



УТВЕРЖДЕНО:

**Педагогическим советом Колледжа
ФГБОУ ВО «РГУТИС»
Протокол № 4 от «24» февраля 2021 г.
с изм. Протокол № 5 от 05.04.2021
с изм. Протокол № 6 от 18.06.2021**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ
ДИСЦИПЛИНЫ
ОП.09. Web-дизайн
основной профессиональной образовательной программы среднего
профессионального образования – программы подготовки специалистов среднего
звена
по специальности: 54.02.01 Дизайн (по отраслям)
Квалификация: дизайнер
год начала подготовки: 2021**

Разработчики:

должность	подпись	ученая степень и звание, ФИО
<i>преподаватель</i>		Цикин В.П.

Методические указания согласованы и одобрены руководителем ППСЗ:

должность	подпись	ученая степень и звание, ФИО
<i>Руководитель ОПОП 54.02.01. Дизайн (по отраслям)</i>		Козьмодемьянская Е.И.



Содержание

1. Общие положения
2. Тематика и содержание лекций
3. Тематика и содержание практических занятий
4. Тематика и содержание самостоятельной работы
5. Информационное обеспечение обучения



1. Общие положения

Методические указания предназначены для обучающихся по ОПОП СПО по специальности 54.02.01 Дизайн (по отраслям), изучающих учебную дисциплину «Web - дизайн», и могут использоваться как на учебных занятиях, которые проводятся под руководством преподавателя, так и для самостоятельного выполнения практических занятий, предусмотренных рабочей программой во внеаудиторное время.

Цели и задачи учебной дисциплины «Web - дизайн».

Содержание учебной дисциплины ориентировано на подготовку студентов к освоению специальности 54.02.01 «Дизайн (по отраслям)» и овладению общими и профессиональными компетенциями (ОК, ПК):

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК 6. Работать в коллективе, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

ПК 1.2. Осуществлять процесс дизайнерского проектирования с учетом современных тенденций в области дизайна.

ПК 1.5. Выполнять эскизы с использованием различных графических средств и приемов.

Формы контроля

В процессе изучения дисциплины предусмотрены следующие формы контроля по овладению общекультурными и профессиональными компетенциями: текущий контроль в форме практических занятий, устных опросов, самостоятельной работы обучающихся промежуточная аттестация в форме зачета.

Текущий контроль осуществляется в течение семестра в виде оценки результатов практических занятий, оценки выполнения самостоятельных работ, оценки устных опросов.

Промежуточная аттестация осуществляется в виде выполнения итогового тестирования

Контроль самостоятельной работы студентов осуществляется в течение всего семестра. Преподаватель самостоятельно определяет формы контроля самостоятельной работы студентов в зависимости от содержания разделов и тем, выносимых на самостоятельное изучение. Такими формами могут являться: тестирование, видеопрезентации, проектные технологии, контрольные работы и др.

2. Тематика и содержание лекций



Лекция – один из методов обучения, одна из основных системообразующих форм организации учебного процесса.

Лекционное занятие представляет собой систематическое, последовательное, монологическое изложение преподавателем-лектором учебного материала, как правило, теоретического характера. Такое занятие представляет собой элемент технологии представления учебного материала путем логически стройного, систематически последовательного и ясного изложения.

Цель лекции – организация целенаправленной познавательной деятельности студентов по овладению программным материалом учебной дисциплины.

Тематика и содержание

Раздел 1. Общие сведения об Интернете

Содержание: История, современное состояние и перспективы развития Интернета.

Раздел 2. Технические основы построения Интернета

Тема 2.1. Основы работы информационных сетей

Содержание: Протоколы передачи данных. Простейшая модель сетевых соединений. Клиент-серверные технологии, виды серверных услуг.

Тема 2.2. Новации Интернета

Содержание: Распределенная архитектура, динамическая маршрутизация. 7-уровневая модель OSI, характеристика уровней. Пакетная передача данных, структура информационных пакетов.

Раздел 3. Основы построения WorldWideWeb

Тема 3.1. Возникновение и развитие ресурса WWW

Содержание: HTTP, HTML, и другие программные средства создания сайтов. История развития World Wide Web, WWW как часть Интернета.

Тема 3.2. Другие прикладные ресурсы Интернета

Содержание: Электронная почта e-mail и web-mail, Службы FTP, TELNET: протоколы, функции в глобальной сети.

Тема 3.3. Структура адреса URL

Содержание: Адрес URL. Предоставляемые ресурсы: http, mailto, ftp, telnet, file. Доменное имя: уровни, географические и функциональные индикаторы, регистрация. Прочие разделы URL.

Тема 3.4. Выбор домена и хостинга

Содержание: Способы создания Web-ресурса: бесплатный хостинг, платный хостинг, регистрация домена. Факторы, учитываемые при выборе домена и хостинга.

Раздел 4. Создание Веб-страниц средствами HTML и CSS

Тема 4.1. Создание HTML документов

Содержание: Программные средства создания HTML-кода. Спецификации HTML. Правила записи тегов, атрибуты и правила их записи. Структура страницы HTML, разделы Head и Body.

Тема 4.2. Содержание раздела Body

Содержание: Теги работы с текстом: управляющие его разметкой на странице, работающие с цветом и фоном, создания списков, анимации текста, специальные символы. Теги создания и форматирования таблиц, вставки графики и мультимедиа, создания гиперссылок.

Тема 4.3. Содержание раздела Head



Содержание: Тег Title, мета теги: типа документа и кодировки, аннотации, ключевых слов, автора, владельца, динамичности страниц, взаимодействия с поисковиками.

Тема 4.4. Каскадные таблицы стилей CSS

Содержание: Стили и их применение, синтаксис CSS. Способы помещения стиля в код страницы: внешние файлы стилей, таблица стилей на странице, внутритекстовые стили. Наследование стилей. Классы стилей.

Раздел 5. Характеристики Веб-сайтов и требования к ним

Тема 5.1. Постановка задачи при создании сайта

Содержание: Важность правильной постановки задачи. Вопросы, прорабатываемые при создании Веб-проектов.

Тема 5.2. Изучение аудитории

Содержание: Профилирование. Демографические, социальные, психологические характеристики пользователей. Значимые характеристики для организаций.

Тема 5.3. Специфика создания Веб-контента

Содержание: Учет психофизических характеристик людей, структурирование информации, меры по повышению доступности сайта.

Тема 5.4. Дизайн и структура сайта

Содержание: Влияние цвета на эмоции человека. Меры, помогающие лучшему усвоению информации: контрастность текста и цветовых тонов, элементы навигации, использование анимации, правила применения шрифтов. Топологические структуры сайтов, области их применения. Правила построения файловой структуры сайта.

Тема 5.5. Интерактивные средства Веб-сайтов

Содержание: Интерактивность, ее виды, реализация. Правила создания распространенных сервисов: регистрации посетителей, проведения опросов и голосований, форумов и гостевых книг, новостной ленты, возвратной формы, почтовых рассылок, FAQ, аккаунтов. Интернет-магазин.

Тема 5.6. Динамические сайты

Содержание: Преимущества динамических сайтов: редактирование без программирования, прямое обращение к базам данных. Серверные средства создания динамических сайтов: языки сценариев (PHP, Perl, ASP, JSP). Клиентские средства: JavaScript, Flash. Многостадийные клиент-серверные технологии.

Раздел 6. Поисковые системы и поисковая оптимизация сайта

Тема 6.1. Развитие и порядок работы поисковых систем

Содержание: История создания поисковиков. Поисковые каталоги. Характеристика ведущих поисковых систем в России и в мире. Поисковые машины, этапы их работы: сбор данных из WWW, индексация, обработка клиентского запроса. Релевантность, ее критерии.

Тема 6.2. Правила составления запросов

Содержание: Пертигентность, языки поисковых запросов. Применение для поиска логических операторов И, НЕ, ИЛИ, другие команды.

Тема 6.3. SEO, внутренние факторы ранжирования

Содержание: Факторы ранжирования страницы: ключевые слова, параметры и теги оптимизации текста. Параметры структуры сайта. Распространенные ошибки, попытки обмана поисковика.

Тема 6.4. SEO, внешние факторы ранжирования

Содержание: Индекс цитирования, содержание и тематическая релевантность внешних ссылок. GooglePageRank: ТИЦ и ВИЦ Яндекса, их использование.



Тема 6.5. Оптимизация при продвижении сайта

Содержание: Индексация сайта в поисковых системах. Оптимизация высокочастотных и низкочастотных ключевых слов. Сабмит в каталоги, DMOZ, Яндекс. Маркетинговые меры повышения популярности сайта.

3. Тематика и содержание практических занятий

Практическое занятие №1

Создание Web-страниц формата .html в текстовом редакторе «Блокнот»

В данной работе, для создания несложных Web-страниц, Блокнот используется по следующей причине. Этим способом можно освоить язык HTML, и в дальнейшем, уже со знанием дела, работать с другими редакторами.

1. HTML и Интернет (очень кратко)

HTML – это язык программирования Web-страниц. Код HTML форматирует содержание страницы (текст, рисунки, звуки и прочее) с помощью стандартных команд – **тегов** (они также называются **дескрипторы**). При форматировании задаются размер, расположение и много других свойств элементов, из которых состоит страница.

При записи программного кода команды-теги заключают в угловые скобки, например: **<HTML>**. Большинство тегов парные, они подразделяются на открывающие и закрывающие. Открывающий тег определяет действие, которое производится с текстом или другим элементом, стоящим после него. Закрывающий указывает на окончание действия и имеет косую черту: **</HTML>**. Команда распространяется на все, что расположено между открывающим и закрывающим тегом.

Кроме парных тегов есть и одиночные, не имеющие закрывающей части. Например, **МЕТА** теги или команда добавления графики: ****.

Теги могут быть вложены друг в друга. Это происходит в том случае, когда к участку текста или другому объекту одновременно применяется несколько команд. При этом работает **принцип стека**: команда, примененная позже, отменяется раньше (и наоборот – самая ранняя команда отменяется позже всех).

Пример. Запись **<P>Это вложенные <I>теги</I></P>** устанавливает следующую разметку:

Это **вложенные теги**

Теги могут иметь **атрибуты**, которые задают различные варианты использования одного и того же тега. Атрибуты всегда записываются внутри угловых скобок. Сначала, после пробела пишется вид атрибута. Потом, ставится знак = и пишется его значение. В общем случае, значения атрибута пишется в кавычках, но для большинства ситуаций кавычки можно не ставить.

Примеры атрибутов:

- **Атрибут** - здесь слово Атрибут будет отображено шрифтом Arial, текстом 4-го уровня (чаще всего это уровень основного текста);
- ** Атрибут** - здесь слово Атрибут будет отображено шрифтом Verdana, текстом 2-го уровня (это заголовок, хотя и не самый крупный).

Важно, что к некоторым командам язык HTML нечувствителен.

• В программном коде между разными тегами может задаваться сколько угодно пробелов и переводов строки, это не имеет никакого значения. Например, записи:

<P>Internet</P><P>Explorer</P> и
<P>Internet</P><P>Explorer</P>



полностью равнозначны.

- Теги и атрибуты (но не значения атрибутов) можно набирать как строчными, так и прописными буквами. Например, обозначение тега **FONT**, **Font** и даже **foNt** будут восприниматься браузерами одинаково.

Как известно, Web-страница в Интернете, обязательно должна иметь адрес **URL** (UniformResourceLocator), без адреса ее в Сети просто нет. Каждый пользователь Интернета знает этот адрес (но не все знают, что это URL).

Пример адреса:

http://www.abc.def.ru/Index.html

Первая часть адреса (до знаков // включительно) обозначает протокол доступа к данному ресурсу. Что это такое - здесь мы рассматривать не будем. Вторая часть (до **ru** включительно) - доменное имя. Каждый Web-сайт имеет свое уникальное, присущее только ему имя. А далее идет адрес страницы в структуре данного сайта (аналогично адресу файла на диске компьютера). В данном случае структура сайта одноуровневая, папок внутри папки сайта **abc** нет. Именем **Index** обычно называют главную страницу сайта. При отсутствии указания на страницу браузер выводит страницу с именем Index.

При создании сайта все файлы, из которых сайт состоит, собирают в одну папку. Это могут быть не только веб-страницы, но и рисунки, скрипты, другие элементы. Компактное размещение требуется потому, что файлы сайта связаны друг с другом множеством гиперссылок. Каждая гиперссылка должна иметь однозначный адрес. Внутри одной папки адреса имеют минимальный путь, и при переносе папки в другое место гиперссылки не рвутся. Если же какие-то файлы окажутся за пределами переносимой папки, то этих материалов в сайте уже не будет.

Файлы, из которых состоит сайт, следует именовать латинскими буквами. До недавнего времени все файлы Интернета могли именоваться только символами английской клавиатуры. В последнее время в Сети появились доменные имена на кириллице. Но это исключения, а в остальных случаях файлы на кириллице Интернет не воспримет.

2. Создание структуры кода страницы и заголовка

2.1. Откройте программу **Блокнот (Notepad)**. Если ярлыка программы нет на рабочем столе, то ее можно найти с помощью команд: **Пуск** → **Все программы** → **Стандартные**.

2.2. Документы HTML имеют следующую структуру:

```
<HTML>  
<HEAD>  
</HEAD>  
<BODY>  
</BODY>  
</HTML>
```

Наберите в Блокноте указанные теги. Здесь и далее по умолчанию набирайте команды и текст обычным, нежирным шрифтом.

Блокнот – примитивный редактор. Единственное существующее в нем средство ускорения набора кода – буфер обмена. Максимально используйте его.

Тег <HTML> указывает на язык программирования HTML. Эта запись нужна потому, что Web-страницы могут создаваться и другими средствами.

