

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ
сборник учебно-методических материалов
для направления подготовки

45.04.01 – Филология

Благовещенск 2017

*Печатается по решению
редакционно-издательского совета
факультета математики и информатики
Амурского государственного
Университета*

Составители: Лебедь О.А., Попова А.М.

Информационные технологии в профессиональной деятельности: сборник учебно-методических материалов для направления подготовки 45.04.01. – Благовещенск: Амурский гос. ун-т, 2017.

Рассмотрен на заседании кафедры общей математики и информатики 03.11.2017, протокол № 3.

© Амурский государственный университет, 2017

© Кафедра общей математики и информатики, 2017

© Лебедь О.А., Попова А.М., составление

ВВЕДЕНИЕ

Изучение дисциплины «Информационные технологии в профессиональной деятельности» для направления подготовки 45.04.01 позволяет сформировать необходимые знания и навыки использования современных базовых компьютерных технологий в качестве инструмента для решения практических задач в своей предметной области.

Задачи дисциплины:

привить умение целенаправленно работать с информацией, развить навыки профессионального использования информационных технологий;

научить использовать формат представления данных сети для создания собственной Web-страницы;

ознакомить студентов с основными положениями теории баз данных, технологией организации БД, содержанием этапов проектирования БД и особенностями реляционной модели.

В результате освоения обучающийся должен демонстрировать следующие результаты образования:

Знать: сущность и значение информации в развитии современного информационного общества;

Уметь: применять на практике базовые навыки сбора и анализа языковых и литературных фактов с использованием традиционных методов и современных информационных технологий;

Владеть: основными методами, способами и средствами получения, хранения, переработки информации, навыками работы с компьютером как средством управления информацией.

В качестве способа по формированию умений и навыков в учебной программе предусмотрены практические занятия. Практические занятия проводятся в компьютерных классах с применением специально разработанных учебно-методических пособий, в которых изложены подробные методические рекомендации по изучению каждой темы и выполнению заданий.

В настоящее время актуальным становятся требования к личным качествам современного обучающегося – умению самостоятельно пополнять и обновлять знания, вести самостоятельный поиск необходимого материала, быть творческой личностью. Ориентация учебного процесса на саморазвивающуюся личность делает невозможным процесс обучения без учета индивидуально-личностных особенностей обучаемых, предоставления им права выбора путей и способов обучения. Появляется новая цель образовательного процесса – воспитание личности, ориентированной на будущее, способной решать типичные проблемы и задачи исходя из приобретенного учебного опыта и адекватной оценки конкретной ситуации.

1 МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРАКТИЧЕСКИМ ЗАНЯТИЯМ

Важной составной частью учебного процесса в университете являются практические занятия.

Задачей преподавателя при проведении практических занятий является грамотное и доступное разъяснение принципов и правил проведения работ, побуждение обучающихся к самостоятельной работе, определения места изучаемой дисциплины в дальнейшей профессиональной работе будущего выпускника.

Прежде чем приступить к выполнению заданий на практических занятиях, обучающемуся необходимо ознакомиться с теоретическим материалом, соответствующим данной теме. Задания представлены конкретно-практическими задачами, при решении которых формируется минимальный набор умений. Преподаватель опосредованно руководит познавательной деятельностью обучающихся, консультирует и подробно разбирает со обучающимися возникшие затруднения в ходе решения задачи, обращает внимание группы на возможные ошибки.

После изучения отдельной темы курса дисциплины, каждый обучающийся получает определенное количество баллов по результатам выполнения практических заданий. Максимальное количество баллов за каждое практическое занятие указано в рабочей программе по дисциплине «Информационные технологии в профессиональной деятельности».

Выполнение заданий на практических занятиях актуально и значимо для текущей и промежуточной аттестации.

Практическое занятие №1. Базы данных

В результате изучения темы «Базы данных» обучающиеся должны:

уметь создавать таблицы различными способами;

уметь работать со схемами данных; уметь работать с запросами, формами, отчетами.

Задания выполняются в СУБД Access.

Составить базу данных по теме “**Библиотека**”, в которой должны иметься сведения:

данные о книгах (год издания, цена, инвентарный номер);

сведения о читателях;

В базе данных обязательно должны быть использованы типы полей:

текстовый;

числовой;

дата;

логический;

Построить и объяснить схему данных.

Создать запросы:

Вывести книги определенного автора.

Создать запрос о прочитанных читателем книгах.

Вывести количество прочитанных книг (параметр).

Отобразить количество книг, стоимость которых меньше 600 рублей.

Вывести фамилии авторов книги, которые состоят из: а) 6 символов; б) более 7-ми символов; с) менее 6-ти.

Создать запрос, вычисляемый среднюю цену данных книг.

Создать отчеты: автоотчет, отчет в режиме конструктора, сложного отчета.

Создать формы: автоформа, форма в режиме конструктора, объединенная форма (включает две созданные формы).

Практическое занятие №2. Создание Web-страницы с помощью языка HTML

В результате изучения темы «Создание Web-страницы с помощью языка HTML» обучающиеся должны:

- уметь создавать Web-страницы;
- уметь работать с гипертекстовыми документами.

Создание Web-страницы с помощью языка HTML

Web-страница представляет собой текстовый документ. Экранное представление Web-страницы формируется с помощью специальных команд языка HTML, интерпретируемых браузером. Web-страница отображается как комбинированный документ.

Язык HTML (HyperText Markup Language) – это язык разметки гипертекста. Гипертекст – это расширенный текст, содержащий дополнительные элементы. HTML представляет собой достаточно простой набор команд, которые описывают структуру документа. Этот язык разметки позволяет выделить в документе отдельные логические части (заголовки, абзацы, списки и т.д.), но не задает конкретные атрибуты форматирования.

Команды языка HTML называются тегами и записываются в виде последовательных букв, заключенных в угловые скобки: <тег>. Теги могут быть одиночными или парными. Парный тег состоит из открывающегося и закрывающегося тега. Закрывающийся тег содержит ту же последовательность букв, но им предшествует косая черта: </тег>. Открывающие теги часто могут содержать атрибуты, влияющие на эффект, создаваемый тегом. Атрибуты – это дополнительные ключевые слова, отделенные от ключевого слова тега и друг от друга пробелами. Закрывающие теги никогда не содержат атрибутов. Если в один тег включается несколько атрибутов, они отделяются от наименования тега и друг от друга одиночными пробелами.

Структура документа HTML

Все документы HTML имеют строго заданную структуру. Документ должен начинаться с тега <HTML> и заканчиваться соответствующим закрывающим тегом </HTML>. Внутри он состоит из двух разделов – раздела заголовков (парный элемент HEAD) и тела документа (парный элемент BODY).

