные  
технологии в сфере логистики обеспечивают целостность и  
сохранность грузов на протяжении всего времени следования, не  
зависимо от расстояния.<br>
```

«АЛЬФА-ТРАНС» - *крупная* транспортно-экспедиционная компания.

На рынке с 2000 года. Оказываем услуги перевозки грузов весом от 800 кг по населенным пунктам республики Адыгея и по регионам всей России автотранспортом любой грузоподъемности.

В работе мы используем как собственный автопарк, так и привлеченный транспорт, что позволяет охватывать все направления грузовых перевозок, в том числе сложные и непопулярные.

Мы хотим, чтобы сайт сделал более удобным и мобильным общение между нами и нашими постоянными клиентами.

7. Сохраните документ под именем **about_2.html** в папке «AlfaTrans.ru» и просмотрите в браузере (рис. 1.20). Обратите внимание, как меняется отображаемый в окне браузера текст документа. Влияют ли теги выделения шрифта **** и *<i>* на текст? Что изменится, если убрать контейнер **... **? Если убрать контейнер *<i>... </i>*? Если убрать атрибут выравнивания абзаца?

8. Внесите изменения по форматированию текста страницы `about_1.html`, например, в заголовочной части внесите следующие изменения (тип, цвет и размер шрифта, фон страницы подберите по своему усмотрению):

```
<body background="fon_01.jpg">
  <font face="Times New Roman, Arial" color="blue">
    <h2><p align="center"><b>О нас</b></p></h2>
  </font>
  <hr size="2" width="100%"><br>
  <font face="Comic Sans MS, Times New Roman, Arial"
color="red">
    <h2><p align="center"><b>ТРАНСПОРТНАЯ
КОМПАНИЯ «АЛЬФА-ТРАНС» </b></p></h2>
  </font>
```

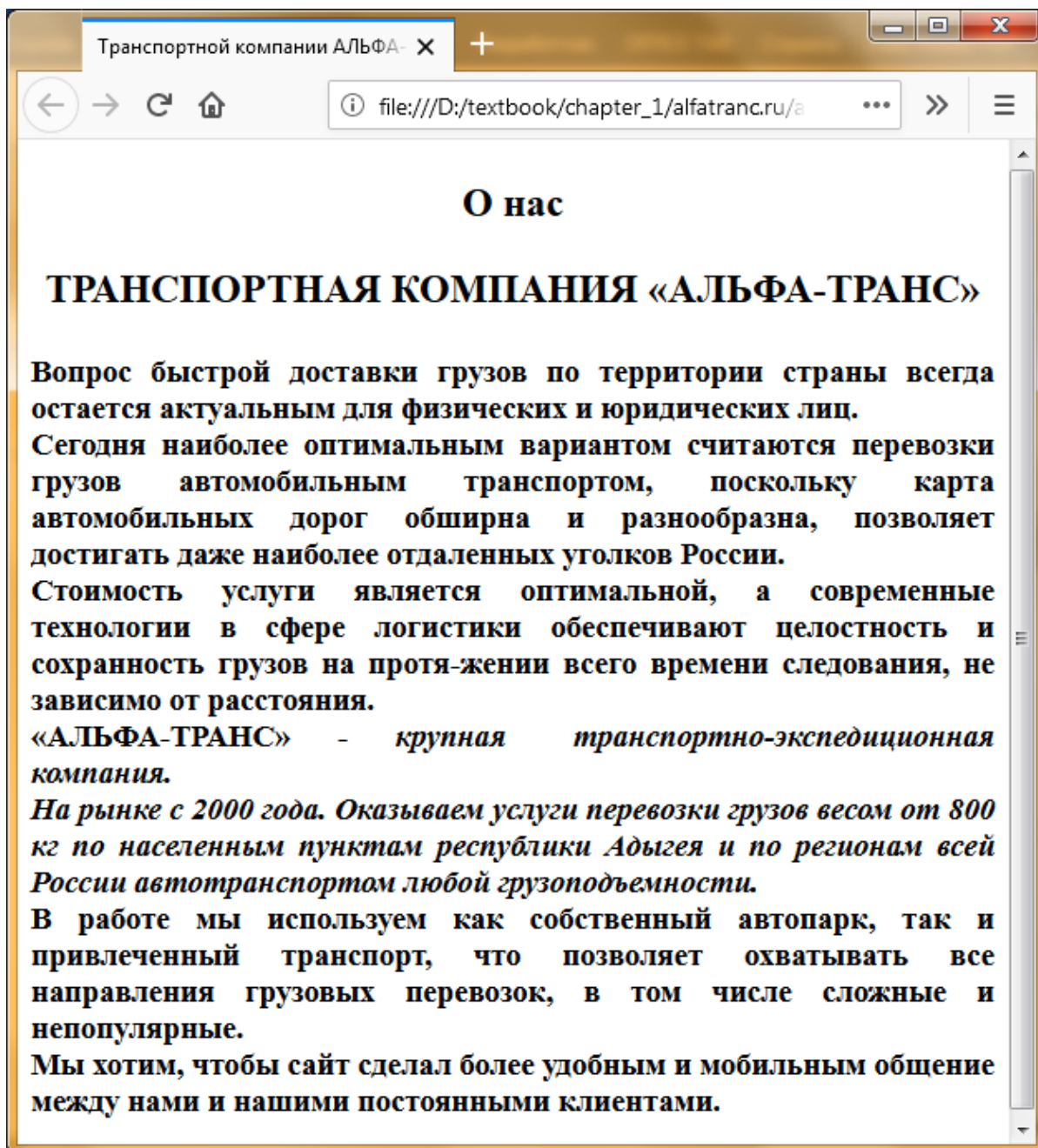


Рисунок 1.20 Результат просмотра страницы about.html после внесенных изменений в браузере Firefox

Окончательно сохраните документ под именем **about.html** в папке «AlfaTrans.ru» и просмотрите в браузере (рис. 1.21).

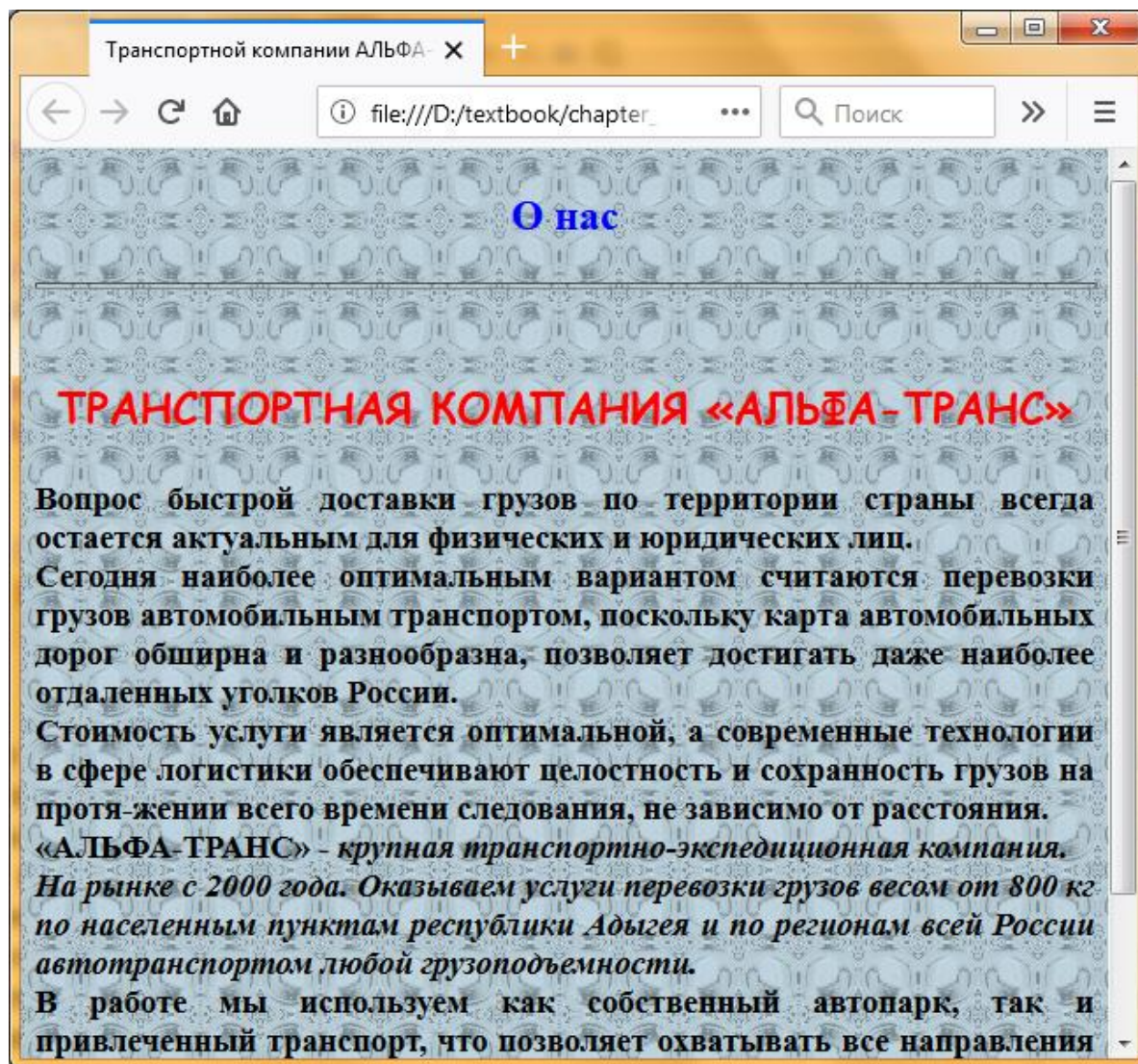


Рисунок 1.21 Окончательный результат страницы about.html

Тестовые задания для самоконтроля

Уровень А

1. Web-страница (документ HTML) представляет собой ...
 - A. текстовый файл с расширением txt или doc
 - B. текстовый файл с расширением htm или html
 - C. двоичный файл с расширением com или exe
 - D. графический файл с расширением gif или jpg
2. Программа для просмотра гипертекстовых страниц называется ...

- A. сервер
- B. протокол
- C. html
- D. браузер

3. С помощью какого мета-тега определяется тип документа, его кодировка и служит для правильного отображения символов браузером?

- A. content-language
- B. content-type
- C. frefresh
- D. robots

4. Укажите какие значения для мета-тега robots нужно выбрать, чтобы разрешать индексацию web-страницы поисковыми машинами, и запрещать переход по ссылкам...

- A. index
- B. noindex
- C. follow
- D. nofollow
- E. all
- F. none

5. Следующий html-код:

```
<meta name="robots" content="all"> ...
```

- A. разрешает индексацию web-страницы поисковыми машинами
- B. запрещает индексацию web-страницы поисковыми машинами
- C. разрешает переход по ссылкам
- D. запрещает переход по ссылкам

6. Какой из перечисленных тегов обозначает начало абзаца?

- A. <h1>
- B.

- C. <p>
- D. <u>
- E. < title>

7. Укажите фрагмент html-документа, задающий подчеркивание текста.

- A. `<u>текст</u>`
- B. `<p>текст</p>`
- C. `текст`
- D. `<i>текст</i>`

8. После просмотра html-документа в браузере текст, заключенный между тегами `<p align="justify">Структура html-документа</p>` выводиться ...

- A. как новый абзац, жирным шрифтом, выровненный по центру
- B. как новый абзац, жирным шрифтом, выровненный по ширине
- C. как новый абзац, наклоненным шрифтом, выровненный по ширине
- D. как новый абзац, наклоненным шрифтом, выровненный по центру

9. Какой тип списка отобразится в окне браузера после интерпретации им следующего фрагмента html-кода?

```
<ol>
  <li>элемент 1</li>
  <li>элемент 2</li>
  <li>элемент 3</li>
</ol>
```

- A. маркированный
- B. нумерованный
- C. список описаний

10. С помощью какого атрибута можно изменить внешний вид маркера в списке?

- A. атрибута `type`
- B. атрибута `ol`
- C. атрибута `ul`
- D. атрибута `start`

Уровень В

1. Для каждого элемента (тега) HTML, указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- | | |
|---------------------------|---|
| А. тег <base> | 1. поисковые системы используют для того, чтобы определить релевантность страницы тому или иному запросу |
| В. мета-тег "keywords" | 2. используется при создании краткого описания страницы сайта |
| С. мета-тег "description" | 3. служит для определения полного базового адреса текущего документа для гипертекстовых ссылок документа, заданных в неполной форме |

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

А	В	С

2. Для каждого элемента (тега) HTML, указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- | | |
|------------|--------------------------------------|
| А. <BODY> | 1. формат документа, тип документа |
| В. <HTML> | 2. заголовок документа |
| С. <HEAD> | 3. текст (тело) документа |
| Д. <TITLE> | 4. название документа, имя документа |

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

А	В	С	Д

3. Для каждого элемента (тега) HTML, указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- | | |
|---------|----------------------------------|
| А. <P> | 1. простое окончание строки |
| В.
 | 2. ввод пустого абзаца, параграф |

- C. <SUB> 3. выделения текста курсивом
- D. <SUP> 4. выделения текста жирным шрифтом
- E. 5. надстрочные индексы
- F. <I> 6. подстрочные символы

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

	A	B	C	D	E	F

4. Для управления гарнитурой, цветом и размером текста html-документа используются соответствующие атрибуты тега . Для каждого атрибута тега , указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- A. color 1. задает значения размера шрифта
- B. size 2. задает гарнитуру шрифта
- C. face 3. задает цвет текста

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

	A	B	C

5. С помощью атрибутов начального тега <body> можно задавать цвет фона и текста html-документа. Для каждого атрибута тега <body>, указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- A. background 1. задает графическое изображение для
- B. bgcolor 2. черепичного заполнения фона
- C. text 2. задает цвет фона
- 3. задает цвет текста по умолчанию

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

	A	B	C

Уровень С

1. Запишите html-код, определяющий в качестве заголовка окна веб-страницы текст "Основы HTML".

Ответ: ...

2. Запишите атрибут для горизонтального выравнивания по центру содержимого абзаца в теге <p>.

Ответ: ...

3. Перепишите фрагмент html-документа: <p>Выравнивание текста абзаца</p>, так чтобы текст абзаца был выровнен по ширине.

Ответ: ...

4. Запишите html-код, форматирующий текст "Транспортная компания АЛЬФА-ТРАНС" следующим образом:

Транспортная – жирный курсив;

компания – курсив;

АЛЬФА-ТРАНС – жирный подчеркнутый.

Ответ: ...

5. Для тега определите гарнитуру шрифта – "Arial" и "Times New Roman"; цвет шрифта – красный (соответствующий ключевому слову) и размер шрифта – 3.

Ответ: ...

6. Запишите html-код, определяющий сине-зеленый (aqua) цвет фона документа.

Ответ: ...

7. Запишите html-код, задающий графическое изображение, которое как черепица заполнит фон документа. Графическое изображение хранится под именем fon.jpg в папке Img, которая размещена в той же папке, что и html-документ.

Ответ: ...

8. Запишите html-код, определяющий на всю ширину экрана окна браузера горизонтальную линию толщиной в 3 пикселя.

Ответ: ...

9. Запишите html-код, определяющий бегущую строку "Я бегущая строка" без атрибутов.

Ответ: ...

10. Запишите html-код, определяющий бегущую строку "Я бегущая строка" в направлении справа-налево.

Ответ: ...

Глава 2. Вставка объектов в html-документ

2.1 Размещение графики на web-страницах

В Интернете трудно найти сайт, который бы не содержал изображений, и это не удивительно, ведь изображения – это главная часть дизайна сайта, а хороший дизайн сайта – залог его успешного продвижения.

Для размещения на странице изображений используется тег **** (от англ. image), с обязательным атрибутом **src**, который в качестве значения принимает полный или относительный адресный путь к файлу изображения:

Тег **** позволяет вставить в документ изображение. Оно появится в том месте документа, где находится этот тег. Данный тег является одиночным.

Графика в Web, как правило, распространяется в следующих форматах: jpg, png, gif. При этом картинка не вставляется непосредственно в текст html-документа, а лишь дается указание браузеру, где картинка лежит, ее имя и как разместить ее на экране.

Ниже даны варианты вставки графических изображений, размещенные в разных местах:

Вариант 1.

где car.jpg - это название файла с графическим изображением автомобиля.

В значение атрибута src не указан путь к файлу изображения, поэтому чтобы браузер правильно отображал рисунок автомобиля, файл car.jpg должен быть размещен в том же каталоге, что и html-документ.

Вариант 2.

Такая запись подразумевает, что в директории, где расположен

html-документ, есть папка Images, в которой расположен файл car.jpg.

Вариант 3.

```

```

В этом варианте рядом с html-документом должна быть расположена папка Images, в ней еще одна папка с названием graphics, и уже в ней нужный файл car.jpg.

Вариант 4.

```

```

Файл рисунка car.jpg размещен на уровень выше от html-документа.

Вариант 5.

```

```

Соответственно на два уровня выше, сколько поставите ..\ настолько и поднимитесь.

Вариант 6.

```

```

```

```

В этом случае указывается абсолютное значение пути, где размещены графические файлы: на диске и адрес интернет - ресурса.

Задание № 2.1.а. Размещение графики на web-странице.

1. Выбор и создание рабочей папки для заданий и графических изображений второй главы.

В задании № 1.1. главы 1 для практических заданий учебного пособия на диске D: была создана папка под название Textbook. Выполненные задания из второй главы будут размещаться в отдельной папке с названием **Chapter_2** (от англ. Глава_2). Поэтому создайте в папке Textbook новую папку с именем **Chapter_2** (абсолютный адрес рабочей папки: D:\Textbook\Chapter_2). Это папка будет рабочей папкой для практических заданий главы 2.

2. Все изображения для сайта, как правило, хранятся в отдельной папке, например, Images. Поэтому создайте в своей рабочей папке новую папку с именем **Images**, в которой будете размещать графические файлы. Во всех последующих заданиях пути

к изображениям будут прописаны, исходя из того, что графические файлы находятся в папке «D:\Textbook\Chapter_2\Images», а web-страницы в папке «D:\Textbook\Chapter_2».

На прилагаемом к учебному пособию диске в папке «Учебные материалы\Глава_2\Графические файлы» размещены графические изображения автомобилей, которые будут использованы в заданиях. Из этой папки скопируете графические изображения автомобилей **car.png**, **car.jpg**, **lamborghini.jpg**, **lada-granta.jpg** и **car.gif** в папку Images рабочей папки.

3. Скопируйте шаблон html-документа temp.html, размещенный по адресу «D:\Textbook\Chapter_1» в новую рабочую папку **Chapter_2** и откройте его в редакторе Notepad++. В его заголовочной части произведите следующие изменения:

- в теге <title> измените текст заголовка на "Варианты размещения графики на web-странице";

- в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы.

Для начала будут рассмотрены первые два варианта размещения графических изображений на web-страницу. Для этого в тело документа добавьте нижеприведенный фрагмент html-кода:

```
<h2>Варианты размещения файлов с рисунками в html-документ</h2>
```

```
<p>Вариант 1.<br>Файл car.jpg размещен в том же каталоге что и html-документ</p>
```

```

```

```
<p>Вариант 2.<br>Файл car.png размещен в папке Images, которая размещена в том же каталоге, что и html-документ</p>
```

```

```

4. Сохраните шаблон html-документа под именем **task-21_a1.html** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 2.1).

В варианте 1 предполагается, что в каталоге, где размещена

страница task-21_a1.html должен быть размещен графический файл car.jpg. Но он размещен в папке Images, поэтому при просмотре данной страницы результатом будет верхняя часть окна, изображенное на рисунке 2.1, т.е. рисунок автомобиля не отображается на web-странице.

Во втором варианте графический файл car.png размещен правильно, поэтому рисунок автомобиля отображается на web-странице.

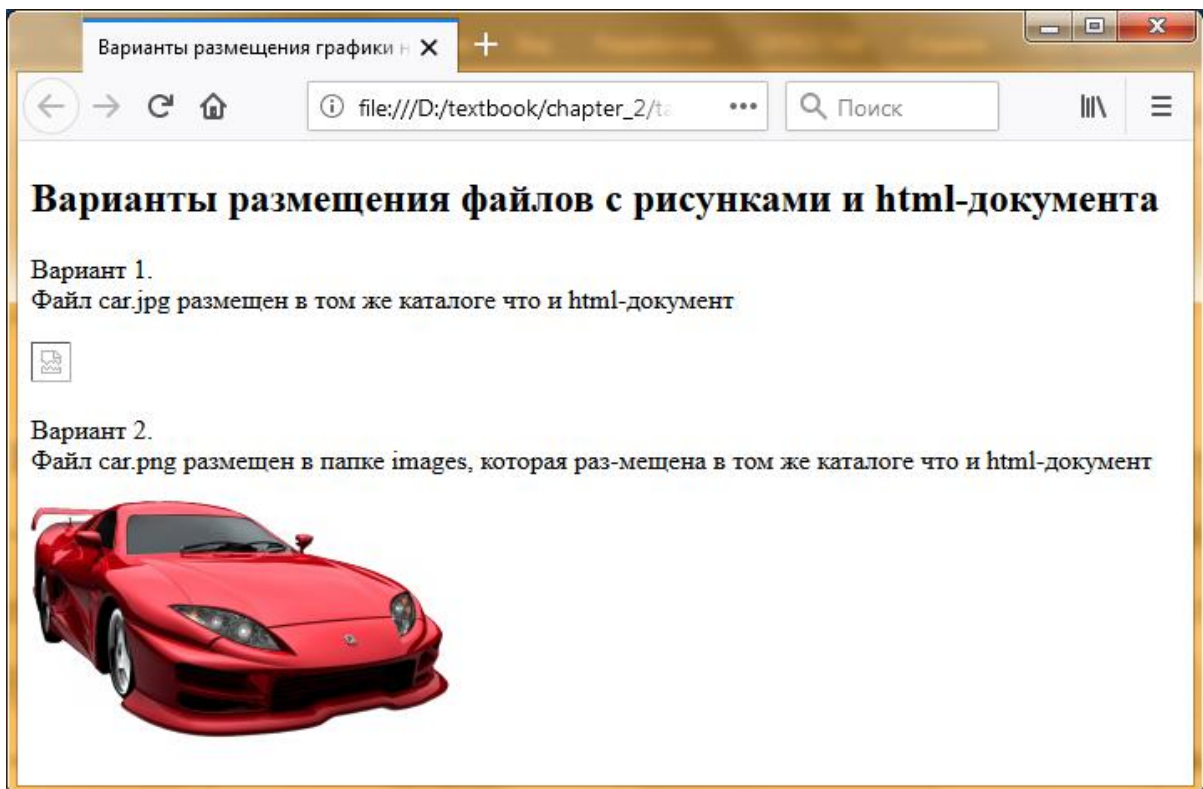


Рисунок 2.1 Результат просмотра страницы task-21_a1.html в браузере

5. Скопируйте графический файл car.png из папки Images в каталог, где размещена страница task-21_a1.html. Сохраните и обновите документ в браузере Firefox (рис. 2.2). Сделайте выводы.

6. Чтобы продемонстрировать вариант 3 создайте в папке Images новую папку с именем **Photocar** и переместите туда графический файл автомобиля **lamborghini.jpg**. В тело документа task-21_a1.html добавьте следующий html код:

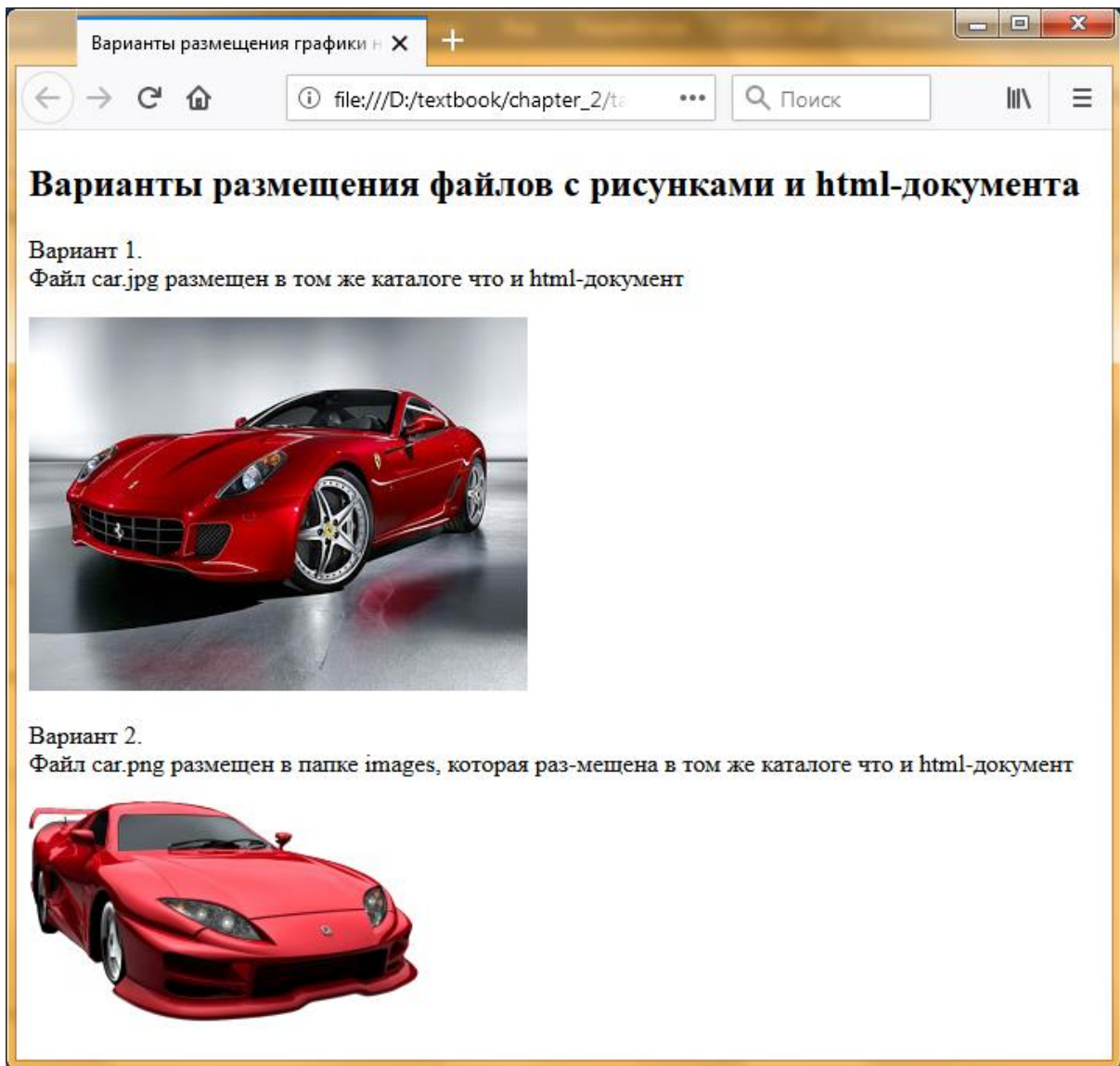


Рисунок 2.2 Результат повторного просмотра task-21_a1.html в браузере

Вариант 3. Файл lamborghini.jpg размещен в папке Images\Photocar