В разделе HEAD задаются базовые параметры страницы, посетитель при загрузке сайта их не видит. Исключение: название страницы выводятся на экран в самой верхней строке браузера, на синем фоне. Базовые параметры используются: браузерами при загрузке страницы на компьютер; поисковыми машинами при



помещении данной страницы в список ссылок, создаваемых по запросу клиента; а также для других целей.

В разделе BODY размещается содержание станицы. Здесь все, что находится за пределами угловых скобок <>, выводится на экран.

2.3. Теперь сохраним сделанное в формате .html. Для этого сначала создадим папку, куда будем помещать все материалы создаваемого сайта. Создайте такую папку в месте, указанном преподавателем, и назовите ее произвольным именем (естественно, пользуясь английской клавиатурой).

2.4. Далее в папку сохраним созданную страницу (под именем Index или любым другим, набранным латинскими буквами). Перейдите опять в Блокнот, вызовите окно **Файл** → **Сохранить как**, и в строке **Имя файла** замените .txt на .html, а * на то имя, которое вы собираетесь дать создаваемой странице. В строке **Тип файла** поставьте **Все файлы**. При запуске файла включается браузер и демонстрируется содержимое страницы, пока это пустое окно.

2.5. Создайте заголовок страницы, отображаемый в строке заголовка браузера. Для этого в разделе **HEAD** надо создать строку: `<TITLE></TITLE>`, и между этими тегами написать название страницы, наша страница будет называться: **Казахская кухня**. Сохраните сделанное изменение командой **Файл** → **Сохранить** или **Ctrl + S**. Теперь при запуске страницы в строке заголовка, на самом верху экрана, появляется ее название.

Название страницы, задаваемое тегом <TITLE> и наименование файла страницы – это разные вещи. Первое используется поисковыми системами для создания базы ссылок, второе входит в адрес URL и используется при доставке страницы пользователю.

В теге <TITLE> часто задают ключевые слова. Набор ключевых слов надо тщательно продумать с точки зрения возможных запросов пользователей. Иначе говоря: при наборе каких слов поисковая система будет загружать пользователю вашу страницу.

2.6. Теперь при запуске страницы в строке заголовка, на синей полосе в верхней части экрана, появляется ее название. Для того, чтобы посмотреть сделанное изменение, надо в Internet Explorer нажать кнопку **Обновить**. Для перехода обратно к программному коду надо применить команду **Файл** → **Править в Блокнот**. В старых версиях Explorer надо нажимать: **Вид** → **Источник**.

В дальнейшем все созданное надо сохранять в Блокноте, потом переходить в Internet Explorer и обновлять страницу. После чего через меню Вид переходить обратно в Блокнот (если используемая версия браузера позволяет это).

2.7. Создадим несколько параметров, не выводимых на экран. В **HEAD** запишите непарный тег с атрибутами: `<METAname='author' contents='Имя Фамилия'>`, при этом поставьте свои имя и фамилию. Этот дескриптор создает имя автора для поисковой системы.

2.8. Далее там же запишем `<METAname='description' contents='Страница создана при выполнении практического занятия по предмету ...'>`, поставьте название предмета. Здесь после contents пишется то, что поисковая система будет приводить в качестве резюме при помещении вашей странице в список вместе с другими найденными страницами. Легко убедиться, что Meta-теги на экране не отображаются (кроме TITLE).

3. Разметка Web-страницы с помощью таблиц

Создание чего-либо обычно начинается с проектирования. Иногда сначала проект Web-страницы рисуют на бумаге. На практике разметку станицы создают с помощью таблиц. Таблицы в Web-дизайне имеют более широкое применение, чем в текстовых документах. Создаваемая нами страница будет иметь несложную структуру, изображенную на рисунке (линии по умолчанию будут невидимы).

Для создания таблиц используются четыре тега:

- `<TABLE></TABLE>` – создает таблицу;
- `<TR></TR>` – создает новую строку;
- `<TH></TH>` – создает ячейку заголовка;
- `<TD></TD>` – создает простую ячейку.

По умолчанию в ячейку заголовка пишется текст полужирного начертания с выравниванием по центру. В обычной ячейке текст не выделяется и имеет выравнивание по левому краю.

Для начинающего создать подобную структуру сложно, можно легко запутаться. Поэтому будем снабжать создаваемый программный код **комментариями** – замечаниями, недоступными для просмотра.

Комментарии вводятся следующим образом: `<!--Текст комментария -->`. Однако комментарии – это не секретный текст, при просмотре программного кода они становятся доступными.

3.1. После `<BODY>` напишите `<TABLE width=100% border=1></TABLE>`. Мы создали таблицу, внутри которой будет помещено все содержание Web-страницы. Надписью `border=1` задали толщину границ ячеек в 1 пиксель. Ширину таблицы задали в 100% экрана монитора. Не имеет значения, каков будет монитор у пользователя, таблица по ширине займет весь экран.

Не забывайте о том, что между тегами можно ставить пробелы и переводы строки. HTML их не чувствует, а для чтения программного кода операции удобно разделять переводом строки.

3.2. После `<TABLE>` напишите `<TR><TH></TH></TR>` – это будет шапка, где будет помещен красочный заголовок страницы. Обратите внимание на то, что в строке обязательно надо создать ячейку, даже если она одна. После `</TR>` создадим комментарий: `<!--Шапка -->`. Чтобы ячейка отобразилась в окне браузера, надо в нее вписать какое-либо слово (между открывающим и закрывающим тегом). Сделайте это.

3.3. Создайте еще одну строку, это будет строка меню. В ней создайте три ячейки заголовка, в каждую впишите что-либо. К строке создайте комментарий: `<!--Строка меню -->`, и посмотрите что получилось.

3.4. Создайте еще одну строку для ячеек (простых) иллюстраций и текста, с двумя ячейками, впишите в ячейки что-либо.

3.5. Завершите разметку страницы созданием строки с одной простой ячейкой для нижней части, но сделайте выравнивание текста по центру: `<TR><TD align='center'></TD></TR>`. Впишите в ячейку что-либо, сделайте комментарий, посмотрите что получилось.

По горизонтали получилось не то, что надо. Дело в том, что HTML стремится сделать все ячейки одинаковыми по ширине, причем к тому же примерно одинаковыми. Придать

Шапка	
Строка меню	
Иллюстрации	Текст
Нижняя часть	



ячейкам разную ширину можно, но это сложный процесс. Проще создавать объединенные ячейки, занимающие несколько столбцов. Это делается командой **COLSPAN**.

3.6. Для ячеек шапки и низа поставьте атрибут **colspan=3** (т.е. эти ячейки занимают три столбца каждая). Для текста поставьте **colspan=2**. Расположение ячеек должно стать таким, как на рисунке.

3.7. Команду **border=1** уберите, границы ячеек нужны были только для наглядности.

4. Импорт и форматирование текста

4.1. В Блокноте откройте файл **Казахская кухня**, выделите начало текста до слов «Деликатесы из конины», и с помощью буфера обмена перенесите этот текст в раздел, отведенный для основного текста (то, что там было написано – удалите). При просмотре видно, что форматирование текста оставляет желать лучшего. По умолчанию HTML выставляет левое выравнивание без абзацев и каких-либо иных параметров.

4.2. Вернемся в Блокнот и установим абзацы. Для этого текст каждого имеющегося абзаца надо заключить между тегами **<P>** и **</P>**. Одновременно эти теги устанавливают шрифт обычного начертания. Для ускорения процесса можно их вводить через буфер обмена (сначала **<P>**, потом **</P>**).

Можно менять и другие параметры текста: шрифт, начертание, размер и др. Возможности изменения вида текста достаточно широкие, сравнимые с Microsoft Word. Но в Блокноте вместо щелчка по меню и кнопкам приходится ставить команды языка HTML, что, конечно, занимает больше времени. Чтобы не пришлось переделывать работу, периодически надо контролировать правильность введенных команд в Internet Explorer.

4.3. Для заголовка и первого абзаца установим центральное выравнивание, добавив в открывающий тег атрибут выравнивания по центру: **<Palign=center>**. Для остальных абзацев сделаем выравнивание по ширине: **<Palign=justify>** Разумеется, здесь тоже полезно использовать буфер обмена.

4.4. Теги и атрибуты можно применять сразу к нескольким абзацам. Перед открывающим тегом заголовка поставьте **<FONTface=Arial>**, после закрывающего тега 2-го абзаца поставьте ****. Внутри обозначенной области шрифт TimesNewRoman сменился на Arial.

4.5. В HTML можно установить 6 уровней заголовков, что логически упорядочивает данные на Web-странице. Кроме того, от умелого использования заголовков зависит эффективность нахождения страницы в Интернете поисковыми роботами. Обычный текст соответствует 4 уровню; 1 уровень имеет самый крупный текст; 5 и 6 используются для сносок и примечаний. На странице рекомендуется использовать не более 3-х уровней.

Примените к заголовку второй уровень: **<H2></H2>**, к 1-му абзацу третий уровень: **<H3></H3>**, это будет подзаголовок. Теги заголовков надо ставить первыми, в самом начале и в конце абзаца. Второй абзац сделаем курсивным начертанием, поставив теги **<I>** и **</I>** внутри, после остальных команд.

Для правильного воспроизведения команд имеет значение порядок их применения. В Блокноте этот порядок легко установить экспериментально. При других способах создания Web-страницы несоблюдение порядка может быть не так заметно, что вызовет трудности.

4.6. С помощью парного тега **<BLOCKQUOTE>** можно выделить любой абзац, за счет появления отступов справа и слева. Это хороший прием выделения текста. Примените к любому абзацу дескриптор **<BLOCKQUOTE></BLOCKQUOTE>**.



4.7. Теги могут применяться не только к абзацу, но и к отдельным словам и выражениям, даже к символам. В тексте выделите любые выражения, слова, символы жирным шрифтом: ``, подчеркиванием: `<U></U>`. Измените размер шрифта для отдельных слов: `<FONTsize=6>`, это крупный шрифт. В HTML номер шрифта не соответствует его кеглю (как в Word). Номер 6 – это 24-й кегль, а 1 – 8-й.

Списки – простой и эффективный способ упорядочивания материала в логической последовательности.

4.8. **Нумерованный список** создается с помощью двух пар тегов: `` и `` устанавливают начало и конец нумерованного списка, а `` и `` отмечают отдельные элементы списка. Найдите в тексте какое-либо из перечислений и ограничьте его тегами ``, а каждое из слов – тегами ``. Посмотрите, получился ли список.

4.9. С помощью различных атрибутов можно установить нумерацию римскими цифрами и буквами, задать начало нумерации не с первого номера, пропустить номера в списке. Задайте `<OLtype=a>`, посмотрите что получилось.

4.10. **Маркированный список** создается тегами `` и ``, элементы списка ограничиваются теми же `` и ``. Создайте маркированный список из слов 3-го абзаца Web-страницы, причем фразы по строкам нужно разместить следующим образом, вставив заглавные буквы и поменяв знаки препинания:

- В первую очередь ... айран.
- Затем чай с молоком или сливками ... куртот.
- Затем следовали закуски из конины ... кабырга.
- На любом столе ... пшеничной муки.

4.11. **Многоуровневые списки** создаются простым вложением тегов ``, ``, `` внутри имеющегося элемента списка, то есть области, ограниченной `` и ``. Создайте многоуровневый список из маркированного списка, поставив в отдельные строки перечисленные напитки и закуски.

4.12. Маркеры в списках могут быть разными, их можно задавать атрибутом `type`. Для одного из вложенных списков поставьте `<ULtype=square>`.

Бегающая строка – анимационный эффект, при котором строка текста перемещается по странице. Это хороший способ привлечения внимания к короткому, но емкому сообщению. Эффект бегающей строки поддерживается только в Internet Explorer, в других браузерах строка будет неподвижной.

4.13. Теги бегающей строки: `<MARQUEE>` и `</MARQUEE>`. Поставьте их до и после слов подзаголовка: **его история, обычаи и традиции**. Эти слова выделятся в отдельную бегающую строку, по умолчанию она движется справа налево и бесконечно повторяется.

4.14. Параметры перемещения текста можно изменять соответствующими атрибутами. Дополните тег атрибутом `<MARQUEEdirection=right>` – направление движения текста сменится на противоположное. Задайте `loop=7` – теперь бегающая строка повторится 7 раз. У этого тега есть и другие интересные атрибуты.

5. Добавление графики

В Интернете применяются 3 растровых графических формата: JPEG, GIF и PNG. Используется также векторная FLASH-графика и анимация, создаваемая с помощью Flash-редакторов.

Используемая в Сети растровая графика проходит специальную подготовку в графических редакторах, чаще всего в Adobe Photoshop. Без этой подготовки иллюстрации в Сеть не попадут, даже если будут вставлены в программный код.



Графические файлы обычно имеют намного больший размер, чем текстовые, и они могут существенно увеличить время загрузки Web-страницы. Поэтому основная цель подготовки заключается в уменьшении размеров графического файла за счет снижения разрешения и глубины цветности. Другая цель – создание анимационных эффектов, украшающих публикацию.

Подготовку графики к Интернету имеет смысл рассматривать при изучении графических и анимационных редакторов, далее же говорится о работе с графикой средствами HTML.

5.1. Графика вставляется непарным тегом ****, вставьте его в ячейку, предназначенную для иллюстраций (убрав то, что в ней было). В середине ячейки появляется значок рисунка без самого рисунка, он ставится, когда браузер не смог найти графический файл.