Минимальный «полный» документ HTML выглядит:

```
<HTML>
  <HEAD>
  <TITLE>
  Моя первая Web - страница
  </TITLE>
  </HEAD>
  <BODY>
    Сегодня учусь составлять HTML документ.
  </BODY>
</HTML>
```

Создав документ в Блокноте, его необходимо сохранить в виде файла с расширением *.html (*.htm) и открыть с помощью браузера.

В теге <BODY> можно задавать следующие цветовые атрибуты:

BGCOLOR – определяет фоновый цвет страницы;

TEXT – определяет цвет текста на странице;

LINK – определяет цвет для не просмотренных ссылок;

ALINK – определяет цвет активной ссылки – ссылки, на которую наведен указатель мыши или выделенной с помощью клавиши TAB;

VLINK – определяет цвет просмотренных ссылок, для которых уже осуществлялся переход по щелчку.

Таблица 1. Таблица цветов

Цвет	Значение	Цвет	Значение	Цвет	Значение
Белый (WHITE)	FFFFFF	Желтый (YELLOW)	FFFF00	Серый (GRAY)	808080
Синий (BLUE)	0000FF	Зеленый (GREEN)	00FF00	Темно-синий (NAVY)	000080
Фиолетовый (PURPLE)	FF00FF	Аква (AQUA)	00FFFF	Оливковый (OLIVE)	808000
Красный (RED)	FF0000	Черный (BLACK)	000000	Серебряный (SILVER)	C0C0C0

Напр., строка `<BODY BGCOLOR="FFFFFF">` идентична записи `<BODY BGCOLOR="WHITE">` или `<BODY BGCOLOR="FFFF00" TEXT="000000">`

Линейки. Горизонтальные линейки задаются одиночным тегом `<HR>`. Атрибут `ALIGN` задает способ выравнивания линейки: `LEFT` – слева, `RIGHT` – справа, `CENTER` – по центру. Атрибут `WIDTH` определяет ширину (длину) линейки относительно страницы (знак % обязателен, напр.: `<HR WIDTH="80%">`). Толщину (реальную ширину) определяет атрибут `SIZE`, его значение задается в пикселях.

Заголовки. Язык HTML позволяет задать шесть разных уровней заголовков, которые задаются парными тегами от `<H1>` до `<H6>`. Выравнивание заголовков задается атрибутом `ALIGN`, принимает значения: `LEFT`, `RIGHT`, `CENTER`, `JUSTIFY`.

<code><H1></code> заголовок 1-го уровня <code></H1></code>	<code><H4></code> заголовок 4-го уровня <code></H4></code>
<code><H2></code> заголовок 2-го уровня <code></H2></code>	<code><H5></code> заголовок 5-го уровня <code></H5></code>
<code><H3></code> заголовок 3-го уровня <code></H3></code>	<code><H6></code> заголовок 6-го уровня <code></H6></code>

Абзацы. Отдельный абзац задается с помощью парного тега `<P>`. Сплошной текст можно разбивать при помощи тега перехода на новую строку `
`.

Логические разделы. Разбиение текста на логические разделы осуществляется при помощи элемента `DIV`. Этот элемент похож на элемент абзаца, но предназначен для выравнивания группы блочных элементов. Выравнивание задается при помощи атрибута `ALIGN`, не поддерживается только выравнивание по ширине. Напр.: `<DIV ALIGN="CENTER">` – выравнивание текста по центру.

Блочная цитата описывается при помощи парного тега `<BLOCKQUOTE>`, т.е. образуется отступ, как у книжных цитат.

Адресные данные. Один из самых важных элементов в документе HTML – это элемент `ADDRESS`. В нем вы называете себя как автора документа. Элемент `ADDRESS` состоит из текста, расположенного между открывающим и закрывающим тегами `<ADDRESS>` `</ADDRESS>`.

Форматирование текста

Для задания размера, цвета и начертания шрифта служит парный тег ``, влияющий на весь текст, заключенный между открывающим и закрывающим тегами. Тег `` должен иметь хотя бы один из трех возможных атрибутов:

1. `SIZE` – задает размер шрифта. Возможны семь заранее заданных размеров шрифта (от 1 до 7). Чем больше значение, тем крупнее шрифт.
2. `COLOR` – задает цвет шрифта;
3. `FACE` – задает вид шрифта.

Задание начертания шрифта

<code></code> текст жирный <code></code>	<code></code> увеличенный текст <code></code>
<code><I></code> текст наклонный <code></I></code>	<code><STRIKE></code> перечеркнутый текст
<code><U></code> текст подчеркнутый <code></U></code>	<code><SUB></code> подстрочный текст <code></SUB></code>
<code><BIG></code> большой текст <code></BIG></code>	<code><SUP></code> надстрочный текст <code></SUP></code>
<code><SMALL></code> маленький текст <code></SMALL></code>	

Списки

Язык HTML поддерживает 5 видов списков: нумерованный; маркированный; список определений; список каталогов; список меню.

Все списки оформляются однотипным образом. Начало и конец списка помечаются парным тегом, описывающим тип списка. Элементы списка – это блочные элементы HTML. Внутри каждого элемента списка можно использовать любые блочные или текстовые элементы HTML.

Маркированный список задается парным тегом и может включать атрибут TYPE, описывающий тип маркера. Напр.: TYPE="DISC" – круглый черный маркер; TYPE="SQUARE" – квадратный черный маркер; TYPE="CIRCLE" – маркер виде колечка.

Нумерованный список задается с помощью тега и может иметь значения атрибута TYPE: TYPE="1" – арабские цифры (1, 2, 3...); TYPE="a" – латинские буквы нижнего регистра (a, b, c...); TYPE="A" – латинские буквы верхнего регистра (A, B, C...); TYPE="i" – римские цифры с использованием нижнего регистра (i, ii, iii...); TYPE="I" – римские цифры с использованием верхнего регистра (I, II, III...).

Элементы списка, задаваемые внутри маркированного и нумерованного списков, описываются парным тегом .

Список определений описывается парным тегом <DL>. Предполагается, что данный список представляет собой «словарь», в котором чередуются термины и определения. Термины выделяются обычно полужирным шрифтом, а определения отображаются с отступом от левого поля.