```

```

Сохраните документ под именем **task-21_a2.html** и просмотрите в браузере Firefox.

7. Чтобы продемонстрировать варианты 4 и 5 скопируйте графический файл **car.jpg** в папки, лежащие на уровень и на два уровня выше от html-документа task-21_a2.html, т.е. в папку Textbook и в корневой каталог диска D:.

В тело документа task-21_a2.html добавьте следующий html-код.

<p>Вариант 4.
Файл car.jpg размещен на уровень выше от html-документа </p>

```

```

<p>Вариант 5.
Файл car.jpg размещен на два уровня выше от html-документа </p>

```

```

Сохраните документ под именем **task-21_a3.html** и просмотрите в браузере Firefox. Сделайте выводы.

8. Вариант 6 продемонстрируйте для графического файла **lada-granta.jpg**. Для этого в тело документа task-21_a3.html добавьте следующий html-код:

<p>Вариант 6.
Отображение рисунка по прямой ссылке на диске</p>

```

```

Замечание. В браузере Firefox для правильного отображения рисунка по прямой ссылке на диске следует путь к файлу начинать со спецификатора "file:///". Поэтому, если ваш браузер не отображает рисунок «lada-granta.jpg», то вышеприведенный html-код замените на следующий код:

```

```

Сохраните документ под именем **task-21_a4.html** и просмотрите в браузере Firefox. Сделайте выводы.

9. Формат .gif поддерживает анимационные изображения. Они представляют собой последовательность из нескольких статичных кадров, а также информацию о том, сколько времени каждый кадр должен быть показан на экране. Для демонстрации анимации на web-странице в тело документа task-21_a4.html добавьте следующий html-код:

<p>Демонстрация анимации на web-странице </p>

```

```

Сохраните документ под именем **task-21_a5.html** и просмотрите в браузере Firefox. Сделайте выводы.

Атрибуты title и alt тега . Тег имеет немало атрибутов, которые можно задавать дополнительно. Они могут располагаться в любом месте после тега img.

Атрибут **title** – задает описание изображения. Это текст, который появляется рядом с курсором, если навести курсор на изображения и подержать его там несколько секунд.

Атрибут **alt** (от англ. alternative) – задает текст, заменяющий изображение в том случае, если браузер не воспринимает это изображение. Содержание атрибута alt активно используется поисковыми системами и произносится голосовыми браузерами, поэтому его надо использовать в теге .

Задание № 2.1.b. Атрибуты title и alt тега .

Создайте страницу, содержащую короткую информацию о трех популярных автомобилях: Toyota Corolla, Ford F-Series и Volkswagen Golf с фотографиями. Для этого выполните следующие действия:

1. Откройте шаблон html-документа temp.html и произведите следующие изменения:

– в теге <title> измените текст заголовка на "Атрибуты title и alt тега img";

– в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;

– в тело документа добавьте фрагмент html-кода, описывающий вышеприведенные автомобили:

```
<font face="Comic Sans MS, Arial" color="red" size="5">
```

```
<p align="center">Toyota Corolla</p></font>
```

```
<p> 
```

Toyota Corolla – компактный автомобиль, выпускаемый компанией Toyota. Появившись в 1966 году, в 1974 году он попал в Книгу рекордов Гиннеса как самая продаваемая модель в мире. </p>

```
<font face="Comic Sans MS, Arial" color="red" size="5">
```

```
<p align="center">Ford F-Series</p></font>
```

```
<p>
```

F-Series – серия полноразмерных пикапов, выпускаемых Ford Motor Company более шестидесяти лет. Первый пикап Ford F-серии - одна из самых успешных моделей в истории компании Ford.</p>

```
<font face="Comic Sans MS, Arial" color="red" size="5">
```

```
<p align="center">Volkswagen Golf</p></font>
```

```
<p>
```

Volkswagen Golf – автомобиль немецкой компании Volkswagen. Гольф стал самой успешной моделью Фольксвагена, занимает третье место среди самых продаваемых автомобилей (по данным на 2007 год произведено более 25 миллионов шт.), является лидером продаж в Европе. </p>

2. Сохраните документ под новым именем **task-21_b1.html** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 2.3).

Из результата просмотра видно, что на web-странице вместо рисунков появляется альтернативный текст "рисунок не найден". Это происходит потому, что указанный в теге относительный адрес не существует.

По вышеприведенному коду предполагается, что в каталоге, где размещена страница task-21_b1.html должна быть папка Images, в ней другая папка Popularcars, а в ней должны быть размещены графические файлы автомобилей: **toyota_corolla.jpg**, **ford_f-150.jpg**, **vwgolf6.jpg**. Но их пока там нет.

Несмотря на это, при наведении курсора на место предполагаемых рисунков, рядом с курсором появляются соответствующие рисункам надписи. Проверьте это.

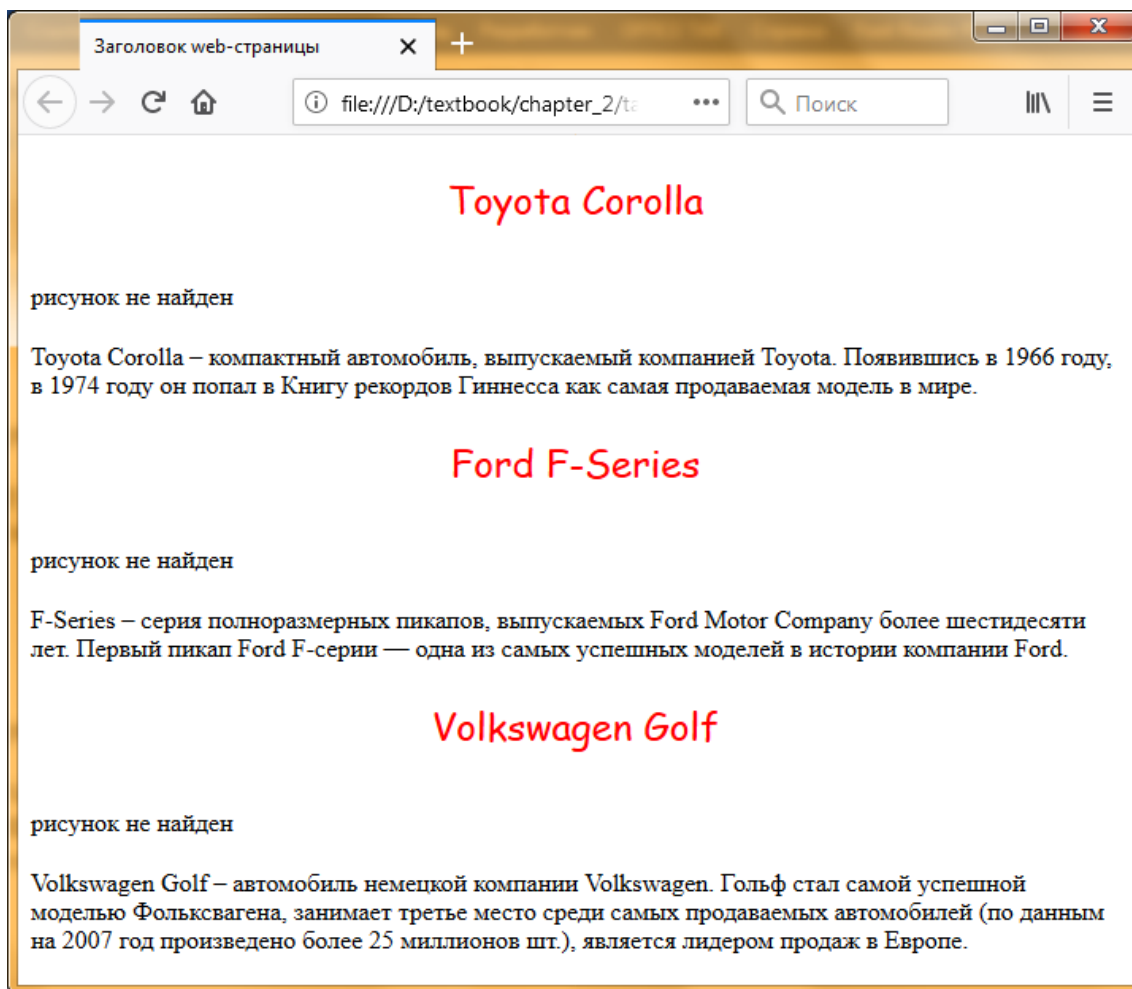


Рисунок 2.3 Результат просмотра task-21_b1.html в браузере

Создайте в папке Images новую папку **Popularcars** и скопируйте туда графические файлы автомобилей **toyota_corolla.jpg**, **ford_f-150.jpg** и **vwgolf6.jpg**, расположенные на прилагаемом к учебному пособию диске в папке «Учебные материалы\Глава_2\Графические файлы».

Обновите документ в браузере Firefox (рис. 2.4).

Атрибут выравнивания align тега **. Атрибут **align="left | right | top | bottom | middle" – задает выравнивание изображения относительно текста:

- **left** – выравнивает изображение по левому краю окна. Текст обтекает справа;

- **right** – выравнивает изображение по правому краю окна. Текст обтекает слева;

– **top** – верхняя граница изображения выравнивается по самому высокому элементу текущей строки;

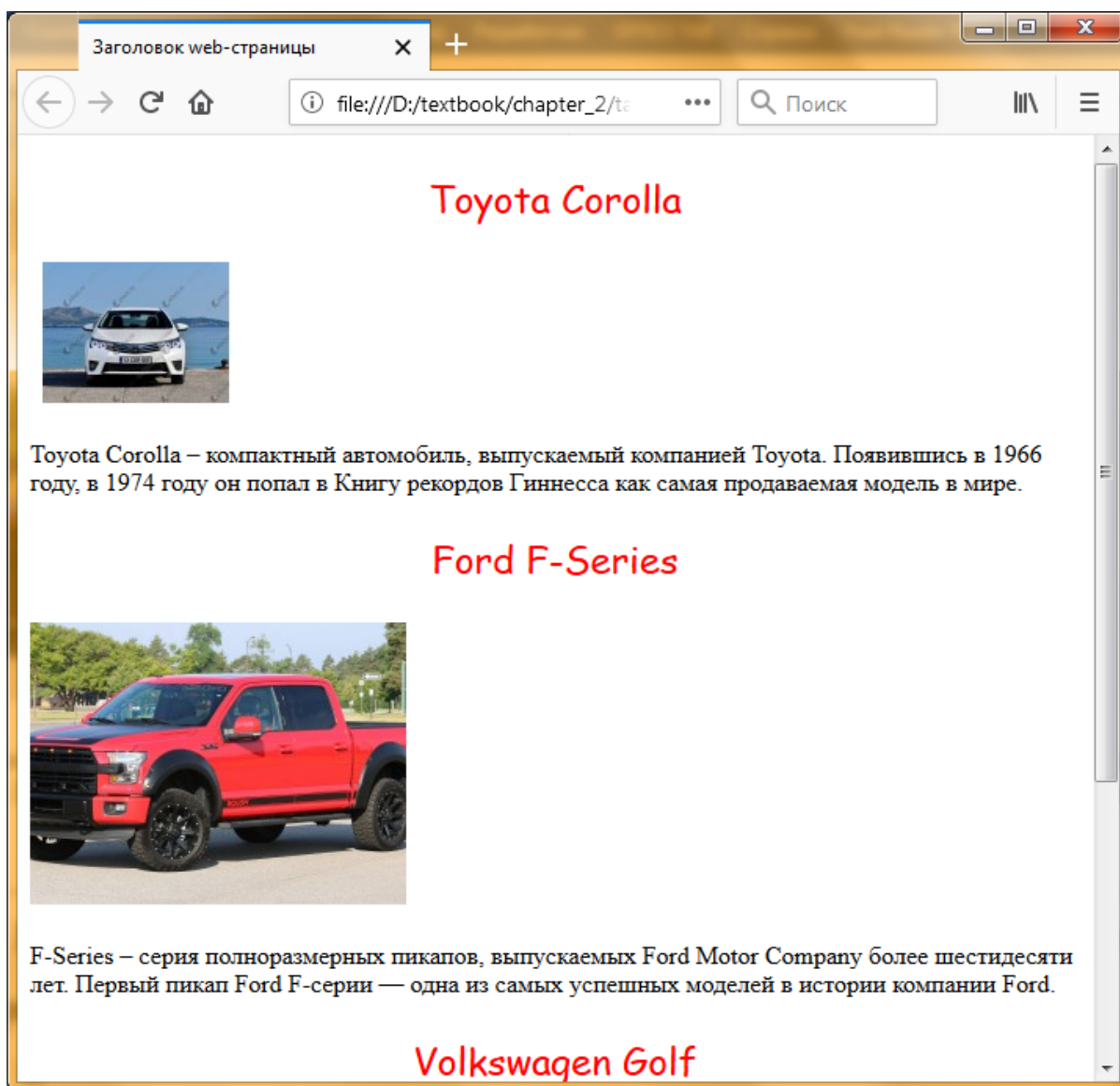


Рисунок 2.4 Результат повторного просмотра task-21_b1.html в браузере

– **bottom** (по умолчанию) – выравнивание нижней границы изображения по окружающему тексту;

– **middle** – выравнивание середины изображения по базовой линии текущей строки.

При форматировании html-документа чаще используют значения `left` и `right`, создающие обтекание текста вокруг изображения.

Задание № 2.1.с. Атрибут выравнивания рисунка align.

В html-документе task-21_b1.html произведите следующие изменения:

– в теге <title> измените текст заголовка на "Атрибут выравнивания align тега img";

– в тегах вставки рисунков добавьте атрибут выравнивания следующим образом:

```

```

```

```

```

```

Сохраните документ под новым именем **task-21_c1.html** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 2.5).

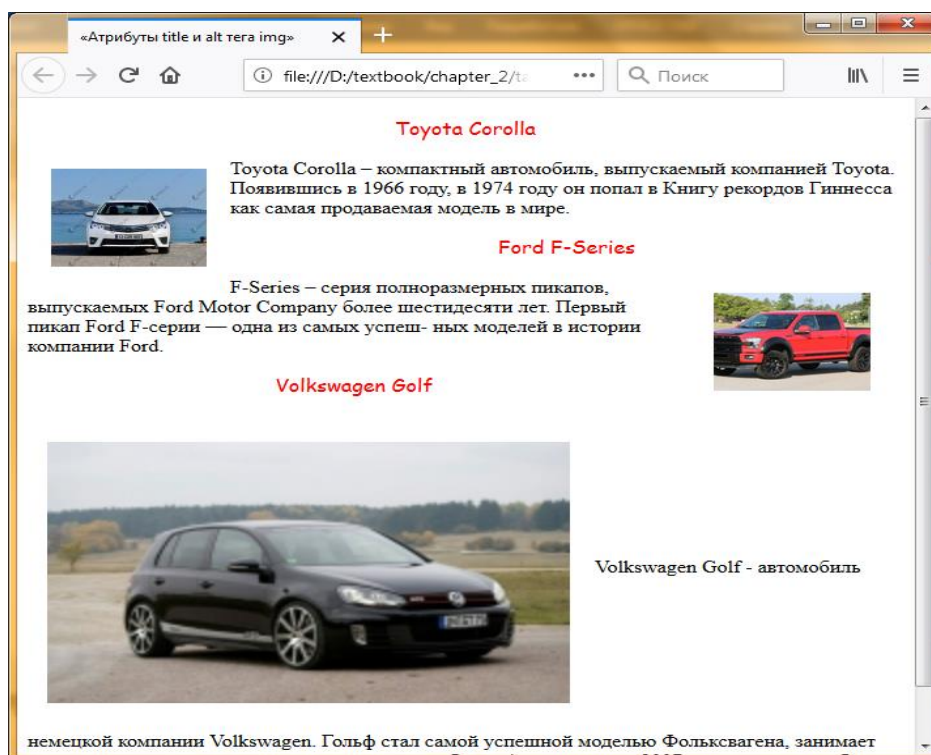


Рисунок 2.5 Результат просмотра страницы task-21_c1.html в браузере

Атрибуты отступа `vspace` и `hspace` тега ``. Атрибуты `vspace = "отступ по вертикали"` и `hspace = "отступ по горизонтали"` – задают соответственно добавление верхнего, нижнего и левого, правого пустых полей. Например, следующий html-код задает расстояние по вертикали от рисунка до текста в 15 пикселей, а по горизонтали от рисунка до текста в 25 пикселей:

```

```

Задание № 2.1.d. Атрибуты отступа `vspace` и `hspace` тега ``.

В html-документе `task-21_c1.html` произведите следующие изменения:

- в теге `<title>` измените текст заголовка на "Атрибуты `vspace` и `hspace` тега `img`";

- в тегах вставки рисунков `` добавьте атрибуты отступа по вертикали и по горизонтали разные для каждого рисунка:

`vspace="10" hspace="10"` – для первого рисунка

`vspace="15" hspace="25"` – для второго рисунка

`vspace="25" hspace="15"` – для третьего рисунка

Сохраните документ под новым именем **`task-21_d1.html`** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 2.6).

Атрибуты размера `width` и `height` тега ``. Атрибуты `width= "значение"` и `height= "значение"` – используют для изменения соответственно ширины и высоты изображения. Значения можно задавать в пикселях или процентах.

Если значение задана в процентах, то размеры изображения вычисляются относительно родительского контейнера, где находится тег ``. В случае отсутствия родительского контейнера, в его качестве выступает окно браузера, т.е. если задать, например, `width="100%"`, то рисунок будет растянут на всю ширину веб-страницы.

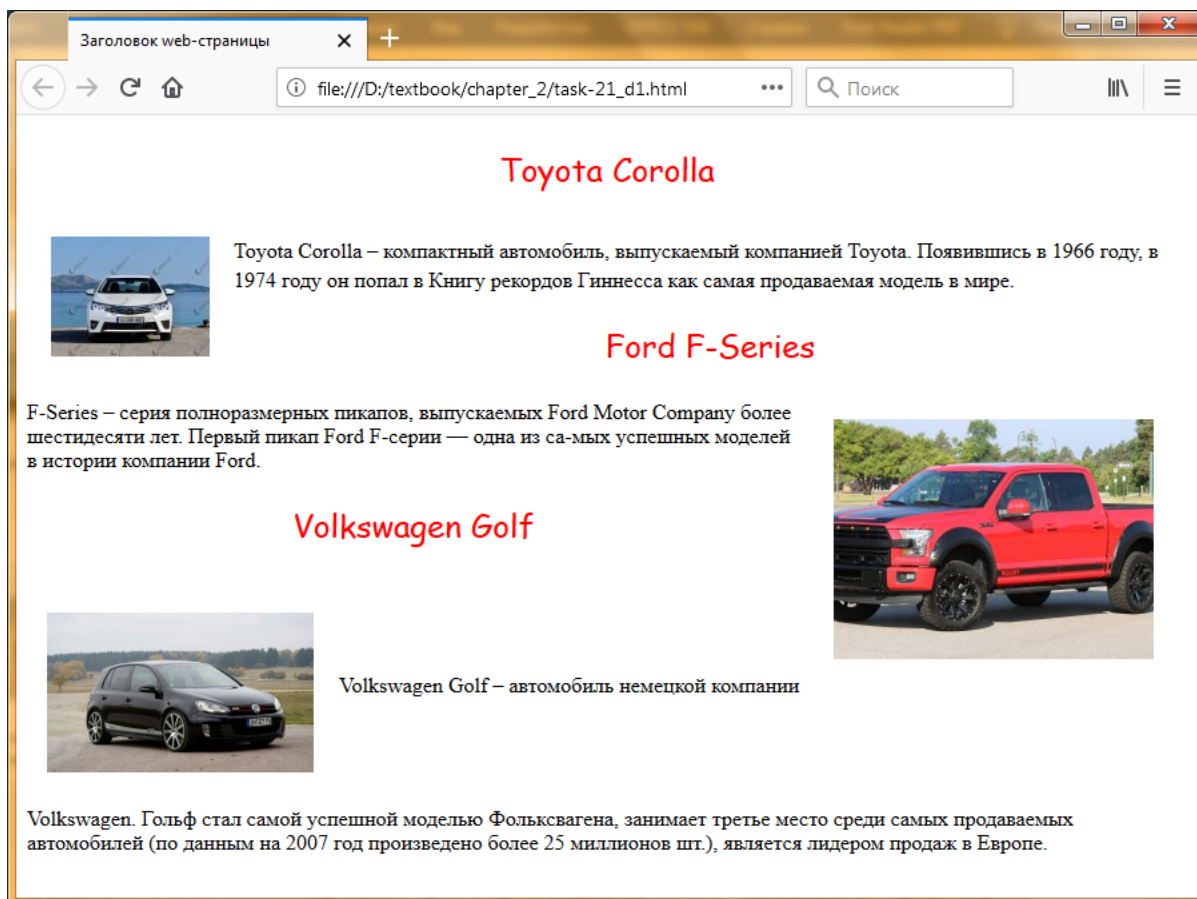


Рисунок 2.6 Результат просмотра страницы task-21_d1.html в браузере

Размеры всех изображений на веб-странице нужно обязательно задавать. Это несколько ускоряет загрузку страницы, потому что браузер загружает изображения после того, как загрузит html-документ и начнет отображать веб-страницу. Если в теге `` не указаны размеры изображения, то браузер не будет их знать до тех пор, пока не загрузит рисунок. В этом случае вместо изображения сначала отображается квадратик, а после загрузки графического файла происходит обновление страницы с соответствующим перемещением текста, других объектов страницы и потерей времени. Если же размеры заданы, то браузер оставляет место для изображения и страница загружается быстрее.

Если атрибуты `width` и `height` не использовать, то ширина и высота изображения по умолчанию будет равна реальным ее размерам, без каких-либо искажений.

При использовании этих атрибутов можно увеличить или уменьшить изображение. Можно указывать значения атрибутов `width` и `height`, которые меньше или больше, чем фактические размеры изображения, и браузер установит пропорции рисунка так, чтобы они соответствовали новым размерам. Если использовать один из атрибутов размера (`width` или `height`), то размеры изображения изменятся пропорционально, без искажений. А если использовать сразу два атрибута, то при неправильном указании размеров изображение исказится.

Задание № 2.1.е. Атрибуты размера `width` и `height` тега ``.

1. В html-документе `task-21_d1.html` произведите следующие изменения:

– в теге `<title>` измените текст заголовка на "Атрибуты `width` и `height` тега `img`";

– в тегах вставки рисунков `` добавьте атрибуты изменения размеров изображения для каждого рисунка (размеры изображений, используемых в заданиях следующие: `toyota_corolla.jpg` – 135x90, `ford_f-150.jpg` – 240x180, `vwgolf6.jpg` – 200x120):

`width="135" height="90"` – для первого рисунка как в оригинале;

`width="120" height="90"` – для второго рисунка, уменьшить в 2 раза;

`width="400" height="240"` – для третьего рисунка, увеличить в 2 раза.

3. Сохраните документ под новым именем **`task-21_e1.html`** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 2.7).

Сравните результаты просмотра страниц `task-21_c1.html` и `task-21_e1.html`. Обратите внимание на то, что при увеличении растровых изображений качество их отображений на web-странице ухудшается.

4. Используя вышеприведенный фрагмент html-кода, а также текстовый документ «Популярные автомобили в мире.txt» (находится на прилагаемом к учебному пособию диске в папке «Учебные материалы\Глава_2»), оригинальные графические файлы автомобилей `Ford_F-150.jpg`, `VWGolf6.jpg` (смотрите на диске в папке

«Учебные материалы\Глава_2\Графические файлы») допишите в документ информацию об автомобилях Ford F-Series и Volkswagen Golf. Измените шрифтовое оформление документа, например для первого рисунка: Шрифт – Times New Roman, размер – 4, цвет – сине-зеленый.

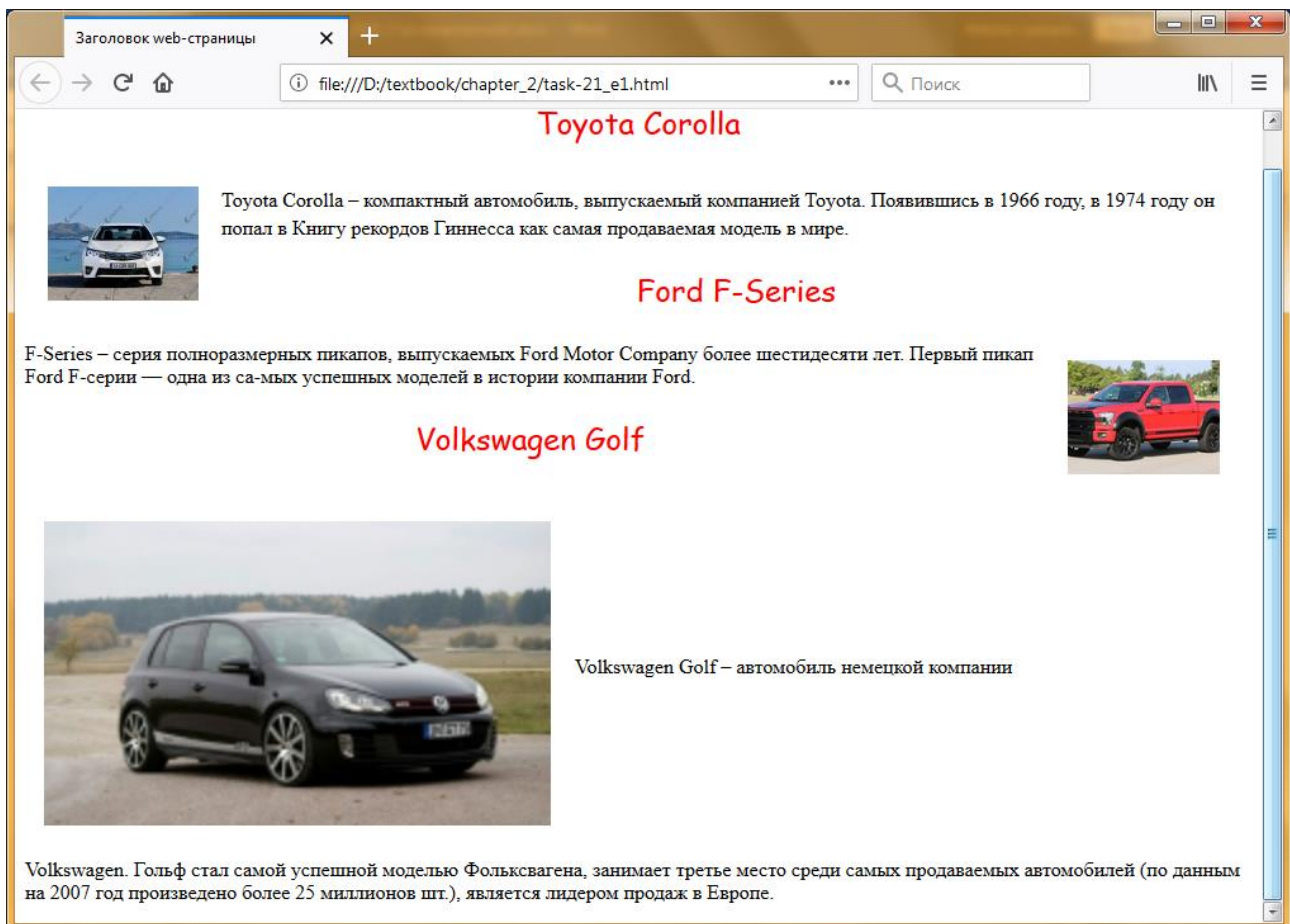


Рисунок 2.7 Результат просмотра страницы task-21_e1.html в браузере

Сохраните документ под новым именем **task-21_e2.html** в рабочей папке. Просмотрите документ в браузере Firefox. Сделайте выводы.

2.2 Вставка таблиц в HTML-документ

На web-страницах таблицы используются как для представления данных, так и как средство форматирования страницы.

Любая таблица состоит из одной или нескольких строк, каждая строчка состоит из одной или нескольких ячеек с данными. Ячейки с данными могут содержать текст, изображения, списки и т.д.

Для добавления таблицы на веб-страницу используется контейнер `<table> ... </table>`. Все элементы таблицы должны находиться внутри этого контейнера. Строки создаются с помощью контейнера `<tr> ... </tr>` (сокр. от англ. «table row» – строка таблицы). Количество встречающихся пар этих тегов определяют количество строк таблицы.

Ячейка таблицы описывается только внутри строки таблицы с помощью контейнеров `<td> ... </td>` (сокр. от англ. «table data» – данные таблицы) или `<th> ... </th>` (сокр. от англ. «table heading» – заголовок таблицы). Контейнер `<tr>` содержит содержимое ячейки, а контейнер `<th>` используется для задания заголовочных ячеек html-таблицы. Текст в заголовочных ячейках выделен жирным шрифтом, и выровнен по центру.

Таблица может иметь заголовок, который заключается в пару тегов `<caption>` и `</caption>` (от англ. caption – подпись). Описание заголовка должно располагаться внутри тегов `<table>` и `</table>` в любом месте, но вне тегов `<tr>`, `<td>` и `<th>`. По умолчанию текст заголовка таблицы располагается над таблицей и размещается в центре вне рамки.

Задание № 2.2.а. Размещение таблицы на web-странице.

Для рассмотрения примера вставки в html-документ таблицы, состоящей из 2 строк и 3 столбцов, выполните следующие действия:

1. Откройте шаблон html-документа temp.html и произведите следующие изменения:

- в теге `<title>` измените текст заголовка на "Размещение таблицы на web-странице";

- в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;

– в тело документа добавьте нижеприведенный фрагмент html-кода, добавляющий в документ таблицу, состоящую из двух строк и трех столбцов, причем ячейки первой строки будут заголовками соответствующих столбцов:

```
<table>
  <caption> Пример вставки в html-документ таблицы
</caption>
  <tr>
    <th> Первая ячейка первой строки </th>
    <th> Вторая ячейка первой строки </th>
    <th> Третья ячейка первой строки </th>
  </tr>
  <tr>
    <td> Первая ячейка второй строки </td>
    <td> Вторая ячейка второй строки </td>
    <td> Третья ячейка второй строки </td>
  </tr>
</table>
```

2. Сохраните шаблон html-документа под именем **task-22_a1.html** в своей рабочей папке. Просмотрите документ в браузере Firefox (рис. 2.8).

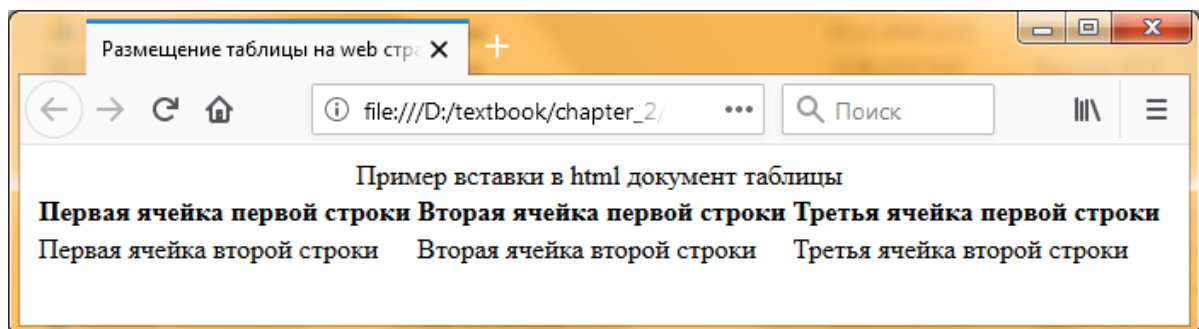


Рисунок 2.8 Результат просмотра страницы task-22_a1.html в браузере

Обратите внимание на то, что:

– по умолчанию ячейки таблицы не обрамлены рамкой и

текст заголовка расположен сверху и выровнен по центру. Некоторые браузеры по умолчанию выводят таблицу с рамкой;

- название таблицы по умолчанию размещается сверху таблицы и выровнен по центру относительно таблицы;

- ячейки первой строки таблицы являются заголовками столбцов. Они выделены по умолчанию жирным шрифтом и выровнены по центру относительно ячеек. Содержимое остальных ячеек по умолчанию выровнены по горизонтали слева, а по вертикали – сверху;

- ширина ячеек определяется содержимым и шириной окна браузера. Если ширину окна браузера сделать меньше ширины таблицы, то ширина столбцов пропорционально уменьшится. Если при этом текст не будет умещаться по ширине в ячейку, то текст разобьется на две, три и т.д. строки. Проверьте это, меняя высоту и ширину окна браузера.

Ячейки таблицы кроме текста могут содержать, например, графические изображения. Например, замените текст в первой ячейке первой строки на "Автомобиль Ford F-150", а в первую ячейку второй строки вставьте изображение из файла Ford_F-150.jpg. Для этого измените заголовок первой ячейки первой строки на следующий html-код:

```
<th>Автомобиль Ford F-150</th>
```

Далее замените первую ячейку второй строки на следующий html-код:

```
<td>img src="Images\Popularcars\ford_f-150.jpg"</td>
```

Сохраните документ под новым именем **task-22_a2.html** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 2.9).

Основные атрибуты тега <table>.

Атрибуты толщины рамок border и цвета фона bgcolor, background тега <table>.

Атрибут **border="число"** – задает толщину рамки таблицы. Если данный атрибут присутствует в теге <table>, граница таблицы

прорисовывается для всех ячеек и для таблицы в целом. Атрибут `border` может принимать числовое значение, определяющее ширину границы, например `<table border="3">`. Возможность независимого управления отображением рамки вокруг всей таблицы и рамками вокруг ячеек данный атрибут не предусматривает.

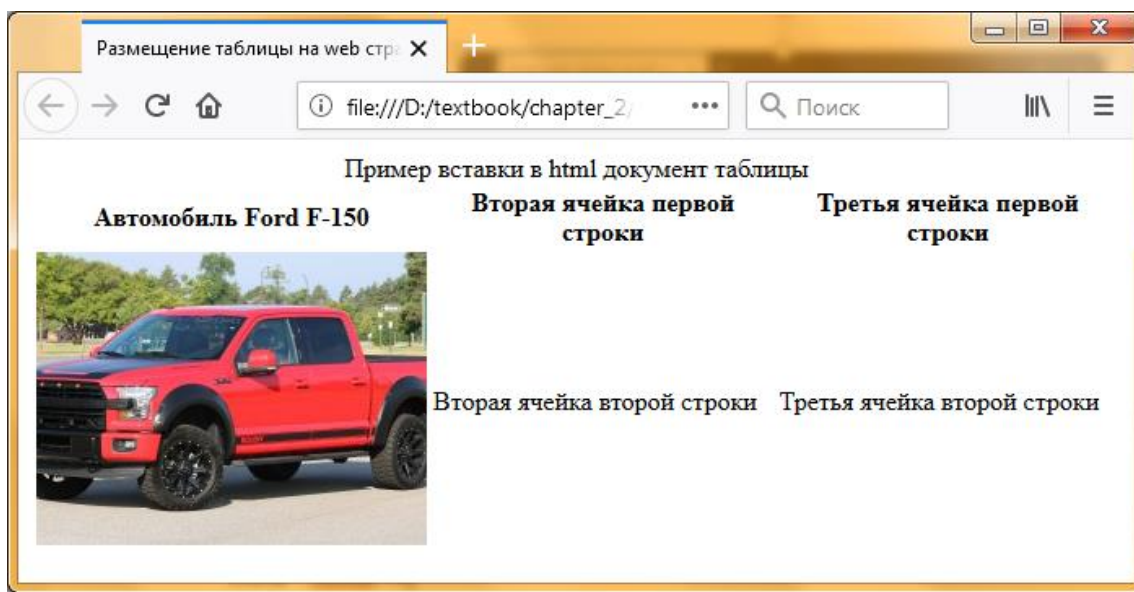


Рисунок 2.9 Результат просмотра страницы `task-22_a2.html` в браузере

Атрибут `bgcolor="цвет"` – в зависимости от того, с каким тегом (`table`, `tr`, `td`) применяется, позволяет задать цвет фона для всей таблицы, для строки или для отдельной ячейки. В качестве цвета можно выбрать любой из всей палитры (см. коды и названия `html-цветов`).

Атрибут `background="URL"` – определяет фоновый рисунок для всей таблицы или ячеек. URL – адрес фонового изображения.

Задание № 2.2.в. Атрибуты толщины рамок `border` и цвета фона `bgcolor`, `background` тега `<table>`.

1. Прорисуйте границу для всей таблицы. Для этого в `html-документе task-22_a2.html` произведите следующие изменения:

– в теге `<title>` измените текст заголовка на "Атрибуты толщины рамок и цвета фона таблицы";

– в тег вставки таблицы `<table>` добавьте атрибут прорисовки границы толщиной в 2 пикселя:

```
<table border="2">
```

Сохраните документ под новым именем **task-22_b1.html**.
Просмотрите документ в браузере Firefox (рис. 2.10).

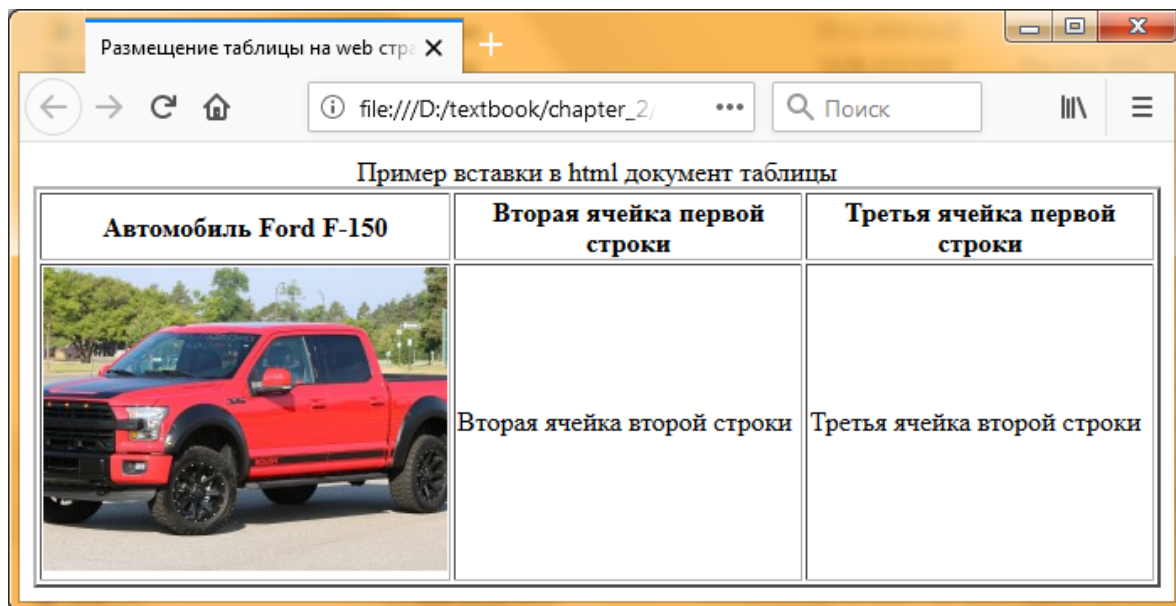


Рисунок 2.10 Результат просмотра страницы task-22_b1.html в браузере

2. Выделите заголовок светло-серым (silver) цветом. Для этого добавьте в тег `<tr>` первой строки атрибут цвета фона `bgcolor`:

```
<tr bgcolor="silver">
```

Во второй строке для первой ячейки сделайте цвет фона желтым (yellow), второй ячейки – оливковым (olive), а для третьей ячейки для заполнения фона используйте фоновый рисунок из файла `Images\Fon\fon.jpg`.

Для этого в соответствующих местах замените html-код:

```
<tr>
```

```
<td bgcolor="yellow">
```

```
img src="Images\Popularcars\ford_f-150.jpg"
```

```
</td>
```

```
<td bgcolor="olive"> Вторая ячейка второй строки </td>
```

```
<td background="Images\Fon\fon.jpg">
```


Третья ячейка второй строки

</td>

</tr>

Сохраните документ под новым именем **task-22_b2.html**.
Просмотрите документ в браузере Firefox (рис. 2.11).

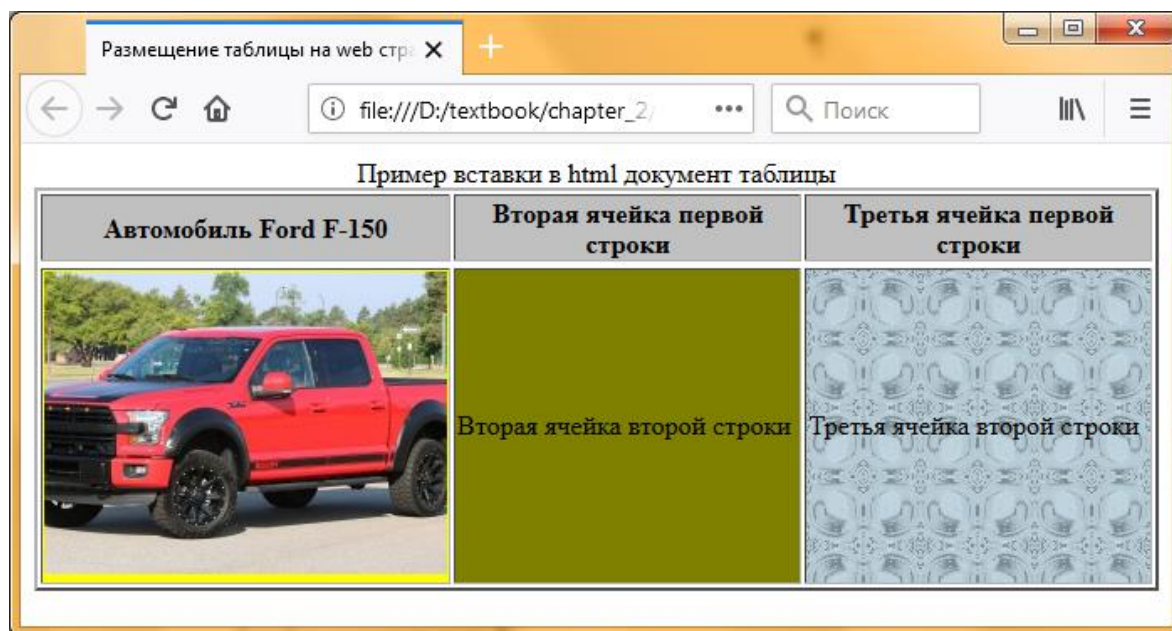


Рисунок 2.11 Результат просмотра страницы task-22_b2.html в браузере

Атрибуты cellpadding и cellspacing. Атрибут **cellspacing** тега <table> задает расстояние между внешними границами ячеек. Если установлен атрибут border, толщина границы принимается в расчет. Синтаксис этого атрибута задается следующим образом:

<table cellspacing="n">...</table>

где n - любое целое положительное число, определяет расстояние между смежными ячейками (точнее между рамками ячеек) в пикселях, как по горизонтали, так и по вертикали. По умолчанию равно 0 (когда нет атрибута border) или 2 (когда атрибут border установлен).

Атрибут **cellpadding** тега <table> определяет расстояние между границей ячейки и ее содержимым. Он добавляет пустое

пространство к ячейке, увеличивая тем самым ее размеры. Без атрибута `cellpadding` текст в таблице «налипает» на рамку, снижая тем самым его восприятие. Добавление тега `cellpadding` позволяет улучшить читабельность текста. При отсутствии границ особого значения этот атрибут не имеет, но может помочь, когда требуется установить пустой промежуток между ячейками. Синтаксис этого атрибута задается следующим образом:

```
<table cellpadding="n">...</table>
```

где `n` - любое целое значение в пикселях или процентах от доступного пространства. По умолчанию значение принимается равным 1. Установка параметра `cellpadding` равным 0 может привести к тому, что некоторые части текста ячейки могут касаться ее рамки, что выглядит не очень эстетично.

Задание № 2.2.с. Атрибуты толщины рамок `cellspacing` и `cellpadding`.

1. В html-документе `task-22_b2.html` произведите следующие изменения:

- в теге `<title>` измените текст заголовка на "Атрибуты `cellspacing` и `cellpadding`";
- задайте расстояние между внешними границами ячеек. Для этого добавьте атрибут `cellspacing` к тегу `<table>`:

```
<table border="2" cellspacing="3">
```

Сохраните документ под новым именем **`task-22_c1.html`** и просмотрите в браузере Firefox. Меняя значения `border` и `cellspacing`, поэкспериментируйте с действием этих атрибутов на таблице. Сделайте выводы.

2. Рассмотрите действие атрибута `cellpadding` на таблицу. Для этого для всей таблицы задайте расстояние между границей ячейки и ее содержимым равным 5:

```
<table border="1" cellspacing="1" cellpadding="5">
```

Сохраните документ под новым именем **`task-22_c2.html`** и просмотрите в браузере Firefox.

Атрибуты ширины width и высоты height. При отображении таблиц их ширина и высота автоматически вычисляются браузером, при этом делается попытка представить таблицу в наиболее удобном виде – расположить таблицу так, чтобы она помещалась в окне просмотра.

При этом браузером делается попытка по возможности форматировать таблицу, чтобы ширина таблицы не превосходила ширины окна просмотра, т.е. ширина таблиц является более важным, первостепенным параметром, расчет которого выполняется в первую очередь по сравнению с высотой.

В большинстве случаев динамическое определение размеров таблицы дает в результате эстетически соразмерное изображение с эффективным использованием реальной площади окна просмотра. Однако бывает необходимо принудительно указывать ширину или высоту таблицы. Для этой цели используются параметры **width** (ширина таблицы) и **height** (высота таблицы) тега **<table>**. Синтаксис этих параметров задается следующим образом:

width="n" height="m"

где n и m – численное значение соответственно ширины и высоты всей таблицы или для отдельных ячеек в пикселях или в процентах от всего размера окна.

Задание № 2.2.d. Атрибуты width и height.

Сохраните html-документ task-22_c2.html под новым именем **task-22_d1.html** и произведите следующие изменения:

– в теге **<title>** измените текст заголовка на "Атрибуты width и height";

– задайте ширину таблицы на все окно. Для этого добавьте атрибут **width="100%"** к тегу **<table>**:

```
<table border="1" cellspacing="1" width="100%" height="100%">
```

Просмотрите html-документ в браузере.

Аналогичные параметры могут задаваться и для отдельных ячеек. Заметим, что задание конкретного значения параметра, например **WIDTH=200**, не означает, что таблица в любом случае

будет иметь указанную ширину, а лишь определяет рекомендуемую ширину, которая будет выдержана по возможности. Например, пусть имеется таблица, которая в данных условиях по умолчанию имела бы ширину, меньшую заданной. В этом случае браузер увеличит ширину таблицы до требуемой путем пропорционального расширения всех колонок таблицы. При сужении окна просмотра ширина таблицы изменяться не будет, и, возможно, для ее просмотра потребуются горизонтальная прокрутка. Если же таблица по умолчанию имеет ширину, большую заданной, то браузер сделает попытку уменьшить ее ширину за счет, во-первых, сокращения ширины отдельных колонок, для которых заданная ширина больше необходимой, во-вторых, разбиением текста в отдельных ячейках на несколько строк с увеличением высоты таблицы. Эти действия могут не обеспечить требуемого размера таблицы, и тогда она будет иметь минимально возможную ширину. Такие же действия предпринимаются для таблиц, у которых не указаны размеры, при сужении окна просмотра.

Конкретные алгоритмы настройки таблиц для различных браузеров могут несколько отличаться.

Атрибуты выравнивания align и valign. Атрибут **align** применяется к таблице и к ее элементам и определяет общее горизонтальное выравнивание:

- `<table align="left/right/center">` – размещение таблицы на странице по левому краю/правому краю/по центру;
- `<tr align="left/right/center">` – размещение элементов строки по левому краю/правому краю/по центру;
- `<th align="left/right/center">` – размещение заголовка таблицы по левому краю/правому краю/по центру (по умолчанию по центру);
- `<td align="left/right/center">` – размещение данных в ячейке по левому краю/правому краю/по центру (по умолчанию по левому краю).

Атрибут **valign** также применяется к таблице и ко всем элементам таблицы и определяет общее вертикальное выравнивание:

- `<table valign="bottom/middle/top">` – размещение элементов таблицы внизу/по центру/вверху (по умолчанию по центру);
- `<tr valign="bottom/middle/top">` – размещение элементов строки внизу/по центру/вверху (по умолчанию по центру);
- `<th valign="bottom/middle/top">` – размещение заголовка таблицы внизу/по центру/вверху (по умолчанию по центру);
- `<td valign="bottom/middle/top">` – размещение данных в ячейке внизу/по центру/вверху (по умолчанию по центру).

В задании №2.2.е рассмотрены примеры использования в различных комбинациях описанных выше атрибутов форматирования таблицы и содержимого ячеек.

Задание № 2.2.е. Атрибуты горизонтального align и вертикального valign выравнивания таблицы и ее элементов.

1. Откройте шаблон html-документа temp.html. В его заголовочной части произведите следующие изменения:

- в теге `<title>` измените текст заголовка на "Атрибуты выравнивания align и valign таблицы и ее элементов";
- в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;
- в тело документа добавьте нижеприведенный фрагмент html-кода, добавляющий в документ таблицу, состоящую из двух строк и одного столбца, причем ячейка первой строки будет заголовком, а вторая ячейка будет содержать рисунок. Таблица будет выровнена с правой стороны страницы.

`<p>`Пример размещения таблицы с картиной на странице по правому краю в html-документе`</p>`

```
<table border="2" align="right">
  <caption> Align=right для table </caption>
  <tr bgcolor="silver">
```

```

        <th>Автомобиль Ford F-150</th>
    </tr>
    <tr>
        <td bgcolor="yellow">
            
        </td>
    </tr>
</table>

```

2. Сохраните документ под новым именем **task-22_e1.html**. Просмотрите документ в браузере Firefox (рис. 2.12). Сделайте самостоятельно так, чтобы таблица с изображением была выравнена по центру. Сохраните документ под новым именем **task-22_d2.html**. Просмотрите документ в браузере Firefox.

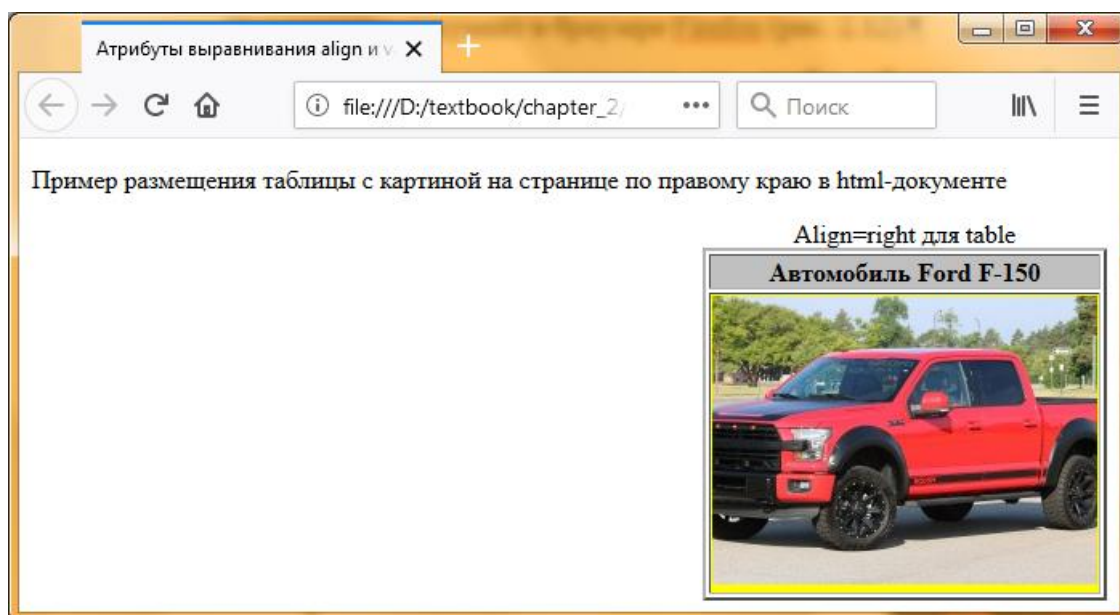


Рисунок 2.12 Результат просмотра страницы task-22_e1.html в браузере

Атрибуты nowrap и colspec. Атрибут **nowrap** указывает на то, что данные в ячейке не могут логически разбиваться на несколько строк и должны быть представлены одной строкой.

Атрибут **colspec** – позволяет задавать фиксированную ширину

колонок либо в символах, либо в процентах, например `colspec="20%"`.

Атрибуты `colspan` и `rowspan`:

– **`colspan`** – указывает, какое количество ячеек будет объединено по горизонтали для указанной ячейки. По умолчанию – 1.

– **`rowspan`** – указывает, какое количество ячеек будет объединено по вертикали для указанной ячейки. По умолчанию – 1.

Задание № 2.2.f. Атрибуты `colspan` и `rowspan`.

Для демонстрации использования атрибутов `colspan` и `rowspan` вставьте в html-документ таблицу без объединений, с которой будете работать дальше. Для этого откройте шаблон html-документа `temp.html` и в его заголовочной части произведите следующие изменения:

– в теге `<title>` измените текст заголовка на "Атрибуты объединения `colspan` и `rowspan`";

– в соответствии с содержанием web-страницы в мета-тегах `"keywords"` и `"description"` соответственно добавьте ключевые слова и описание страницы;

– в тело документа добавьте нижеприведенный фрагмент html-кода, добавляющий в документ таблицу, состоящую из четырех строк и четырех столбцов:

```
<table border="1" cellspacing="0" cellpadding="5">
```

```
<tr>
```

```
<td>Ячейка 11</td>
```

```
<td>Ячейка 12</td>
```

```
<td>Ячейка 13</td>
```

```
<td>Ячейка 14</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Ячейка 21</td>
```

```
<td>Ячейка 22</td>
```

```

        <td>Ячейка 23</td>
        <td>Ячейка 24</td>
    </tr>
    <tr>
        <td>Ячейка 31</td>
        <td>Ячейка 32</td>
        <td>Ячейка 33</td>
        <td>Ячейка 34</td>
    </tr>
    <tr>
        <td>Ячейка 41</td>
        <td>Ячейка 42</td>
        <td>Ячейка 43</td>
        <td>Ячейка 44</td>
    </tr>
</table>

```

Сохраните документ под новым именем **task-22_f1.html**.
Просмотрите документ в браузере Firefox (рис. 2.13).

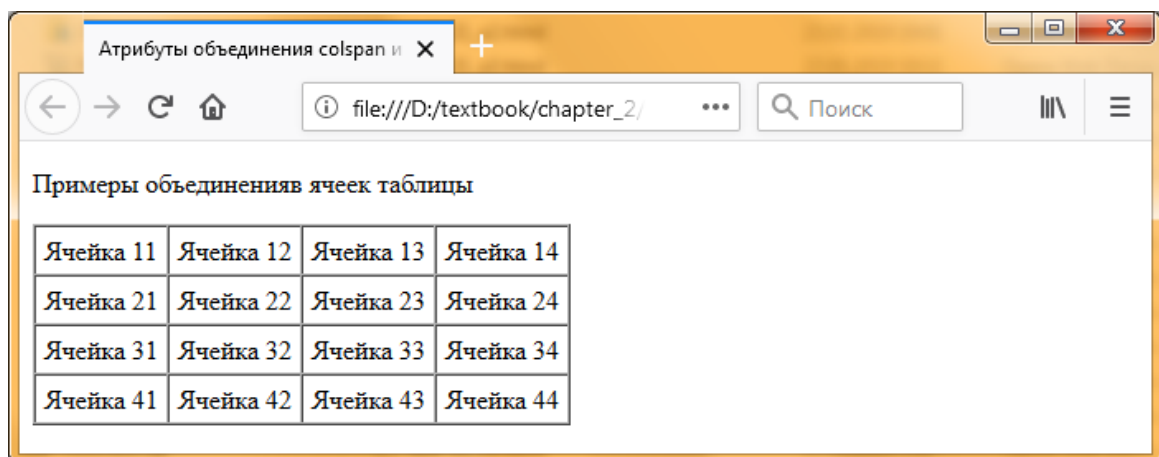


Рисунок 2.13 Результат просмотра страницы task-22_f1.html в браузере

Расширьте ячейку 11 на два ряда (объедините ячейки 11 и 21), задав ей атрибут `rowspan="2"`. Для этого добавьте атрибут `rowspan="2"` для ячейки 11:

<td rowspan="2">Ячейка 11</td>

Сохраните документ под новым именем **task-22_f2.html**.
Просмотрите документ в браузере Firefox (рис. 2.14 (а)).

В результате просмотра видно, что при добавлении атрибута `rowspan="2"` к первой ячейке, эта ячейка займет два ряда по высоте таблицы. При этом ячейка 21 из ряда ниже никуда не девается. Она и весь ряд сдвигается вправо, таблица расширится на один столбец, добавив пустые ячейки. Чтобы этого не произошло, нужно удалить одну из ячеек из ряда ниже.

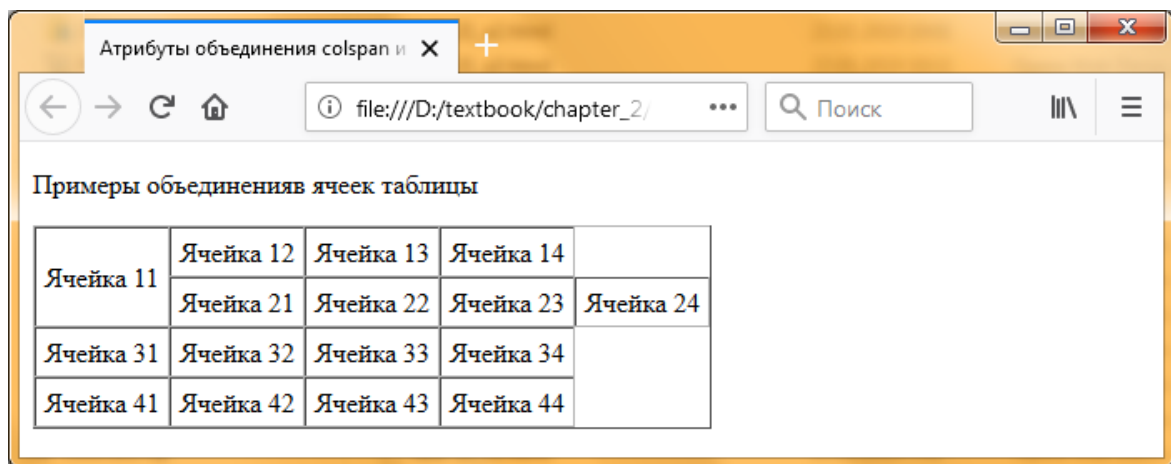


Рисунок 2.14 (а) Результат просмотра страницы `task-22_f2.html` в браузере

Удалите одну из ячеек второго ряда, не обязательно Ячейку 21, которая находится непосредственно под Ячейкой 11. Можно удалять любую из ряда:

```
<tr>  
  <td>Ячейка 22</td>  
  <td>Ячейка 23</td>  
</tr>
```

Сохраните документ под новым именем **task-22_f3.html**.
Просмотрите документ в браузере Firefox (рис. 2.14 (б)).

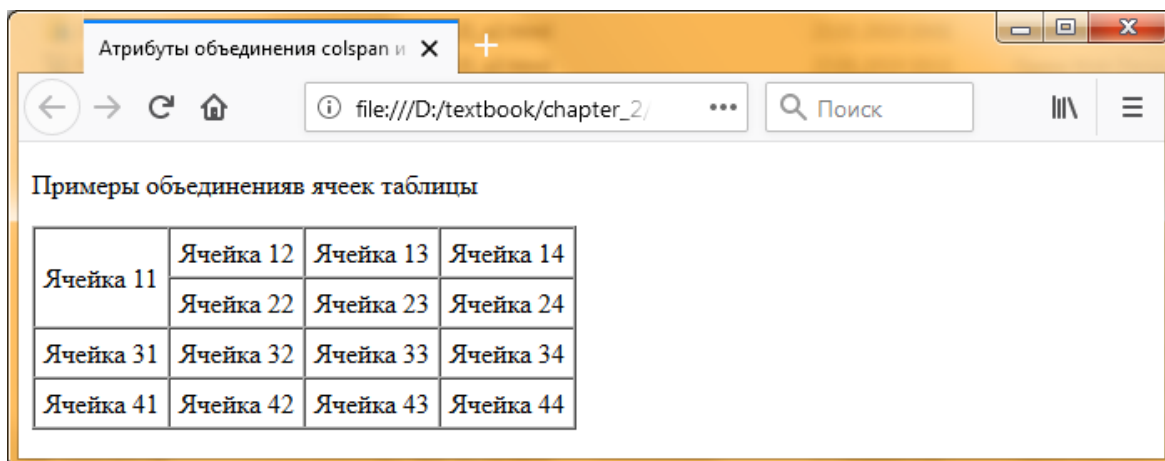


Рисунок 2.14 (б) Результат просмотра страницы task-22_f3.html в браузере

Рассмотрите объединение ячеек по строке на примере той же таблицы (рис. 2.13).

В редакторе Notepad++ откройте html-документ task-22_f1.html и для объединения Ячеек 11 и 12 для первой строки произведите следующее изменение:

```
<td colspan="2">Ячейка 11</td>
```

Сохраните документ под новым именем **task-22_f4.html** и просмотрите документ в браузере Firefox (рис. 2.15).

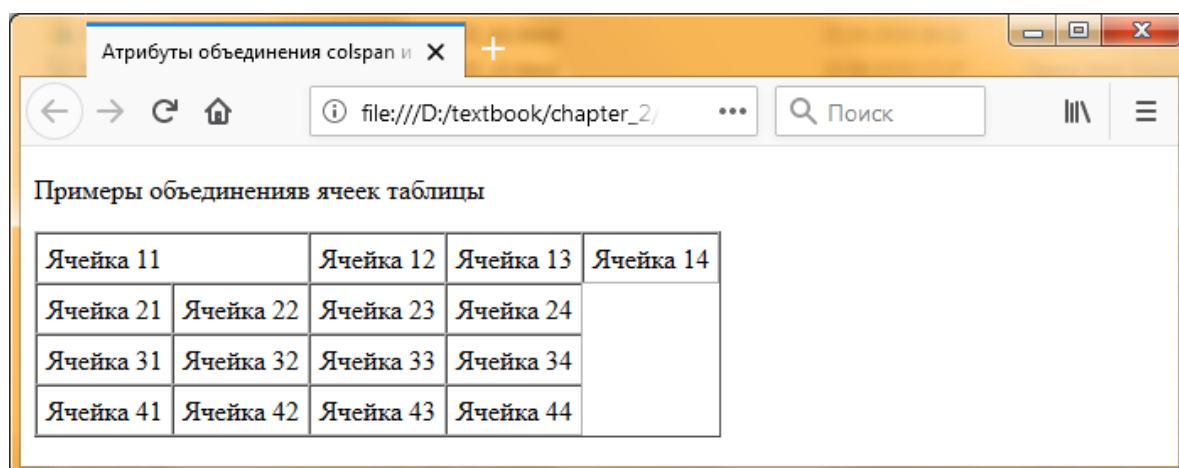


Рисунок 2.15 Результат просмотра страницы task-22_f4.html в браузере

В результате просмотра видно, что при объединении Ячеек 11 и 12 результат такой же, как и при объединении ячеек по столбцам: первая ячейка займет два столбца по ширине таблицы. При этом ячейки, лежащие правее, выдвигаются вправо, расширив таблицу. Чтобы этого не произошло, нужно удалить одну из ячеек из ряда, например Ячейку 12.

```
<tr>
  <td colspan="2">Ячейка 11</td>
  <td>Ячейка 13</td>
  <td>Ячейка 14</td>
</tr>
```

Отредактируйте первую строку как показано выше, сохраните html-документ под новым именем **task-22_f5.html** и просмотрите в браузере Firefox (рис. 2.16).

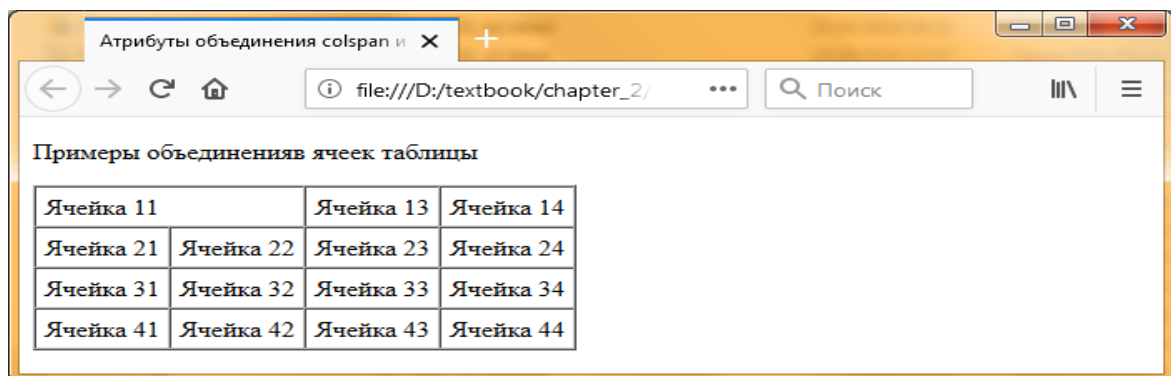


Рисунок 2.16 Результат просмотра страницы task-22_f5.html в браузере

Самостоятельно на основе html-документа task-22_f4.html правильно объедините три ячейки первой строки: Ячейки 12, 13 и 14. Сохраните html-документ под именем **task-22_f6.html** и просмотрите (рис. 2.17). Сделайте выводы.

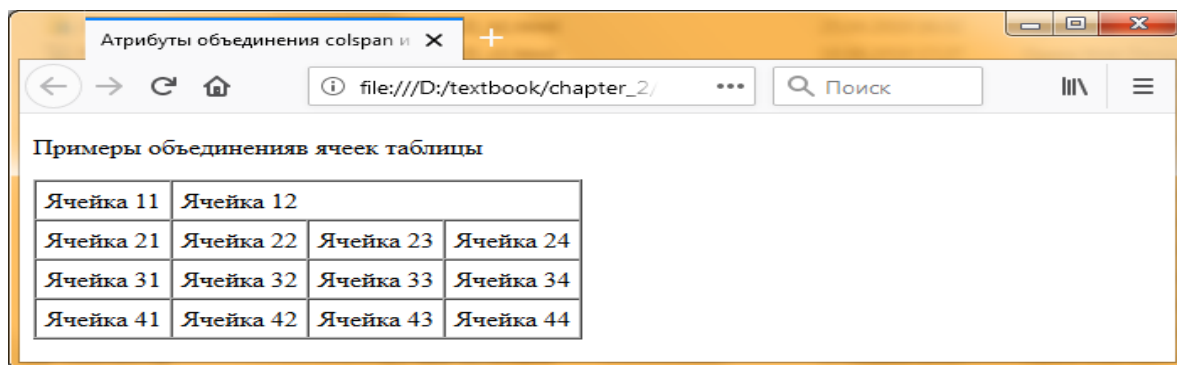


Рисунок 2.17 Результат просмотра страницы task-22_f6.html в браузере

Самостоятельно на основе html-документа task-22_f4.html правильно объедините: Ячейки 22, 23, 32 и 33. Сохраните html-документ под именем **task-22_f7.html** и просмотрите в браузере Firefox (рис. 2.18). Сделайте выводы.

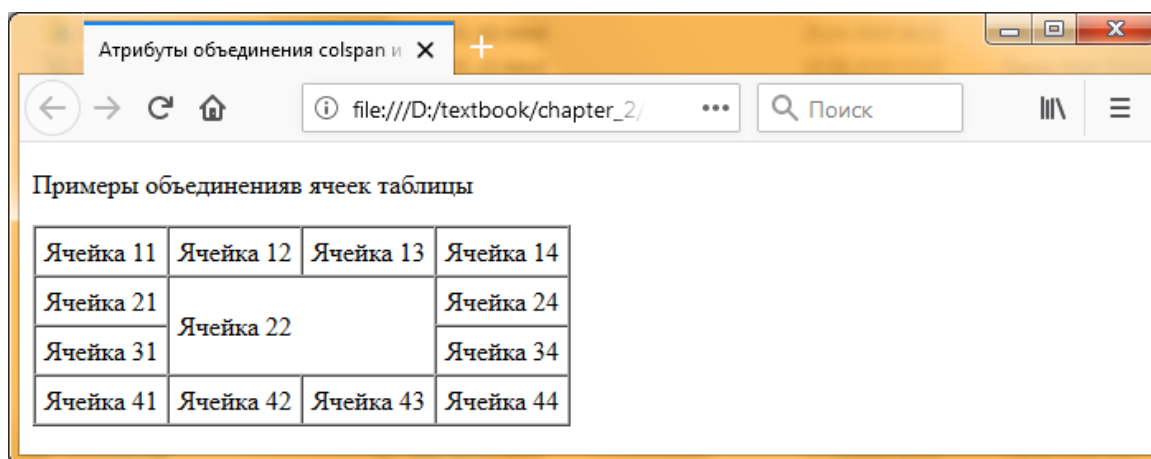


Рисунок 2.18 Результат просмотра страницы task-22_f7.html в браузере

Во второй главе в качестве сквозного примера практической части будет разрабатываться небольшой Web-сайт транспортной организации по перевозке грузов и других услуг под условным названием «Альфа-Транс». Поэтому создайте в папке «D:\Textbook\Chapter_2» папку **AlfaTrans.ru**, в которой будете сохранять разрабатываемые в дальнейшем html-страницы.

Для начала скопируете в созданную папку файл about.html,

содержащий информацию о транспортной компании, файл фона fon_01.jpg и файл шаблона temp.html, созданные в первой главе. Просмотрите html-страницу в браузере, чтобы проконтролировать правильность ее отображения в браузере.

Дальше в заданиях будут создаваться остальные html-страницы сайта. И первым создайте html-страницу прайс-листа **price.html**, с помощью которой можно демонстрировать использование таблиц как средство для представления табличных данных.

Задание № 2.2.g. Использование таблиц для представления числовых данных.

На рисунке 2.19 показан типичный пример использования таблиц для представления числовых данных, разбитых по строкам и столбцам.

The screenshot shows a web browser window with the address bar containing 'file:///D:/textbook/chapter_2/alfatranc.ru'. The main content area displays a table titled 'Тарифы на грузоперевозки Для юридических лиц' (Freight rates for legal entities). Below it is another table titled 'Для физических лиц' (For physical entities). Both tables have columns for vehicle type, weight, minimum order time, hourly price, minimum order amount, and a surcharge for Maykop.

Тарифы на грузоперевозки Для юридических лиц					
Тип Т/С	Вес, тонн	Мин. заказ, часы	Цена 1 часа	Мин. заказ, руб.	за Майкопом Км,руб.
Газель	1,5	3+1	600	3600	16
Удлинённая Газель	1,5	3+1	700	4600	18
ЗИЛ «Бычок»	3	5+1	800	5600	20
Для физических лиц					
Тип Т/С	Вес, тонн	Мин. заказ, часы	Цена 1 часа	Мин. заказ, руб.	за Майкопом Км,руб.
Газель	1,5	3+1	500	3000	14
Удлинённая Газель	1,5	3+1	600	4000	16
ЗИЛ «Бычок»	3	5+1	700	5000	18

Рис. 2.19 Использование таблиц для представления числовых данных

Ниже приведен фрагмент html-кода этой страницы.

Фрагмент html-кода страницы прайс-листа

```

<table border="1">
<th colspan="6">
    Тарифы на грузоперевозки <br>Для юридических лиц
</th>
<tr bgcolor="azure">
    <th> Тип Т/С</th>
    <th> Вес, тонн</th>
    <th>Мин. заказ, часы</th>
    <th>Цена 1 часа</th>
    <th>Мин. заказ, руб.</th>
    <th>за Майкопом км,руб.</th>
</tr>
<tr align="center">
    <td>Газель</td>
    <td>1,5</td>
    <td>3+1</td>
    <td>600</td>
    <td>3600</td>
    <td>16</td>
</tr>
</table>

```

Комментарий. Вначале шесть ячеек первой строки объедините и сделайте надпись, оформив ее как заголовок: фон второй строки выделите цветом и оформите как заголовок.

Откройте шаблон html-документа temp.html из папки «AlfaTrans.ru». В тело документа добавьте вышеприведенный фрагмент html-кода (Листинг 2.1). Сохраните шаблон html-документа под именем **price.html** в папке «AlfaTrans.ru». Просмотрите документ в браузере Firefox.

Используя данные из файла «Тарифы на грузоперевозки.docx» самостоятельно добавьте недостающие строки в html-документ price.html (рис. 2.19). Сохраните и обновите документ в браузере.

Тарифы на грузоперевозки Для юридических лиц					
Тип Т/С	Вес, тонн	Мин. заказ, часы	Цена 1 часа	Мин. заказ, руб.	за Майкопом км,руб.
Газель	1,5	3+1	600	3600	16

Рисунок 2.20 Результат просмотра фрагмента html-документа price.html

Следующий html-документ, который нужно создать для будущего сайта - **autopark.html**. Данный html-документ продемонстрирует, что использование таблицы является средством форматирования Web-страниц.

Задание № 2.2.н. Использование таблицы как средство форматирования Web-страниц.

На рисунке 2.21 представлено использование таблицы в целях форматирования документа. При просмотре такого документа сразу не видно, что для его построения используется таблица, так как рамки вокруг ее ячеек не прорисовываются.

Листинг 2.2

Фрагмент html-кода страницы автопарка

```

<table>
<th colspan="3"> Автопарк транспортной компании АЛЬФА-ТРАНС
</th>
<tr>
<td> </td>
</td>

```

```

        <td> </td>
    </tr>
    <tr>
    <td align="center">Газель</td>
    <td align="center">Газель удлинённый</td>
    <td align="center">Зил Бычок</td>
    </tr>
</table>

```

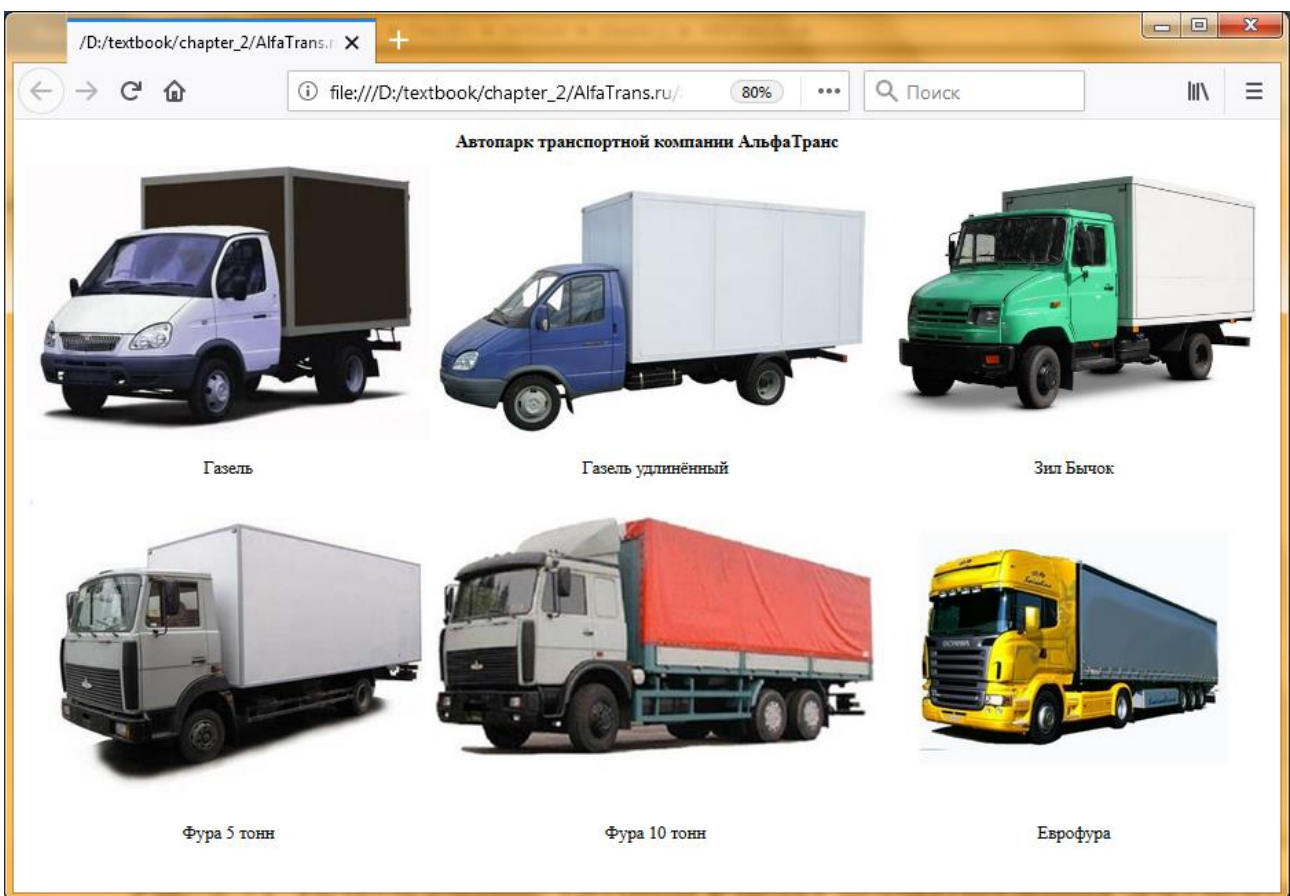


Рисунок 2.21. Использование таблицы в целях форматирования документа.

Комментарий. Вначале три ячейки первой строки объедините и сделайте надпись: `<th colspan = "3">Автопарк транспортной компании</th>`.

В трех ячейках второй строки размещены фотографии автомобилей. При этом желательно, чтобы они были одинакового

размера, поэтому ширина всех рисунков устанавливаете равным `width="200"`.

Остальные строки оформите аналогично.

Далее выполните следующие действия:


– из папки «Учебные материалы» скопируйте файлы **gazel.jpg**, **gazel_udlinennaya.jpg**, **zil_bychok.jpg** в папку AlfaTrans.ru\Images. Откройте шаблон html-документа temp.html. В заголовочной части добавьте строку:

```
<title>Автопарк транспортной компании АЛЬФА-ТРАНС</title>
```

– в тело документа добавьте приведенный в Листинге 2.2 фрагмент html-кода. Сохраните шаблон html-документа под именем **autopark.html**. Просмотрите результат в браузере Firefox.

Используя рисунок 2.21, скопируйте недостающие графические файлы **5_tonn.jpg**, **10_toon.jpg** и **eurofura.jpg** из папки «Учебные материалы» в папку «AlfaTrans.ru\Images» и самостоятельно добавьте недостающие строки в html-документ autopark.html. Сохраните и обновите документ в браузере.

2.3 Вставка гиперссылок в HTML-документ

Гипертекст – это текст, обладающий возможностями навигации, а гиперссылки – ключевой, практически самый главный элемент гипертекста. Связь между html-документами и фрагментами документов организуется с помощью гиперссылки. Наличие ссылок является фундаментальным свойством веб-страниц. Внешне отличить гиперссылку от обычного текста просто: при наведении указателя мыши на гиперссылку указатель принимает вид руки с указательным пальцем , а в строке состояния (в нижней части окна программы браузера) отображается URL – адрес ссылки. Если указателем является текст, то ссылка подчеркивается.

При щелчке мышью по гиперссылке осуществляется переход к тому объекту, на который ссылается гиперссылка.

Гиперссылка может ссылаться на:

– любой участок web-страницы, то есть осуществляется переход к определенной главе, разделу, находящимся на этой же странице;

– любую web-страницу;

– документ любого формата (документы совместимых форматов открываются непосредственно в окне обозревателя), либо используются соответствующие (ассоциированные) программы, например, для просмотра видео, прослушивания звуковых файлов и т.п.;

– любой файл – в этом случае начинается его загрузка и запись на жесткий диск вашего компьютера (локальный диск);

– адрес электронной почты – открывается окно для создания сообщения.

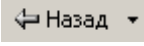

Гиперссылка имеет 4 состояния:

– непосещенная гиперссылка (перехода по данной гиперссылке не было);

– посещенная гиперссылка (был переход по данной гиперссылке, либо был переход на страницу, на которую ссылается гиперссылка);

– активная гиперссылка – щелчок по гиперссылке был сделан, но новая страница еще не начала загружаться;

– подсвеченная гиперссылка – гиперссылка, на которую наведен указатель мыши.

Браузер постоянно отслеживает все переходы по гиперссылкам и регистрирует ваш путь, поэтому возвратиться на прежнюю страницу можно нажав кнопку **Назад** , если ее нажать несколько раз, то вы пройдете весь путь в обратном направлении. После того как вы отошли назад, можно по тому же самому пути пройти вперед с помощью кнопки **Вперед** . С помощью треугольников, имеющих на этих кнопках, можно открыть меню, в которых перечислены пройденные страницы, и сразу выбрать нужную страницу.

Гиперссылка состоит из двух частей: указателя ссылки и адреса ресурса, на который необходимо осуществить переход.

Указатель ссылки описывается тегом `<a>`, а адрес перехода реализуется с помощью параметра `href`, значением которого является путь к тому или иному интернет - ресурсу. Закрывающий тег `` является обязательным. Формат этого параметра задается следующим образом:

` выделенный фрагмент текста`

Цвет гиперссылок. По умолчанию ссылки выделяются синим цветом и подчеркиваются. Ссылки на документы, которые вы уже посетили, выделяются фиолетовым цветом. Изменить эти цвета можно в параметрах тега `<body>`:

`<body link="red" vlink="#00ff00" alink="blue">`

Параметр `link` обозначает цвет обычной ссылки, `vlink` – цвет посещенной ссылки, `alink` – цвет ссылки, над которой находится курсор мыши.

Далее будут рассмотрены различные способы использования гиперссылок.

2.3.1 Гиперссылки внутри документа

В случае если у вас имеется большой по объему документ, то для навигации по нему бывает удобно указать ссылки внутри этого документа. Данный подход будет полезен, если вы будете составлять, например, техническую документацию или юридический договор, где пользователю понадобится быстрый переход от одного пункта документа к другому.

Структура внутренней гиперссылки имеет две части – это сама ссылка и ее именной идентификатор, данная запись позволит вам переместиться в нужное место вашего электронного документа.

Далее дается пример составления внутренних ссылок, который рассмотрен в листинге ниже.

Задание № 2.3.а. Использование гиперссылки внутри большого документа.

В качестве примера необходимо разработать для сайта страницу, содержащую информацию о сервисах транспортной компании «АЛЬФА-ТРАНС».

Этот документ будет содержать общую информацию о компании, оглавление и описание оказываемых услуг: сборные перевозки по России, квартирный переезд, дачный переезд и услуги грузчиков.

Вначале текста поместите общую информацию о компании. После общей информации поместите оглавление – список сервисов компании. Оглавление оформите в виде нумерованного списка. Вслед за оглавлением поместите описание каждого сервиса.

Ниже приведен фрагмент варианта оформления html-страницы сервисами компании. На рисунке 2.22 результат просмотра данной html-страницы в браузере Firefox.

Листинг 2.3

Фрагмент html-документа service.html

```
<font face="Comic Sans MS, Arial" color="green" size="4">
<p align="center"><a name="service">ТРАНСПОРТНАЯ
КОМПАНИЯ АЛЬФА-ТРАНС <br>Наши услуги</a></p>
</font>
<ol><font color="#400040" size="4">
<li><a href="#service_1"> «Сборные перевозки по России»
</a></p>
<li><a href="#service_2"> «Квартирный переезд» </a></p>
<li><a href="#service_3"> «Дачный переезд» </a></p>
<li><a href="#service_4"> «Услуги грузчиков» </a></p>
</ol> </font>
<font face="Comic Sans MS, Arial" color="red" size="4">
<p align="center"><a name="service_1"><b>«Сборные перевозки
по России»</b></a></p>
</font>
```

```
<font face="Comic Sans MS, Arial" color="green" size="3">
```

```
<p align="justify">
```

На сегодня доставка сборным грузом представляет собой отличный вариант транспортировки малогабаритных грузов (в одной машине от 15 до 60 получателей). Этот способ подходит для людей и организаций, нуждающихся в отправке небольшой партии продукции. Он отлично подходит для интернет-магазинов, которые могут отправить посылку, весом от 1 кг.

```
<p><a href="#service">Оглавление</a></p>
```

```
<p><a name="service_2"> «Квартирный переезд» </a></p>
```

```
<p>Описание сервиса «Квартирный переезд» </p>
```

```
<p><a href="#service">Оглавление</a></p>
```

```
<p><a name="service_3"> «Дачный переезд» </a></p>
```

```
<p>Описание сервиса «Дачный переезд» </p>
```

```
<p><a href="#service">Оглавление</a></p>
```

```
<p><a name="service_4"> «Услуги грузчиков» </a></p>
```

```
<p>Описание сервиса «Услуги грузчиков» </p>
```

```
<p><a href="#service">Оглавление</a></p>
```

В этом примере тегом **name** задается имя для каждого сервиса (service_1, service_2, service_3 и т.д.), и идет ссылка на эти имена в начале документа. Когда посетитель щелкает мышью на ссылку, браузер автоматически прокручивает страницу на то место, где расположено соответствующее этой ссылке имя.

Чтобы создать html-документ **service.html** выполните следующие действия:

1. Откройте шаблон сайта html-документа temp.html, размещенный в папке «D:\Textbook\Chapter_2\AlfaTrans.ru» и произведите в нем следующие изменения:

- в теге <title> измените текст заголовка на:

```
<title> Сервис транспортной компании АЛЬФА-ТРАНС</title>;
```

– данный html-документ должен содержать рисунки.

Скопируйте в папку Images файлы рисунков **kvartirnniy_pereezd.jpg, dachny_pereezd.jpg, gruzchiki_dly_pereezda.jpg** из папки «Учебные материалы\Рисунки» прилагаемого диска;

– в тело документа добавьте приведенный в Листинге 2.3 фрагмент html-кода. Сохраните шаблон html-документа под именем **service.html**. Просмотрите результат в браузере.

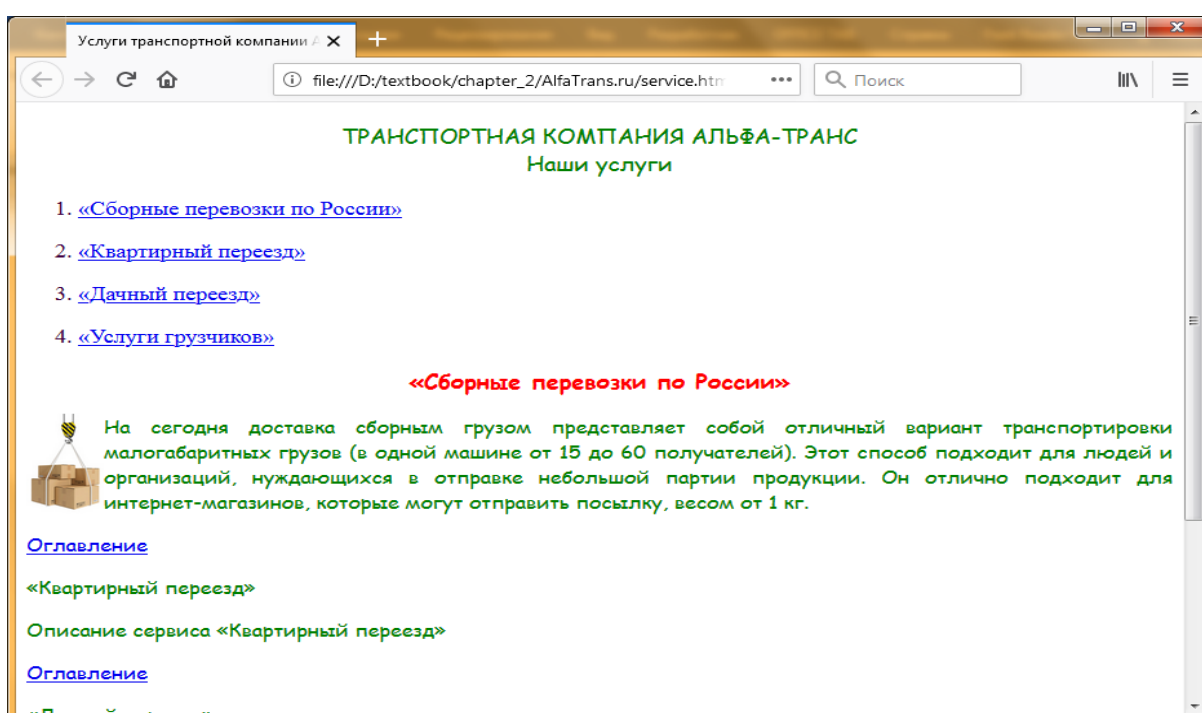


Рисунок 2.22 Результат просмотра страницы service.html

В примере дано описание лишь одного сервиса «Сборные перевозки по России». Описание остальных сервисов добавьте самостоятельно. Материалы возьмите из файлов «Услуги.docx», «Услуги.txt» или из Интернета.

Стилевое оформление и форматирование текста документа подберите самостоятельно. Сохраните и обновите документ в браузере. Щелкая по ссылкам, проверьте правильность их работы.

2.3.2 Гиперссылки на другие документы

Как было показано выше, общий синтаксис вставки в документ гиперссылки имеет вид:

** выделенный фрагмент текста**

Если в качестве значения параметра href указать абсолютный или относительный адрес html-страницы, то эти ссылки позволят переходить с одного html-документа на другой.

По умолчанию при переходе по ссылке на эти документы, они будут открываться в текущем окне, перекрывая собой главную страницу, с которой были проставлены гиперссылки на внешние страницы. При необходимости это условие может быть изменено атрибутом **target** (назначение) тега <a>. Ниже представлены все значения этого атрибута (табл. 2.1).

Таблица 2.1

Значения атрибута **target**

Атрибут	Значение
_blank	Загружает страницу в новое окно браузера.
_self	Загружает страницу в текущее окно. Это значение по умолчанию.
_parent	Загружает страницу во фрейм-родитель, если фреймов нет, то это значение работает как _self.
_top	Отменяет все фреймы и загружает страницу в полном окне браузера, если фреймов нет, то это значение работает как _self.
framename	Открывает ссылку в указанном фрейме.

Последние три значения атрибута target используются при использовании фреймов (см. главу 3).

Далее рассматривается процесс окончательного создания сайта условной транспортной компании «АЛЬФА-ТРАНС». В ходе этой работы будет продемонстрирован механизм использования гиперссылок на другие документы.

Сайт будет содержать следующие страницы:

- **index.html** – основная страница, содержит краткую

информацию о транспортной компании «АЛЬФА-ТРАНС» и ссылки на другие страницы;

- **about.html** – содержит общую информацию;
- **service.html** – страница, содержащая информацию об оказываемых услугах транспортной компанией;
- **news.html** – страница с новостями;
- **autopark.html** – фотогалерея автопарка транспортной компании «АЛЬФА-ТРАНС»;
- **price.html** – прайс-лист содержит цены оказываемых услуг.

В ходе выполнения практических заданий большинство из этих страниц (**about.html**, **service.html**, **autopark.html** и **price.html**) было уже подготовлено, а некоторые будут создаваться по ходу работы.

Задание № 2.3.б. Использование гиперссылки на другие документы.

Рассмотрение пошагового создания сайта:

1 шаг. Создайте главную страницу – файл **index.html**.

С точки зрения веб-разработки на главной странице должны быть следующие элементы:

1. Шапка сайта (или хедер). Это самая верхняя часть страницы, на которой обычно размещен логотип и название компании (в правом верхнем углу). Может включать в себя телефоны компании, рабочие часы, кнопку обратного звонка, адрес электронной почты.

2. Меню. Главная цель меню – дать посетителю понять, как перейти в другие разделы сайта. Оно может быть горизонтальным, вертикальным, комбинированным.

3. Форма обратной связи. Позволит посетителю сразу же связаться с вами.

4. Подвал сайта или футер - это завершающий блок страницы, в который обычно выносят полезные ссылки (например, на пользовательское соглашение), копирайт, иногда – карту проезда и контакты, могут перечислить филиалы и т.п.

На рисунке 2.23 изображена схема начальной страницы сайта транспортной компании «АЛЬФА-ТРАНС».

Логотип	Название компании
Главное меню	Общая информация
Сведения об авторских правах	

Рисунок 2.23 Схема начальной страницы сайта index.html

Разместите на этой странице общую информацию о транспортной компании, его задачах, целях создания сайта, логотип, название сайта и он должен содержать ссылки на внешние html-документы – about.html, autopark.html, service.html, news.html, gazel.html, price.html и ссылку на электронную почту компании.

Для того чтобы при открывании с начальной страницы сайта index.html внешние html-документы не перекрывали главную страницу, в теге <a> будет использоваться атрибут target со значением _blank (Листинг 2.4). Тогда внешние html-документы будут открываться в новых вкладках браузера.

Сайт любой компании должен иметь обратную связь (e-mail), чтобы посетители могли задавать интересующие их вопросы прямо на сайте, поскольку не всегда у них есть время на личную встречу.

HTML позволяет создать гиперссылку, указывающую на адрес электронной почты (почтовую гиперссылку). Щелчок на ней запустит программу почтового клиента, установленную в системе по умолчанию. Интернет-адрес такой гиперссылки записывается особым образом.

Пусть вы хотите создать гиперссылку, указывающую на почтовый адрес: **alfatrans@mail.ru**.

Согласно стандарту html, этот почтовый адрес должен быть записан так:

mailto: alfatrans@mail.ru

Тогда почтовая гиперссылка будет выглядеть так:

Отправить письмо

Адрес электронной почты в этой гиперссылке выдуман.

Разумеется, вы можете заменить его на реальный электронный адрес.

Главную страницу сайта реализуйте с использованием табличного оформления сайта.

Страница будет разбита на три горизонтальные строки. При этом первые две строки разбиты на два столбца, а последние две объединены. Поэтому создайте таблицу, состоящую из трех строк и двух столбцов. После ячейки третьей строки объедините в одну.

Ниже приведен фрагмент варианта оформления html-документа главной страницы index1.htm:

Листинг 2.4

Фрагмент html-документа главной страницы

```
<table border="0" cellspacing="0" cellpadding="16">
<tr align="center">
<td bgcolor="green">
        
</td>
<td bgcolor="teal">
        <h1> Транспортная компания АЛЬФА-ТРАНС</h1>
</td>
</tr>
<tr>
<td bgcolor="azure"> Главное меню <br>
        <a href ="about.html" target="_blank"> О нас </a> <br>
        <a href ="news.html" target="_blank"> Новости </a> <br>
        <a href ="service.html" target="_blank">Наши услуги </a>
<br>
```

```
<a href = "autopark.html" target = "_blank" > Автопарк</a> <br>
<a href = "price.html" target = "_blank" > Прайс-лист</a> <br>
<a href = "mailto: alfatrans@mail.ru" >
<img src = "Images\postbox.gif" >
<br> на наш ящик </a><br>
```

```
</td>
```

```
<td valign = "top" align = center >
```

```
<font face = "Comic Sans MS, Arial" color = "red" size = "5" >
```

```
<p align = "center" >
```

```
Грузоперевозки с компанией «АЛЬФА-ТРАНС»
```

```
</p>
```

```
<font face = "Comic Sans MS, Arial" color = "green" size = "3" >
```

```
<p align = "justify" >
```

Транспортная компания «АЛЬФА-ТРАНС» осуществляет перевозки в большое количество городов России и р. Адыгея. У нас в реестре более 2000 населенных пунктов. Первоочередным направлением деятельности компании является организация широкого спектра грузоперевозок. Нашим клиентам предлагаются выгодные условия сотрудничества, организация может выполнить погрузку, разгрузку, экспедирование, своевременную обработку грузов, в нашем распоряжении имеются складские помещения. Благодаря тому, что компания не первый день на рынке, у нас сложилась профессиональная команда специалистов, наша ценовая политика заинтересует все категории клиентов, что в свою очередь отличает нас от других фирм, занимающихся перевозками на территории России и Республики Адыгея.

Почему вы должны выбрать именно нашу компанию: </p>


```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td colspan = "2" >
```

Политика конфиденциальности
Все права защищены © 2013-2017
Транспортная компания «РУСТРАНС»
ОГРН: 1155029012610 ИНН: 5029206268
8 (926) 805-98-67 / 8 (495) 668-06-96
г. Майкоп, улица Учебная, д. 12

```
</td>  
</tr>  
</table>
```

Далее выполните следующие задания:

1. Откройте шаблон сайта html-документа temp.html, размещенный в папке «D:\Textbook\Chapter_2\AlfaTrans.ru» и произведите в нем следующие изменения:

– данный html-документ должен содержать рисунки. Скопируйте в папку Images файлы рисунков **logo.png** - логотип компании и **postbox.gif** – анимация почтового ящика из папки «Учебные материалы\Рисунки» прилагаемого диска;

– в тело документа добавьте приведенный в Листинге 2.4 фрагмент html-кода. Сохраните шаблон html-документа под именем **index.html**. Просмотрите результат в браузере (рис. 2.24).

В примере дано частичное описание деятельности транспортной компании «АЛЬФА-ТРАНС». Полная информация хранится в файле index.docx (index.txt), который находится в папке «Учебные материалы\АЛЬФА-ТРАНС». Используя материал из этого документа, самостоятельно добавьте в html-документ index.html недостающую информацию. Стилевое оформление и форматирование текста документа подберите самостоятельно. Сохраните и обновите документ в браузере Firefox.

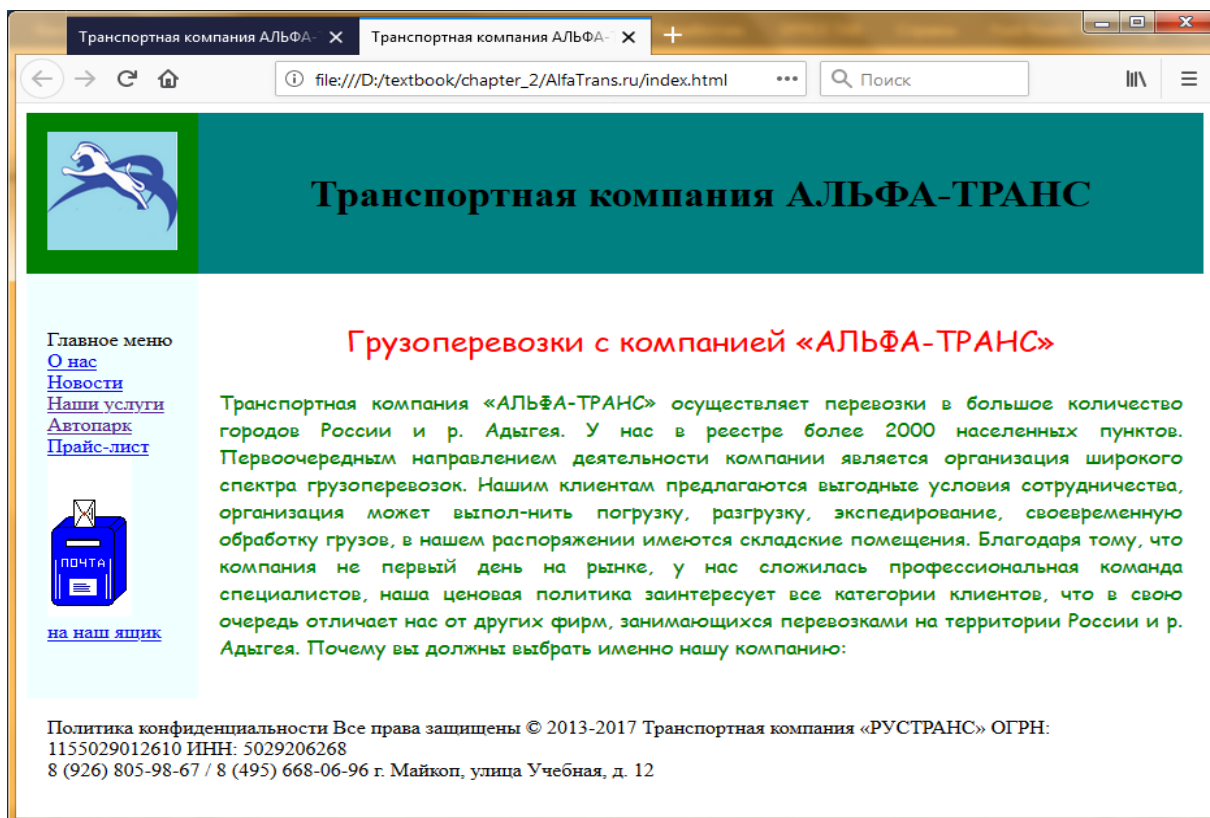


Рисунок 2.24 Результат просмотра страницы index.html

Как было указано выше, вами уже созданы большинство внешних html-документов, на которые должны быть ссылки. Поэтому результатом щелчка на ссылки для несозданных html-документов, например, "Новости" будет окно, изображенное на рисунке 2.25.

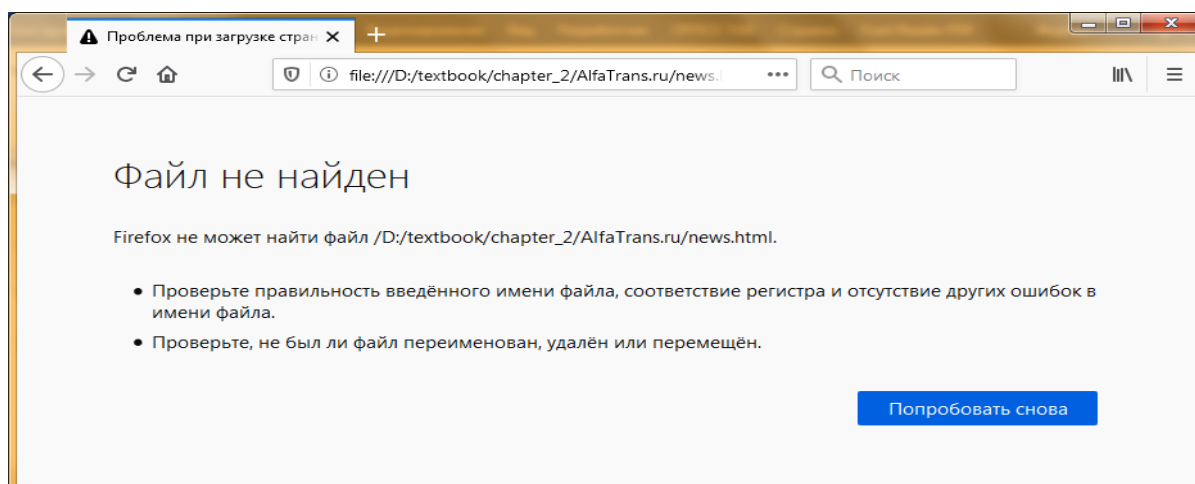


Рисунок 2.25 Невозможность отображения страницы

А результатом щелчка на ссылки "О нас" (html-документ `about.html`), "Наши услуги" (html-документ `service.html`), "Автопарк" (html-документ `autopark.html`) и "Прайс-лист" (html-документ `price.html`) будет открывание соответствующих web-страниц.

Сразу бросается в глаза два недостатка организации такого сайта:

1. При щелчке на гиперссылку окно с меню исчезает и вместо него появляется соответствующая страница. В дочерних web-страницах отсутствует механизм возврата в главную страницу. Как это делается, можно увидеть при создании недостающей web-страницы с новостями.

2. При организации такого сайта удобнее было бы, если меню все время присутствовало на экране. В html это организовывается с помощью фреймов. Тему фреймы вы рассмотрите в следующей главе. Там же будут рассмотрены общие требования к элементам главной страницы, а именно к логотипу, меню, основному содержанию сайта и подвалу.

2 шаг. Создайте недостающую web-страницу с новостями – html-документ **news.html**. Вы создаете лишь заготовку будущего сайта, поэтому эта страница пока будет содержать минимум информации. В листинге 2.5 приводится фрагмент html-документа новостной страницы:

Листинг 2.5

Фрагмент html-документа `news.html`

```
<a href ="index.html"> На главное </a> <br>
<p><font size="5" color="red">
  <marquee>Последние новости</marquee>
</font></p>
<table border="1" width="100%" cellspacing="0" cellpadding="16">
<tr>
  <td width="100" rowspan="2">
```

```

        <font size="5" color="red">6.02.2019</font>
    </td>
    <td>
        <font size="5" color="blue">Заголовок</font>
    </td>
</tr>
<tr>
    <td>
        <font size="5" color="black">Текст новости</font>
    </td>
</tr>
</table>

```

Выполните следующие действия:

- откройте шаблон сайта html-документа temp.html, размещенный в папке «D:\Textbook\Alfatrans.ru» и произведите в нем следующие изменения: в теге <title> измените текст заголовка на:

```
<title> Новости транспортной компании АЛЬФА-ТРАНС</title>
```

- в тело документа добавьте приведенный в Листинге 2.5 фрагмент html-кода. Сохраните шаблон html-документа под именем **news.html**. Просмотрите результат в браузере (рис. 2.26).

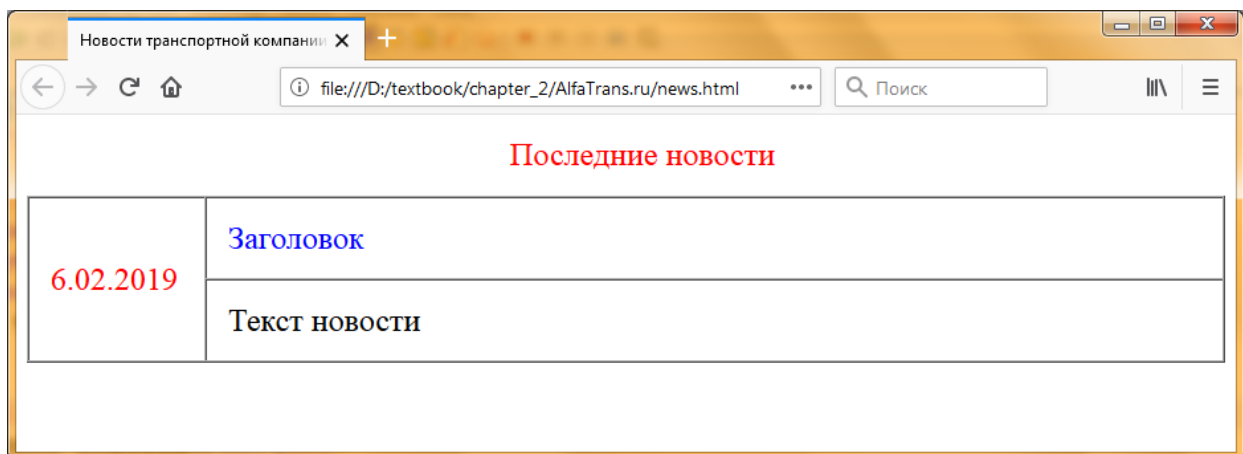


Рисунок 2.26 Результат просмотра страницы news.html

– в каждом из html-документов about.html, service.html, autopark.html price.html, сразу же после тега <body> добавьте ссылку на главную страницу сайта:

```
<a href = "index.html"> На главное </a> <br>
```

Сохраните эти документы. Попробуйте теперь с главной страницы сайта делать с помощью меню переходы на нужные web-страницы и возвращаться обратно.

Самостоятельно попробуйте изменить с помощью соответствующих тегов цвет гиперссылок, выбранных и уже посещенных гиперссылок. Измените цвет, формат текста, а также фон. Добавьте на страничках изображения, фото и необходимую информацию.

2.3.3 Гиперссылки с помощью карты изображения

HTML позволяет превратить в гиперссылку часть графического изображения. Можно разбить изображение на части, каждая из которых будет представлять собой гиперссылку, указывающую на свой интернет-адрес. Такие изображения называют картами изображения, а ее части-гиперссылки – «горячими» областями.

Карты изображения создают в три этапа.

Первый этап – создание самого изображения с помощью тега :

```
<img src = "map.gif">
```

Второй этап – создание карты - особого элемента Web-страницы, который описывает набор «горячих» областей карты изображения. Сама карта на Web-странице никак не отображается.

Карту создают с помощью парного тега <map>:

```
<map name = "имя карты"> </map>
```

С помощью обязательного атрибута **name** тега <map> задается уникальное в пределах Web-страницы имя карты. Оно однозначно идентифицирует данную карту, может содержать только латинские буквы, цифры и знаки подчеркивания и начинаться должно с буквы:

<map name="samplemap"> </map>

После создания карты следует привязать ее к созданному на первом этапе изображению. Для этого применяется обязательный в данном случае атрибут **usemap** тега ****. Его значение – имя привязываемой к изображению карты, причем в начале этого имени обязательно следует поставить символ **#** ("решетка"). (В имени, заданном атрибутом **name** тега **<map>**, символ **#** присутствовать не должен).

Третий этап - создание описания самих «горячих» областей в карте. Их помещают внутрь соответствующего тега **<map>** и формируют с помощью одинарных тегов **<area>**:

<area [shape="rect|circle|poly"] coords="набор параметров" -href="интернет-адрес гиперссылки"|nohref -target="цель гиперссылки">

Необязательный атрибут **shape** задает форму "горячей" области. Обязательный атрибут **coords** перечисляет координаты, необходимые для построения этой области.

Значения атрибута **shape**:

– **"rect"** – прямоугольная "горячая" область. Атрибут **coords** в этом случае записывается в виде **coords="<X1>,<Y1>,<X2>,<Y2>"**, где **X1** и **Y1** – координаты верхнего левого, а **X2** и **Y2** – правого нижнего угла прямоугольника. *Замечание.* Если атрибут **shape** не указан, создается именно прямоугольная область.

– **"circle"** – круглая "горячая" область. В этом случае атрибут **coords** имеет вид **coords="<X центра>,<Y центра>,<радиус>"**;

– **"poly"** – "горячая" область в виде многоугольника. Атрибут **coords** должен иметь вид **coords="<X1>,<Y1>,<X2>,<Y2>,<X3>,<Y3>..."**, где **Xn** и **Yn** – координаты соответствующей вершины многоугольника.

Атрибут **href** задает интернет-адрес гиперссылки, атрибут **target** задает цель гиперссылки.

Рассмотрите пример использования карты изображения.

Задание № 2.3.с. Использование гиперссылки с помощью карты изображения.

Имеется карта - изображение Адыгеи **adygeya.gif** размером 1519x1604 пикселей (рис. 2.27). Для трех населенных пунктов: Майкоп, Каменноостский и Гузерипль создайте три html-документа, на которые надо организовать гиперссылки на карте Адыгеи.

Создайте вначале вспомогательные html-документы: **majkop.html**, **khadgoh.html**, **guseripl.html**, содержащие соответственно информацию о туристических достопримечательностях Майкопа, Каменноостского и Гузерипля. Ниже приводится фрагмент html-кода страницы **guseripl.html**.



Рисунок 2.27 Карта изображение Республики Адыгея

Фрагмент html-документа guseripl.html

```
<p align="center"><font color="#400040" size="5"><b>
```

```
Поселок Гузерипль</b></font></p>
```

```
<p align="justify"><b>Место расположения:</b> поселок
Гузерипль. Левый берег реки Белой, не входящий в территорию
заповедника. Высота над уровнем моря 665, 4 м.</p>
```

```
<p align="justify">Поселок Гузерипль основан в 1924 году как
кордон Кавказского государственного биосферного природного
заповедника. В экскурсионную программу показа в поселке
Гузерипль входят несколько объектов.</p>
```

```
<p align="center"><font color="#A00070"
size="3"><b>Экскурсионные объекты поселка
Гузерипль.</b></font></p>
```

Выполните следующие операции:

1. Откройте шаблон html-документа temp.html из папки AlfaTrans.ru. В его заголовочной части произведите следующие изменения:

- в теге <title> измените текст заголовка на "Экскурсионные маршруты: п. Гузерипль";

- в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;

- в тело документа добавьте вышеприведенный фрагмент html-кода (Листинг 2.6). Сохраните шаблон html-документа под именем **guseripl.html**. Просмотрите документ в браузере (рис. 2.28);

- в примере дано описание части информации о туристических достопримечательностях поселка Гузерипль. Используя материалы из файла «Поселок Гузерипль.docx» сделайте страницу полноценной. Стилизованное оформление и форматирование текста документа подберите самостоятельно.

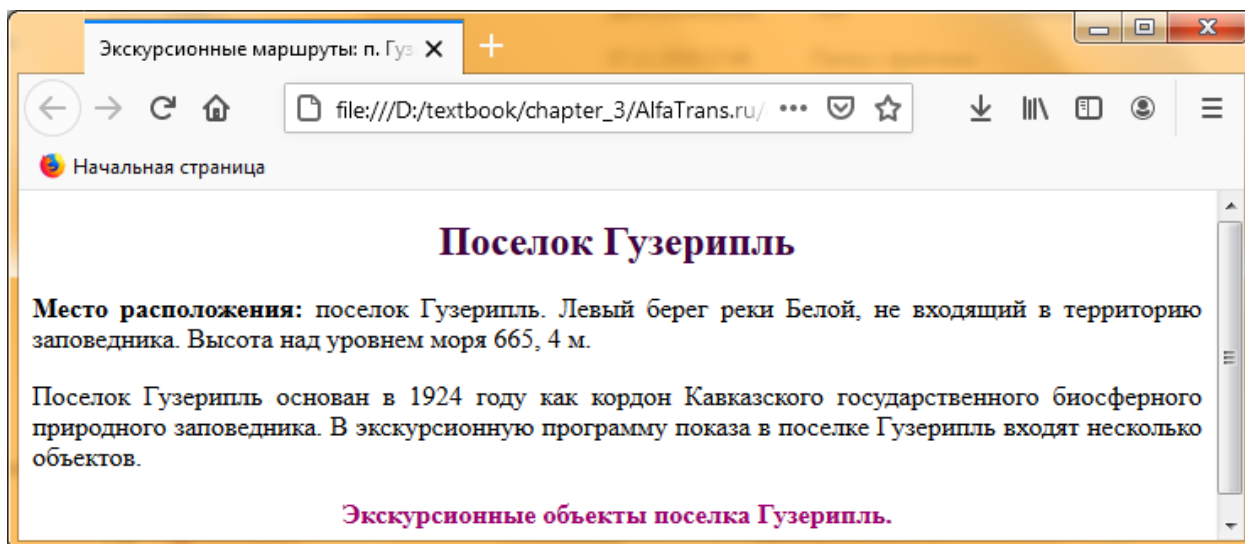


Рисунок 2.28. Результат просмотра страницы guseripl.html в браузере

2. Аналогично создайте самостоятельно страницы **majkop.html** и **khadgoh.html**. Для этого можно использовать документы из прилагаемых файлов. Просмотрите созданные Web-страницы с помощью браузера.

Активными зонами изображения сделайте кружки, условно отмечающие местоположение на карте трех населенных пунктов. На карте активные зоны выделите красными кружочками. Определите координаты этих населенных пунктов: Майкоп, Каменомосткий и Гузерипль. Для этого загрузите карту в графический редактор Paint и, наведя курсор на соответствующие точки определите координаты. Центры кружков будут находиться примерно на уровнях: (990, 760) – Майкоп, (1040, 1050) – Каменномостский, (1015, 1310) – Гузерипль. Выберите в качестве формы активной зоны окружность радиусом равным 9 пикселям и реализуйте гиперссылки с помощью следующего html – кода:

Листинг 2.7

Фрагмент html-документа adygeya.html

```

<map name="adygeya">
  <area shape="circle" coords="990, 760, 9" href="majkop.html"
  target="_blank" alt="Майкоп" title="Майкоп">
```

```
<area shape="circle" coords="1040, 1050, 9" href="khadgoh.html"
target="_blank" alt="Каменномоостский"
title="Каменномоостский">
<area shape="circle" coords="1015, 1310, 9" href="guseripl.html"
target="_blank" alt="Гузерипись" title="Гузерипись">
</map>
```

3. Откройте шаблон html-документа temp.html из папки AlfaTrans.ru. В его заголовочной части произведите следующие изменения:

- в теге <title> измените текст заголовка на "Карта экскурсионных маршрутов по Адыгее";
- в соответствии с содержанием web-страницы в мета-тегах "keywords" и "description" соответственно добавьте ключевые слова и описание страницы;
- в тело документа добавьте вышеприведенный фрагмент html-код (Листинг 2.7). Сохраните шаблон html-документа под именем **adygeya.html**. Просмотрите документ в браузере (рис. 2.29);
- наведите курсор на населенные пункты Майкоп, Каменномоостский и Гузерипись и проверьте, что по щелчку на выделенную область открывается соответствующая web-страница.

Тестовые задания для самоконтроля

Уровень А

1. Из нижеприведенных адресов выберите универсальные указатели ресурса с абсолютной адресацией:

A. http://Textbook/unit_2/task-2.html

B. :/Textbook/unit_2/task-2.html

C. <../..image>

D. <courses/foto>

E. <./courses/foto>

2. Из нижеприведенных адресов выберите универсальные указатели ресурса с относительной адресацией:

- A. http://Textbook/unit_2/task-2.html
- B. D:/Textbook/unit_2/task-2.html
- C. [.././image](#)
- D. [courses/foto](#)
- E. [./courses/foto](#)

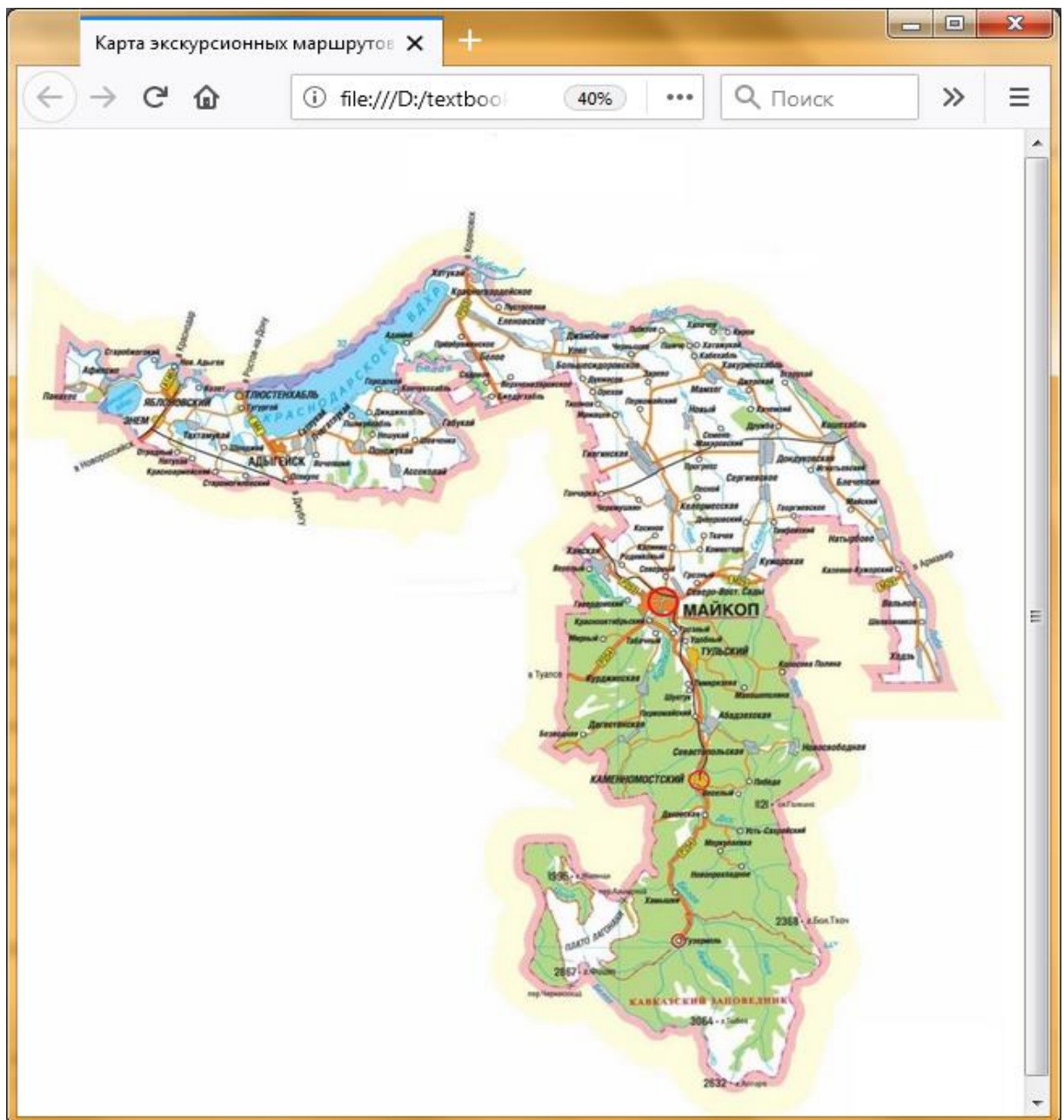


Рисунок 2.29 Результат просмотра страницы adugeya.html в браузере

3. Укажите правильные утверждения относительно абсолютного адреса:

А. точка отсчета абсолютного адреса начинается с адреса сайта

В. точка отсчета абсолютного адреса начинается с места, где находится страница

С. абсолютный адрес никогда не меняется

Д. абсолютный адрес может изменяться в зависимости от того, где находится ссылающаяся страница

Е. абсолютные адреса можно использовать на страницы и файлы разных сайтов

Ф. абсолютные адреса можно использовать только в пределах одного сайта

4. Укажите правильные утверждения относительно относительного адреса:

А. точка отсчета относительного адреса начинается с адреса сайта

В. точка отсчета относительного адреса начинается с места, где находится страница

С. относительный адрес никогда не меняется

Д. относительный адрес может изменяться в зависимости от того, где находится ссылающаяся страница

Е. относительный адрес можно использовать на страницы и файлы разных сайтов

Ф. относительный адрес можно использовать только в пределах одного сайта

5. На странице list.html нужно вставить графическое изображение car.jpg. Файл car.jpg расположен в той же папке, что html-документ list.html. Html-документ list.html размещен по абсолютному адресу: C:\mysite\product\soft. Укажите html-код(ы), который(е) правильно вставляет(ют) рисунок в html-документ list.html.

- A.
- B.
- C.
- D.
- E.

6. Определите количество строк и столбцов в таблице следующего фрагмента html-документа:

```
<table>
<tr>
  <td> Привет </td>
  <td> Привет </td>
  <td> Привет </td>
</tr>
<tr>
  <td> Привет </td>
  <td> Привет </td>
</tr>
</table>
```

- A. 2 строки 3 столбца
- B. 3 строки 2 столбца
- C. 2 строки 2 столбца
- D. 3 строки 3 столбца

Уровень В

1. Для управления изображением, вставляемым в текст html-документа, используются соответствующие атрибуты тега . Для каждого атрибута тега , указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- | | | | |
|----------|---------------------|--------------|-------------|
| A. title | 1. задает | выравнивание | изображения |
| B. alt | относительно текста | | |

- C. align 2. задает описание изображения
 3. задает текст, заменяющий изображение в том случае, если браузер не воспринимает это изображение

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

A	B	C

2. Для каждого атрибутом тега <table>, указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- A. height 1. определяет ширину таблицы
 B. cellspacing 2. определяет способ вертикального
 C. valign выравнивания для содержимого таблицы
 D. bgcolor 3. определяет расстояние (в пикселях) между
 E. width границами соседних ячеек
 4. определяет высоту таблицы.
 5. определяет цвет фона ячеек таблицы

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

A	B	C	D	E

Уровень С

1. В какой абсолютный адрес преобразуется адрес в ссылке image, если текущий адрес задается в виде <http://mysite.ru/news/1>?

Ответ: ...

2. В какой абсолютный адрес преобразуется адрес в ссылке ../image, если текущий адрес задается в виде <http://mysite.ru/users/profile/news?>

Ответ: ...

3. В какой абсолютный адрес преобразуется адрес в ссылке ..., если текущий адрес задается в виде <http://mysite.ru/users/profile/news?>

Ответ: ...

4. Добавьте в html-код `` атрибут, задающий текст «Файл рисунка не найден», который заменяет изображение в том случае, если браузер по каким-то причинам не воспринимает изображение.

Ответ: ...

5. Добавьте в html-код `` атрибут выравнивания изображения по правому краю.

Ответ: ...

6. Измените html-код `` так, чтобы расстояние по вертикали от рисунка до текста было 20 пикселей, а по горизонтали от рисунка до текста – 15 пикселей.

Ответ: ...

Глава 3. Работа с фреймами

Фреймы делят область отображения информации в браузере на прямоугольные части (или подокна), в каждой из которых отображается отдельный html-документ. Использование фреймов значительно улучшает внешний вид и функциональность информационных систем и Web-приложений. Каждый фрейм может иметь следующие свойства:

- имеет свой URL, что позволяет загружать его независимо от других фреймов;
- имеет собственное имя (параметр name), позволяющее переходить к нему из другого фрейма;
- размер фрейма может быть изменен пользователем прямо на экране при помощи мыши (если это не запрещено указанием специального параметра).

Данные свойства фреймов позволяют создавать продвинутые интерфейсные решения, такие как:

- размещение статической информации, которую автор считает необходимым постоянно показывать пользователю, в одном статическом фрейме. Это может быть графический логотип фирмы, copyright, набор управляющих кнопок;
- помещение в статическом фрейме оглавления всех или части WEB-документов, содержащихся на WEB-сервере, что позволяет пользователю быстро находить интересующую его информацию;
- создавать окна результатов запросов, когда в одном фрейме находится собственно запрос, а в другом результаты запроса;
- создавать формы типа "мастер-деталь" для WEB-приложений, обслуживающих базы данных.

При создании страницы с фреймами разрабатывается несколько html-документов, которые отличаются по своему назначению. Документ (файл) раскладки определяет структуру окна браузера, то есть количество и вид областей, на которые разделено это окно.

Необходимое количество областей и их вид (или горизонтальные, или вертикальные, или комбинация) определяется разработчиком WEB-страницы. Html-документы содержания предназначены для заполнения информацией каждой из областей.

Большинство новых версий браузеров поддерживают фреймы в обязательном порядке. Для случая, когда поддержка фреймов в браузере отсутствует, с помощью элемента `<noframes>...</noframes>` предусматривается ввод предупреждающего сообщения и (или) гиперссылки, адресующей к версии страницы без фреймов.

3.1 Синтаксис фреймов

Формат html-документа, использующего фреймы, внешне похож на формат обычного html-документа, только вместо тега `<body>` используется контейнер **frameset**, содержащий описание внутренних html-документов:

```
<frameset>
  <frame src="url_1">
  ...
  <frame src="url_n">
</frameset>
```

Общий контейнер **frameset** описывает все фреймы, на которые делится экран. Можно разделить экран на несколько вертикальных или несколько горизонтальных фреймов. Для этого можно использовать атрибуты:

- **cols = "список параметров"** – определяет горизонтальную структуру фреймов слева направо;
- **rows = "список параметров"** – определяет вертикальную структуру фреймов сверху вниз.

Это два основных атрибута, определяющие структуру фрейма. Список значений параметров **rows** и **cols** представляет собой разделенный запятыми список значений, которые могут задаваться в

пикселях или в процентах. Символ "*" означает выделение под данный фрейм всего оставшегося на экране пространства. Например, рассмотрите несколько вариантов разбиения web-страницы на фреймовые области:

1. запись `<frameset cols="200, *, 100">` определяет три горизонтальных фрейма: первый – 200 пикселей, последний 100 пикселей, второй (средний) – все оставшееся пространство;

2. запись `<frameset cols="20%, 100, 50,* ">` определяет четыре горизонтальных фрейма: первый – 20% экрана, второй – 100 пикселей, третий – 50 пикселей, четвертый – все оставшееся пространство;

3. запись `<frameset rows="50%, 50%">` определяет два вертикальных фрейма по 50% экрана в высоту каждый. Не трудно догадаться, что того же эффекта можно добиться, задав значение параметра rows равным 50%, *.

В отдельном теге `<frameset>` имеет смысл использовать только один из атрибутов – cols или rows. Это означает, что фреймовая структура будет состоять или только из столбцов, или только из строк. Чтобы создать строки внутри столбцов или столбцы внутри строк, вам потребуется вложенные контейнеры `<frameset>...</frameset>`.

Например, организуйте фреймовую структуру макета web-страницы, состоящая из двух горизонтальных строк, так чтобы вторая строка была бы поделена на две части в отношении 1:5 (рис. 3.1).

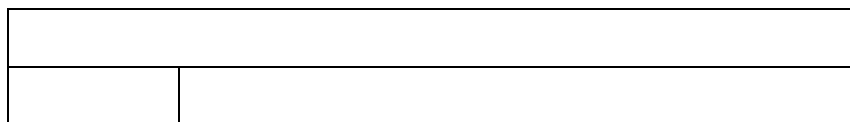


Рисунок 3.1 Макет web-страницы

В Листинге 3.1 дан фрагмент варианта html-кода, создающий фреймовую структуру, изображенную на рисунке 3.1.

Фрагмент варианта html-кода, создающий фреймовую структуру.