На странице можно разместить рисунки, из любой папки компьютера и даже те, которые находятся в Сети. Для этого надо правильно прописать путь доступа к ним. Но при указании пути доступа могут встретиться различные трудности, о которых разработчик сайта может даже и не знать. Поэтому обычно все рисунки переносят в папку **Image**, которую размещают в папке, где находятся компоненты сайта.

5.2. Пользуясь указаниями преподавателя, перенесите все необходимые рисунки в папку вашего сайта. Теперь вставьте рисунок **KazKuh1**, дополнив тег **** атрибутом **src=Images/KazKuh1.gif**– должна появиться картинка. **Src**– обязательный атрибут, без него неясно, какой файл надо вставлять. Вставьте другую картинку: ****

Для вставки рисунка необходимо указать его формат. Формат должен читаться браузером, что бывает не всегда, даже если это GIF или JPEG. Читаемый формат создается при подготовке рисунка для Web, что можно сделать во многих программах. В **Photoshop** надо применить команду **Файл → Сохранить для Web**. В открывшемся окне можно менять различные параметры, в том числе формат.

Не каждый графический файл может быть помещен на Web-страницу, а только подготовленный специальным образом.

5.3. Рисунки располагаются горизонтально, подобно соседним символам текста, и это расположение «съедает» очень много места на странице. Для вертикального расположения надо каждый рисунок расположить в своем абзаце, то есть до каждого тега **** поставить **<P>**, а после – поставить **</P>**. Сделайте это.

5.4. Еще одно неудобство: рисунки попадают в центр ячейки, а лучше их располагать сверху. Для этого надо ячейку дополнить атрибутом: **<TD valign=top>**.

Для расположения содержимого ячейки по горизонтали используется атрибут **ALIGN** со значениями: **left, center, right** (соответственно влево, по центру, вправо). Для расположения содержимого по вертикали используется атрибут **VALIGN** со значениями: **top, middle, boottom** (соответственно вверху, по центру, внизу).

5.5. Страница будет лучше смотреться, если рисунки сделать одинаковыми по ширине. Сначала определим размеры рисунков. Щелкните правой кнопкой мыши по файлу **KazKuh1.gif**, выберите **Свойства** → **Сводка**. Рисунок имеет ширину **283 точки**, высоту – **378 точек**. Аналогично можно узнать, что **KazKuh2.jpg** имеет ширину **186 точек**, высоту – **259 точек**.

5.6. С помощью простых вычислений можно определить, каковы должна быть высота **KazKuh2.jpg**, если мы увеличим его ширину до 283 пикселей: **(283:186) x 259 = 394 пикселя**. Размеры любого графического файла можно скорректировать атрибутами



width(длина)и**height**(высота). Поставьте в тег вызова **KazKuh2.jpg** атрибуты **width=283**
height=394.

5.7. Добавьте файл **KazKuh3.jpg**. Он оказался слишком велик. Поставьте ему ширину **283**
пикселя, а необходимую высоту вычислите самостоятельно. Поставьте этому рисунку
атрибут **valign=bottom**

5.8. Дополните рисунки рамками толщиной в 2 пикселя, поставив каждому атрибут
border=2.

*В виде рисунка можно вставить текст необычного начертания, выполненный в мало
распространенной шрифтовой гарнитуре.*

5.9. В самой верхней ячейке таблицы, в шапке, вставьте рисунок **TextKazKuh**. Задайте
ему атрибут центрального выравнивания **align=center**. Добавьте еще атрибут
альтернативного текста: **alt='Казахская кухня'**. Обратите внимание на то, что значение
атрибута, состоящее из нескольких слов, обязательно надо выделять апострофами: ‘’.

*Для чего нужен альтернативный текст? Он читается поисковыми системами, а рисунки
поисковая система не читает. Часто в шапке Web-страницы задаются самые главные,
ключевые слова в виде красочного рисунка. Тогда без атрибута **ALT** сильно снижаются
шансы страницы быть замеченной пользователем.*

6. Работа с цветом и фоном

*В HTML можно **менять цвет текста, фона, различных объектов**. Причем можно
изменить цвет любого участка текста: от всего содержимого до отдельных символов. А
фон можно задать как для всей страницы, так и для ее части.*

*Количество возможных цветов очень велико, оно соответствует цветовой модели RGB
(red, green, blue). Двумя шестнадцатеричными цифрами задаются уровни яркости
красного, зеленого или синего цвета. Цвет задается комбинацией из этих трех основных
цветов, т.е. требуется шесть шестнадцатеричных цифр. Всего таким способом можно
создать около 17 миллионов цветов.*

*16 цветов имеют собственные имена, их можно задавать по имени. Например: **red**–
красный (**#FF0000**), **blue**– синий (**#0000FF**), **silver**– серебряный, (**#C0C0C0**) – оливковый
(**#808000**). В скобках приведены шестнадцатеричные коды этих цветов. Перед числовым
кодом цвета всегда ставится знак #.*

6.1. Раскрывающий тег **<BODY>** дополните атрибутом: **bgcolor=#FFFF80**, что
соответствует светло-желтому цвету фона.

6.2. Для любых слов или выражений поменяйте цвет шрифта с черного на красный,
поставив: **<FONTcolor=red>**.

6.4. Окрасьте светло-желтым цветом, фон бегущей строки и фон последней строки
таблицы, поставив атрибут **bgcolor=#FFFFD0** в теги **<TR>** и **<MARQUEE>**. А для фона
строки меню выберите голубой цвет: **bgcolor=# D0D0FF**

6.5. Страницу можно украсить фоновым рисунком. Если рисунок меньше окна браузера,
то он многократно копируется. Чтобы не мешать чтению, фоновый рисунок должен быть
светлым, малоконтрастным, не очень сложным. Замените в теге **BODY** атрибут фонового
цвета на атрибут фонового рисунка **background** и адрес файла **ForFon**:
background=Images/ForFon.jpg.

7. Специальные символы

Специальные символы бывают двух видов.

*1. Различные знаки, которые сложно или невозможно набрать прямо с клавиатуры.
Одни из них имеют специальное и служебное назначение, другие представляют собой
картинки, украшающие публикацию.*



2. Знаки, используемые в языке HTML. Они нужны для того, чтобы на Web-странице можно было написать команды HTML в виде текста. При этом неправильно набранная команда и так появится на Web-странице в виде текста.

Команды специальных символов всегда начинаются с **&**, например: `<`, `×`, `&trac34`. Для этих команд угловые скобки `<>` не ставятся.

Команды могут иметь **именной и числовой код**. Именной код присвоен командам вывода самых важных символов, например: `>` (`>`, больше), `¢` (`€`, евро), `©` (`©`, знак авторского права). Свой числовой код имеет любая команда вывода символа, включая буквы русского и английского алфавитов, цифры, различные знаки и пр. То есть числовой код позволяет вывести любой символ, установленный в компьютере. Числовой код записывается в следующем виде: **&#номер символа**.

7.1. При импорте текста HTML игнорирует множественные пробелы, между всеми словами остается только один пробел. Команда ** ** вставляет **неразрывные пробелы**, каждый из которых в обязательном порядке присутствует на Web-странице. В начале любого абзаца поставьте подряд 5 раз символ ** **. Получится абзацный отступ первой строки.

7.2. Верхняя иллюстрация располагается слишком близко к строке меню. Для того, чтобы ее немного сдвинуть вниз, создайте в столбце картинок сверху еще одну строку с пробелом: `<P> </P>`.

7.3. В любом месте страницы поставьте знаки `<>`, **align=justify**, **&abcd**. Все эти неправильно введенные команды HTML отобразятся в виде текста.

7.4. В нижней строке страницы наберите: **©Миреев В.А.**, имеющийся там текст удалите. Отобразится копирайт автора этого текста.

7.5. Введем символ-иллюстрацию из специального шрифта. В шапке, после рисунка **Казахская кухня** создадим абзац текста с помощью следующего набора команд: `<P align=left>ä</P>`. Написанное означает введение символов шрифта Webdings, 16 кегля, с левым выравниванием. Из этого шрифта набираем один символ с кодом 228. В результате в левом верхнем углу страницы отобразится картинка со столовыми приборами.

Таким образом, на страницу можно загрузить как рисунок, который выглядит как текст (см. пункт 5.9.), так и текст в виде рисунка.

7.6. Коды символов можно найти в Microsoft Word. Вызываем **Word**, создаем там пустой файл, после чего командами **Вставка** → **Символ** открываем окно, в котором можно найти все имеющиеся в данном компьютере знаки и их коды. Лучше выбирать распространенные шрифты, например Webdings, он имеется на всех компьютерах с ОС Windows.

8. Создание гиперссылок

Гиперссылки являются важнейшим элементом Web-страниц, они обеспечивают логическую структуру всемирной паутины WWW. Без гиперссылок Интернет в его нынешнем виде невозможен.

Гиперссылка создается парными тегами `<A>` и ``, между которыми заключен текст или картинка, служащие гиперссылкой. Адрес, на который ведет гиперссылка, задается атрибутом **href**: `<Ahref=адрес>`.

Гиперссылкой может служить:

- участок текста любого размера, от буквы до нескольких абзацев;
- баннер – графический или анимированный объект, ставший гиперссылкой.

Гиперссылка может вести на:

- закладку на текущей Web-странице;
- другую Web-страницу данного сайта, как правило на том же носителе;
- Web-страницу, находящуюся в Интернете, а также в локальной сети;
- мультимедийный ресурс в сети или на данном компьютере;
- документ FTP сервера;
- адрес электронной почты;
- сценарий на языке JavaScript.

Есть и другие варианты гиперссылок.

8.1. В строке меню создадим гиперссылки на рецепты нескольких блюд. Для этого в каждой ячейке меню напишем соответствующие ссылки: `<A>Рецепт 1`, `<A>Рецепт 2`, `<A>Рецепт 3` (вместо того, что там было). Но пока эти гиперссылки никуда не ведут, надо создать соответствующие страницы.

8.2. Скопируйте файл **Kazyi.html** в ту же папку, где находится ваша главная страница, после чего для ссылки **Рецепт 1** поставьте атрибут адреса гиперссылки: `<Ahref=Kazyi.html>`. Надпись **Рецепт 1** должна приобрести синий цвет и стать подчеркнутой (так по умолчанию обозначаются текстовые гиперссылки).

Но каждая страница должна тоже иметь ссылки куда-то, иначе она будет являться «тупиком», откуда не выбраться. Для страницы **Kazyi.html** сделаем обратную ссылку на **Index.html**.

8.3. В конце текста сделайте новую строку, придайте ей центральное выравнивание, и напишите: **НА ГЛАВНУЮ**. Поставьте также тег жирного начертания: ``. Затем поставьте тег гиперссылки: `<Ahref=Index.html>`. Не забудьте, что после надписи должны идти завершающие теги, в порядке, обратном по отношению к открывающим: `</P>`.

8.4. Создайте еще одну страницу-рецепт для любого выбранного вами блюда. Раскройте код страницы **Казы** (см. пункт **2.6**) и сотрите все, что содержится между тегом **BODY** и ссылкой на главную страницу. Раскройте файл **Казахская кухня** и скопируйте оттуда любой рецепт вместо стертого текста.

8.5. Сохраните файл в свою папку под другим именем в виде страницы html, так, как это описано в п. **2.4**. В теге **TITLE** замените **Казына** другое название.

8.6. Просмотрите созданную страницу. Отсутствуют абзацы, все абзацы надо обозначить тегами `<P>` и `</P>` (что удобнее сделать с использованием буфера обмена). Для названия блюда сделайте центральное выравнивание.

8.7. Пропишите ссылку **Рецепт 2** на созданный файл так, как это описано в п. **8.2**. Проверьте, работает ли созданная вами ссылка.

8.8. Аналогично создайте третью страницу для выбранного блюда. Ссылкой на нее будет надпись **Рецепт 3**.

8.9. С помощью атрибута **target** можно поменять способ открытия документа, указанного в гиперссылке. Добавьте в тег `<Ahref=Kazyi.htmltarget=blank>`, тогда при открытии внешнего документа текущая страница не исчезает, и можно работать с двумя страницами.

8.10. Создадим гиперссылку – графический баннер с рисунка пельменей на страницу **Pelmeni**. Сначала эту страницу скопируйте к себе в папку. Затем найдите в HTML-коде страницы тег ``, управляющий вставкой рисунка пельменей и ограничьте его тегами создания гиперссылки `<A ...>`.



Рамка иллюстрации должна поменять свой цвет на сиреневый. По умолчанию все баннеры имеют такую рамку. Курсор на рисунке приобретает вид руки, при щелчке должна открыться страница **Пельмени**.

8.11. Однако с этой страницы никуда не ведет гиперссылка, получился тупик. Откройте страницу любого рецепта, вызовите ее код и скопируйте оттуда надпись **НА ГЛАВНУЮ** со всеми управляющими тегами.

8.12. Бывает так, что размер изображения велик и его включение в Web-страницу существенно замедлит загрузку документа. Тогда на этот файл создают гиперссылку.

Создайте ссылку на рисунок **Baursaki.jpg**. Скопируйте рисунок в свою папку, после чего создайте гиперссылку с элемента списка **баурсаками** на рисунок этого кушанья. Атрибуты не должны отличаться от ранее созданных гиперссылок. При щелчке по слову должен открываться рисунок. Чтобы вызванный рисунок служил иллюстрацией к тексту, дополните тег гиперссылки атрибутом **target=blank**.

8.13. Если страница длинная, то в ней можно создать систему навигации с переходом от одного участка к другому. Для создания закладки то место, куда ведет гиперссылка следует обозначить тегами гиперссылки с атрибутом **name: <Aname=#имя закладки> ... **. Имя закладки может быть любым. Сама гиперссылка пишется обычным образом, но добавляется значок **#:<Ahref=#имя закладки> ... **.