Пример использования маркированного и нумерованного списка (рис. 1):

```
<HTML>
<HEAD> <TITLE>Вложенные списки</TITLE>
</HEAD>
<BODY>
<H3> По назначению можно выделить следующие виды ВУ:</H3>
<OL> <LI>Устройства хранения данных (ВЗУ или внешняя память ПК):
<UL> <LI>Магнитные диски;
<LI> Оптические диски.
</UL> <LI>Устройства ввода информации:
<UL> <LI>клавиатура;
<LI>графические планшеты;
<LI>сканеры;
<LI>цифровые фотокамеры;
<LI>манипуляторы.
</UL><LI>Устройства вывода информации:
<UL><LI>принтер;
<LI>плоттер;
<LI>монитор.
</UL> <LI>Средства связи и теле-
коммуникации
<LI>Устройства речевого ввода-
вывода </OL> </BODY> </HTML>
```

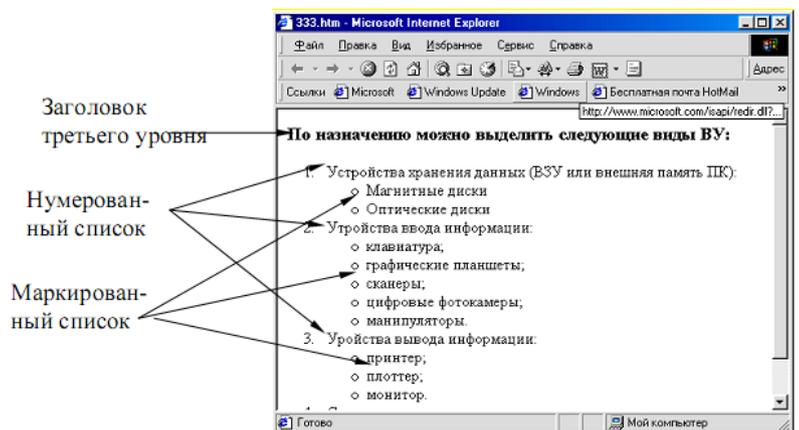


Рис. 1. Пример вложенного списка

Использование рисунков в оформлении Web-страниц

Важную роль в оформлении Web-страниц играют рисунки, которые хранятся в отдельных файлах вне документа, но они отображаются внутри Web-страницы браузером. Для отображения рисунков на странице служит тег , который обязательно содержит атрибут SRC, значением которого является адрес используемого изображения. Для задания размера рисунка используются атрибуты WIDTH (ширина) и HEIGHT (высота).

Например:

Если браузер не может отобразить рисунок, тогда вместо него выводится альтернативный текст, задается тегом ALT.

Например: .

ALIGN – служит для выравнивания заданных объектов в документе.

Значения:

top – верхний край объекта выравнивается по верхнему краю строки;

middle – центр объекта выравнивается по базовой линии строки;

bottom – нижний край объекта выравнивается по базовой линии строки;

left – объект выравнивается по левому краю, при этом возможно обтекание объекта текстом;

right – объект выравнивается по правому краю, при этом возможно обтекание объекта текстом.

Изображение на страницах Web-страницы можно использовать в качестве фонового рисунка страницы. Используя рисунок в теге <BODY> с атрибутом BACKGROUND, в этом случае рабочая область окна заполнится этим рисунком.

1. На диске C: создать папку, назвать ее номером вашей группы. Открыть Блокнот, в нем набрать текст:

```
<HTML>
```

```
<HEAD> <TITLE> Различные подходы прогнозирования объектов </TITLE> </HEAD>
```

```
<BODY> Проанализируем два принципиально различных подхода прогнозирования экономических объектов: телеологический и генетический.
```

Телеологический подход основан на целевых установках развития объекта прогнозирования и степени его приближения к поставленным задачам.

Сущность генетического подхода, есть анализ развития предыстории региона, фиксируются основополагающие факторы, которые позволяют определить особенности и тенденции развития состояния прогнозируемого экономического объекта в будущем.

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

Сохранить файл под именем 1 с расширением html в вашу папку. Просмотреть вашу первую Web-страничку в браузере.

2. Создать Web-страницу 2.html, сохранить ее в вашу папку:

```
<HTML> <HEAD> <TITLE> Стихотворение Анны Ахматовой </TITLE> </HEAD>
```

```
<BODY>
```

```
<H2> А. Ахматова </H2>
```

```
<H2> Белой ночью </H2>
```

```
<BR> Ах, дверь не запирала я,
```

```
<BR> Не зажигала свеч,
```

```
<BR> Не знаешь, как, усталая,
```

```
<BR> Я не решалась лечь.
```

```
<BR> Смотреть, как гаснут полосы
```

```
<BR> В закатном мраке хвой,
```

```
<BR> Пьянея звуком голоса,
```

```
<BR> Похожего на твой.
```

```
<BR> И знать, что все потеряно,
```

```
<BR> Что жизнь – проклятый ад!
```

```
<BR> О, я была уверена,
```

```
<BR> Что ты придешь назад.
```

</BODY> </HTML>

3. Создать Web-страницу 3.html, сохранить ее в вашу папку:

<HTML>

<HEAD> <TITLE> Цвет шрифта </TITLE> </HEAD>

<BODY>

 Красный

 Зеленый

 Синий </BODY> </HTML>

4. Создать Web-страницу 4.html, сохранить ее в вашу папку:

<HTML>

<HEAD> <TITLE> Стили шрифта </TITLE> </HEAD>

<BODY>

 Текст жирный

 Текст <I> наклонный </I>

 Текст <U> подчеркнутый </U>

 Текст <BIG> большой </BIG>

 Текст <SMALL> маленький </SMALL>

 Текст увеличенный

 Текст <STRIKE> перечеркнутый </STRIKE>

 Текст _{подстрочный}

 Текст <SUP> надстрочный <SUP> </BODY> </HTML>

5. Создать Web-страницу 5.html, сохранить ее в вашу папку:

<HTML>

<HEAD> <TITLE> Заголовки </TITLE> </HEAD>

<BODY> <H1> Заголовок 1 </H1>

<H2> Заголовок 2 </H2>

<H3> Заголовок 3 </H3>

<H4> Заголовок 4 </H4>

<H5> Заголовок 5 </H5>

<H6> Заголовок 6 </H6> </BODY>

</HTML>

6. Создать Web-страницу 6.html, сохраните ее в вашу папку:

<HTML>

<HEAD> <TITLE> Линии </TITLE> </HEAD>

<BODY> <HR>

<HR ALIGN=CENTER WIDTH="20%" noshade size=40>

<HR ALIGN=CENTER WIDTH="30%" noshade size=30>

<HR ALIGN=CENTER WIDTH="40%" noshade size=20>

<HR ALIGN=CENTER WIDTH="50%" noshade size=10>

<HR ALIGN=CENTER WIDTH="60%" noshade size=5>

</BODY> </HTML>

7. Создать Web-страницу 7.html, сохранить ее в вашу папку:

<HTML>

<HEAD> <TITLE> Линии, выровненные по правому краю </TITLE> </HEAD>

<BODY BGCOLOR="#00FF00">

<HR ALIGN=LEFT WIDTH="35%" size=5>

<HR ALIGN=LEFT WIDTH="45%" size=10>

<HR ALIGN=LEFT WIDTH="55%" size=15>

<HR ALIGN=LEFT WIDTH="65%" size=20> </BODY> </HTML>

8. Создать Web-страницу 8.html, сохранить ее в вашу папку:

<HTML>

<HEAD> <TITLE> Абзацы и разрывы строк </TITLE> </HEAD>

<BODY>

<P ALIGN=CENTER>

Язык HTML является стандартным языком, предназначенным для создания гипертекстовых документов в среде WWW. </P>

 HTML-документы (или Web-документы) могут просматриваться Web-браузерами различных типов.