```
<frameset rows="100, *">
  <frame src="logo.htm">
  <frameset cols="150, *">
    <frame src="menu.htm">
    <frame src="content.htm">
  </frameset>
</frameset>
```

Вначале страница делится на два горизонтальных фрейма:

```
<frameset rows="100, *">.
```

В верхнюю часть загрузите страницу logo.htm:

```
<frame src="logo.htm">
```

Нижнюю часть разделите еще на два вертикальных фрейма:

```
<frameset cols="20%, *">.
```

Левая загружается страницей – menu.htm, а правая – content.htm:

```
<frame src="menu.htm">
```

```
<frame src="content.htm">
```

Задание 3.1.а. Фреймовая организация макета web-страницы.

1. Выбор и создание рабочей папки для заданий третьей главы.

В задании №1.1. главы 1 для практических заданий учебного пособия на диске D: была создана папка под названием Textbook. Выполненные задания из третьей главы будете размещать в отдельной папке с названием **Chapter_3** (от англ. Глава_3). Поэтому создайте в папке Textbook новую папку с именем **Chapter_3** (абсолютный адрес рабочей папки «D:\Textbook\Chapter_3»). Это папка в дальнейшем будет рабочей папкой для практических заданий главы 3.

2. Скопируйте шаблон html-документа temp.html, размещенный по адресу «D:\Textbook\Chapter_1» в новую рабочую папку **Chapter_3** и откройте его в редакторе Notepad++. В его

заголовочной части произведите следующие изменения:

- в теге <title> измените текст заголовка на "Фреймовая организация макета web-страниц";

- в соответствии с содержанием web-страницы в мета-теге "keywords" определите ключевые слова:

```
<meta name="keywords" content="фрейм, frame">
```

- в соответствии с содержанием web-страницы в мета-теге "description" добавьте описание страницы:

```
<meta name="description" content="Страница посвящена рассмотрению использования фреймов в макетировании web-страниц">
```

Сохраните документ.

3. Для начала рассмотрите приведенные выше варианты разбиения web-страницы на фреймовые области размещения графических изображений на web-страницу. Для этого выполните следующие действия:

- на основе html-документа temp.html создайте три html-документа с именами соответственно **frame_1.html**, **frame_2.html**, **frame_3.html**, которые будут использоваться в дальнейшем для загрузки в окна фрейма. Чтобы их визуально различать в тело каждого документа добавьте соответственно строки: "Окно фрейма 1", "Окно фрейма 2", "Окно фрейма 3". Открывая по очереди, проверьте правильность работы этих web-страниц в браузере Firefox (рис. 3.2);

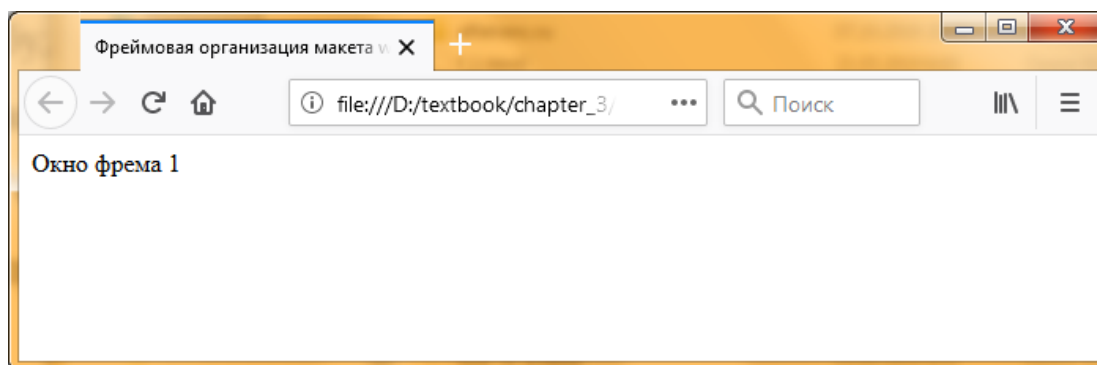


Рисунок 3.2 Результат просмотра страницы frame_1.html в браузере

– откройте html-документ temp.html, удалите из него тело документа, т.е. контейнер <body> и вместо него добавьте фрагмент html-кода, определяющий три горизонтальных фрейма: первый – 250 пикселей, последний 300 пикселей, второй (средний) – все оставшееся пространство:

```
<frameset cols="250, *, 300">  
  <frame src="frame_1.html">  
  <frame src="frame_2.html">  
  <frame src="frame_3.html">  
</frameset>
```

– сохраните шаблон html-документа под именем **task-31_a.html** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 3.3);

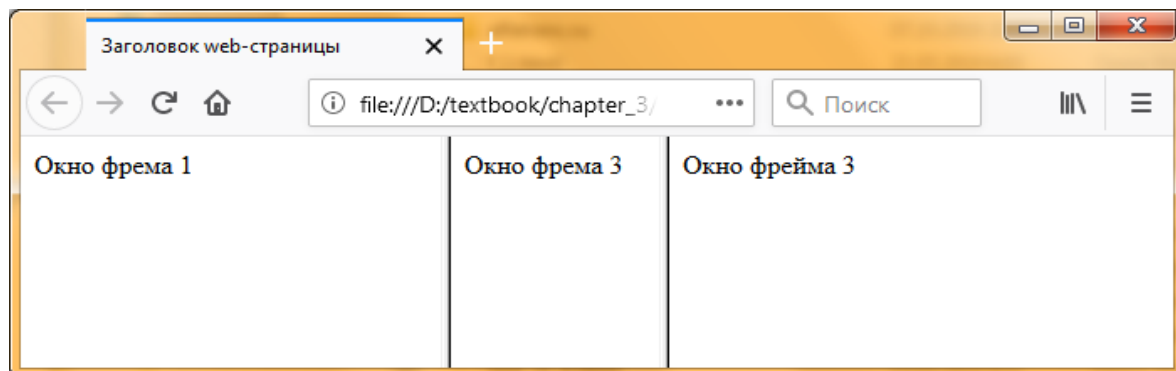


Рисунок 3.3 Результат просмотра страницы task-31_a.html в браузере

– самостоятельно измените html-документ task-31_a.html, чтобы определить два вертикальных фрейма по 50% экрана в высоту каждый. В первый загрузите html-документ frame_1.html, а во второй – frame_2.html. Сохраните html-документ под именем **task-31_b.html** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 3.4);

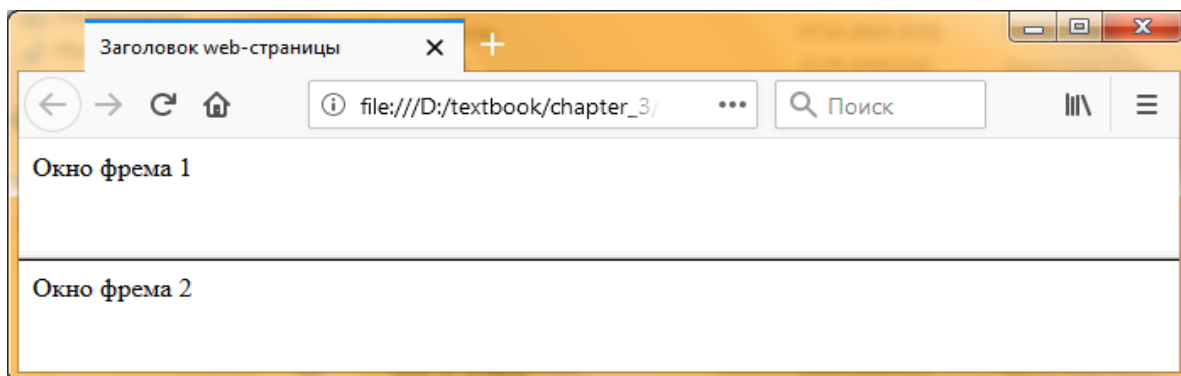


Рисунок 3.4 Результат просмотра страницы task-31_b.html в браузере

– самостоятельно измените html-документ task-31_b.html, чтобы определить фрагмент варианта html-кода (Листинг 3.1), создающий фреймовую структуру, изображенную на рисунке 3.1. В верхней части макета загрузите html-документ frame_1.html, в нижней левой части – frame_2.html, а в нижней правой части – frame_3.html. Сохраните html-документ под именем **task-31_c.html** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 3.5).

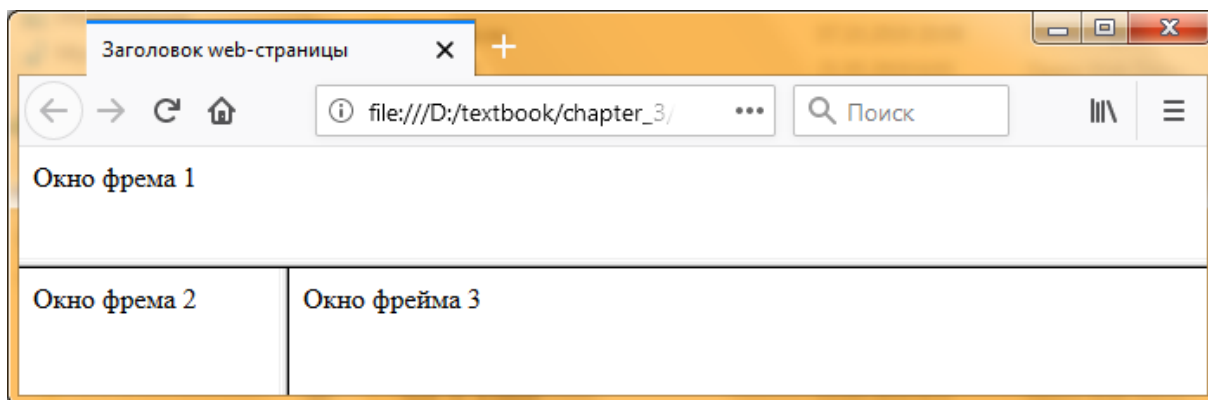


Рисунок 3.5 Результат просмотра страницы task-31_c.html в браузере

***Замечание.** Совместное использование данных параметров может привести к непредсказуемым результатам. Например, строка: `<frameset rows="50%,50%" cols "50%,50%">` может привести к ошибочной ситуации.*

Рассмотрите другие атрибуты тега frameset:

– **frameborder** – определяет наличие рамок, т.е. границ фреймов, применяется как в теге <frameset>, так и в теге <frame>. Может принимать следующие значения: 1 или yes – граница будет отображена; 0 или no – граница будет скрыта (по умолчанию).

Пример 1.

```
<frameset cols = "200, *" frameborder = "1">
```

Если параметр определен в теге <frameset>, то его действие распространяется на все фреймы этой группы. Для отдельного фрейма значение может быть переопределено.

По умолчанию в большинстве браузеров параметр принимает значение Yes и граница отображается серым цветом. Ее вид по своему усмотрению можно настроить с помощью атрибутов border и bordercolor:

– **border** – определяет толщину рамок между фреймами в пикселях. Данный параметр применяется только в теге <frameset>. Для того чтобы границы не было вообще, нужно установить значение этого параметра равным 0.

Пример 2.

```
<frameset cols = "200, *" border = "0">
```

Параметры border и frameborder работают независимо друг от друга, т.е. если в качестве значения frameborder задано no, а для border задано значение, отличное от нуля, то рамка между фреймами прорисовываться не будет, но место под нее, определенное значением параметра border, все равно будет отведено.

– **bordercolor** – определяет цвет рамок, применяется как в теге <frameset>, так и в теге <frame>. По умолчанию (если используется стандартная цветовая схема системы Windows) границы фреймов имеют тусклый, серый оттенок. При желании можно выбрать любой другой цвет. При определении цвета можно выбрать как его название, так и числовой эквивалент в системе RGB. Например, blue или 0000FF.

Пример 3.

```
<cols= "200,*" frameborder="1" border="4" bordercolor="blue">
```

Задание 3.1.b. Атрибуты `frameborder`, `border` и `bordercolor`.

1. Откройте html-документ `task-31_a.html`. Для фреймов установите следующие параметры: наличие границ, толщина границы рамок сделайте равным 8 пикселям и красного (`red`) цвета. Для этого добавьте в основной тег `<frame>` соответствующие параметры:

```
frameset cols="250, *, 300" frameborder="yes", border="8" bordercolor="red">
```

Сохраните html-документ под именем **task-31_a1.html** в рабочей папке. Просмотрите документ в браузере Firefox (рис. 3.6).

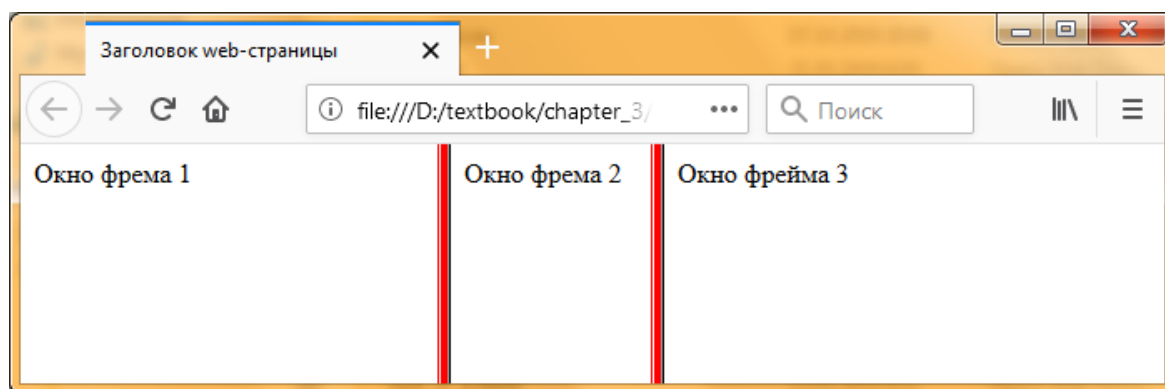


Рисунок 3.6 Результат просмотра страницы `task-31_a1.html` в браузере

2. Самостоятельно измените html-документ `task-31_b.html` так, чтобы фреймовая структура имела толщину рамок в 12 пикселей, цвета – `navy` (темно-синий). Сохраните html-документ под именем **task-31_b1.html** в рабочей папке. Просмотрите документ в браузере.

3. Самостоятельно подберите толщину и цвет рамок фреймовой структуры html-документ `task-31_c.html` и сохраните этот документ под именем **task-31_c1.html** в рабочей папке. Просмотрите документ в браузере.

Внутри набора фреймов отдельное окно определяется с помощью тега `<frame>`.

Тег `<frame>` определяет внешний вид и поведение фрейма. Этот

тег не имеет закрывающего тега. Вся суть тега <frame> в его атрибутах. Их шесть: **name**, **marginwidth**, **marginheight**, **scrolling**, **noresize** и **src**.

Атрибут **marginwidth** задает горизонтальный отступ между содержимым кадра и его границами. Наименьшее значение этого атрибута равно 1, нельзя указать 0, по умолчанию атрибут равен 6.

Атрибут **marginheight** действует так же, как и **marginwidth**, но задает поля в верхней и нижней частях фрейма.

Атрибут **scrolling** используется для управления полосами прокрутки, которые появляются, если содержимое фрейма не помещается в отведенное окно. Возможные варианты:

- `scrolling = "yes"` – во фрейме всегда будут полосы прокрутки, даже если это не нужно;

- `scrolling = "no"` – полосы прокрутки не будет, даже когда это необходимо. Если документ слишком большой и задан режим без прокрутки, документ просто будет обрезан;

- `scrolling = "auto"` – предоставляет браузеру самому решать требуются полосы прокрутки или нет. Если атрибут `scrolling` отсутствует, результат будет таким же, как при использовании `scrolling=auto`.

Атрибут **noresize** устанавливает запрет пользователю изменять размер фреймов. По умолчанию пользователь может изменять размер фреймов с помощью мыши.

Задание 3.1.с. Атрибуты `marginwidth`, `marginheight` и `scrolling`.

1. Откройте html-документ `task-31_a.html`. Для фрейма 1 установите горизонтальный и вертикальный отступы между содержимым кадра и его границами равным отличным от параметров по умолчанию. Для этого добавьте в тег <frame> первого окна соответствующие параметры:

```
<frame src="frame_1.html" marginwidth="12" marginheight="12">
```

2. Сохраните html-документ под именем **task-31_a2.html** в рабочей папке. Просмотрите документ в браузере.

Практическая часть

В практической части рассмотрите процесс создания макета основной страницы сайта с использованием фреймов, а также требований к оформлению элементов основной страницы сайта.

На рисунке 3.7 изображен макет начальной страницы сайта транспортной компании «АЛЬФА-ТРАНС». Реализуйте ее с использованием фреймов.

Предварительно подготовьте содержимое фреймов, т.е. страницу заголовка с логотипом, страницу главного меню и страницу сведений об авторских правах.

Страница заголовка с логотипом. Заголовок сайта – это достаточно важный элемент сайта. Можно сказать, что это лицо сайта. Поэтому его часто помещают в горизонтальный фрейм, расположенный сверху, вдоль верхней стороны окна Web-обозревателя. Таким образом, он хорошо виден, но и не бросается в глаза, если специально на него не смотреть. Иногда его разделяют на два фрейма, в одном из которых находится логотип организации, а в другом – текст названия.

Логотип (файл logo.html)	Заголовок сайта (файл header.html)
Меню сайта (файл menu.html)	Полезное содержимое сайта (файл content.html)
Сведения о правах разработчика сайта (файл copyright.html)	

Рисунок 3.7 Макет главной страницы транспортной компании

Html-страница с логотипом – это документ, содержащий рисунок логотипа транспортной компании «АЛЬФА-ТРАНС». В качестве логотипа будете использовать файл рисунка **logo.png**.

Для начала в папке «D:\Textbook\Chapter_3» создайте папку, в

которой будете сохранять разрабатываемые web-страницы. В дальнейшем с русским языком в названии папок и файлов могут возникнуть проблемы, поэтому имя папки нужно писать в англоязычной раскладке клавиатуры. Например, в качестве папки для сайта можно определить папку с именем «AlfaTrans.ru» – по названию транспортной компании.

1. В папке «D:\Textbook\Chapter_3» создайте папку с именем **AlfaTrans.ru**. Скопируете в эту папку шаблон сайта html-документа temp.html, размещенный в папке «D:\Textbook\Chapter_1\AlfaTrans.ru» и папку Images вместе с файлами.

2. Откройте шаблон сайта html-документа temp.html и произведите в нем следующие изменения:

- в тело документа добавьте следующий фрагмент html-кода:

- сохраните шаблон html-документа под именем **logo.html** в папке AlfaTrans.ru;
- проверьте правильность отображения страницы в браузере (рис.3.8).

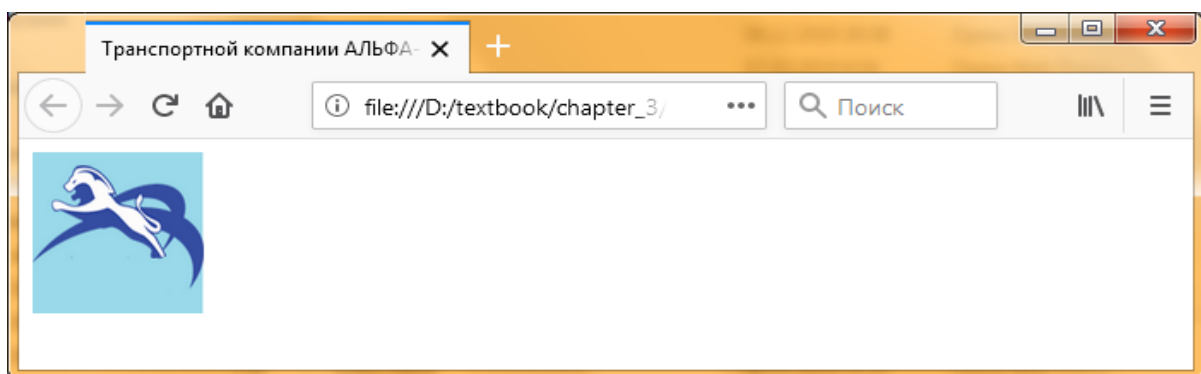


Рисунок 3.8 Результат просмотра страницы logo.html

Замечание. Вместо отдельной html-страницы с логотипом можно при определении фрейма сразу же загрузить рисунок – логотип компании. Для этого фрейм нужно описать следующим образом:

```
<frame scrolling="no" src="Images/logo.png">
```

Заголовок сайта.

3. Откройте шаблон сайта html-документа temp.html, размещенный в папке «D:\Textbook\AlfaTrans.ru» и произведите в нем следующие изменения:

– в тело документа добавьте фрагмент html-кода:

```
<font face="Comic Sans MS, Arial" color="aqua" size="5">
```

```
<p align="center">АЛЬФА-ТРАНС</p>
```

```
</font>
```

```
<font face="Comic Sans MS, Arial" color="red" size="3">
```

```
<p align="center"> Сайт транспортной компании</p>
```

```
</font>
```

Сохраните шаблон html-документа под именем **header.html** в папке AlfaTrans.ru.

Проверьте правильность отображения страницы в браузере (рис. 3.9).

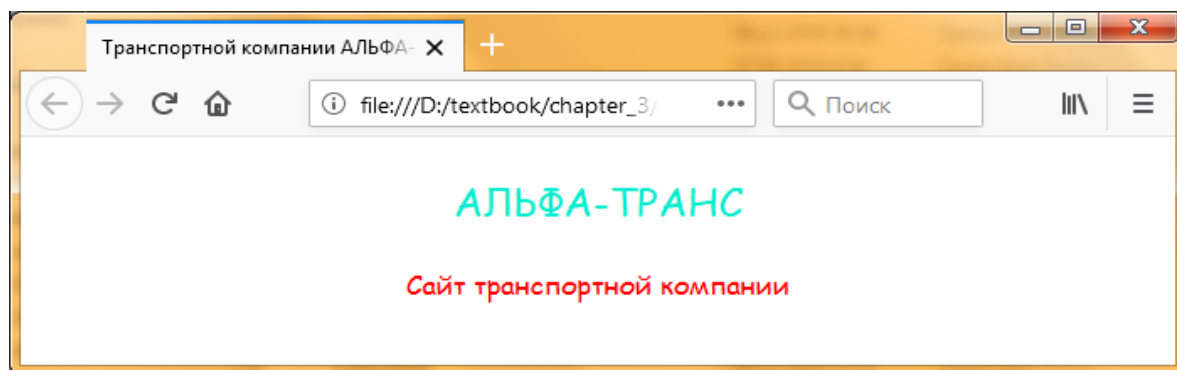


Рисунок 3.9 Результат просмотра страницы header.html

Страница меню. Страница меню – это html-страница, построенная как оглавление. Этот фрейм узкий и высокий, так что страница, которая будет в него загружаться, должна быть соответствующим образом спроектирована. Ниже приводится вариант оформления html-страницы меню с использованием маркированных списков:

```
<h3 align="center">меню</h3>
```

```

<hr>
<ul><font color="#400040" size="4">
  <li><a href="about.html" target="content">главная</a>
  <li><a href="news.html" target="content">новости</a>
  <li><a href="autopark.html" target="content">наш
автопарк</a>
  <li><a href="price.html" target="content"> прайс-лист</a>
</ul>
<a href="mailto:turfirma@mail.ru">  <br> на наш ящик </a><br>
</font>

```

4. Откройте шаблон сайта html-документа temp.html, размещенный в папке «D:\Textbook\Chapter_3\AlfaTrans.ru» и в тело документа добавьте вышеприведенный фрагмент html-кода.

Сохраните шаблон html-документа под именем **menu.html** в папке AlfaTrans.ru. Проверьте правильность отображения страницы в браузере (рис.3.10).

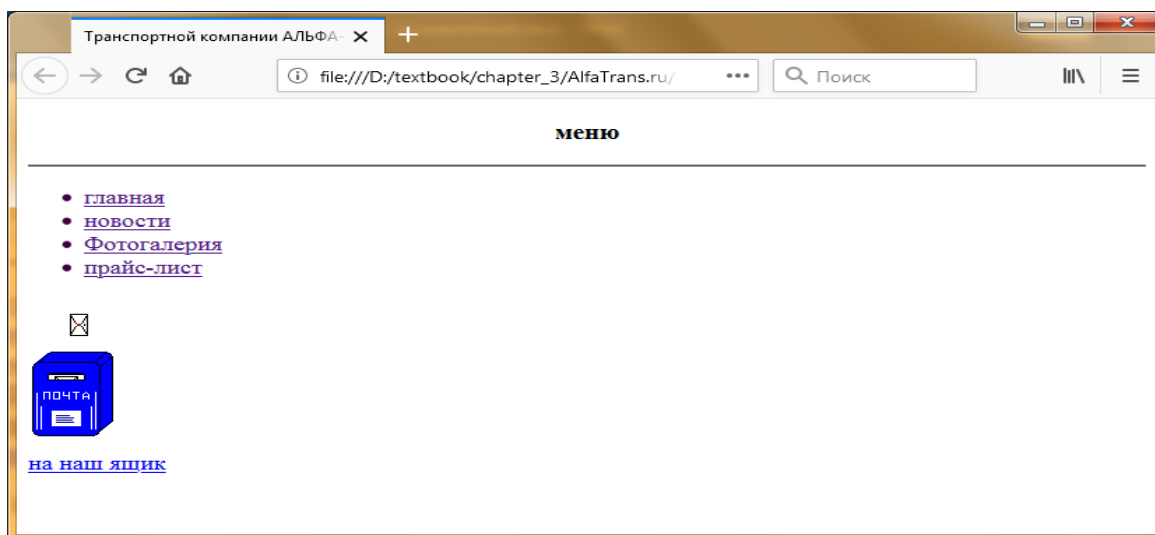


Рисунок 3.10 Результат просмотра страницы menu.html

Страница сведений об авторских правах. Что касается сведений об авторских правах, то потребность в том, чтобы они

постоянно присутствовали на экране, не так велика. Пожалуй, их помещают в отдельный фрейм только те, кто озабочен своими правами на сайт и опубликованные на нем материалами. В этом случае лучше расположить их ненавязчиво – в нижнем горизонтальном фрейме.

В большинстве случаев сведения об авторских правах располагают либо в тексте начальной страницы, что отображается по умолчанию в основном фрейме, либо помещают в один фрейм с набором гиперссылок в самом его низу. Также можно поместить их в заголовок сайта, если он расположен в отдельном фрейме.

5. Откройте шаблон сайта html-документ temp.html, размещенный в папке «D:\Textbook\Chapter_3\AlfaTrans.ru» и в тело документа добавьте нижеприведенный фрагмент html-кода, содержащий сведения об авторских правах:

```
<font color="#400040" size="4">
```

```
Политика конфиденциальности. Все права защищены © 2018-2019 <br>
```

```
Транспортная компания «РУСТРАНС» ОГРН: 1155029012610  
ИНН: 5029206268<br>
```

```
8 (926) 5-98-67/8 (495) 5-06-96 г. Майкоп, улица Учебная, д. 12
```

```
</font>
```

Сохраните шаблон html-документа под именем **copyright.html** в папке AlfaTrans.ru. Проверьте правильность отображения страницы в браузере (рис.3.11).

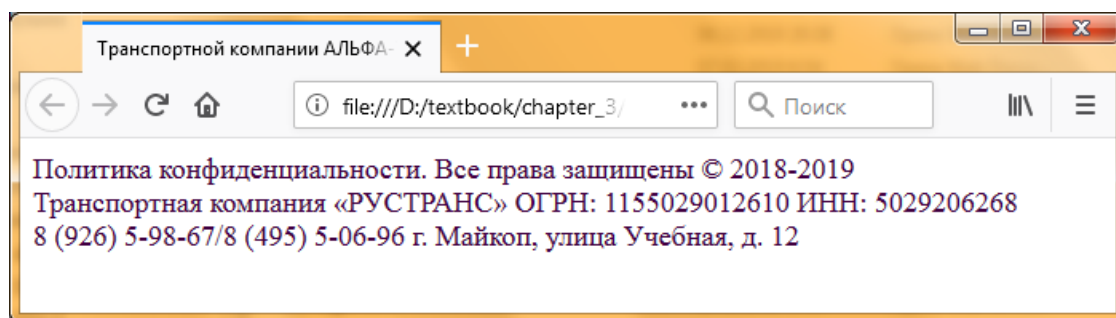


Рисунок 3.11 Результат просмотра страницы copyright.html

Теперь создайте страницу, организующую фреймовую структуру.

Из рисунка 3.7 видно, что главная страница должна быть разбита на три горизонтальных фрейма. При этом первые два дополнительно разбиты на два фрейма. Поэтому создайте три горизонтальных фрейма и внутри первых двух еще по два вертикальных фрейма:

Листинг 3.2

Фрагмент главной страницы.

```
<frameset rows="100, *, 100">
  <frameset cols="120, *">
    <frame src="logo.html" scrolling="no" noresize>
    <frame src="header.html" scrolling="no">
  </frameset>
  <frameset cols="200, *">
    <frame src="menu.html" noresize>
    <frame src="content.html" name="content">
  </frameset>
  <frame src="copyright.html" scrolling="no">
</frameset>
</noframes>
```

Вы видите, эту страницу браузером, не поддерживающим фреймы.