В самом конце страницы заключите **©В.А.Миреев** между тегами **<Aname=#chapter1>и**. В самом начале заключите **ВВЕДЕНИЕ** между тегами **<Ahref=#chapter1>и**. При щелчке по слову **ВВЕДЕНИЕ** должно появляться окончание страницы.

8.14. Web-сайт обычно имеет ссылки на внешние ресурсы Интернета. Сделайте ссылку на главную страницу поисковой системы **Яндекс**. Для этого скопируйте в папку **Images** значок поисковика **Yandex.gif** и поместите его в конце страницы **Index**, после надписи **©В.А.Миреев**. Снабдите этот значок гиперссылкой на **Яндекс**: **<Ahref=http://www.yandex.ru>**, разделите надпись и значок несколькими неразрывными пробелами.

Если компьютер подключен к Интернету, то при щелчке по значку, появится страница поисковика **Яндекс**. Для возвращения нажмите в меню **Internet Explorer** кнопку **Назад**.

8.15. Интересна ссылка на адрес электронной почты. Она запускает окно создания письма по адресу, указанному в этой ссылке. После значка **Yandex** наберите **Написать письмо** (тоже отделив несколькими пробелами). Поставьте к этой надписи гиперссылку **<Ahref=mailto:abc@def.ru>**. При щелчке запустится окно для создания письма, хотя вряд ли такой адрес существует.

При создании сайта с помощью этой операции организуют возможность посетителю написать письмо на e-mail владельца сайта.

Практическое занятие №2

Создание Web-сайта с помощью программного пакета Dreamweaver

Редактор Web-приложений Dreamweaver пользуется заслуженной популярностью. Огромный арсенал его средств и инструментов предоставляет профессионалу широкие возможности, как в реализации требований заказчиков, так и в разработке собственных идей. Но и новичок с помощью этой программы без особых трудностей может создавать сайты, и может постепенно пройти путь до опытного разработчика.

Первая версия Dreamweaver была выпущена компанией Macromedia в 1998 году. Постепенно редактор завоевывал популярность, став самым известным средством



разработки Web-приложений. В 2007 году компания Macromedia была приобретена компанией Adobe и с тех пор приложение называется AdobeDreamweaver.

Интерфейс редактора типичен для всех продуктов Macromedia: окна и панели Dreamweaver похожи на окна и панели Flash. Но теперь эти программные пакеты интегрированы с другими многочисленными приложениями Adobe. Для Web-дизайна самым важным, конечно, является Photoshop.

Для создания сайтов в Dreamweaver, как и в других Web-редакторах, не требуется набирать код HTML. Для этого существуют множество панелей, инспекторов, вкладок, диалоговых окон и других элементов управления. Они позволяют реализовать принцип WYSIWYG: whatyouseeiswhatyouget (что вы видите, то и получите). То есть, на экране редактора мы видим, как будут выглядеть наши Web-страницы после применения тех или иных команд.

Однако при создании сайта, особенно если этим занимается новичок, а также бригада дизайнеров, возможны многочисленные ошибки: неправильные команды, отсутствие связей и др. Dreamweaver имеет мощные средства поиска и устранения ошибок, непосредственного просмотра страниц в браузерах.

Но даже такой мощный редактор, как Dreamweaver не гарантирует полного отсутствия ошибок, поскольку современные средства Web-дизайна разнообразны и постоянно совершенствуются. Поэтому отказываться от классических, проверенных средств разумно лишь в том случае, если они не работают. Тем более что нередко одни и те же команды по-разному интерпретируются в разных браузерах.

Dreamweaver позволяет использовать все средства HTML, включая таблицы, фреймы, каскадные листы стилей, слои, формы. Имеется внушительный список готовых шаблонов, можно создавать собственные шаблоны и библиотеки. Кроме HTML, программный пакет работает со средствами, позволяющими создавать и добавлять на страницы мультимедийные и интерактивные элементы, такими как DHTML, Flash, Java, ActiveX, внедрять серверные технологии: ASP, JSP, PHP, AJAX.

К сожалению, официальной русской версии пакета нет (это касается CS4 и ниже). Но в данной работе использован корректный перевод интерфейса.

В данной практической работе предлагается создать с помощью Dreamweaver несложный сайт-визитку компании. Это сайт компании **Строймаркет**, оказывающей информационно-технические услуги в области строительства. Реально такой фирмы не существует, но похожая фирма есть, и ее сайт взят за образец.

1. Определение параметров Web-проекта

Первые, очень важные стадии создания сайта – это:

- продумывание концепции, то есть определение содержания;
- планирование структуры, то есть определение страниц и связей между ними.

Тщательное планирование является залогом успешной работы сайта в дальнейшем. Ошибки на этом этапе могут привести к большим неудобствам при пользовании сайтом, либо к его переделке.

В данном случае структура и содержание сайта определены, остается только его создать. Сайт будет состоять из 7 страниц: **главная, услуги, цены, о нас, программы, выставки и семинары, литература**. Известны и связи между ними.

На следующем этапе работы следует задать параметры создаваемого веб-проекта, подготовить Dreamweaver к управлению вашим сайтом.

1.1. Прежде всего, создайте папку, где будут храниться все файлы вашего Web-сайта, произвольно назовите ее.

1.2. Откройте программу **Dreamweaver**, появляется исходное окно программы. В столбце **CreateNew (Создать)** собраны команды создания различных объектов. Видно, что возможности редактора широки. Открыв папку **More**, можно познакомиться с множеством предлагаемых готовых шаблонов.

Нажмите **DreamweaverSite...**, открывается окно **SiteDefinition (Определение сайта)**. Параметры сайта можно задать с помощью мастера на вкладке **Basic (Основные)**. Но мы будем создавать на вкладке **Advanced (Дополнительные)**, так как при этом можно лучше понять, что происходит.

1.3. На вкладке **Advanced** выделите верхнюю категорию **LocalInfo (Локальная информация)**, где задаются параметры для информации, хранящейся на компьютере. Категории **RemoteInfo (Удаленная информация)** и **TestingServer** используются при публикации сайта на сервере и при загрузке материалов из сети. Нам они не нужны. С некоторыми из остальных категорий мы познакомимся далее.

В окне **SiteName** выделена надпись: **UnnamedSite (Сайт без имени)**. Произвольно назовите сайт.

1.4. В окне **LokalRootFolder (Локальная корневая папка)** с помощью расположенного рядом значка папки вызовите адрес своей папки. Далее для расположенной ниже папки хранения изображений **DefaultImagesFolder** создайте внутри корневой папки папку **Image** и сохраните ее. Если все сделано правильно, то в расположенной справа вкладке **LocalFiles** появляется папка для файлов вашего сайта, и в ней папка для изображений.

Все файлы сайта надо хранить в одной папке. Тогда при переносе файла на сервер или на другой носитель связи не нарушатся. По умолчанию изображения тоже хранятся вместе.

1.5. Для **LinksRelativeto:** должен быть выбран вариант **Document**. **HTTPaddress** вводить не надо, сюда пишется адрес URL сайта, опубликованного в Интернете.

Полезно также включить флажки:

- учета регистра при проверке ссылок: **Casesensitivelinks** – тогда не будет возникать проблем с регистрами ссылок;
- **Cache (Кэш)** – в программе есть внутренняя кэш-память с данными о файлах и ресурсах, ее включение ускоряет работу.

1.6. К окну настроек можно добраться другим путем: через панель **Site** → **ManageSites (Управление сайтами)** → **Edit (Редактирование)**. Помимо редактирования, в **ManageSites** есть и другие возможности. Имеется и третий путь: через вкладку **LocalFiles**, где есть окно, показывающее файловую систему и открытые сайты.

То есть, при необходимости настройки всегда можно изменить, а сам сайт удалить, продублировать, экспортировать и т.д.

2. Создание пустых страниц

Для пользователя веб-сайт всегда открывается с одной страницы. Чаще всего это начальная страница сайта. Ее также называют главной, или домашней. С нее и надо начинать.

Полезно также создать остальные страницы сайта, пусть они даже при этом будут пустыми. Это поможет избежать сложностей и путаницы в дальнейшем, ведь гиперссылка может быть создана только на существующую страницу.

2.1. В исходном окне программы нажмите кнопку **HTML**, появляется чистая HTML-страница. В режиме **Design (Проект)** страница выглядит так же, как в браузере, кнопка **Code (Код)** выводит код HTML, кнопка **Split (Разделение)** показывает и код, и вид страницы.



Нажатие одной кнопки привело к появлению нескольких строк кода, что конечно, удобнее, чем работа в Блокноте.

2.2. Созданная страница называется **Untitled-1 (Безымянная-1)**, и она не относится к какому-либо сайту. Откройте окно **File** → **SaveAs**, по умолчанию откроется папка, где будут собираться материалы вашего сайта. Назовите страницу **Index** и сохраните ее. Многие серверы сконфигурированы на имя **Index.html** как на начальную страницу, поэтому так чаще всего называют главную страницу сайта. Убедитесь, что страница присутствует в папке сайта на вкладке **LocalFiles**.

Главной странице сайта необходимо дать еще одно очень важное имя: название страницы для Сети, читаемое браузерами и поисковыми системами. Поисковые системы будут давать сведения о вашем сайте по запросам пользователей, в первую очередь руководствуясь словами, упомянутым в этом названии. Поэтому при его создании надо подумать о том, какие слова могут употребляться в запросах те, кому вы предназначаете свой сайт.

2.3. Имя для Интернета пишется в текстовом поле **Title: (Название)**, по умолчанию там стоит **UntitledDocument**. Поставьте название: **Строительный информационный портал**. Убедитесь, что в описании кода HTML это имя разместилось между тегами **<title>**.

2.4. Создайте еще одну страницу. Для этого с помощью **File** → **New** надо выйти в диалог создания объектов, и во вкладке **BlankPage (Пустая страница)** выбрать **HTML** и затем **Create (Создать)**. Назовите страницу **Service (Услуги)** и сохраните ее в папку сайта. Названия для браузера этой странице давать не надо, оставьте **UntitledDocument**. На нее выходить из сети ни к чему.

2.5. Аналогично создайте и сохраните остальные 5 страниц сайта, назовите их: **Цены – Price, Программы – Program, Выставки и семинары – Show, Литература – Book, О нас – Address**. Здесь тоже не надо давать имен для браузера.

2.6. Для всех страниц выберите одинаковый светло-зеленый фон. Для этого надо во вкладке **Properties** выбрать **PageProperties...**, в открывшемся окне раскрыть палитру **BackgroundColor** и выбрать нужный цвет. Цвет должен быть светлым, поскольку на темном фоне плохо будет читаться текст. Для создания стилевого единства все страницы сайта должны иметь одинаковый фон.

3. Импорт и правка текста

Импортировать текст в Dreamweaver очень просто: его можно перенести из Word и Excel с помощью буфера обмена. Имеются также команды специальной вставки: **Edit (Редактирование)** → **PasteSpecial**, с различными вариантами сохранения параметров форматирования исходного текста. Нажав кнопку **PastePreferences**, можно познакомиться с многочисленными установками процесса вставки данных – не только текста, но и других объектов.

3.1. Пользуясь указаниями преподавателя, откройте файл **Для Главной**, выделите весь содержащийся там текст и через буфер обмена перенесите его в рабочее окно страницы **Index**. Форматирование исходного текста сохранилось.

3.2. Аналогично перенесите текст файла **Для услуги** на страницу **Service**. Здесь форматирование исходного текста тоже в основном сохранилось, хотя есть небольшие расхождения, которые затем надо будет исправить.

В рабочем окне имеются возможности для правки и несложного форматирования текста. Эти возможности больше, чем у редактора Блокнот, но существенно меньше, чем у Word.



Однако форматирование непосредственно в рабочем окне носит вспомогательный характер: для исправления ошибок и мелких недочетов. А основное форматирование сайта лучше всего проволить с помощью таблиц и CSS– каскадных листов стилей, о чем будет сказано далее.


3.3. На обеих страницах надо удалить из текста все точки и точки с запятой (;), они на веб-странице не нужны. Для этого удобно применить контекстную замену. Откройте **Edit**→**FindandReplace (Поиск и Замена)**, в окно **Find:** внесите знак ;, окно **Replace:** оставьте пустым. После команды **ReplaceAll** все точки с запятой из текстов исчезнут. Аналогично удалите точки.

3.4. На странице **Service** удалите пустую строку перед фразой: **А также в проект «Строймаркет» входит:**. Здесь же надо бы восстановить многоуровневые списки в пункте 4, но это сделаем далее с помощью CSS.

3.5. Перенесите текст файла **Для выставки** на страницу **Show**. Здесь тоже надо убрать точки и точки с запятой. И поставить точки в указаниях времени работы выставок: **10.00** и **19.00**.

3.6. Откройте файл **price+**, найдите в его первом листе (**Strinf**) раздел **Печать** (под разделы отведен столбец **A**) и перенесите все его содержимое на страницу **Price**. Скопировался не только текст, но и поля таблиц, которые были в Excel. После этого на страницу **Bookиз Price** надо перенести наименования книг без цен на них.

Если на экране одновременно находится несколько страниц, то материал со страницы на страницу можно просто перетаскивать, выделяя нужный элемент и перенося его при

*нажатом **Ctrl**. Для одновременного показа всех страниц сайта надо нажать кнопку , расположенную в конце строки с перечислением страниц сайта (под главным меню). В исходное состояние можно придти, переведя любую страницу в полноэкранный режим.*

Сначала скопируйте заголовок таблицы, потом ее первый столбец.

