

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»**

Колледж радиоэлектроники имени П.Н. Яблочкова



УТВЕРЖДАЮ

_____ 2021 г.

Дополнительная профессиональная программа
повышения квалификации

Web-дизайн. Начальный уровень

Автор и составитель программы:
Белицкая В.С., преподаватель Колледжа радиоэлектроники имени
П.Н. Яблочкова СГУ имени Н.Г. Чернышевского

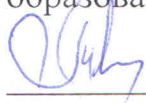
Саратов – 2021

СОГЛАСОВАНО

Зам. директора Института

дополнительного профессионального

образования СГУ



Г.А. Ташпеков

«14» 02 2021 г.

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1. Нормативно-правовые основания разработки программы

Нормативную правовую основу разработки программы составляют:

- Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказ Минтруда России от 12 апреля 2013 г. № 148н «Об утверждении уровней квалификаций в целях разработки проектов профессиональных стандартов»;
- Приказ Минобрнауки России от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам»;
- Приказ Минобрнауки России от 9 декабря 2016 г. № 1547 «Об утверждении федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование,

Программа разработана на основе профессионального стандарта ^{06.035} «Разработчик Web и мультимедийных приложений» (утв. приказом Министерства труда и социальной защиты РФ от 18.01.2017 № 44н).

1.2. Область применения программы

Настоящая программа направлена на совершенствование умений и приобретение навыков необходимых для эффективного осуществления профессиональной деятельности в сфере Web-дизайна. Целевая группа: руководители дизайн студий, рекламные компании, интернет-магазины, иные заинтересованные лица, желающие повысить уровень своей квалификации в области Web-дизайна.

1.3 Требования к слушателям (категории слушателей)

Лица, имеющие среднее профессиональное образование; лица, получающие среднее профессиональное образование (студенты старших курсов). Наличие указанного образования должно подтверждаться документом государственного или установленного образца.

1.4. Цель и планируемые результаты освоения программы

Программа направлена на совершенствование следующих профессиональных компетенций по видам профессиональной деятельности

Виды деятельности	Профессиональные компетенции	Практический опыт (навыки)	Умения	Знания
1	2	3	4	5
ВД 1 верстка страниц	ПК 8.1. Разрабатывать	Создание, модификация и	Создавать, использовать	Нормы и правила выбора

информационного ресурса	дизайн-концепции веб-приложений в соответствии с корпоративным стилем заказчика.	сопровождение веб-сайтов, корпоративных порталов организаций, мультимедиа и интерактивных приложений, информационных ресурсов.	и оптимизировать изображения для веб-приложений; выбирать наиболее подходящее для целевого рынка дизайнерское решение; создавать дизайн с применением промежуточных эскизов, требований к эргономике и технической эстетике.	стилистических решений; современные методики разработки графического интерфейса.
ВД 2 Кодирование на языках веб-программирования	ПК 8.3. Осуществлять разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки.	Создание, модификация и сопровождение веб-сайтов, корпоративных порталов организаций, мультимедиа и интерактивных приложений, информационных ресурсов.	Создавать дизайн с применением промежуточных эскизов, требований к эргономике и технической эстетике; разрабатывать интерфейс пользователя для веб-приложений с использованием современных стандартов.	Требования и нормы подготовки и использования изображений в информационно-телекоммуникационной сети "Интернет"; разработку интерфейса пользователя для веб-приложений с использованием современных стандартов.

1.5. Цель и планируемые результаты обучения

Слушатель в результате освоения программы должен обладать следующими профессиональными компетенциями:

ПК 8.1. Разрабатывать дизайн-концепции веб-приложений в соответствии с корпоративным стилем заказчика.

ПК 8.3. Осуществлять разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки.

Основная цель вида профессиональной деятельности:

Создание, модификация и сопровождение web-сайтов, корпоративных порталов организаций, мультимедиа и интерактивных приложений, информационных ресурсов.

Виды деятельности:

- верстка страниц информационного ресурса;
- проверка и отладка программного кода;
- работа с системой контроля версий;
- кодирование на языках web-программирования;
- тестирование IP с точки зрения логической целостности (корректность ссылок, работа элементов форм).

Владеть:

- основными приемами создания и редактирования веб-приложений;

уметь:

- создавать, использовать и оптимизировать изображения для веб-приложений;
- выбирать наиболее подходящее для целевого рынка дизайнерское решение;
- создавать дизайн с применением промежуточных эскизов, требований к эргономике и технической эстетике;
- разрабатывать интерфейс пользователя для веб-приложений с использованием современных стандартов.

знать:

- нормы и правила выбора стилистических решений;
- современные методики разработки графического интерфейса;
- требования и нормы подготовки и использования изображений в информационно-телекоммуникационной сети "Интернет";
- разработку интерфейса пользователя для веб-приложений с использованием современных стандартов.