<P ALIGN=RIGHT>

Основное преимущество HTML заключается в том, что документ может быть просмотрен на Web-браузерах различных типов и на различных платформах. HTML - один из наиболее простых языков создания Web-страниц. </P>

<P ALIGN=LEFT>

WWW - это сообщество Web-серверов (сетевых компьютеров), входящих в Интернет, на которых хранятся текстовые, графические, видео-, аудио- и другие информационные файлы. </P>
</BODY> </HTML>

9. Создать Web-страницу 9.html, сохранить ее в вашу папку:

<HTML>

<HEAD> <TITLE> Нумерованный список </TITLE> </HEAD>

<BODY>

<H3> Правила создания Web-страниц </H3>

 Web-страницы не должны быть слишком широкими, чтобы пользователям не приходилось пользоваться прокруткой.

Каждая Web-страница должна иметь заголовок.

Не перегружайте страницу графикой.

Попытка визуально выделить все означает не выделить ничего.

Не забывайте житейскую мудрость: "Все гениальное - просто".

 </BODY>

</HTML>

10. Создать Web-страницу 10.html, сохранить ее в вашу папку:

<HTML>

<HEAD> <TITLE> Маркированный список </TITLE> </HEAD>

<BODY>

<H3> Правила создания Web-страниц </H3>

 Web-страницы не должны быть слишком широкими, чтобы пользователям не приходилось пользоваться прокруткой.

Каждая Web-страница должна иметь заголовок.

Не перегружайте страницу графикой.

Попытка визуально выделить все означает не выделить ничего.

Не забывайте житейскую мудрость: "Все гениальное - просто".

 </BODY> </HTML>

11. Создать Web-страницу 11.html, сохранить ее в вашу папку:

<HTML>

<HEAD> <TITLE> Список определений </TITLE> </HEAD>

<BODY>

<DL>

<DT> WWW

<DD> – широкомасштабная гипермедиа-среда, ориентированная на предоставление универсального доступа к огромному количеству документов, расположенных на серверах.

</DL>

<DL>

<DT> Web-сайт (web site)

<DD> – совокупность Web-документов, организованных в виде титульной страницы и нескольких связанных с ней страниц. Каждая такая совокупность, доступная в Интернете, может иметь любые размер и содержание, информационную и эмоциональную направленность.

</DL>

<DL>

<DT> Браузер

<DD> – программа просмотра HTML-файлов, программа-интерпретатор языка HTML. Браузер показывает на экране текст и графику, интерпретируя теги.

</DL> </BODY> </HTML>

12. Создать Web-страницу 12.html, сохранить ее в вашу папку:

<HTML>

<HEAD>

<TITLE>Список-указатель</ TITLE> </ HEAD>

<BODY>

<DIR>

А-Л

М-У

Ф-Я </DIR> </BODY> </HTML>

Таблицы

В HTML для разметки таблиц используются несколько тегов (одновременно). Это напоминает разметку списков, когда один тег определяет список целиком, а другой – отдельный элемент списка.

Тег <TABLE> размещает таблицу. Закрывающий тег обязателен. Любая таблица состоит из ячеек. В HTML ячейки таблицы сгруппированы по рядам (строкам). Для разметки ряда таблицы используется тег <TR>. Для разметки отдельных ячеек в строке используется <TD>. Спецификацией HTML допускается отсутствие закрывающих тегов </TD> и </TR>.

Основные атрибуты элементов таблицы

Атрибут ALIGN – выравнивание содержимого ячеек (LEFT (влево), RIGHT (вправо) и CENTER (по центру)).

Атрибут BACKGROUND – задает фоновое изображение для ячеек строки.

Атрибут BGCOLOR – задает фоновый цвет для ячеек строки.

Атрибут BORDER – задает толщину рамки, обрамляющей ячейки строки. Нулевое значение данного атрибута означает отсутствие обрамления.

Атрибут VALIGN (встречается внутри тегов <TR>, <TH> и <TD>) – определяет вертикальное выравнивание данных в ячейках: TOP (вверху), BOTTOM (внизу), CENTER (по центру), MIDDLE (по середине) и BASELINE (по базовой линии).

Атрибут COLSPAN=n определяет, сколько табличных столбцов займет ячейка в пределах строки.

Атрибут ROWSPAN=m определяет, сколько табличных строк займет ячейка в пределах столбца.

Пример создания таблицы (рис.2), в которой столбцы и строки сгруппированы:

```

<HTML>
<HEAD>
<TITLE> Таблица, в которой строки и столбцы сгруппированы
</TITLE>
</HEAD>
<BODY>
<H3> Таблица </H3>
<TABLE BORDER=4>
<TR><TD ROWSPAN=2> </TD> <TH
COLSPAN=2> Среднее значение </TH></TR>
<TR><TH> Рост </TH> <TH> Вес </TH> </TR>
<TR><TD> Мужчины </TD> <TD ALIGN=CENTER> 184 </TD> <TD
ALIGN=CENTER> 88 </TD> </TR>
<TR><TD> Женщины </TD> <TD ALIGN=CENTER> 168 </TD> <TD
ALIGN=CENTER> 63 </TD> </TR>
</TABLE> </BODY> </HTML>

```

Таблица

	Среднее значение	
	Рост	Вес
Мужчины	184	88
Женщины	168	63

Рис.2. Создание таблицы

Гипертекстовые ссылки

Гипертекст – обеспечивает переход между гипертекстовыми ссылками. Гипертекстовая ссылка – это фрагмент текста (изображения), активация которых отображает связанные с ними документы. Задается она с помощью парного тега <A>. Важный атрибут - HREF или NAME.

Например: название ссылки .