3.7. Далее скопируйте из **price+** на страницу **Price** небольшие фрагменты продаваемого программного обеспечения не менее чем из трех листов, с указанием цен и примечаний. Скорее всего, поля таблиц будут отличаться по ширине, в дальнейшем их надо будет скорректировать.

3.8. На страницу **Program** скопируйте со страницы **Price** наименования программ и сведения о них, без указания цен. Должна получиться таблица с одним столбцом. Для этого копируем заголовок верхней таблицы, потом ее левый столбец, потом заголовок и левый столбец следующей таблицы и т.д.

3.9. После этого, если в **Price** для программы в правом столбце есть подробности по ее возможностям и установке, то их надо скопировать в **Program** в клеточку с названием программы. А затем командой **Shift + Enter** в этих клетках сделайте 2 строки: одну с названием программы, другую с комментарием.

*Клавиша **Enter** создает большое межстрочное расстояние, а **Shift + Enter** – меньшее. В первом случае генерируется парный тег **<P></P>**, во втором – одиночный **

3.10. В файле **price+**, лист **Strinf**, в правом верхнем углу указан адрес НТС «Строймаркет». Скопируйте его и через буфер обмена перенесите на страницу **Address**. Затем вместо НТС «Строймаркет» напишите **Наш адрес:**. Сверху напишите отдельной строкой **СХЕМА ПРОЕЗДА**, снизу: **e-mail: soft@stroika.co**.

3.11. Сохраните все страницы. Для этого надо перевести их в полноэкранный режим, и, нажимая правой кнопкой названия страниц, выбрать **Save**.

4. Работа с таблицами

HTML обладает довольно ограниченным набором средств расположения текста и графики на странице. Этих средств намного меньше, чем, например, в Microsoft Word.

Использование таблиц позволяет преодолевать эти ограничения. Текст и графику можно располагать в разных ячейках таблицы, а затем, объединяя и разделяя ячейки, решать многие задачи. При необходимости таблицы можно скрыть, сделав рамки невидимыми.

Для создания таблиц используют теги:

<Table> ... </Table> – устанавливает начало и конец таблицы;

<Caption> ... </Caption> – устанавливает заголовок таблицы;


<Tr> ... </Tr> – добавляет новую строку в таблицу;

<Th> ... </Th> – добавляет в строку ячейку заголовка;

<Td> ... </Td> – добавляет в строку обычную ячейку.

Код HTML, выводящий таблицу – это длинная последовательность указанных дескрипторов с их атрибутами. Вводить их с клавиатуры утомительно. Dreamweaver радикально упрощает эту задачу, создавая сложный код с помощью простых действий.

4.1. Раскройте страницу **Price** и сделайте ширину всех имеющихся там таблиц в 100%


экрана. Для этого надо щелкнуть по углу таблицы так, чтобы появился значок , и она вся выделится (не столбец и не строка). После этого на вкладке **Properties**, в окнах правее **W**, поставьте **100 %**.

Ширину таблицы можно задавать в пикселях и в процентах. Последний способ предпочтительней, так как при этом таблица будет всегда занимать одну и ту же долю экрана, независимо от размера и разрешения.

4.2. С таблицей удобнее работать в режиме **Макет (Layout)**. Для простоты выбора из списка вариантов рабочего стола выберите **CLASSIC** (эти варианты расположены в самой верхней строке программы, правее меню). В старых версиях только этот вариант и присутствует. В **CLASSIC** непосредственно под меню расположена вкладка **Insert**, и в ней можно выбрать **Layout**. Более наглядно таблица выглядит не в стандартном режиме, а в расширенном: **Expanded**.


В режиме макета можно легко создать новую таблицу, добавить столбцы, строки, фреймы. Но у нас другая задача: надо оставить две колонки – с названием продукта и его ценой (разумеется, при сохранении строк с заголовками). Нужно убрать лишние столбцы, а оставшиеся колонки сделать одинаковыми.

4.3. Для удаления столбца ставим курсор на любую его ячейку и в режиме правой кнопки выбираем: **Table** → **DeleteColumn**. Для удаления строки надо применять **Table** → **DeleteRow**.

4.4. Границы столбцов легко передвигаются мышкой. Размеры столбцов можно посмотреть в конце таблицы при появлении надписи **Presscontrolkeytoseeablestructure**. Перемещайте курсор по вертикали до появления этой надписи, затем нажмите кнопку мыши. По умолчанию размеры измеряются в пикселях, но лучше их перевести в проценты. Для этого щелкните по линии, указывающей всю ширину таблицы, при этом нижняя часть вкладки **Properties** должна измениться. Щелкните по кнопке , после этого можно увидеть ширину колонок в процентах. Сделайте ширину колонки с описанием товара 88%, а с ценой – 12%.


На странице **Price** имеется несколько таблиц, их надо объединить в одну. В Dreamweaver нет такой команды, а если есть, то она малодоступна. Но не обязательно искать нужную

кнопку, можно работать непосредственно с кодом! В данном случае надо найти теги закрытия предыдущей таблицы и открытия следующей и удалить их.

4.5. Перейдите в режим **Split**, где демонстрируется вид страницы и ее код. Переведите курсор в угол первой таблицы до появления значка , и щелчком выделите таблицу. В записи кода выделяется часть, относящаяся к таблице.

4.6. Выделите теги окончания таблицы `</table>` и начала следующей таблицы `<table width="100%" cellpadding="0" cellspacing="0">` и сотрите их. Таблицы должны объединиться в одну. Проведите аналогичные операции с другими таблицами.

Тот, кто разбирается в HTML, может при необходимости совершать действия, не предусмотренные в Dreamweaver.

4.7. На панели документов имеется кнопка **Preview/Debuginbrowser (Просмотр/Отладка в браузере)** , посмотрите с ее помощью, как будет выглядеть страница в Internet Explorer. Сделайте ее более привлекательной, выбрав подходящие значения **Border** (толщина рамок) и **CellPad** (отступ текста внутри ячейки).

4.8. Сделайте центральное выравнивание для столбца с ценами и для заголовков, поставив в списке **Horz** выравнивание **Center**.

4.9. На странице **Program** запланируем несколько таблиц, причем с одним столбцом. Сделайте все таблицы одинаковыми по ширине. Затем переведите размеры в проценты. Удалите лишние строки и столбцы (если они есть), объедините таблицы с общим заголовком. Поставьте толщину рамок и отступ текста такие же, как в **Price**.


4.10. Если под заголовком находится только один элемент списка, то его надо превратить в таблицу. Для этого на вкладке **Layout** включите пиктограмму **Table** и создайте под заголовком таблицу с одной строкой, одним столбцом, 100% шириной, такими же рамками и отступом текста, как и в остальных. После этого через буфер обмена перенесите туда содержимое элемента, а пустую строку удалите.

4.11. Сделайте для заголовков таблиц центральное выравнивание. Для этого поставьте курсор на строку и в режиме правой кнопки примените **Align** → **Center**. Проверьте страницу в Internet Explorer и устранили недочеты.

4.12. Аналогично **Program**, с такими же параметрами, отформатируйте страницу **Book**. Здесь должна быть одна таблица с одним заголовком и двумя подзаголовками. Проверьте страницу в браузере.

4.13. Вызовите страницу **Index** текстом, импортированным из Word. Легко убедиться, что возможности форматирования текста во вкладке **Properties** ограничены, и в основном текст изменяется целиком. А вот маркированные списки какому-либо изменению не поддаются. Страницу надо разбить на невидимые ячейки таблицы, и изменять как единое целое уже текст в ячейках. На вкладке **Layout** включите **Table** и создайте таблицу с 10-ю строками (**Rows**), одним столбцом (**Columns**), 100% шириной и **CellPadding** (отступ текста) таким же, как на странице **Price** (см. 4.7.). Рамки не должно быть, то есть клетка **Borderthicknes** должна остаться пустой.

4.14. С помощью буфера обмена распределите текст следующим образом: каждая строка немаркированного текста должна иметь свою ячейку, а маркированные списки надо помещать в одну ячейку. Может потребоваться вручную уменьшать высоту строк таблицы.

4.15. Кнопкой  **TextIntent** сместите маркированные списки вправо. Уберите несколько пробелов между точкой маркера и текстом. Проверьте страницу в браузере.

4.16. Аналогично **Index**, разбейте на ячейки и отформатируйте текст на странице **Service**, **Show** и **Address** так, как вам покажется оптимальным. Проверяйте страницы в браузере.

4.17. На странице **Service** имеются многоуровневые маркированные списки. Для повышения уровня списка надо всего лишь применить к соответствующим строкам кнопку **TextIndent** на вкладке **Properties**. При этом сами маркеры становятся другими.

4.18. В этих четырех страницах остаются пустые строки. Их нужно удалить из записи кода, стерев все, что располагается между **</table>** и **</body>**.

4.19. Далее надо выделить заголовки. На всех страницах к заголовкам примените уровни **Heading2** и **Heading3** (окно **Format**). Пояснения в тексте лучше выделять курсивом, где-то можно применить и жирный шрифт. Не забывайте о том, что с клавишей **Ctrl** можно выделять несколько ячеек, расположенных в разных местах.

5. Создание и применение стилей CSS


Каскадные листы стилей CSS (Cascading Style Sheets) являются одним из самых популярных средств дизайна Веб-страниц. Трудности освоения CSS многократно окупаются преимуществами, которые они предоставляют для управления любыми элементами страницы.

Для каждого вида текста (заголовки, подзаголовки, основной текст и пр.) создается свой стиль с определенными параметрами (гарнитура, начертание, размер, выравнивание и др.). После включения в листы стилей этот вид текста выводится единообразно. При создании внутреннего стиля он поддерживается на текущей странице, внешний стиль распространяется на все страницы. Можно также стандартизовать параметры иллюстраций.

5.1. Команды, относящиеся к созданию, представлению, изменению стилей сосредоточены во вкладке **CSSSTYLES**, раскройте ее. Вкладка делится на две части: **All** и **Current** (для всех страниц и для текущей), оставьте **All**. В верхнем окошке **AllRulers** (**Все правила**) перечислены созданные стили. Пока создан только один стиль, управляющий цветом фона. Ниже, в окне **Properties** (**Свойства**) приводятся свойства выделенного стиля.

В самом низу расположена панель с кнопками управления. Левые кнопки могут показать свойства стиля с подробностями и без. Если нажать кнопку **Show category view**, то можно убедиться, что общее число возможных параметров для различных стилей велико. Правые кнопки открывают окна присоединения, создания, редактирования стиля и команду его удаления.

5.3. Создадим стиль, который поменяет шрифт основного текста на всех наших страницах. Откройте страницу, где есть маркированные списки, например, **Index**.

Создание стиля начинается с кнопки **New CSS Rule** (**Новое правило CSS**) , она открывает окно, задающее базовые свойства стиля. Для создания внешнего стиля в окне **Rule Definition** (**Определение правила**) поставьте **New Style Sheet File** (**Новый файл таблицы стилей**).

5.4. Далее в окне **Selector Type** надо выбрать тип правила. Имеются следующие типы:

- **Class**– стиль может применяться к любым элементам **HTML**;
- **ID**– применяется только к одному элементу **HTML**;
- **Tag**– изменяет определенный элемент **HTML**;
- **Compound**– смешанный тип стиля, он используется для гиперссылок.

Наш стиль должен применяться ко всем элементам, то есть, выбираем **Class**.

После этого в окне **Selectorname** надо дать стилю имя. Имя может быть произвольным, но у стиля типа **Class** оно должно начинаться с точки. Пробел не допустим, вместо пробела ставится знак **_**. Назовем наш стиль **.new_font1**.


5.5. Следующим включается окно, предлагающее **SaveStyleSheetFileAs (Сохранить файл листов стиля как)**. Надо создать файл, в котором можно будет хранить этот стиль и другие, однотипные ему. Дайте файлу имя, например, **.new_fonts.css**, после чего сохраните его в папку сайта. Имя файла появляется на вкладках **FILES** и **CSSCTYLES**.

5.6. На следующей стадии включается окно определения правил стиля: **CSSRuledefinitionfor .new_font1**. В окне содержится много вкладок, и в каждой вкладке свой набор параметров. В данном случае определим только два параметра шрифта на вкладке **Type: Fontsize (Размер шрифта) → 14 px (пикселей)** и **Font-family (Семейство шрифта) → ThrebuchetMS, Arial, Helvetica, sans-serif**. Выбирается не конкретный шрифт, а семейство шрифтов. При отсутствии первого шрифта у посетителя сайта, включается второй и т.д. Sans-serif означает «лишенный засечек» (шрифты serif имеют засечки на концевых и угловых элементах букв).


Остальные, не определенные параметры в ходе применения стиля не меняются. После определения стиль и его правила можно посмотреть, найдя и выделив стиль в **new_fonts.css**.

5.7. Пользоваться созданным стилем очень просто: надо выделить ячейки, к которым требуется применить данный стиль, а затем выбрать его в **Properties**, в списке **Class**. На странице **Index** примените созданный стиль ко всему тексту, имеющему **FormatParagraf**.

5.8. Однако на других страницах созданный стиль отсутствует, его требуется присоединять к страницам. Раскройте страницу **Service.html** и нажмите на вкладке

CSSCTYLES кнопку  **AttachStyleSheet (Присоединить лист стилей)**. Открывается окно присоединения, в котором по умолчанию будет созданный стилизованный лист, других просто нет. В режиме **Link** присоедините лист стилей к странице (если в окне **File/URL** нет надписи **new_fonts.css**, то ее надо туда поставить). Здесь примените стиль к тексту формата **None**.

5.9. Присоедините лист стилей ко всем остальным страницам и примените стиль ко всему мелкому тексту, форматов **Paragraf, None, Heading 4**. Несмотря на один и тот же стиль, текст выглядит по-разному в зависимости от формата и других параметров вкладки **Properties**.