1.6. Форма обучения – очная, с применением дистанционных образовательных технологий.

Режим занятий: не более 4 часов в день.

1.7. Форма документа, выдаваемого по результатам освоения программы - удостоверение о повышении квалификации установленного образца.

1.УЧЕБНЫЙ ПЛАН

Таблица 1

№	Наименование модулей/дисциплин	Всего, час.	В том числе:			Форма промежуточной/итоговой аттестации
			Лекции	Практические занятия (семинары, лабораторные работы, выездные занятия)	Самостоятельная работа	
1	Основные понятия Internet- технологий. Введение в HTML.	8	2	4	2	зачет
2	Структура HTML- документа. Теги форматирования текста	12	2	6	4	зачет
3	Табличные теги	12	2	6	4	зачет
4	Создание мультимедийных объектов	12	2	6	4	зачет
5	Создание HTML- формы	17	1	12	4	зачет
6	Фреймы	9	1	4	4	зачет
7	Итоговая аттестация:	2				Зачет
	Итого:	72	10	38	22	

2. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Таблица 2

Дата, проведения занятий (неделя обучения)	Модуль	Тема / Примечания
1,2 неделя	Основные понятия Internet- технологий. Введение в HTML.	Самостоятельная работа с информацией в оболочке Ipsilon.uni с использованием ДОТ
1,2 неделя	Структура HTML-документа. Теги форматирования текста	
1,2 неделя	Табличные теги	
3,4 неделя	Создание мультимедийных объектов	
3,4 неделя	Создание HTML-формы	
3,4 неделя	Фреймы	
	Итоговая аттестация	Зачет в форме тестирования с использованием ДОТ

3. РАБОЧИЕ ПРОГРАММЫ МОДУЛЕЙ

Таблица 3

3.1. Учебно-тематический план программы

№ пп	Наименование модулей, разделов и тем	Всего, час.	В том числе:			Форма промежуточной/итоговой аттестации
			Лекции	Практические занятия (семинары, лабораторные работы, выездные занятия)	Самостоятельная работа	
1	2	3	4	5	6	7
1.	Модуль 1. Основные понятия Internet-технологий. Введение в HTML	8	2	4	2	зачет
1.1	Тема 1. Основные понятия Internet-технологий. Языки разметки гипертекста	1	1			-
1.2	Тема 2. Синтаксис HTML	1	1			-
1.3	Тема 3 Основы работы в HTML.	6		4	2	
2.	Модуль 2. Структура HTML-документа. Теги форматирования текста	12	2	6	4	зачет
2.1	Тема 1. Теги форматирования документа. Теги логического и физического форматирования текста	1	1			-
2.2	Тема 2.2 Использование нумерованных и маркированных списков	1	1			
2.3	Тема 3 Основы работы в HTML. Работа с текстом	10		6	4	
3.	Модуль 3. Табличные теги	12	2	6	4	зачет
3.1	Тема 1. Теги описания таблиц. Особенности построения таблиц	2	2			-
3.2	Тема 2. Создание таблиц. Объединение ячеек	10		6	4	-

4	Модуль 4. Создание мультимедийных объектов	12	2	6	4	зачет
4.1	Тема 1. Работа с изображением. Создание закладок и гиперссылок	2	2			-
4.2	Тема 2. Добавление рисунков. Создание закладок и гиперссылок	10		6	4	-
5	Модуль 5. Создание HTML-формы	17	1	12	4	зачет
5.1	Тема 1. Теги и атрибуты тегов создания полей формы	1	1			-
5.2	Тема 2. Создание различных полей формы в html-документе	16		12	4	-
6	Модуль 6. Фреймы	9	1	4	4	зачет
6.1	Тема 1. Правила описания фреймов	1	1			-
6.2	Тема 2. Фреймы. Создание навигационных карт	8		4	4	-
8.	Итоговая аттестация	72	10	38	22	Зачет
	Итого					