13. Создать Web-страницу 13.html, сохранить ее в вашу папку:

```

<HTML><HEAD>
<TITLE> Таблица без рамок, надписи в которой центрированы </TITLE> </HEAD>
<BODY> <H3> Таблица цветов </H3> <TABLE WIDTH=100%>
<TR ALIGN=CENTER> <TD> Цвет </TD> <TD> Значение </TD></TR>
<TR ALIGN=CENTER> <TD> Белый </TD> <TD> FFFFFFF </TD></TR>
<TR ALIGN=CENTER> <TD> Желтый</TD><TD>FFFF00 </TD></TR>
<TR ALIGN=CENTER><TD>Фиолетовый</TD><TD>FF00FF </TD></TR>
<TR ALIGN=CENTER> <TD> Оливковый </TD> <TD> 808000 </TD></TR>
<TR ALIGN=CENTER> <TD> Серый </TD> <TD> 808080 </TD></TR>
<TR ALIGN=CENTER> <TD> Аква </TD> <TD> 00FFFF </TD></TR>
<TR ALIGN=CENTER> <TD> Серебряный </TD><TD> C0C0C0 </TD></TR>
</TABLE> </BODY> </HTML>

```

14. Создать Web -страницу 14.html, сохранить ее в вашу папку:

```

<HTML>
<HEAD> <TITLE> Таблица, в которой строки и столбцы сгруппированы </TITLE> </HEAD>
<BODY> <H3> Таблица </H3>
<TABLE BORDER=6>
<TR> <TD ROWSPAN=2> </TD> <TH COLSPAN=2> Система счисления </TH></TR>
<TR> <TH> Основание </TH> <TH> База </TH> </TR>
<TR> <TD> Десятичная </TD> <TD> 10 </TD> <TD> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 </TD> </TR>
<TR> <TD> Двоичная</TD> <TD> 2 </TD> <TD> 0, 1 </TD> </TR>
<TR> <TD> Восьмеричная </TD> <TD> 8 </TD> <TD> 0, 1, 2, 3, 4, 5, 6, 7 </TD> </TR>
<TR> <TD> Шестнадцатеричная</TD> <TD> 16 </TD> <TD> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C,
D, E, F </TD> </TR>

```

```
</TABLE> </BODY> </HTML>
```

15. Создать Web-страницу 15.html, сохранить ее в вашу папку:

```
<HTML>
```

```
<HEAD> <TITLE> Современные информационные технологии </TITLE> </HEAD>
```

```
<BODY>
```

```
<BODY BACKGROUND=FFFFFF>
```

```
<CENTER> <H1> Задания по курсу "Современные информационные технологии в социаль-  
ных науках" </H1> </CENTER>
```

```
<HR NOSHADE SIZE=10 WIDTH="100%" ALIGN=CENTER>
```

```
<P> <H2> <A HREF="2.html"> Задание №2 </A> </P>
```

```
<P> <H2> <A HREF="3.html"> Задание №3 </A> </P>
```

```
<P> <H2> <A HREF="4.html"> Задание №4 </A> </P>
```

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

16. Создать Web-страницу 16.html, сохранить ее в вашу папку: добавить рисунок в файл 2.html

Фреймы

Если Web-станция разбита на подокна, каждое окно может иметь вертикальную или горизонтальную полосы прокрутки, и каждое подокно может функционировать независимо от других, что позволяет управлять содержанием данного подокна, которое называется фреймом.

Свойства фреймов

Обращаться к фрейму и загружать его независимо от других фреймов можно с помощью своего URL-адреса. Для того чтобы обратиться к фрейму из другого фрейма, необходимо задать собственное имя, которое задается с помощью атрибута NAME. В любом фрейме на Web-станции, можно отобразить любую информацию.

В документе, можно создать набор фреймов (количество фреймов, относительные и абсолютные размеры), которые будут на Web-станции, тогда необходимо вместо тега <BODY> использовать парный тег <FRAMESET>. Окно можно разделить на вертикальные и горизонтальные фреймы, для описания данного фрейма используется тег <FRAME>.

Вид структуры документа:

```
<HTML> <HEAD>...</HEAD> <FRAMESET>...</FRAMESET> </HTML>
```

Тег <FRAME>, может находиться между тегами <FRAMESET> и </FRAMESET>.

<FRAMESET> имеет следующие атрибуты: BORDER, FRAMEBORDER, COLS, ROWS.

Атрибут COLS задает количество и размеры столбцов.

Атрибут ROWS задает количество и размеры строк.

Например:

```
<FRAMESET COLS = «число, число%. число*»>
```

```
<FRAMESET ROWS= «число, число%; число*»>
```

Задавая атрибут ROWS (COLS) с единичным масштабным коэффициентом означает, что все оставшееся место будет принадлежать данному фрейму.

Делится в равных частях окно Web-страницы в том случае, если единичные масштабные коэффициенты указываются для двух или более фреймов.

Атрибут BORDER – толщина обрамления в пикселях (<FRAMESET BORDER="число">)

Данная запись: <FRAMESET FRAMEBORDER="1"> - обрамление; <FRAMESET FRAMEBORDER="0"> - отсутствие обрамления.

17. Пример Web-страницы с набором фреймов:

```
<HTML> <HEAD> <TITLE> Фреймы </TITLE> </HEAD> <FRAMESET COLS="60%,*">
```

```
<FRAMESET ROWS="60%,*"> <FRAMESET COLS="50%,*"> <FRAME SRC="2.html">
```

```
<FRAME SRC="4.html"> </FRAMESET> <FRAME SRC="5.html">
```

```
</FRAMESET> <FRAME SRC="6.html">
```

```
</FRAMESET> </HTML>
```

Практическое занятие №3. Формы HTML-документов

В результате изучения темы « Формы HTML-документов» обучающиеся должны:
уметь создавать структуру пользовательских форм;
уметь работать с тегами, которые представляют форму.

Совокупность стандартных HTML-конструкций ввода информации и программы работающей на сервере, для обработки введенной информации, называется пользовательской формой. Просмотр данных конструкций возможен с помощью браузера, в этом случае пользователь получает необходимую информацию.

Структура пользовательских форм

Конструкция формы описывается с помощью тега <FORM>, внутри данного тега располагаются теги и параметры, которые составляют HTML-форму, но в данном теге не может содержаться другая форма.

Параметры тега <FORM>:

1. ACTION показывает путь к программе обработчика данных форм на сервере.
2. METHOD используется для пересылки данных с сервера, с помощью протокола.
3. NAME присваивает имя для идентификации пользовательских данных.