5.10. Дополним лист **.new_fonts** стилем для заголовков. Нажимаем кнопку , и в окне **SelectorType** создаем стиль **.new_font2** (больше ничего не меняем). В окне определения правил надо оставить то же семейство шрифта, а в списке **Font-style** выбрать **oblique**. Такое начертание в Word отсутствует, это наклонный, но не курсивный шрифт.

Применим созданный стиль ко всем заголовкам на всех страницах. Теперь уже лист стилей присоединен, и новый стиль автоматически создается на всех страницах. При различии в параметрах **Properties** заголовки тоже не одинаковы.

5.11. Созданный стиль всегда можно отредактировать, причем различными путями. Один из путей: щелкните 2 раза на названии стиля **.new_font2**, откроется окно определения его свойств. Измените цвет шрифта с черного на другой, например, на какой-нибудь красно-коричневый оттенок. Цвет всех заголовков на всех страницах поменялся!

Изменение параметров созданного стиля приводит к автоматическому изменению этих параметров на всех объектах, во всех страницах, где использован этот стиль.

5.12. Теперь создадим стиль главного заголовка страниц **.new_font3** с шрифтом **Font-family** → “**ArialBlack**”, **Gadget, sans-serif** и центральным выравниванием **Text-align** → **center** (на вкладке **Block**). Цвет текста можно сделать каким-либо оттенком синего.

5.13. На страницу **Program** надо добавить заголовок.. Создайте сверху еще одну строку, наберите: **Программное обеспечение** и примените к этому заголовку **Format: Heading 2** и стиль **.new_font3**. Аналогично добавьте заголовок **Выставки и семинары** на страницу **Show**.

5.14. Примените **.new_font3** (все заголовки должны иметь **Format** → **Heading 2**):

- на странице **Service** к фразе **НТС «Строймаркет»** предлагает следующие виды услуг;
- на **Book** к фразе **Книги и справочные издания НТС «Строймаркет»**;
- на **Index** надо сверху написать: **НТС «Строймаркет» предлагает Вашему вниманию**;
- на **Address** написать: **О нас**;
- на **Price** написать: **Прайс-лист «Строймаркет»**;
- на **Program** написать: **Программы**;
- на **Show** написать: **Выставки и семинары**.

Если где-то первой строки нет, ее надо создать с помощью **Enter**.

5.15. После применения текстовых стилей просмотрите страницы в браузере, исправьте ошибки, внесите необходимые коррективы.

6. Добавление иллюстраций

С помощью **Dreamweaver** можно легко размещать изображения, осуществлять обрезку, масштабирование, обтекание текстом. Программа может также создавать карты изображений, фоновую графику, осуществлять базовые операции редактирования графики. Но все же иллюстрации лучше заранее подготовить в графических редакторах.

При вставке изображения на страницу создается ссылка на его файл. Если файл переместить, то на месте иллюстрации останется лишь ярлык указывающий на то, что нужный файл не найден. Поэтому, если сайт не отличается особой сложностью, лучше все изображения помещать в отдельную папку **Image**, расположенную в корневой папке сайта. Это лучший и простейший способ обеспечить целостность связей.

Эффективный способ размещения – помещать изображение в отдельную ячейку таблицы.

6.1. Раскройте страницу **Address**, выделите нижние строки таблицы (за исключением заголовка), после чего нажмите правую кнопку и объедините ячейки командой **Table** → **Merge Cells (Объединить ячейки)**. После этого в режиме **Layout** нажмите пиктограмму **Insert Column to the Left (Вставить колонку слева)**.

6.2. Мы подготовили место под иллюстрацию, теперь подготовим саму иллюстрацию. Найдите файл **карта.jpg**, загрузите его в **Adobe Photoshop**. Рисунки для Интернета должны иметь оптимальные размеры. С целью оптимизации раскройте окно **Изображение** → **Размер изображения** и уменьшите ширину карты до **300 пикселей** (флажок **Сохранить пропорции** должен быть включен).

6.3. Далее примените к файлу команду **Файл** → **Сохранить для Web**. В открывшемся окне параметры не меняйте, по умолчанию файл должен сохраняться в формате **GIF** (если формат другой, поставьте **GIF**). Сохраните файл в **Image**, но дайте файлу имя латинским шрифтом: **chart.gif**.

Вся графика для сайтов должна быть оптимизирована по размеру и сохранена в режиме «для Web». Если не сделать последнюю операцию, то картинка может не открыться. Кириллицу для именования файлов лучше не употреблять.

6.4. Для вставки графики надо в режиме **Common** найти и включить пиктограмму **Image** (в виде дерева), по умолчанию она первая в списке **Images**. Откроется окно, с помощью которого можно легко вставить карту в нужную ячейку таблицы, ниже заголовка. В окне **Alternatetext** напишите по русски **Карта** для поисковой системы. В дальнейшем также пишете альтернативный текст иллюстраций.

6.5. Просмотрите страницу в браузере и исправьте недочеты. В частности, можно поместить надпись **Схема проезда** в другую ячейку, непосредственно над самой схемой. Можно органично совместить текст с изображением и в одной ячейке таблицы.

6.6. Раскройте страницу **Индекси** поместите курсор в начало текста **Вероссийское еженедельное издание «Стройка» ...** Вставьте на это место файл **stroyka.gif**. Изображение можно вставить простым переносом его файла из вкладки **FILES**. А теперь в контекстном меню мыши, в списке **Align (Выравнивание)** поставьте **Right (Справа)**. Текст будет обтекать изображение, не примыкая к нему.

6.7. Изображение лучше уменьшить. Для пропорционального уменьшения нужно перемещать правый нижний угол точно по диагонали. Это сделать довольно сложно, но диагональ можно построить, приложив к картинке листок бумаги. Уменьшите логотип газеты примерно в 2 раза.

6.8. Ниже, в ячейку с текстом: **Серия справочников «Строитель»...** поместите переносом файла из вкладки **FILES** два рисунка, слева и справа: один из файлов серии «Строитель» и файл **z_sk_02.jpg** «Современный коттедж». Если текст к рисункам будет примыкать вплотную, нужно поставить отступ в 10 пикселей: **HSpace 10**.

6.9. На странице **Book** вставьте по два справочника **Строитель** и **Застройщик** так, как показано на рисунке. Как это сделать?

Сначала надо разбить все строки на две колонки, пиктограммой **InsertColumnToTheRight (Ввести колонку справа)**. Потом объединить ячейки заголовков, а в основном тексте сделать справа две большие ячейки.

6.10. В эти ячейки надо перенести по две картинки, в верхнюю из серии **Строитель**, в нижнюю – **Застройщик**. Между картинками надо поставить зазор в 10 пикселей (**HSpace 10**). Чтобы уменьшить размеры строк до оптимального, надо картинки несколько раз переносить в различные ячейки (и, разумеется, обратно). Поставить для картинки **Align (Выравнивание) → Middle**, а для всей линейки **Align → Center**.

Далее в верхнюю часть страницы **Index** вставим логотип **Строймаркет**.

6.11. Включите программу **Photoshop**, загрузите файл **Строймаркет.tif**, сохраните его для Web по имени **Stroumarket.gif** в папку **Image**.

6.12. Раскройте пиктограмму **Table** (вкладка **Common**) и создайте в верхней части страницы **Index** таблицу с одной строкой и одним столбцом. Параметр **Tablewidthth (Ширина таблицы)** поставьте **100%** в окне **Borderthickness (Толщина рамки)** значений быть не должно; т.е. рамка должна быть невидимой.

6.13. Перенесите в эту таблицу рисунок **Stroumarket.gif**. Щелкните по части ячейки, не занятой рисунком, затем поставьте центральное выравнивание по горизонтали: **HorzCenter**.



6.13. Далее надо сделать фоновый цвет ячейки таким же, как и фон логотипа. Во вкладке **Properties** есть квадратное окошко **Bg**, задающее фоновый цвет. При удержании на нем кнопки мыши появляется пипетка. Перенесите пипетку на фон логотипа и отпустите кнопку, цвет фона ячейки приобретет тот цвет, на котором находится пипетка.

6.14. Аналогично поставьте логотип на все остальные страницы сайта. Это можно сделать переносом таблицы с логотипом через буфер обмена.


Оживить и украсить сайт может фоновое изображение, как правило его делают одинаковым для всех страниц. Если в качестве фонового берется небольшое изображение, то оно повторяется по горизонтали и вертикали, заполняя всю страницу. Тогда на скорость загрузки наличие картинки влияет слабо.

Фоновое изображение не должно вызывать затруднений при чтении текста. Чаще всего его делают светлым и малоконтрастным.

6.15. Для начала вставим новый фон на страницу **Index**. При включении на вкладке **Properties** кнопки **Page properties** открывается диалоговое окно, задающее свойства страницы. Нам требуется **Category: Appearance (Внешний вид)**. Здесь в окне **BackgroundImage (Фоновое изображение)** кнопкой **Browse** из папки **Images** надо поставить файл **YELLOUBRK.gif**.

6.16. Но теперь надо восстановить исходный фон таблицы, имеющейся на странице. Для этого надо выделить все строки и столбцы таблицы, после чего рядом с окном **Bg** поставить код цвета фона. Если вы его не помните, то можно посмотреть код HTML любой другой страницы: код фонового цвета является атрибутом тега **body**.

6.17. Поставьте фоновое изображения на все страницы сайта, а фоновый цвет примените ко всем его таблицам. Просмотрите страницы в браузере. Если вместо цвета фона таблица окрашивается в черный цвет, то надо сначала поставить любой другой цвет строк таблицы, и проверить в браузере. Если черный цвет исчез, то снова установите цвет фона. С помощью Dreamweaver на веб-страницы можно также вставить флэш-анимацию. Вставим ее правее логотипа.

6.18. На любой странице выделим ячейку с логотипом и кнопкой  **Splittsellsintorowsandcolumns (Разделить ячейки на строки и столбцы)** разобьем ячейку на два столбца. Правый столбец очень мал по ширине, но перенесите в него логотип, и тогда маленьким станет уже левый столбец, немного увеличьте его.

6.19. Далее во вкладке **Common** найдите пиктограмму **SWF**, и с ее помощью вставьте в левый столбец флэш-баннер **Klip**. Появляется серое окно, в котором будет демонстрироваться баннер. Оно очень велико, уменьшите его размеры примерно в 3 раза (поставьте **W 180, H 130**). Уменьшите также высоту таблицы и ширину ячейки до величины, когда баннер целиком займет ячейку, поставьте левое выравнивание баннера в ячейке.

6.20. Для демонстрации баннера щелкните по кнопке **Play** во вкладке **Property**. Проверьте работу баннера в браузере. Internet Explorer может потребовать новый Flash-плеер. Или заблокировать активное содержимое, о чем сообщает надпись, возникающая над сайтом. Для разблокировки надо щелкнуть по этой надписи и далее по **Разблокировать активное содержимое...**

6.21. Сохраните баннер в буфере обмена, чтобы в дальнейшем не изменять его размеры. А далее на каждой странице создавайте соответствующую ячейку и вставляйте туда баннер.

7. Создание навигационной панели и гиперссылок

Навигационные панели представляют собой мгновенно узнаваемые и понятные в использовании инструменты для перехода по сайту. На странице они обычно



располагаются горизонтально, под логотипом или вертикально, в виде крайнего левого столбца.

Создадим горизонтальную панель навигации на всех веб-страницах. Эту панель можно сделать в виде обычной таблицы.

7.1. Вызовите на экран любую страницу поставьте курсор после логотипа, и с помощью пиктограммы **Table** создайте таблицу, состоящую из одной строки и 7 столбцов, шириной 100%. Параметры, устанавливающие рамку таблицы и рамки ячеек, сделайте равными 0.

7.2. Выделите все ячейки поставьте в них фон основного текста (также как в **6.13**, только цвет зеленый) и центральное выравнивание.

7.3. В каждую ячейку вставьте изображения пунктов меню: **Index, Service, Program, Price, Address, Book, Show**. При вставке размеры ячеек могут меняться, на это не надо обращать внимания. В браузере при увеличении окна страницы размеры кнопок меню остаются неизменными, остающееся пространство заполняется фоном.

7.4. Создадим гиперссылки, в Dreamweaver это сделать очень просто. В **Properties** правее окна **Links** расположена кнопка **Browse for File**. Через нее находим нужную страницу и вставляем ее в **Links**. В иллюстрации-гиперссылке появляется рамка.

7.5. Выделите всю панель навигации и через буфер обмена перенесите ее на все страницы сайта. Проверьте в браузере действие панелей навигации.

Можно создать гиперссылку не на всю страницу, а на **анкер** – заданное место на странице, куда приходит гиперссылка. Анкер можно установить как на другую страницу, так и на ту же, где расположена сама гиперссылка. Анкеры полезны при просмотре длинных страниц, а также для указания на какую-то конкретную информацию.

Создадим несколько ссылок со страницы **Index** на определенные места страницы **Show**. Для этого нужно сначала создать анкеры на странице **Show**.

7.6. Анкер – это точка в определенном месте страницы. Раскройте **Show**, поставьте курсор слева от слова «**Стройка**», и нажмите пиктограмму **Named Anchor** на вкладке **Common**. В открывшемся окошке надо дать анкеру имя, например **stroika** (оно должно быть без пробелов и спецсимволов, чем проще – тем лучше). В выбранном месте появляется пиктограмма якоря (**Anchor** по-английски якорь). Создайте такие же анкеры перед словами «**Домотехника**» и **Научно-практические**.