Таблица 4

3.2. Учебная программа по модулям/дисциплинам

№ п/п	Наименование модуля /дисциплины, разделов и тем	Содержание обучения (по темам в дидактических единицах), наименовании и тематика лабораторных работ, практических занятий (семинаров), самостоятельной работы, используемых образовательных технологий и рекомендуемой литературы	Уровень освоения учебного материала
1	2	3	4
1	Модуль 1. Основные понятия Internet- технологий. Введение в HTML		
1.1	Тема 1. Основные понятия Internet-технологий. Языки разметки гипертекста	Основные понятия Internet-технологий. Языки разметки гипертекста	<i>ознакомительный</i>
1.2	Тема 2. Синтаксис HTML	Синтаксис HTML. Проверка html-документа на валидность	<i>ознакомительный</i>
1.3	Тема 3 Основы работы в HTML.	Практическая работа. Основы работы в HTML	<i>репродуктивный</i>
2	Модуль 2. Структура HTML-документа. Теги форматирования текста		
2.1	Тема 1. Теги форматирования документа. Теги логического и физического форматирования текста	Теги форматирования документа. Теги логического и физического форматирования текста	<i>ознакомительный</i>
2.2	Тема 2.2 Использование нумерованных и маркированных списков	Использование нумерованных и маркированных списков	<i>ознакомительный</i>
2.3	Тема 3 Основы работы в HTML. Работа с текстом	Практическая работа. Основы работы в HTML. Работа с текстом	<i>репродуктивный</i>
3	Модуль 3. Табличные теги		
3.1	Тема 1. Теги описания таблиц. Особенности построения таблиц	Теги описания таблиц. Особенности построения таблиц	<i>ознакомительный</i>
3.2	Тема 2. Создание таблиц. Объединение ячеек	Практическая работа. Создание таблиц. Объединение ячеек	<i>репродуктивный</i>
4	Модуль 4. Создание мультимедийных объектов		
4.1	Тема 1. Работа с изображением. Создание закладок и гиперссылок	Работа с изображением. Создание закладок и гиперссылок	<i>ознакомительный</i>
4.2	Тема 2. Добавление рисунков. Создание закладок и гиперссылок	Практическая работа. Добавление рисунков. Создание закладок и гиперссылок	<i>репродуктивный</i>

5	Модуль 5. Создание HTML-формы		
5.1	Тема 1. Теги и атрибуты тегов создания полей формы	Теги и атрибуты тегов создания полей формы	ознакомительный
5.2	Тема 2. Создание различных полей формы в html-формы в html-документе	Практическая работа. Создание различных полей формы в html-документе	репродуктивный
6	Модуль 6. Фреймы		
6.1	Тема 1. Правила описания фреймов	Правила описания фреймов	ознакомительный
6.2	Тема 2. Фреймы. Создание навигационных карт	Практическая работа. Фреймы. Создание навигационных карт	репродуктивный
	Самостоятельная работа	Самостоятельная работа с информацией в оболочке Ipsilon.uni с использованием ДОТ	продуктивный
	Используемые образовательные технологии	Дистанционные образовательные технологии	
	Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы	<p><i>Основная литература</i></p> <ol style="list-style-type: none"> 1. Баканов В.М. Программное обеспечение компьютерных сетей и информационных систем. – Москва: МГАПИ, 2017 2. Дронов В. А. Java Script в Web-дизайне. – Санкт-Петербург: «БХВ-Петербург», 2016 3. Прохоренко Н.А. HTML, JavaScript, PHP и MySQL. Джентельменский набор web-мастера. – Санкт-Петербург: «БХВ-Петербург», 2018. <p><i>Интернет-ресурсы</i></p> <ol style="list-style-type: none"> 1. Интернет источник с познавательной информацией по HTML и CSS - http://htmlbook.ru/ 2. Интернет источник с познавательной информацией по JavaScript - http://javascript.ru/ 	

4. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ

4.1. Материально-техническое обеспечение

При реализации программы с применением исключительно электронного обучения, дистанционных образовательных технологий должны быть созданы условия для функционирования электронной информационно-образовательной среды, включающей в себя электронные информационные ресурсы, электронные образовательные ресурсы, совокупность информационных технологий, телекоммуникационных технологий, соответствующих технологических средств и обеспечивающей освоение слушателями программы в полном объеме независимо от места их нахождения.

4.2. Организация образовательного процесса

При реализации дополнительной профессиональной программы преподавание модулей в очной форме может осуществляться в форме авторских лекционных курсов и разнообразных видов коллективных и индивидуальных практических занятий, заданий и семинаров, обеспечивающих квалифицированное освещение тематики и минимума содержания, определенных настоящим документом.

4.3. Кадровое обеспечение образовательного процесса

Требования к квалификации педагогических кадров: реализация дополнительной профессиональной программы повышения квалификации должна обеспечиваться педагогическими кадрами, имеющими опыт профессиональной деятельности в оболочке Ipsilon.uni с использованием ДОТ.