Тег <INPUT> - способствует созданию элементов управления в виде: текстовой строки, поле ввода пароля, опции переключения, графический вариант кнопки отправления и сброса введенных данных. Параметры в данном теге задаются атрибутом type (type = «text» - надпись текстовой строки).

Параметры атрибута type:

1. size – представляет собой максимальное количество символов;
2. maxlength – максимальное количество символов, допустимых для ввода в текстовой строке.
3. name – идентификатор имени
4. value – значение текстовой строки.

Пример оформления параметра текстовой строки (рис. 3):

```
<H4> Введите текст </H4>
```

```
<INPUT TYPE = «text» NAME= «S001» VALUE= «Любое текстовое значение»>
```

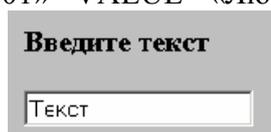


Рис.3. Текстовая строка

Если в атрибуте type указан ввод пароля, то используется значение password (type= «password»).

Пример оформления пользовательского пароля (рис. 4):

```
<H2> Введите пароль </H2>
```

```
<INPUT TYPE = «password» NAME= «S001» VALUE= «пять»>
```

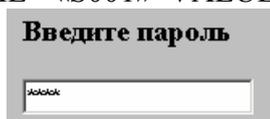


Рис.4. Пользовательский пароль

Атрибут type=«checkbox» используется для выбора значения, используя флажок.

Пример оформления выбора с помощью флажка (рис.5):

```
<H4> Логические операции </H4>
```

```
<INPUT type= «checkbox» name= «operation» value= «inversion»> инверсия
```

```
<INPUT type= «checkbox» name= «operation» value=«conjunction»> конъюнкция
```

<INPUT type= «checkbox» name= «operation» value= «disjunction»> дизъюнкция
 <INPUT type= «checkbox» name= «operation» value= «implication»> импликация
 <INPUT type=«checkbox» name=«operation» value=«equivalence»> эквиваленция

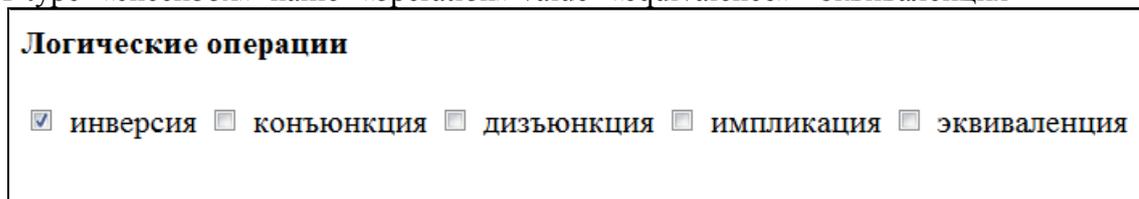


Рис.5. Параметры опции выбора флажком

Атрибут type= «radio» – предназначен для переключения выбора одного значения между различными вариантами. Параметр checked – позволяет выбрать данный параметр по умолчанию.

Пример оформления выбора с помощью переключателя (рис.6):

```
<H2> Переключатели </H2>
<INPUT TYPE=«radio» NAME=«S002» VALUE= «Первый»>
<INPUT TYPE=«radio» NAME=«S002» VALUE = «Второй» checked >
<INPUT TYPE=«radio» NAME=«S002» VALUE = «Третий» >
```

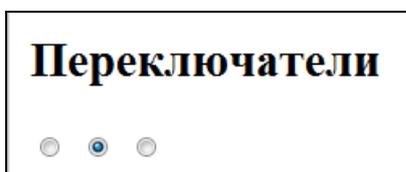


Рис.6. Параметры опции выбора переключателя

Параметр submit атрибута type позволяет создать кнопку отправления данных на сервер (<INPUT TYPE="submit" VALUE="Отправить запрос">).

Параметр reset атрибута type позволяет создать кнопку сброса введенный данных на форме (<INPUT TYPE="reset">).

Пример использование кнопок отправления и сброса данных (рис.7):

```
<HTML>
<HEAD><TITLE> Использование кнопок отправления и сброса
данных</TITLE></HEAD>
<BODY bgcolor="silver">
<H3> Контактные данные</h3>
<FORM>
ФИО: <INPUT NAME = "name"
SIZE="44">
<BR>
Адрес электронной почты: <INPUT NAME = "email" SIZE =: "40">
</FORM>
<INPUT type = "submit" value="Отправить">
<INPUT type="reset">
</BODY> </HTML>
```

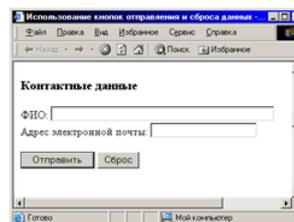


Рис.7. Кнопки отправления и сброса данных

Тег <TEXTAREA> - параметр текстового поля, который позволяет задать высоту и ширину данного текста.

Параметры:

1. name – идентификатор имени
2. cols (rows) – число столбцов (строк) текстового поля

Пример использования текстового поля (рис. 8):

```
<HTML>
<HEAD> <TITLE> Использование текстового поля </TITLE> </HEAD>
<BODY bgcolor="silver">
<H3> Сообщите свои данные</h3>
<FORM> Ваше имя: <BR>
<INPUT name ="name" SIZE="30">
<BR> Ваш E-mail: <BR>
<INPUT name ="e-mail" SIZE="30">
<BR> Комментарий:
<BR>
<TEXTAREA name="comment" rows=6
cols=23>Ваш комментарий... </textarea>
</FORM>
<INPUT type ="submit" value="Отправить">
<INPUT type="reset" value="Очистить">
</BODY>
</HTML>
```

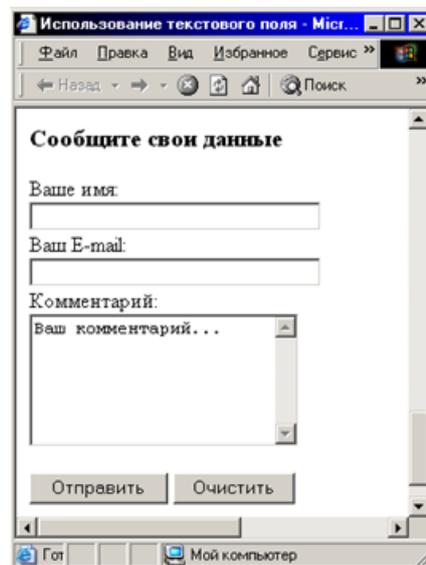


Рис. 8. Использование текстового поля

Тег **<SELECT>** используется для группировки большого количества элементов формы.