```
</noframes>
```

Обратите внимание на то, что для фрейма, куда будете загружать полезное содержимое сайта, вы определяете параметр `name="content"`. В дальнейшем с помощью ссылок из меню вы туда будете загружать соответствующие страницы.

6. Откройте шаблон сайта html-документа `temp.html`, размещенный в папке «D:\Textbook\Chapter_3\AlfaTrans.ru» и замените тело документа фрагментом Листинга 3.2.

Сохраните шаблон html-документа под именем **index.html** в папке AlfaTrans.ru.

Чтобы ссылки меню правильно работали, скопируйте из папки «D:\Textbook\Chapter_2\AlfaTrans.ru» файлы **about.html**, **news.html**, **autopark.html**, **price.html** в папку «D:\Textbook\Chapter_3\AlfaTrans.ru».

Проверьте правильность отображения страницы в браузере (рис. 3.12).

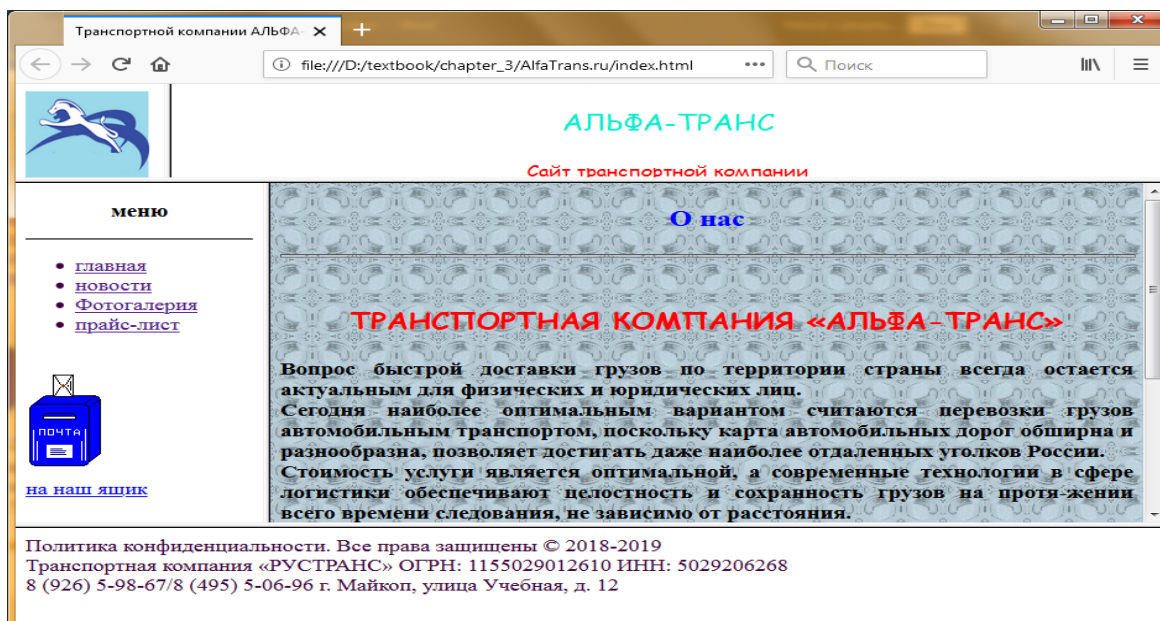


Рисунок 3.12 Результат просмотра страницы index.html

Тестовые задания для самоконтроля

Уровень А

1. Фреймы применяются для...
 - А. отображения в браузере сразу нескольких страниц
 - В. представления информации в табличном виде
 - С. облегчения навигации по сайту
 - Д. повышения наглядности
2. Какие атрибуты тега `<frameset>` позволяют создавать вертикальные фреймы?
 - А. name
 - В. frameborder
 - С. cols

D. rows

3. Какие атрибуты тега `<frameset>` позволяют создавать горизонтальные фреймы?

A). name

B). frameborder

C). cols

D). rows

4. Значениями обязательного атрибута `src` тега `<frame>` могут быть ...

A. ширина верхнего и нижнего полей фрейма

B. полное имя документа, отображаемого в области фрейма

C. ширина верхнего поля фрейма

D. величина расстояния между фреймами

5. Основным отличительным признаком структуры html-документа для фреймов по сравнению с обычным html-документом является...

A. отсутствие тела документа

B. наличие заголовка документа

C. наличие тега

D. все вышеперечисленные признаки

6. Укажите атрибут тега `<frame>`, позволяющий управлять полосами прокрутки во фрейме:

A. scrolling

B. src

C. noresize

D. name

E. нет правильного ответа

7. Запись `<frameset cols="20, *, 100">` определяет ...

A. три горизонтальных фрейма: первый – 20 пикселей, последний – 100 пикселей, второй (средний) – все оставшееся пространство

В. три вертикальных фрейма: первый – 20 пикселей, последний – 100 пикселей, второй (средний) – все оставшееся пространство

С. три горизонтальных фрейма: первый – 20% экрана, последний – 100 пикселей, второй (средний) – все оставшееся пространство

Д. три вертикальных фрейма: первый – 20% экрана, последний – 100 пикселей, второй (средний) – все оставшееся пространство

8. Запись `<frameset rows="50, *, 10%">` определяет ...

А. три горизонтальных фрейма: первый – 50 пикселей, последний 0,1 часть экрана, второй (средний) – все оставшееся пространство

В. три вертикальных фрейма: первый – 50 пикселей, последний – 0,1 часть экрана, второй (средний) – все оставшееся пространство

С. три горизонтальных фрейма: первый – 50% экрана, последний – 10 пикселей, второй (средний) – все оставшееся пространство

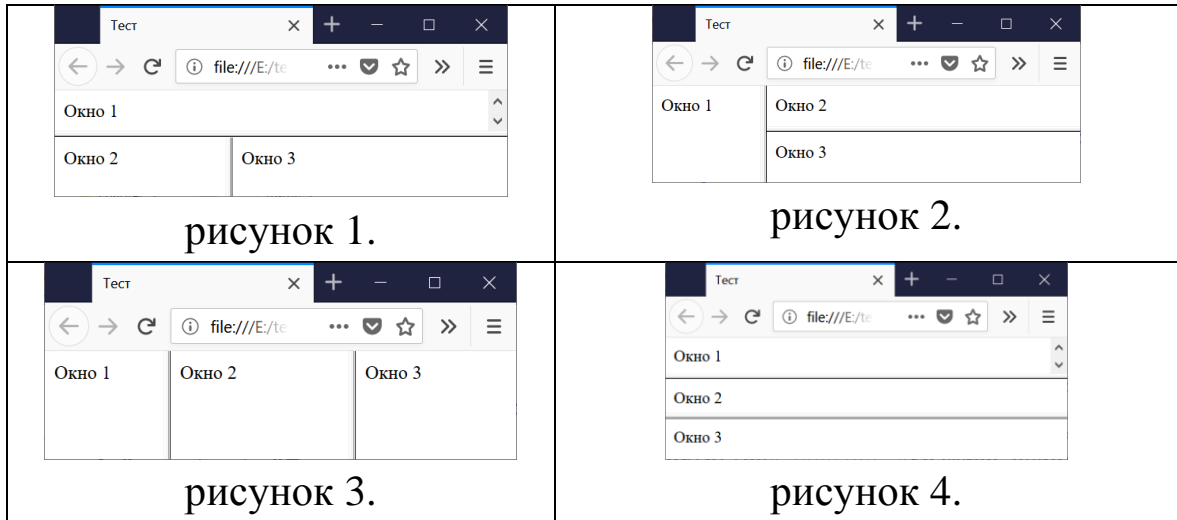
Д. три вертикальных фрейма: первый – 50% экрана, последний – 10% экрана, второй (средний) – все оставшееся пространство

9. Дан фрагмент html-кода, создающий фреймовую структуру:

```
<frameset rows="100, *">  
  <frame src="f1.html">  
<frameset cols="150, *">  
  <frame src="f2.html"> <frame src="f3.html">  
</frameset>  
</frameset>
```

Какому рисунку соответствует результат просмотра данного фрагмента в браузере?

- А. рисунок 1
- В. рисунок 2
- С. рисунок 3
- Д. рисунок 4



10. Значениями обязательного атрибута src тега <frame> могут быть ...

- А. ширина верхнего и нижнего полей фрейма
- В. полное имя документа, отображаемого в области фрейма
- С. ширина верхнего поля фрейма
- Д. полное имя графического файла, отображаемого в области фрейма
- Е. величина расстояния между фреймами

Уровень В

1. Для каждого атрибута тега frameset, указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- А. frameborder 1. определяет толщину рамок между фреймами
- В. border в пикселях
- С. bordercolor 2. определяет наличие границ фреймов
3. определяет цвет границ фреймов

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

А	В	С

2. Для каждого атрибута тега frameset, указанного в левой колонке, установите соответствующее ему описание, приведенное в правой колонке:

- A. `marginwidth` 1. используется для управления полосами
- B. `scrolling` прокрутки, которые появляются, если
- C. `noresize` содержимое фрейма не помещается в
отведенное окно
- 2. задает горизонтальный отступ между
содержимым кадра и его границами
- 3. устанавливает запрет пользователю
изменять размер фреймов

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

A	B	C

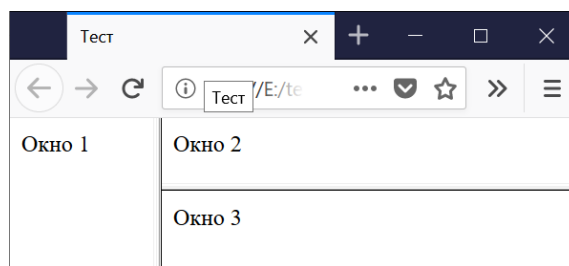
3. Дан фрагмент html-кода, создающий фреймовую структуру.

```

<frameset cols="100, *">
  <frame src="f_1.html">
<frameset rows="150, *">
  <frame src="f_2.html"> <frame src="f_3.html">
</frameset>
</frameset>

```

На рисунке показан результат просмотра данного фрагмента в браузере:



Для каждого окна, указанного в левой колонке, установите соответствующий ему файл html-документа, который в нем будет загружаться, приведенное в правой колонке.

- A. окно 1 1. f_1.html
- B. окно 2 2. f_2.html
- C. окно 3 3. f_3.html

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

A	B	C

4. Дан фрагмент варианта html-кода, создающий фреймовую структуру:

```
<frameset rows="100, *">
  <frame src="f_1.html">
  <frameset cols="150, *">
    <frame src="f_2.html">
    <frame src="f_3.html">
  </frameset>
</frameset>
```

Каждой записи html-кода, указанного в левой колонке, установите соответствующее ей описание, приведенное в правой колонке:

Фрагмент html-кода	Описание, выполняемое фрагментом
A. <frameset rows="100, *">	1. делит страницу на два горизонтальных фрейма
B. <frameset cols="150, *">	2. делит нижнюю часть на два вертикальных фрейма
C. <frame src="f_1.html">	3. загружает в верхнюю часть html-страницу
D. <frame src="f_3.html">	4. загружает в нижнюю левую часть html-страницу

В таблице под каждой буквой укажите соответствующий номер.

Ответ:

A	B	C	D

5. Для каждого описания параметра тега frameset, указанного в левой колонке, установите соответствующий ему html-код, приведенное в правой колонке:

A. предоставляет браузеру 1. scrolling = "auto"

- | | |
|---------------------------|------------------------|
| самому решать, требуются | 2. scrolling = "no" |
| полосы прокрутки или нет. | 3. scrolling = "yes" |
| В. определяет отсутствие | 4. frameborder = "1" |
| границ у фрейма (ов). | 5. frameborder = "yes" |
| | 6. frameborder = "0" |
| | 7. frameborder = "no" |

К каждой позиции первого столбца подберите соответствующую позицию из второго столбца и запишите в таблицу выбранные цифры. Цифры в ответе могут повторяться.

Ответ:

А	В

Уровень С

1. Для тега <frameset> задайте атрибут, определяющий обязательное наличие полосы прокрутки для окна фрейма.

Ответ: ...

2. Для тега <frameset> задайте атрибут, запрещающий наличие полосы прокрутки для окна фрейма.

Ответ: ...

3. Для тега <frameset> задайте атрибут, предоставляющий браузеру самому решать, требуются полосы прокрутки или нет.

Ответ: ...

4. Для тега <frameset> задайте атрибут, запрещающий пользователю изменять размер окна фрейма.

Ответ: ...

5. Для тега <frameset> задайте атрибуты, задающие горизонтальный отступ между содержимым кадра и его границами равным 12 пикселей и поля в верхней и нижней частях фрейма равной 18 пикселей.

Ответ: ...

Ответы к тестовым заданиям

Уровень А

Глава 1		Глава 2		Глава 3	
1.	B	1.	A, B	1.	A
2.	D	2.	C, D, E	2.	D
3.	B	3.	A, C, E	3.	C
4.	A, D	4.	B, D, F	4.	B
5.	A, C	5.	C, D	5.	A
6.	C	6.	A	6.	A
7.	A			7.	A
8.	B			8.	B
9.	B			9.	A
10.	A			10.	B

Уровень В

Глава 1						
1.	A	B	C			
	3	1	2			
2.	A	B	C	D		
	3	1	2	4		
3.	A	B	C	D	E	F
	2	1	6	5	4	3
4.	A	B	C			
	3	1	2			
5.	A	B	C			
	3	1	2			

Глава 2					
1.	A	B	C		
	2	3	1		
2.	A	B	C	D	E
	3	1	2	5	4
Глава 3					
1.	A	B	C		
	2	1	3		
2.	A	B	C		
	2	1	3		
3.	A	B	C		
	1	2	3		
4.	A	B	C	D	
	1	2	3	4	
5.	A	B			
	1	6,7			

Уровень С

Глава 1	
1.	<title>Основы HTML</title>
2.	align="center"
3.	<p align="justify">Выравнивание текста абзаца</p>
4.	<i>Транспортная </i><i> компания </i><u>АЛЬФА-ТРАНС </u>
5.	
6.	<body bgcolor="aqua">
7.	<body background="Img\fon.jpg">

8.	<hr size="3" width="100%">
9.	<marquee>Я бегущая строка</marquee>
10.	<marquee direction="right">Я бегущая строка</marquee>
Глава 2	
1.	http://mysite.ru/news/image
2.	http://mysite.ru/image
3.	http://mysite.ru/users/
4.	
5.	
6.	
Глава 3	
1.	<frameset scrolling = "yes">
2.	<frameset scrolling = "no">
3.	<frameset scrolling = "auto">
4.	<frameset noresize>
5.	<frameset marginwidth = "12" marginheight = "18">

Заключение

Учебное пособие имеет техническую направленность и предусматривает знакомство обучающихся со спецификой технологии разработки сайтов - гипертекстовым языком разметки HTML.

Отличительной чертой от других пособий сходной тематики является знакомство обучающихся с технологиями SEO-продвижения и поисковой оптимизации Интернет-ресурсов, которые в настоящее время являются одной из наиболее динамично развивающихся отраслей ИКТ-технологий и определяют последующее развитие любого веб-ресурса.

Педагогическая целесообразность пособия заключается в том, что оно позволяет реализовать связь с другими дисциплинами, выраженную в более эффективном и успешном освоении обучающимися общеобразовательных программ благодаря развитию у обучающихся информационной культуры и широкого круга компетенций применения ИКТ в различных сферах деятельности.

Учебное пособие имеет модульную структуру, и обучающиеся могут использовать различные схемы для изучения материала. Различные разделы могут представлять интерес для преподавателей при подготовке традиционных занятий информатики.

Во время изучения пособия «Информатика: основы сайтостроения» обучающиеся познакомятся с материалом, который поможет им заниматься сайтостроением профессионально, а также они смогут создавать небольшие, но собственные сайты и будут использовать приобретенные навыки в дальнейшей работе.

Глоссарий

`<!--...-->` - используется для добавления комментариев.

`<!DOCTYPE>` - объявляет тип документа и предоставляет основную информацию для браузера - его язык и версия.

`<a>` - создает гипертекстовые ссылки.

`<area>` - представляет собой гиперссылку с текстом, соответствующей определенной области на карте изображения или активную область внутри карты изображения, всегда вложен внутри тега `<map>`.

`` - задает полужирное начертание отрывка текста, не придавая акцент или важность выделенному.

`<base>` - задает базовый адрес (URL), относительно которого вычисляются все относительные адреса.

`<blockquote>` - выделяет текст как цитату, применяется для описания больших цитат.

`<body>` - представляет тело документа (содержимое, не относящееся к метаданным документа).

`
` - перенос текста на новую строку.

`<caption>` - добавляет подпись к таблице. Вставляется сразу после тега `<table>`.

`<code>` - представляет фрагмент программного кода, отображается шрифтом семейства monospace.

`<col>` - выбирает для форматирования один или несколько столбцов таблицы, не содержащих информацию одного типа.

`<colgroup>` - создает структурную группу столбцов, выделяющую множество логически однородных ячеек.

`<dd>` - используется для описания термина из тега `<dt>`.

`` - помечает текст как удаленный, перечеркивая его.

`<dl>` - тег-контейнер, внутри которого находятся термин и его описание.

`<dt>` - используется для задания термина.

**** - выделяет важные фрагменты текста, отображая их курсивом.

<footer> - определяет завершающую область (нижний колонтитул) документа или раздела.

<h1-h6> - создают заголовки шести уровней для связанных с ними разделов.

<head> - элемент-контейнер для метаданных html-документа, таких как **<title>**, **<meta>**, **<script>**, **<link>**, **<style>**.

<header> - секция для вводной информации сайта или группы навигационных ссылок. Может содержать один или несколько заголовков, логотип, информацию об авторе.

<hr> - горизонтальная линия для тематического разделения параграфов.

<html> - корневой элемент html-документа. Сообщает браузеру, что это html-документ. Является контейнером для всех остальных html-элементов.

<i> - выделяет отрывок текста курсивом, не придавая ему дополнительный акцент.

**** - встраивает изображения в html-документ с помощью атрибута **src**, значением которого является адрес встраиваемого изображения.

<kbd> - выделяет текст, который должен быть введен пользователем с клавиатуры, шрифтом семейства monospace.

**** - элемент маркированного или нумерованного списка.

<link> - определяет отношения между документом и внешним ресурсом. Также используется для подключения внешних таблиц стилей.

<map> - создает активные области на карте-изображении. Является контейнером для элементов **<area>**.

<meta> - используется для хранения дополнительной информации о странице. Эту информацию используют браузеры для обработки страницы, а поисковые системы - для ее индексации. В блоке **<head>** может быть несколько тегов **<meta>**, так как в

зависимости от используемых атрибутов они несут разную информацию.

**** - упорядоченный нумерованный список.

<p> - параграфы в тексте.

<pre> - выводит текст без форматирования, с сохранением пробелов и переносов текста. Может быть использован для отображения компьютерного кода, сообщения электронной почты и т.д.

<q> - определяет краткую цитату.

<small> - отображает текст шрифтом меньшего размера.

**** - контейнер для строчных элементов. Можно использовать для форматирования отрывков текста, например, выделения цветом отдельных слов.

**** - расставляет акценты в тексте, выделяя полужирным.

<sub> - задает подстрочное написание символов, например, индекса элемента в химических формулах.

<sup> - задает надстрочное написание символов.

<table> - тег для создания таблицы.

<td> - создает ячейку таблицы.

<th> - создает заголовок ячейки таблицы.

<thead> - определяет заголовок таблицы.

<title> - заголовок html-документа, отображаемый в верхней части строки заголовка браузера.

<tr> - создает строку таблицы.

<u> - выделяет отрывок текста подчеркиванием, без дополнительного акцента.

**** - создает маркированный список.

HTML - HyperText Markup Language (язык разметки гипертекста), основной язык создания сайтов.

HTML-атрибут - свойство тега, назначающее ему какую-либо характеристику, влияющие на внешность: цвет, размер, фон, выравнивание и т.д.

HTML-документ - это html-файл, открытый с помощью

текстового редактора, например Блокнота. Html-документ содержит html-код.

HTML-код - это весь код или часть кода html-документа.

HTML-страница - это html-файл, открытый с помощью браузера. Другие названия: страница, веб-страница, web-страница.

HTML-тег - основная часть html-языка. У каждого тега есть свое имя на английском языке, до и после которого ставятся угловые скобки. Теги бывают парными и одиночными.

HTML-файл – это файл, имеющий расширение .html или .htm

HTML-элемент – это какой-либо объект страницы, созданный с помощью тегов.

URL (Uniform Resource Locator) или универсальный указатель ресурса - адрес некоторого объекта в Интернете.

Web-страница – документ (файл), подготовленный в формате гипертекста и размещенный в World Wide Web.

World Wide Web, WWW или просто Web – всемирная паутина, распределенная система доступа к гипертекстовым документам, существующая в Интернете. HTML является основным языком для создания документов в WWW.

Абсолютный адрес – полный адрес документа с указанием протокола Интернет (http, ftp и др.).

Атрибут (attribute) – параметр или свойство элемента.

Базовый URL – часть адреса, которая является общей для всех ссылок текущей Web-страницы.

Браузер (browser) – программа для просмотра Web-страниц.

Гиперссылка – фрагмент текста, который является указателем на другой файл или объект. Гиперссылки необходимы для того, чтобы обеспечить возможность перехода от одного документа к другому.

Значение HTML-атрибута - в качестве значений атрибута выступают: числовые величины, ключевые слова, адреса.

Относительный адрес - путь к файлу относительно текущего документа.

Протокол - набор соглашений для передачи данных по сети. В Интернете наиболее распространен HTTP (Hyper Text Transfer Protocol, протокол передачи гипертекста).

Сайт (site) – набор Web-страниц, принадлежащих одному владельцу.

Тег (tag) – начальный или конечный маркеры элемента. Теги определяют границы действия элементов и отделяют элементы друг от друга. В тексте Web-страницы теги заключаются в угловые скобки, а конечный тег всегда снабжается косой чертой.

Фрейм (frame) – технология HTML, позволяющая разбить окно браузера на отдельные области, расположенные рядом друг с другом.

Список использованной литературы

1. Бен Фрейн, HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств. 2-е изд. Питер, 2017, 272 с.
2. Введение в HTML5 [Электронный ресурс]: учебное пособие / К. Миллз [и др.]. - Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. - 133 с. - ЭБС «IPRbooks» - Режим доступа: <http://www.iprbookshop.ru/89424.html>
3. Дронов, В.А. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов. [Электронный ресурс]: практическое пособие / В.А. Дронов. – БХВ-Петербург, 2011. - 414 с. - ЭБС «Znaniium.com» - Режим доступа: <https://new.znaniium.com/catalog/document?id=159076>
4. Кравченко, С.В., Основы сайтостроения: Учеб. пособие / С.В. Кравченко. - Томск, 2012, 300 с.
5. Кириченко А., Хрусталеv А. HTML5 + CSS3. Основы современного WEB-дизайна. Наука и Техника, 2018, 354 с.
6. Лабораторный практикум. Интернет - продвижение гостиничного продукта, Ч. 1 [Электронный ресурс]: учебно-методическое пособие / [сост.: С.К. Меретукова, Ш.Т. Меретуков]. - Майкоп: Магарин О.Г., 2013. - 28 с. - Режим доступа: <http://lib.mkgtu.ru:8002/libdata.php?id=2100000789>
7. Лабораторный практикум. Интернет - продвижение гостиничного продукта, Ч. 2 [Электронный ресурс]: учебно-методическое пособие / [сост.: Ш.Т. Меретуков, С.К. Меретукова]. - Майкоп: Магарин О.Г., 2013. - 36 с. - Режим доступа: <http://lib.mkgtu.ru:8002/libdata.php?id=2100000788>
8. Лабораторный практикум. Интернет - продвижение гостиничного продукта, Ч. 3 [Электронный ресурс]: учебно-методическое пособие / [сост.: Ш.Т. Меретуков, С.К. Меретукова]. -

Майкоп:Магарин О.Г., 2013. - 18 с. – Режим доступа:
<http://lib.mkgtu.ru:8002/libdata.php?id=2100001968>

9. Малышева, Е.Н. Web-технологии [Электронный ресурс]: учебное пособие / Е.Н. Малышева. - Кемерово: Кемеров. гос. ин-т культуры, 2018. - 116с. - ЭБС «Znanium.com» - Режим доступа:
<https://new.znanium.com/catalog/product/1041185>

10. Основы работы с HTML [Электронный ресурс]. - Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 208 с. - ЭБС «IPRbooks» - Режим доступа:
<http://www.iprbookshop.ru/73698>

11. Титов, В.А. Разработка WEB-сайта средствами языка HTML [Электронный ресурс]: учебное пособие / В. А. Титов, Г. И. Пещеров. - Москва: Институт мировых цивилизаций, 2018. - 184 с. - ЭБС «IPRbooks» - Режим доступа:
<http://www.iprbookshop.ru/80643.html>

12. Шапошников, И.В. Самоучитель HTML 4. [Электронный ресурс]: практическое пособие / И.В. Шапошников, - БХВ-Петербург, 2014. - 288 с. - ЭБС «Znanium.com» - Режим доступа:
<https://new.znanium.com/catalog/document?id=302060>

13. <https://www.wikipedia.org> – общедоступная многоязычная универсальная интернет-энциклопедия со свободным контентом, реализованная на принципах вики.

14. <http://ts-maykop.ru/> – сайт ООО Транс Сервис г. Майкоп.

15. <https://1-trk.ru/> – сайт ООО «1-я Транспортная».

16. <https://cdek.ru> – сайт курьерской службы доставки.

Учебное издание

**ИНФОРМАТИКА:
ОСНОВЫ САЙТОСТРОЕНИЯ**

Учебное пособие

Составители

**МЕРЕТУКОВА Сусана Касеевна
ЧУНДЫШКО Вячеслав Юрьевич
МЕРЕТУКОВ Шумаф Туркубиевич
КОЗЛОВА Наталья Шумафовна**

Подписано в печать 13.09.2020 г. Бумага офсетная. Гарнитура Таймс. Формат 60 x 84 ¹/₁₆.
Печать цифровая. Усл. п. л. 8,6. Уч.-изд. л. 9,25. Тираж 100. Заказ № 089.

Отпечатано с готового оригинал-макета в типографии ИП Магарин О.Г.
385008, г. Майкоп, ул. 12 Марта, 146. Тел. 8-906-438-28-07. E-mail: olemag@yandex.ru