5. ФОРМЫ АТТЕСТАЦИИ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Таблица 5

Формы и методы контроля и оценки результатов освоения модулей

Наименование модулей	Основные показатели оценки	Формы и методы контроля и оценки
<p>Основные понятия Internet- технологий. Введение в HTML.</p>	<p><i>Знает:</i></p> <ul style="list-style-type: none"> – нормы и правила выбора стилистических решений; – современные методики разработки графического интерфейса; <p><i>Умеет:</i></p> <ul style="list-style-type: none"> – создавать, использовать и оптимизировать изображения для веб-приложений; – выбирать наиболее подходящее для целевого рынка дизайнерское решение; <p><i>Владеет</i> – основными приемами создания и редактирования веб-приложений</p>	<p><u>Форма контроля:</u> зачет</p> <p><u>Метод контроля:</u> тестирование</p>
<p>Структура HTML-документа. Теги форматирования текста</p>	<p><i>Знает:</i></p> <ul style="list-style-type: none"> – нормы и правила выбора стилистических решений; – современные методики разработки графического интерфейса; <p><i>Умеет:</i></p> <ul style="list-style-type: none"> – создавать, использовать и оптимизировать изображения для веб-приложений; – выбирать наиболее подходящее для целевого рынка дизайнерское решение; <p><i>Владеет:</i> основными приемами создания и редактирования веб-приложений.</p>	<p><u>Форма контроля:</u> зачет</p> <p><u>Метод контроля:</u> тестирование</p>
<p>Табличные теги</p>	<p><i>Знает:</i></p> <ul style="list-style-type: none"> – требования и нормы подготовки и использования изображений в информационно-телекоммуникационной сети "Интернет"; – разработку интерфейса пользователя для веб-приложений с исполь- 	<p><u>Форма контроля:</u> зачет</p> <p><u>Метод контроля:</u> тестирование</p>

	<p>зованием современных стандартов; <i>Умеет:</i></p> <ul style="list-style-type: none"> – создавать дизайн с применением промежуточных эскизов, требований к эргономике и технической эстетике; – разрабатывать интерфейс пользователя для веб-приложений с использованием современных стандартов; <p><i>Владеет:</i></p> <p>основными приемами создания и редактирования веб-приложений.</p>	
Создание мультимедийных объектов	<p><i>Знает:</i></p> <ul style="list-style-type: none"> – требования и нормы подготовки и использования изображений в информационно-телекоммуникационной сети "Интернет"; – разработку интерфейса пользователя для веб-приложений с использованием современных стандартов <p><i>Умеет:</i></p> <ul style="list-style-type: none"> – разрабатывать интерфейс пользователя для веб-приложений с использованием современных стандартов; <p><i>Владеет:</i></p> <p>основными приемами создания и редактирования веб-приложений.</p>	<p><u>Форма контроля:</u> зачет</p> <p><u>Метод контроля:</u> тестирование</p>
Создание HTML-формы	<p><i>Знает:</i></p> <ul style="list-style-type: none"> – требования и нормы подготовки и использования изображений в информационно-телекоммуникационной сети "Интернет"; – разработку интерфейса пользователя для веб-приложений с использованием современных стандартов <p><i>Умеет:</i></p> <ul style="list-style-type: none"> – создавать дизайн с применением промежуточных эскизов, требований к эргономике и технической эстетике; <p><i>Владеет:</i></p>	<p><u>Форма контроля:</u> зачет</p> <p><u>Метод контроля:</u> тестирование</p>

	основными приемами создания и редактирования веб-приложений.	
Фреймы	<p><i>Знает:</i></p> <ul style="list-style-type: none"> – требования и нормы подготовки и использования изображений в информационно-телекоммуникационной сети "Интернет"; – разработку интерфейса пользователя для веб-приложений с использованием современных стандартов <p><i>Умеет:</i></p> <ul style="list-style-type: none"> – создавать дизайн с применением промежуточных эскизов, требований к эргономике и технической эстетике; – разрабатывать интерфейс пользователя для веб-приложений с использованием современных стандартов. <p><i>Владеет:</i></p> <p>основными приемами создания и редактирования веб-приложений.</p>	<p><u>Форма контроля:</u> зачет</p> <p><u>Метод контроля:</u> тестирование*</p>

Промежуточная аттестация проводится в форме зачета. Аттестация по результатам изучения модуля проводится в форме тестирования. Итоговый балл формируется суммированием баллов за промежуточную аттестацию. Приведение суммарной балльной оценки к четырех-балльной шкале производится следующим образом:

Перевод 100-балльной рейтинговой оценки по дисциплине в традиционную четырех-балльную

100-балльная система оценки	Традиционная четырех-балльная система оценки
85 – 100 баллов	оценка «отлично»
70 – 84 баллов	оценка «хорошо»
50 – 69 баллов	оценка «удовлетворительно»
менее 50 баллов	оценка «неудовлетворительно»

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Контрольно-измерительные материалы для проведения
промежуточной аттестации в форме тестирования.

**Комплект заданий по Модулю 1 Основные понятия Internet- технологий.
Введение в HTML.**

**Тестовая проверочная работа по теме « Понятие языка разметки гипер-
текста»**

1) Что такое HTML?