Параметры:

1. Name
2. Size

3. Multiple – параметр тега SELECT, позволяющий выбрать несколько позиций из соответствующего списка данных

Тег **<OPTION>** способствует созданию соответствующего пункта списка.

Параметры:

1. Value
2. Selected – предназначен для текущей позиции ниспадающего меню.

Пример использования ниспадающего меню (рис.9):

```
<HTML>
<HEAD><TITLE>Использование ниспадающего меню</TITLE>
</HEAD><BODY>
<H3> В каком городе вы живете?
<BR> <BR>
<SELECT>
<OPTION selected=2 value=a> Москва
<OPTION value=b> Санкт-Петербург
<OPTION value=c> Казань
<OPTION value=d> Ростов
<OPTION value=d> Краснодар
</SELECT></H3> </BODY> </HTML>
```

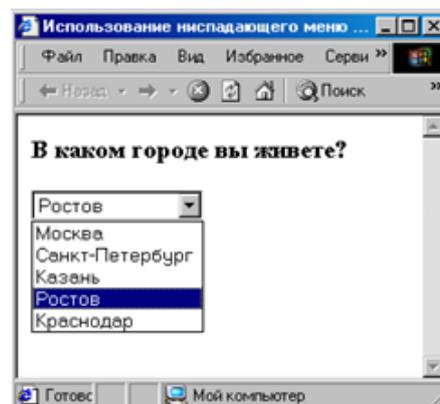
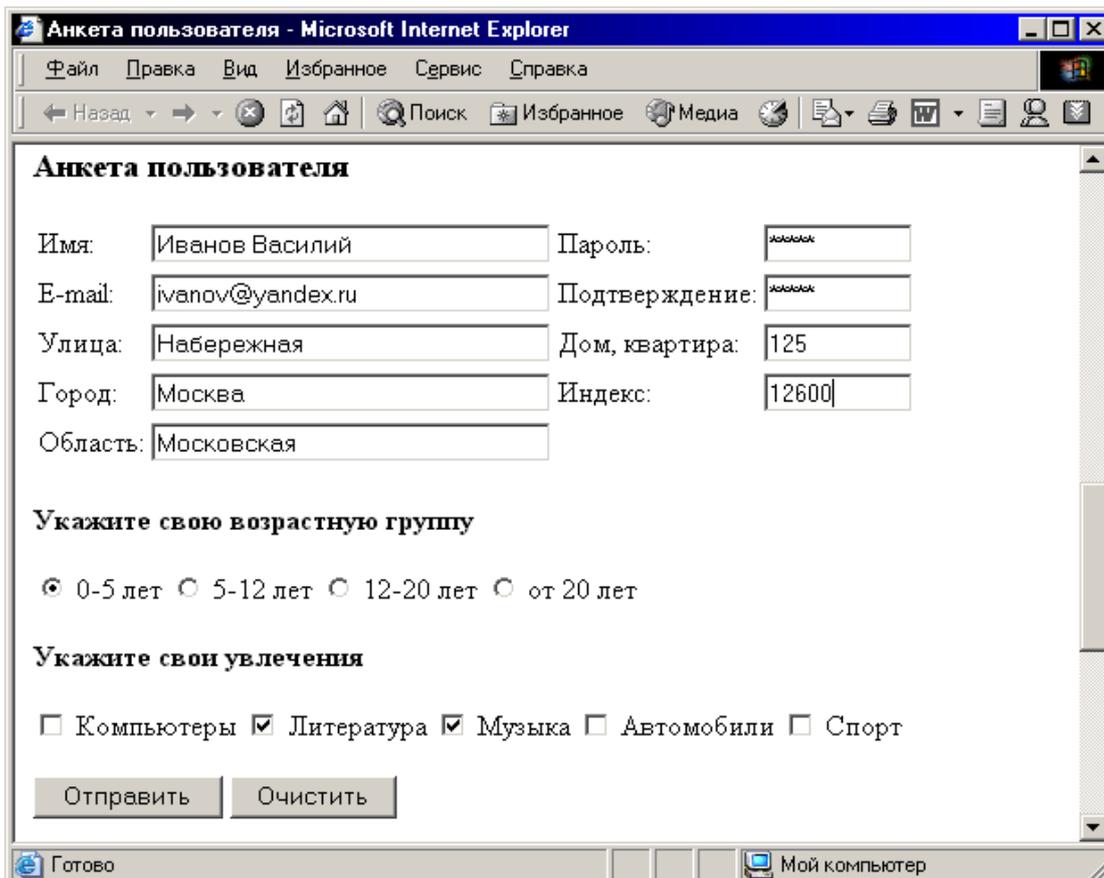


Рис. 9. Ниспадающее меню

18. Используя приведенные формы, создать анкету пользователя, представленную на рисунке 10.



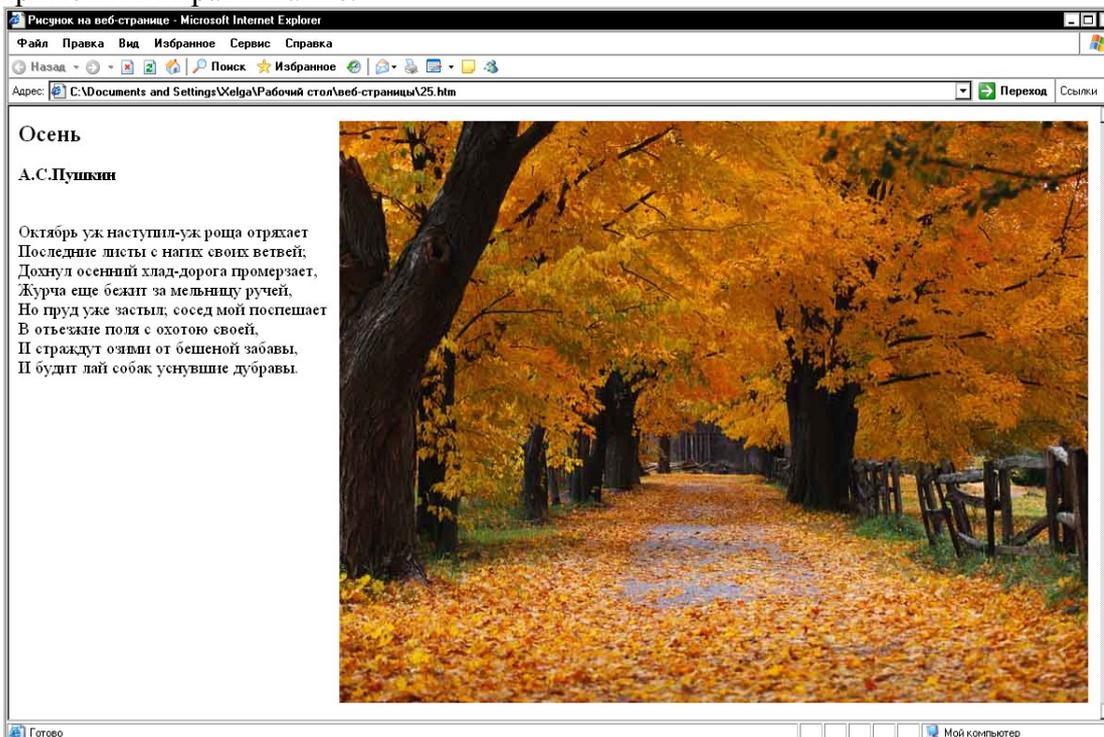
The screenshot shows a Microsoft Internet Explorer window with the title 'Анкета пользователя - Microsoft Internet Explorer'. The browser's address bar is empty, and the menu bar includes 'Файл', 'Правка', 'Вид', 'Избранное', 'Сервис', and 'Справка'. The toolbar contains standard navigation icons. The main content area displays a registration form with the following fields and values:

Имя:	Иванов Василий	Пароль:	*****
Е-mail:	ivanov@yandex.ru	Подтверждение:	*****
Улица:	Набережная	Дом, квартира:	125
Город:	Москва	Индекс:	12600
Область:	Московская		

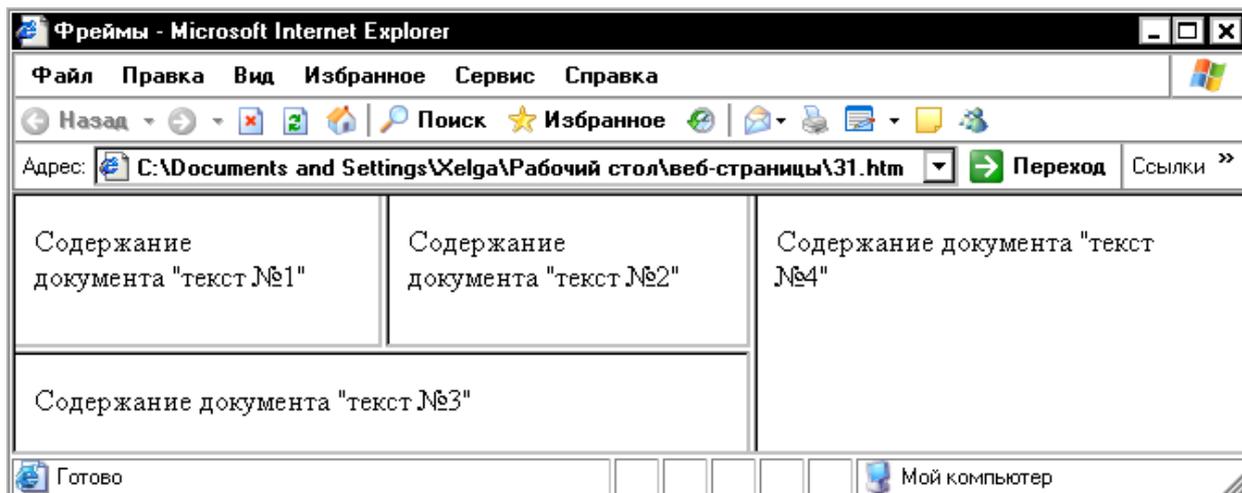
Below the form, there is a section titled 'Укажите свою возрастную группу' with radio buttons for '0-5 лет', '5-12 лет', '12-20 лет', and 'от 20 лет'. The '0-5 лет' option is selected. Another section, 'Укажите свои увлечения', has checkboxes for 'Компьютеры', 'Литература', 'Музыка', 'Автомобили', and 'Спорт'. The 'Литература' and 'Музыка' options are checked. At the bottom of the form are two buttons: 'Отправить' and 'Очистить'. The browser's status bar at the bottom shows 'Готово' and 'Мой компьютер'.

Рис.10. Анкета пользователя

19. Создать следующую страницу, на ней разместить: заголовок окна; заголовок страницы; текст документа; к тексту документа применить выравнивание; на странице поместить рисунок; к рисунку применить выравнивание.



20. Создать Web-страницу с фреймами, как показано на рисунке. И разместить содержимое пунктов 11, 14, 18, 19 в каждом подокне.



21. Оформить текст на странице, как показано ниже:

Октябрь уж наступил,

Уж роца отряхает

Последние листы

С нагих своих ветвей.

Дохнул осенний *хлад,*

Дорога промерзает,

Журча, ^{еще} бежит

За мельницу *ручей*.

2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Самостоятельная работа студента является основным средством овладения учебным материалом во время, свободное от обязательных учебных занятий.

Содержание самостоятельной работы студента определяется учебной программой дисциплины, методическими материалами, заданиями и указаниями преподавателя.

Студентами практикуется два вида самостоятельной работы: аудиторная; внеаудиторная.

Аудиторная самостоятельная работа по дисциплине выполняется на учебных занятиях под непосредственным руководством преподавателя и по его заданию. В этом случае студенты обеспечиваются преподавателем необходимой учебной литературой, дидактическим материалом, в т. ч. методическими пособиями и методическими разработками.

Внеаудиторная самостоятельная работа выполняется студентом по заданию преподавателя, но без его непосредственного участия.

Видами заданий для внеаудиторной самостоятельной работы могут быть:

для овладения знаниями:

чтение текста (учебника, методической литературы);

составления плана текста;

конспектирование текста;

использование компьютерной техники, интернета и др.;

для закрепления систематизации знаний:

повторная работа над учебным материалом;

составление плана выполнения работы в соответствии с планом, предложенным преподавателем; тестирование, выполнение самостоятельных работ;

для формирования умений: выполнение заданий по образцу.

Темы самостоятельной работы представлены в таблице 1.

Таблица 1

№ п/п	Тема дисциплины	Форма (вид) самостоятельной работы
1	Базы данных	Подготовка к лабораторной работе с использованием обучающего теста
2	Создание Web-страницы с помощью языка HTML	Изучение рекомендованной литературы. Подготовка к лабораторной работе с использованием обучающего теста
3	Формы HTML-документов	Изучение рекомендованной литературы

СОДЕРЖАНИЕ

Введение	3
1 Методические указания к практическим занятиям	4
2 Методические указания для самостоятельной работы	19

Ольга Анатольевна Лебедь,
старший преподаватель каф. общей математики и информатики АмГУ
Ангелина Михайловна Попова,
старший преподаватель каф. общей математики и информатики АмГУ