7.7. Теперь надо создать ссылки на анкеры. Раскройте **Index**, в самом конце найдите слово «**Стройка**», выделите его. Затем надо в окошке **Links** набрать точный адрес того места, куда будет направлена ссылка: **Show.html#stroika**. Знак **#** указывает на то, что ссылка будет осуществляться на выделенное место страницы – анкер. Проверьте в браузере, будет ли работать ссылка.

7.8. Аналогично создайте ссылки со слов «**Домотехника**» и **Специализированные семинары** на соответствующие анкеры.

7.9. Анкером может служить и иллюстрация. Раскроем страницу **Address** и поставим анкер на схему проезда, лучше его поставить после схемы. Теперь создадим на него ссылку со слова **Проезд**: относящемуся к выставке «**Стройка**». При проверке в браузере возможна блокировка активного содержимого.

8. Создание интерактивных элементов

В Dreamweaver можно создавать динамические элементы страницы, они способны изменяться в зависимости от действий посетителя сайта или от ситуации на странице. Для создания изменений статический HTML не подходит, используются средства динамического языка DHTML. Чаще всего для реализации динамических эффектов


применяются сценарии на языке JavaScript. Средства Dreamweaver позволяют генерировать сценарии без знания JavaScript.


DHTML имеет и недостатки. Браузеры иногда по-разному интерпретируют динамическое содержимое. Устаревшие браузеры могут его совсем не воспроизвести. Необходимо тестировать активные элементы в наиболее распространенных браузерах. Другой недостаток состоит в том, что создавать динамическое содержимое намного труднее, чем статическое.


В качестве примера для главной страницы создадим вертикальную динамическую навигационную панель. Здесь для кнопок можно будет установить до четырех положений (не обязательно устанавливать все):

- кнопка не нажата – неактивное положение;
- указатель мыши наведен – подготовительное положение;
- кнопка нажата – активное положение;
- кнопка нажата и указатель мыши наведен – подготовительное положение для кнопки в нажатом состоянии.

Сначала надо создать место, на котором будет располагаться навигационная панель. Для вертикальной панели требуется создать еще один столбец слева. Но предварительно надо поработать с ячейками заголовков. Иначе все заголовки будут распространяться на навигационную панель.

8.1. На странице **Index** поставьте курсор на заголовок и в **Properties** нажмите кнопку , в открывшемся окне задайте 2 столбца.

8.2. Создайте новый столбец таблицы, проще всего это сделать кнопкой **Insertcolumnntothelleft** на вкладке **Layout**. После чего выделите все ячейки нового столбца и объедините их кнопкой  **Mergesselectedcellusingspan**.

8.3. Объединенную ячейку нужно сделать прозрачной. Для этого надо раскрыть окно фоновых цветов **Bgi**, не отпуская мышки, выбрать команду  **DefaultColor**.

8.4. Доступ к окну создания кнопки навигационной панели осуществляется из меню командой **Insert** → **ImageObjects** → **NavigationBar**. Не обязательно заполнять все поля, заполним то, что нам необходимо. Дадим название элементу (кнопке) в списке **Elementname:Index**.

8.5. Далее поставим адреса рисунков кнопки, которые будут появляться, когда кнопка не нажата (**UpImage**) и когда курсор наведен на кнопку (**OverImage**). Для первого случая возьмите **Index**, для второго – **Index2**.


8.6. Обязательно надо заполнить поле **Whenclicked, GotoURL (При нажатии перейти на URL-адрес)**. Это должен быть адрес страницы **Index**. Загрузка будет осуществляться в **MainWindow**, поскольку другого окна и нет.

8.7. Полезно поставить флажок **PreloadImages** – при этом изображения панели будут загружены в браузер еще до их использования, что уменьшит скорость выполнения команд. В поле **Insert** надо поставить **Vertically**, поскольку создается вертикальное меню.



8.8. Остальные поля можно не заполнять, или лучше не заполнять. Посмотрите созданное в браузере, проверьте изменение рисунка кнопки при наведении курсора.

В Internet Explorer браузер может блокировать активные элементы, надо будет его разблокировать, как описано ранее.



8.9. Создадим ссылки на остальные страницы сайта. Для этого надо включить **Modify** → **NavigationBar**, откроется то же окно создания навигационной панели. Кнопкой  можно создать новый элемент панели, **Unnamed1**.

8.10. Переименуйте его в одну из страниц сайта, например, в **Service**, поставьте рисунки и гиперссылку, соответствующие этой странице. Нажав **ОК** вы увидите второй элемент навигации: **Услуги**.

8.11. Далее в окне можно создать элементы-кнопки для остальных 5 страниц. При этом не обязательно каждый раз закрывать окно **ModifyNavigationBar**, все можно создать за один сеанс. Кнопками  и  можно менять порядок расположения кнопок в списке **Navbarelements**, таким же он будет и на веб-странице.

9. Финальные операции

Гиперссылки можно создавать и на другие сайты. В данном сайте удобным местом для этого является пространство ниже интерактивной панели, в том же столбце.

9.1. Найдите среди исходных файлов **Rambler'sTop100** и перенесите его в папку **Image**. Это ссылка на главный каталог поисковой системы Rambler. Затем из вкладки **Files** скопируйте ее в **Index** на строку, расположенную ниже интерактивной панели.

9.2. Сделайте гиперссылку более заметной: создайте ниже таблицу из одной ячейки с рамкой, перенесите туда ссылку, сделайте центральное выравнивание и жирное начертание, светлый фон в ячейке. В браузере щелчок по этой ссылке (при работающем Интернете) будет вызывать каталог Rambler.

9.3. Просмотрите в браузере все созданные страницы. Прежде всего, нужно позаботиться о рациональном использовании экрана. Верхняя часть страниц, доступная пользователю без прокрутки, должна быть максимально заполнена информацией, а не фоном. По возможности сдвиньте вверх таблицы и другие элементы.

С главной страницы удалите горизонтальное меню – оно лишнее.

9.4. Все надписи должны быть не на фоне кирпичной кладки, а на светлом фоне, контрастном по отношению к тексту. Для этого надо создавать новые таблицы, или строки таблиц, и туда перемещать заголовки, воспроизводя нужные стили, фон и другие параметры.

9.5. Надо позаботиться об эстетичности. Если на каких-то страницах какие-то элементы вам не нравятся, а на других они получились удачно, то внимательно посмотрите на все характеристики «хороших» страниц и в точности воспроизведите их на неудачных страницах. Если это не помогает, то придется что-то удалить и сделать заново. Часто требуется:

- поэкспериментировать со стилями текста и цветами фона, чтобы добиться приемлемых сочетаний;
- в таблицах в окошке Cell Space поставить 0, тогда исчезнут просветы между ячейками;
- чтобы анимация везде была на белом фоне, в списке **Wmode** должно стоять **Window**.

Практическое занятие №3

Создание Web-сайта в конструкторе WordPress

Система управления сайтом или конструктор WordPress является популярным простым и эффективным средством создания и размещения информации в Интернете. Здесь можно создавать различные Web-материалы: сайт, блог, портфолио творческих работ. При этом



работа с WordPress не требует специальных знаний. В конструкторе имеется множество готовых шаблонов, в которых можно менять текст, иллюстрации, видео и пр. на свое содержание.

В современных условиях создать насыщенной информацией, красочный сайт недостаточно. Нужно сделать его популярным, чтобы к ним ознакомилось максимальное число потенциальных посетителей, для которых предназначена размещенная информация. Для этого проводят поисковую оптимизацию Web-страниц. То есть, делают так, чтобы созданные Web-страницы оказывались в верхней части поисковых запросов по ключевым словам (лучше всего на первой экранной странице списка выдачи). Полезно также иметь гиперссылки с ресурсов, которые могут просматривать потенциальные посетители.

WordPress обладает набором необходимых средств поисковой оптимизации, благодаря которым можно повысить популярность создаваемого материала. Причем эти средства применить довольно просто, используя команды интерфейса. Нажимая соответствующие кнопки, и не вдаваясь в подробности программирования тех или иных действий.

Существуют бесплатная и платные версии этого конструктора. В платных версиях существенно больше возможностей, они стоят несколько тысяч или около 15 тысяч рублей в год. Ну а здесь рассмотрена бесплатная версия, доступная всем, кто может подключиться к Интернету. Как будет показано далее, её возможности ограничены.

Работать с WordPress можно в любой установленной на компьютере операционной системе: Windows, MacOS, Linux и др. А также на любом компьютере: ПК, планшете, смартфоне. Главное, чтобы был нормальный браузер и хорошая скорость доступа в Сеть.

Однако это не значит, что в любой программно-аппаратной среде конструктор будет работать одинаково. Различия есть, о чем будет сказано далее.

WordPress популярен не только у начинающих пользователей, которые не имеют специальных знаний или не хотят вникать в тонкости Web-программирования. Те, кто знаком с HTML, CSS, JavaScript, PHP и прочими средствами, могут добиться большей гибкости и свободы в оформлении сайта, редактируя готовый программный код.

Для специалиста удобнее создавать сайты не с нуля, а на базе имеющихся скриптов и стилей, используя шаблоны WordPress как исходную точку своей работы. Полезным также может оказаться использование многочисленных плагинов, которые можно найти не только в на сервере WordPress, но и в Интернете.

В данной практической работе предлагается создать с помощью WordPress несложный сайт-визитку компании. Это сайт компании **Строймаркет**, оказывающей информационно-технические услуги в области строительства. Реально такой фирмы не существует, но похожая фирма есть, и ее сайт взят за образец.

Образец выполнения работы (страницы сайта) может быть выдан студенту в виде распечатки. Если распечатки нет, то можно ознакомиться с образцом по ссылке **uchebnysite.wordpress.com**.

1. Регистрация на wordpress.com, установка темы

WordPress не является программой, установленной на компьютере. Все файлы этой системы расположены на ресурсе **wordpress.com**, и туда же закачивается то, что создано пользователем. Хотя можно сохранить результаты работы на компьютере в виде файла формата .htm (это будет показано далее).

1.1. Откройте страницу **ru.wordpress.com**, выберите вариант **Начать с сайта**, появляется окно **Выберите тему**. Тему пока выбирать не надо, пропускаем. Если заставка другая, переходите к **1.3**.

1.2. Произвольно выберите имя файла для своего сайта, оно должно быть на строке **Бесплатно**. Кликаем **Выбрать**, потом снова: **Начните с бесплатно**. В следующем окне укажите адрес своей электронной почты, придумайте себе имя и пароль.

1.3. Откройте страницу **ru.wordpress.com**, ознакомьтесь с ее содержанием, затем нажмите **Приступайте**. Открывается страница на английском языке, здесь напишите адрес своей электронной почты, придумайте себе имя и пароль. В последней строке, с адресом блога должно стоять: **.wordpress.com.free**.

*Вы создали аккаунт на ресурсе **wordpress.com**. В вашей электронной почте появилась запись "**Добро пожаловать на Wordpress**". Через эту запись можно всегда попасть на свой сайт.*

Описанная процедура регистрации действует на ПК с установленной Windows 10. Для другой программно-аппаратной среды возможна другая последовательность команд, близкая по содержанию.

1.4. Имя и пароль запомните, а лучше запишите. Нажмите **Посмотреть свой сайт**. По умолчанию загружается тема **TwentySixteen**. Это довольно удобная тема, но в бесплатном варианте ее возможности ограничены.

1.5. Нажмите **Мой сайт → Темы**, найдите тему **Penscratch 2** загрузите ее. После работы конфигуратора эта тема должна появиться на экране.

2. Создание главной страницы


2.1. Для начала проведем загрузку необходимых материалов в отведенное вам хранилище. Нажмите **Посмотреть сайт**, затем **Медиафайлы → Изображение → Добавить**.

2.2. Пользуясь указаниями преподавателя, найдите на компьютере (не в Сети) папку **WordPress → Иллюстрации** раскройте папку, выделите все иллюстрации и заполните ими хранилище, нажав **Открыть**.

2.3. Приступим к редактированию шаблона. Откройте меню **Мой сайт**, и в строке **Темы** нажмите кнопку **Настроить**. После конфигурирования нажимаем **Свойства сайта**.

2.4. В открывшемся окне произвольно заполняем поля: **Название сайта**, **Краткое описание**. Можете написать любое название и описание (разумеется, руководствуясь здравым смыслом). Далее выбираем из библиотеки **Строймаркет** в качестве логотипа, и **Эмблема** в качестве иконки.

2.5. Сохраним и опубликуем созданное. В верхней части колонки настройки имеется

синее окошко, действий с созданным материалом. Нажмите кнопку , откроется меню возможных действий. Выберите и нажмите сначала **Сохранить**, потом **Опубликовать**.

Обратите внимание, что можно всегда отменить что-то неверно сделанное, запланировать изменения, поделиться ими.

2.6. Сделаем рамку в виде фона. Нажмите **Цвета и фоновые изображения → Выбрать изображение**. Зайдите в хранилище, выберите там рисунок кирпичной кладки.

2.7. Теперь добавим текст. На странице нажмите **Edit (Править)**, открывается окно **Редактировать запись**. В папке **WordPress** (на компьютере) раскройте файл **Для главной**, и перенесите через буфер обмена весь текст в нижнюю часть окна, ниже панели с командами редактирования.

2.8. С помощью имеющихся команд сделайте жирный курсив и красный цвет шрифта там, где это сделано в образце, нажмите: **Обновить**.

2.9. Текст слишком крупный, не очень хорошо читается. Снова выйдите в режим правки выделите весь текст командой **CtrlA** и в списке **Абзац** выберите **Отформатированный**. Обновите страницу, текст станет мельче и лучше по дизайну.

В верхней части страницы есть надпись: **Оставьте комментарий**. В старых версиях пишется не комментарий, а **цитата**.