- а) язык разметки
- б) гипертекстовая библиотека
- в) язык программирования
- г) сценарный язык

2) Из каких частей состоит HTML- документ?

- а) глаз и рта
- б) Заголовка и подвала
- в) голова и тела
- г) рук и ног

3) Тег содержащий визуальную информацию о веб-странице?

- а) body
- б) head
- в) p
- г) br

4) Самый распространённый редактор HTML-кода

- а) Total Commander
- б) Notepad
- в) MS Word
- г) Code Playground

5) Какое расширение у файлов HTML

- а) .txt
- б) .exe
- в) .doc
- г) .html

6. Где ставиться тег title

- а) между тегами body
- б) Перед тегом html
- в) После тега html
- г) Между тегами head

7. С помощью чего в языке разметки программируется контент?

- а) сценариев
- б) тегов
- в) функций
- г) команд

8. С помощью какого тега делается перенос строки без дополнительного промежутка между частями текста

а) break

б) lb

в) linebreak

г) br

9. Какие теги используют для обозначения жирного шрифта

а) sub и sup

б) ins и del

в) b и strong

г) I и em

10) Сколько размеров заголовком в HTML

а) 8

б) 5

в) 6

г) 3

Комплект заданий по Модулю 2. Структура HTML-документа. Теги форматирования текста.

Лабораторная работа «Основы работы в Html. Форматирование шрифта и абзаца.»

Для публикации информации и её глобального распространения необходим универсальный язык типа основного всеобщего языка, который потенциально понятен всем компьютерам. Таким языком, используемым в World Wide Web, является HTML (от HyperText Markup Language/Язык Гипертекстовой Разметки).

HTML даёт авторам средства для того, чтобы:

- публиковать online-документы с оглавлением, текстом, таблицами, списками, фотографиями и т.д.;
- запрашивать online-информацию, переходя по гиперссылкам, нажимая кнопки;
- создавать формы для взаимодействия с удалённым сервисом, для поиска информации, создания архивов, продажи товаров и т.д.;
- включать таблицы данных, видео и аудиоклипы и другие приложения прямо в документы.

HTML-документ-это просто текстовый файл с расширением *.htm. Основой всякого сайта является язык **HTML (HyperTextMarkupLanguage)**-язык разметки гипертекста. Всякий HTML-документ составлен из конкретных HTML-меток или тэгов. Большинство тэгов парные, т.е. на каждую открывающую метку есть закрывающая метка вида.

Обязательные метки.

<html>...</html>

Метка <html> должна открывать HTML-документ. Аналогично, метка </html> должна завершать HTML-документ.

<head>...</head>

Эта пара меток указывает на начало и конец заголовка документа.

<title>...</title>

Всё, что находится между этими метками, толкуется браузером как название документа. Ни в коем случае нельзя путать имя вашего HTML-документа (в расширении *.htm) и его название между метками <title> и </title>.

<body>...</body>

Определяют начало и конец тела HTML-документа, внутри этих тэгов расположено основное содержание HTML-документа. Сам тэг может содержать в себе различные параметры, например bgcolor (цвет фона документа), text (цвет текста), link (цвет гиперссылки) и другие. О этих и других параметрах поговорим подробно позже.

Итак, структуру всякого html-документа можно представить следующим образом:

```
<html>
<head>
<title>...</title>
</head>
<body>...</body>
</html>
```

Заголовки и абзацы.

<h1>...</h1>-<h6>...</h6>

Метки вида <h_i>, где i=1..6 описывают **заголовки шести различных уровней (размеров)**. Заголовок первого уровня самый крупный.

<p>...</p>

Такая пара меток описывает **абзац**. Закрывающий тэг </p> не обязателен. Метки <h_i> и <p> могут содержать дополнительный атрибут **align** (выравнивание) он может принимать значения **left**, **center**, **right** и **justify** (justify-выравнивание по ширине). Например, выравнивание заголовка второго уровня по центру записывается следующим образом: <h2 align=center>текст заголовка</h2>. Отсутствие в тэгах заголовка или абзаца параметра align приводит к выравниванию текста по левому краю аналогично записи align=left.

Непарные тэги.

Кроме парных тэгов или тэгов - контейнеров существуют и непарные, т. е. не требующие закрывающих меток типа </tag>.

**
** - используется, если необходимо перейти на новую строку, не прерывая абзаца. Очень удобна эта метка при публикации стихов в html – документах. Данный тэг ставится не в начале строки, а только в её конце. Для примера рассмотрим фрагмент html – документа, содержащего стих.

```
<p>Ты жива ль ещё, моя старушка?<br>
```

```
Жив и я, привет тебе, привет!</p>
```

<nobr> - запрещает перевод строки. Текст, размеченный этим тэгом, будет гарантированно располагаться в одной строке, независимо от ее длины. Если при этом получающаяся строка будет выходить за пределы окна просмотра браузера, то появится горизонтальная полоса прокрутки.

<hr> - описывает горизонтальную линию, удобную для логического разделения частей html – документа. Данная метка может содержать параметры: **size** (толщина-указывается в пикселях:1,2,..), **width** (длина или ширина - в пикселях или процентах), **noshade**-отменяет рельефность линии (не имеет значений),**color**-указывает цвет линии,**align**-выравнивание линии (значения:left,right,center). Например, строка `<hr size=4 width=100%>` определяет в html – документе горизонтальную линию толщиной 4 и длиной во весь экран.

Некоторые горизонтальные линии:

линия толщины 1, шириной 50%: `<hr size=1 width=50%>`

линия толщины 2, шириной 50%: `<hr size=2 width=50%>`

линия толщины 4, шириной 50%, красного цвета: `<hr size=4 width=50% color=#ff0000>`

Включение комментариев в документ

В HTML-документ можно включать комментарии, которые не будут видны читателю. Они могут состоять из произвольного числа строк и должны начинаться тэгом `<!--` и заканчиваться тэгом `-->`. Все, что заключено внутри этих тэгов, при просмотре страницы не будет отображаться на экране. Комментарии обычно используются автором документа для заметок, предназначенных только для собственного пользования. Их применение целесообразно в случае больших HTML-кодов. Также комментарии применяются для скрытия от браузера скриптов.

Специальные символы

Некоторые символы, непосредственно введенные в HTML-документ, будут интерпретированы не так, как задумал автор. Рассмотрим некоторые из них.

< - Знак «меньше» <

> - Знак «больше» >

 - Знак «неразрывный пробел»

© - Знак «знак копирайта»

“” - Знак «кавычки»

1. Форматирование шрифта.

Для форматирования текста HTML-документов предусмотрена целая группа тэгов, которые условно можно разделить на тэги физического и логического форматирования.

Физические стили.

Под **физическим стилем** принято понимать прямое указание браузеру на модификацию текущего текста. Рассмотрим некоторые тэги физического форматирования.

 - отображает текст полужирным шрифтом;

<i> - отображает текст курсивом;

<tt> - отображает текст моноширинным шрифтом;

<u> - отображает текст подчёркнутым;

<strike> или <s> - отображает перечёркнутый текст;

<big> - выводит текст шрифтом большего (чем непомеченная часть текста) размера;

<small> - выводит текст шрифтом меньшего размера;

<sub> - сдвигает текст ниже уровня строки и выводит его (если возможно) шрифтом меньшего размера. Удобно использовать для математических индексов.

<sup> - сдвигает текст выше уровня строки и выводит его (если возможно) шрифтом меньшего размера.

 - указывает параметры шрифта. Он относится к тэгам физического форматирования уровня текста. Различные параметры шрифта в одном документе нарушают его эстетический вид, поэтому данный тэг в версии HTML 4.0 отнесён к отменённым. Для тэга могут задаваться следующие параметры: face, size и color. Параметр **face** служит для указания типа шрифта, например, Times New

Roman, Arial, Courier и т.д. Программа будет выводить текст указанным типом только в том случае, если он имеется на компьютере. Можно указать как одно, так и несколько названий шрифтов, разделяя их запятыми. Список шрифтов просматривается слева на право. Если на компьютере пользователя нет шрифта, указанного в списке первым, то делается попытка найти следующий шрифт и т.д. Если не найден ни один из указанных шрифтов, то браузер по умолчанию отображает текст шрифтом Times New Roman. Параметр **size** служит для указания размера шрифта в условных единицах от 1 до 7. Размер нормального шрифта считается равным 3. Параметр **color** служит для задания цвета шрифта, например, color=blue означает, что заданный текст отобразится синим.

Тэги форматирования являются парными, т. е. имеют закрывающий тэг типа </tag>. Тэги форматирования могут быть вложенными друг в друга. При этом нужно внимательно следить, чтобы один контейнер находился целиком в другом контейнере. Например, полужирный и курсивный шрифт, получен использованием следующего фрагмента HTML – кода:

```
<b><i>полужирный и курсивный шрифт</i></b>
```

Логические стили

С выходом спецификации HTML 4.0 был провозглашён принцип отделения структуры документа от его представления, благодаря которому **логическое форматирование** стало более предпочтительным, чем физическое. Разные браузеры толкуют одни и те же метки логических стилей по-разному. Рассмотрим самые распространённые логические стили.