Комментарий (цитата) – это небольшое текстовое описание, служащее для привлечения внимания пользователя. Если текст комментария заинтересовал пользователя, то он кликает на заголовок поста и попадает внутрь для изучения материала.

2.10. Создайте комментарий. Нажмите **Edit (Править)**, в окне **Редактировать запись** → **Дополнительные параметры** → **Отрывок** пишем: **НТС «Строймаркет» предлагает Вам справочники и программное обеспечение**.

2.10. Интересно, что созданную страницу можно отображать в полном и сокращенном видах: **Полный текст** и **Цитата (комментарий)**. Выбор одного из вариантов можно сделать, нажав в строке **Темы** кнопку **Настроить**, и далее **Параметры содержимого**.

2.11. Ознакомьтесь с видом страницы в режимах **Полный текст** и **Цитата из записи**. Сравните полученную страницу с образцом. Если замечаний нет, то сохраните и опубликуйте созданную страницу, как описано в **2.5**.

В опубликованной странице полный текст из цитаты можно получить, щелкнув по дате ее создания.

3. Создание других страниц

У нас имеется главная страница, заполненная содержанием. В созданном сайте есть ещё две страницы: **О нас** и **Контакты**, они созданы программой автоматически.

3.1. Добавим содержание в страницу **О нас**. Нажмите **Мой сайт** → **Страницы сайта** → **О нас**, сотрите имеющийся текст. В папке **WordPress** (на компьютере) раскройте файл **Для О нас**, и пернесите весь имеющийся там текст на страницу ниже панели с командами.

3.2. Внизу страницы нажмите **Править**, и в открывшемся окне поставьте жирное и курсивное начертания, измените цвет текста так, как это сделано в образце.

3.3. Здесь тоже надо выделять текст и в списке **Абзац** выбирать строку **Отформатированный**. Но эту операцию лучше проводить не целиком, а отдельно по спискам. А потом надо будет вручную выравнивать маркированные списки, удаляя лишние пробелы (чтобы было как в образце).

Обратите внимание, что главной страницей у нас является блог, которого нет в меню. В блог можно попасть, нажав на логотип: **Строймаркет** или на название сайта: **Работа 3№**. Теперь заполним страницу **Контакты**.

3.4. По умолчанию уже имеется страница **Контакты**. Там есть форма, но ее практически невозможно заполнить. Удалим эту страницу командой **Отправить в корзину**, которая расположена в правом нижнем углу экрана, без возможности восстановления.

3.5. Создадим новую страницу командой **Мой сайт** → **Добавить** на строке **Страницы сайта**. В поле **Заголовок** напишите: **Контакты**, а под окошком редактирования введите информацию из файла **Контакты**, который находится в папке **WordPress**.

3.6. Сделаем адреса почты, сайта и телефоны гиперссылками. Для этого нажмите вкладку **HTML**. Выделите адрес почты и нажмите команды **link** → **Добавить ссылку**. То же сделайте с адресом сайта и телефонами.

3.7. Добавим в контакты карту с проездом к компании. Загрузите новую вкладку в браузере **Мой сайт** → **Медиафайлы**, выберите рисунок с картой проезда. В контекстном

меню мыши нажмите **Копировать**. Вернитесь в окно редактирования страницы **Контакты** и вставьте эту картинку ниже текста.

3.8. Выделите текст контактов и примените к нему команду **Абзац → Отформатированный**. Обновите страницу.

3.9. Так же, как написано в п. 3.4 и 3.5, создайте ещё одну страницу и назовите ее **Цены**. Здесь возникает одна трудность: ассортимент предлагаемых товаров и их цены обычно пишут в виде таблицы. Но в выбранной нами теме нет готовых инструментов для создания таблиц! **Однако есть возможность записи кода HTML.** Можно создать все, что доступно сделать средствами HTML. В том числе и таблицу.

К сожалению, выбранная тема не обладает богатыми возможностями, при создании сайта постоянно возникают сложности. *Таковы практически все бесплатные темы. А доступ к платным темам WordPress возможен при оплате довольно значительной сумм. Однако в Интернете есть множество интересных шаблонов для WordPress, предлагаемых бесплатно или за умеренную сумму. Тот, кто займется поисками, сможет найти, то, что ему требуется.*

3.10. На панели команд редактирования откройте вкладку **HTML** и нажмите **Code**. Наберите:

```
<table width="100">
<tr>
<th style="text-align:center;" width="80%"></th>
<th style="text-align:center;" width="20%"></th>
</tr>
<tr>
<td width="80%"></td>
<td style="text-align:center;" width="20%"></td>
</tr>
</table>
```

Это код таблицы, состоящий из строки заголовка и одной строки для данных. В таблице два столбца: один с наименованием товара занимает 80% ширины, другой с ценой 20%. Далее повторите несколько раз строку для данных:

```
<tr>
<td width="80%"></td>
<td style="text-align:center;" width="20%"></td>
</tr>
```

3.11. Раскройте в Excel файл **price+**, скопируйте в ячейки заголовка: **Наименование программной и печатной продукции** (где 80%) и **Цена с учетом НДС** (где 20%). К сожалению, ячейки можно копировать только по одной. Далее заполните строки таблицы на странице **Цены**, произвольно выбрав продукцию и ее цену из файла **price+**. Обновите и посмотрите страницу.

3.12. У созданной таблицы есть недостаток: надписи слишком крупные. Войдите в меню **Настройка → Шрифты**, поставьте для заголовков размер **Обычный (Нормальный)** и начертание **Bold** (жирное). Для базового шрифта надо установить **Обычный (Нормальный)**. Изменения распространяются на все страницы.

3.13. Просмотрите созданные страницы, сравните их с образцом. Если замечаний нет, сохраните и опубликуйте их.

4. Установка виджетов

В WordPress есть полезное средство: Виджеты. Как и везде, *виджетом называют элемент интерфейса (иконку или иную понятную команду), запустив который можно выполнить какие-то дополнительные полезные функции.*

4.1. Загрузим окно виджетов последовательностью команд: **Темы (Настроить) → Виджеты.** Выбранная нами тема (Penscratch 2) относится к тем немногим, где можно бесплатно работать с виджетами, они по умолчанию ставятся в подвал. Если раскрыть **Подвал 1 → Добавить виджет**, то можно увидеть внушительный список готовых подпрограмм, которые можно разместить на Веб-странице.

4.2. Выберем простой вариант: добавление иллюстраций в виде галереи. Раскрываем **Галерея** или **Gallery (Original)**, в **Заголовок** напишите **Образцы справочной литературы.** Далее нажмите **Выбрать изображение → Редактировать галерею → Добавить в галерею.** Открывается заполненное ранее хранилище медиафайлов. Добавьте оттуда в галерею газету **Стройка** и два журнала: один из серии **Строитель**, другой из серии **Застройщик.** Далее надо скомандовать: **Вставить в галерею**, затем **Сохранить и опубликовать.** Требуемые иллюстрации должны появиться в галерее.

4.3. Ещё раз откройте: **Выбрать изображения → Редактировать галерею** и поставьте всплывающие подписи к иллюстрациям: **Газета «Стройка», Журнал «Строитель», Журнал «Застройщик».** Обновите, сохраните и опубликуйте созданный материал.

Обратите внимание, что созданные виджеты демонстрируются на всех страницах.

Добавим ещё один виджет: Авторы. Вообще-то авторы по умолчанию в виде виджета уже указываются. Но с помощью операции добавления можно указать конкретного автора.

4.4. Выполните **Подвал 1 → Добавить виджет → Авторы.** Раскройте окно **Авторы**, нажав кнопку ▼, и в окне **Заголовок** сотрите то, что там есть и напишите свою фамилию и инициалы. Это надо делать очень медленно: следующую букву вставлять только после того, как на странице появилась предыдущая.

4.5. Больше ничего делать не надо, процесс довольно капризный. Закройте окно, нажав ▲, сохраните созданное и опубликуйте.



4. Тематика и содержание самостоятельной работы

Трудоемкость освоения дисциплины «Web-дизайн» составляет 108 часов, из них 72 часа аудиторных занятий и 36 часов, отведенных на самостоятельную работу студента.

Формы самостоятельной работы.

1. Подготовка к опросу/коллоквиуму. Коллоквиум назначается по одному/двум разделам дисциплины. Вопросы для подготовки строго соответствуют лекционному курсу и выдаются не позднее, чем за неделю до опроса. Одновременно студентам может быть выдан лекционный курс по опрашиваемому материалу в электронном виде.

2. Изучение теоретических разделов практикума. Каждый студент до выполнения практической работы должен изучить вводный раздел к ней, в форме самостоятельной работы. Этот раздел в виде распечатки выдается студенту на предыдущем занятии. А перед выполнением работы проводится опрос, в устной форме. Вводные разделы приведены ранее, при описании практических занятий.

Форма контроля: опрос на оценку: студент устно отвечает по одному из вопросов, затем выясняется уровень его знаний по всему разделу.

Перечень тем самостоятельной работы студентов соответствует тематическому плану рабочей программы дисциплины, трудоемкости.

Вопросы для подготовки к опросу-коллоквиуму

Раздел 1. История и современное состояние Интернета

1. История Интернета: глобальные сети, ARPANET и его превращение в Интернет.
2. Перспективы развития интернета. Интернет вещей.
3. Информационный обмен в Глобальной сети: статистика, доступ к ресурсам, техническое руководство Сетью.

Раздел 2. Технические основы построения Интернета.

Вопросы для самостоятельного изучения:

4. Понятия: протокол передачи данных, сервер и клиент.
5. Распределенная архитектура и динамическая маршрутизация в Интернете.
6. Уровни сетевого соединения: упрощенная модель и модель OSI.
7. Основы пакетной передачи данных.

Раздел 3. Основы построения WorldWideWeb.

Вопросы для самостоятельного изучения:

1. Возникновение и развитие WorldWideWeb, браузеров.
2. Структура адреса URL: протокол, доменное имя, адрес порта, путь к файлу и параметры файла.
3. Протоколы FTP и Telnet, их назначение. Доступ к файлам компьютера и почте.
4. Электронная почта e-mail и web-mail. Сравнение, применение.
5. Регистрация интернет-домена, выбор домена и хостинга.

Раздел 4. Характеристики Web-сайтов и требования к ним.

Вопросы для самостоятельного изучения:

1. Планирование Web-проекта: этапы, модели, приемы.
2. Изучение целевой аудитории Web-сайта.
3. Специфика создания Web-контента: психофизические аспекты, структурирование информации, повышение доступности сайта.



4. Дизайн сайта: цветовые решения, текст, элементы навигации, анимация.
5. Топологические и файловая структуры Web-сайтов.
6. Статичные и интерактивные сайты, средства обратной связи с пользователем.
7. Интернет-магазин, аккаунт, новостная лента: цели и средства создания.
8. Динамические сайты, их преимущества.
9. Серверные средства создания динамических Web-ресурсов.
10. Клиентские средства создания динамических Web-ресурсов.

Раздел 5. Создание Web-страниц средствами HTML и CSS.

Вопросы для самостоятельного изучения:

1. Основные программные средства создания Web-страниц: набор программного кода, Web-редакторы, Web-мастера, CMS-средства и др.
2. Язык HTML, его версии. Правила написания тегов и их атрибутов.
3. Назначение и синтаксис важнейших мета тегов.
4. Теги HTML, управляющие разметкой текста на странице.
5. Работа с цветом и фоном в HTML.
6. Создание в HTML дополнительных элементов текста: списки, анимация текста, специальные символы.
7. Подготовка иллюстраций для Web, вставка графики на Web-страницу.
8. Создание различных видов гиперссылок, комментариев в HTML.
9. Роль таблиц в форматировании Web-страниц. Теги создания и форматирования таблиц.
10. Стили CSS: назначение, возможности, способы помещения в код HTML.

Раздел 6. Поисковые системы и поисковая оптимизация сайта.

Вопросы для самостоятельного изучения:

1. Поисковые системы: история и современное состояние
2. Поисковые машины, этапы их работы, релевантность и пертинентность.
3. Правила составления поисковых запросов. Команды языков поисковых запросов.
4. Поисковая оптимизация. Внутренние и внешние факторы ранжирования Web-страниц.
5. Приемы и правила поисковой оптимизации текста и структуры Web-страниц.
6. Ошибки, махинации при работе с внутренними факторами ранжирования Web-страниц.
7. GooglePageRank, ТИЦ и ВИЦ Яндекса: принципы расчета, использование.
8. Индексация сайта в поисковых системах, определение значимости ключевых слов.
9. Сабмит в различные каталоги, правила обмена ссылками.



5. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, дополнительной литературы

Основные источники:

1. Практикум по информатике. Компьютерная графика и web-дизайн : учебное пособие / Т.И. Немцова, Ю.В. Назарова ; под ред. Л.Г. Гагариной. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2020. — 288 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-16-105768-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1073058>
2. Компьютерная графика и web-дизайн : учебное пособие / Т.И. Немцова, Т.В. Казанкова, А.В. Шнякин ; под ред. Л.Г. Гагариной. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2020. — 400 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-16-106582-2. - Текст : электронный. - URL: <https://new.znanium.com/catalog/product/1041338>

Дополнительные источники:

1. Компьютерная графика : курс лекций / М. А. Дорощенко. - Москва : ФГОУ СПО "МИПК им. И.Федорова", 2018. - 236 с. - Текст : электронный. - URL: <https://new.znanium.com/catalog/product/1040946>
2. Иоханнес Иттен. “Искусство цвета”
3. Джемс Р. Крейг, Ирина Скала. “Шрифт и дизайн. Современная типографика”
4. Сьюзан Уэйншенк. “100 главных принципов дизайна”