**** - (от английского emphasis – акцент) используется для выделения важных фрагментов текста. Обычно браузеры отображают такой текст курсивом.

**** - используется для выделения важных фрагментов текста, отображается полужирным шрифтом.

<code> - как правило, отображается моноширинным шрифтом. Рекомендуется использовать для фрагментов исходных текстов.

<samp> - отмечает текст как образец. Используется также для выделения нескольких символов моноширинным шрифтом.

<kbd> - рекомендуется использовать для указания того, что надо ввести с клавиатуры. Обычно отображается моноширинным шрифтом.

<var> - рекомендуется использовать для написания имён переменных, отображается курсивом.

<cite> - используется для отметки цитат или названий книг и статей, ссылок на другие источники. Текст, заключенный в этот контейнер, выводится курсивом.

**** - этот тэг отмечает свой текст как удаленный (перечеркнутый). Полезно этот тэг использовать для отметки изменений, вносимых в документ от версии к версии.

Параметры тэга <body>.

Как уже говорилось, между тэгами <body> ...</body> заключается содержательную часть HTML-документа. Большинство тэгов, должно располагаться в данном разделе документа. Тэг <body> имеет ряд параметров, ни один из которых не является обязательным, но может пригодиться для разработки интересно оформленных HTML-страниц. Как Вы уже смогли заметить, если в тэге <body> параметры отсутствуют, то браузер отображает такую страницу белым цветом, текст отображается черным цветом, гиперссылки – синим. Это сделано не просто так. Черный текст на белом фоне считается классикой с точки зрения Web-дизайна (хотя и ужасно заезженной). Это весьма объяснимо, т.к. наиболее читабельный именно такой вариант. Представьте себе желтый текст на красном фоне, или синий на черном. В первом случае оформление – вырви глаз, раздражающее и тяжело воспринимаемое; во втором – вообще читать невозможно, т.к. буквы почти сливаются с цветом страницы. Пожалуй, становится понятным одно из основных правил оформления Web – страниц: Оформление страницы не должно затруднять восприятие информации, находящейся на ней. Это, однако, не означает то, что надо использовать белый фон документа и черный текст, просто надо всегда думать не только о себе, но и о тех людях, которые, возможно посетят Ваш сайт.

Итак, мы вплотную подошли к первому параметру тэга <body> **bgcolor**. Как Вы уже могли догадаться, данный параметр определяет цвет фона HTML – документа (bgcolor, от английского background color задний фон). В языке HTML цвета определяются цифрами в шестнадцатиричном коде. Цветовая гамма базируется на трех основных цветах – красном (Red), зеленом (Green) и синем (Blue) – и обозначается RGB. Для каждого цвета задается шестнадцатиричное значение в пределах от 00 до FF, что соответствует диапазону 0 – 255 в десятичном исчислении. Затем эти значения объединяются в одно число, перед которым ставится символ #. Например, зеленый цвет имеет код #008000. Чтобы не запоминать совокупность цифр или букв, можно воспользоваться таблицей цветов (с их кодами), либо просто называть цвета своими именами (по английски), что менее желательно. Например, запись параметра bgcolor=red внутри тэга <body> эквивалентна записи bgcolor=#FF0000. Если с помощью параметра bgcolor можно определить цвет фона документа, то, естественно, можно определить цвет текста, для этого служит параметр **text**. Например, запись text=#0000FF внутри тэга <body> будет задавать текст синего цвета. Заметим, что в данном случае, весь текст HTML-документа станет синим, в отличие от применения тэга , которым выделяются лишь отдельные фрагменты. Выбирая цвета фона документа и текста, не забывайте о читаемости Вашей страницы: как только Вас начинает что-то смущать, раздражать, либо плохо восприниматься, попробуйте подобрать наиболее удачное сочетание цветов. Здесь главное чувство меры и эстетический вид страницы, о чем говорилось ранее.

Продолжим знакомство с параметрами тэга. Как известно, во всяком HTML-документе применяются гиперссылки. Существуют несколько параметров, определяющих цвет ссылки.

link – определяет цвет еще не просмотренной ссылки,

alink – определяет цвет активной ссылки (ссылки, которая работает в данный момент),

vlink – определяет цвет уже просмотренной ссылки.

Например: `<body link=red alink=yellow vlink=blue>`.

Здесь цвет не просмотренной ссылки - красный, активной – желтый, просмотренной – синий. По умолчанию браузеры выводят не посещенную ссылку - синим цветом, посещенную – фиолетовым.

Обратимся теперь к параметрам, задающим границы полей HTML – документа. Таких параметров четыре:

topmargin - устанавливает границу верхнего поля документа в пикселях,

bottommargin - устанавливает границу нижнего поля документа в пикселях,

leftmargin - устанавливает границу левого поля документа в пикселях,

rightmargin - устанавливает границу правого поля документа в пикселях.

Например: `<body topmargin=0 bottommargin=0 leftmargin=30 rightmargin=30>`.

Здесь верхнее и нижнее поля отсутствуют (равны 0), а правое и левое поле по 30 пикселей.

Теперь поговорим о фоновых изображениях. Наряду с заданием цвета фона, HTML – страницу можно замостить так называемыми "обоями" или фоновым изображением. Для этого используется параметр **background** тэга `<body>`. Параметр `background` указывает на URL – адрес изображения, которое используется в качестве фонового. В качестве изображения должна использоваться "картинка" сохраненная в формате .Gif или .Jpeg (.jpg). Удобно использовать небольшое изображение (например, 5X5 см.) в качестве фонового, т.к. в этом случае оно застилает всю страницу без видимых швов (если оно удачно подготовлено для этих целей). Самой удобной программой для подготовки таких "кубиков" (текстуры) служит Photoshop (на худой конец Paint).

Комплект заданий по Модуль 3 Табличные теги

Средства описания таблиц в HTML.

По мере развития WWW стало ясно, что средств, которые заложены в HTML, недостаточно для качественного отображения различного типа документов. Недостатком HTML было отсутствие в его составе средств отображения таблиц. Для этой цели обычно использовался предформатированный текст (тег `<PRE>`), в котором таблица обрисовывалась символами ASCII. Но такая форма представления таблиц была недостаточно высокого качества и выбивалась из общего стиля документа. После введения таблиц в HTML у Web-мастеров появился не просто инструмент для размещения текстовых и числовых данных, а мощное средство дизайна для размещения в нужном месте экрана графических образов и текста.

Тег для создания таблиц `<table></table>`

`<tr></tr>` - строчка таблицы

`<td></td>` - столбец (ячейка) таблицы

Для задания выравнивания и ширины бордюра используем

`<table align=center border=1> </table>`

1x1	1x2	1x3
2x1	2x2	2x3

```
<table>
<tr>
<td>1x1</td>
<td>1x2</td>
<td>1x3</td>
</tr>
<tr>
<td>2x1</td>
<td>2x2</td>
<td>2x3</td>
</tr>
</table>
```

Задание высоты и ширины ячеек: атрибуты `height` и `width`

```
<table>
<tr>
<td height="35" width="50" bgcolor="#FFCC33"> 1x1 </td>
<td width="50" bgcolor="#336699"> 1x2 </td>
<td width="50" bgcolor="#FFCC33"> 1x3 </td>
</tr>
<tr>
<td height="35" width="50" bgcolor="#336699"> 2x1 </td>
<td width="50" bgcolor="#FFCC33"> 2x2 </td>
<td width="50" bgcolor="#336699"> 2x3 </td>
</tr>
</table>
```

```
<table>
<tr>
<td height="35" width="50" bgcolor="#FFCC33"> <center> 1x1 </center> </td>
<td width="50" bgcolor="#336699"> <center> 1x2 </center> </td>
<td width="50" bgcolor="#FFCC33"> <center>1x3 </center> </td>
</tr>
<tr>
```



```

<td height="35" width="50" bgcolor="#336699"> <center> 2x1 </center> </td>
<td width="50" bgcolor="#FFCC33"> <center> 2x2 </center> </td>
<td width="50" bgcolor="#336699"> <center> 2x3 </center> </td>
</tr>
</table>

```

Объединение ячеек

<COLSPAN=num>, где num – это числовое значение, определяющее на сколько столбцов нужно расширить текущую ячейку по горизонтали.

<ROVSPAN=num>, где num - это числовое значение, определяющее на сколько строк нужно расширить текущую ячейку по вертикали.

Заголовки столбцов таблицы — тег <TH>

Заголовки для столбцов и строк таблицы задаются с помощью тега заголовка <TH></TH> (Table Header, заголовок таблицы). Эти теги подобны <TD></TD>. Отличие состоит в том, что текст, заключенный между тегами <TH></TH>, автоматически записывается жирным шрифтом и по умолчанию располагается посередине ячейки. Центрирование можно отменить и выровнять текст по левому или правому краю. Если воспользоваться <TD></TD> с тегом и атрибутом <ALIGN=center>, текст тоже будет выглядеть как заголовок. Однако следует иметь в виду, что не все браузеры поддерживают в таблицах жирный шрифт, поэтому лучше задавать заголовки таблиц с помощью <TH>.

```

<TABLE >
  <TR>
    <TH>Заголовок центрирован по умолчанию</TH>
    <TH COLSPAN=2>Заголовок может объединять столб-
цы</TH>
  </TR>
  <TR>
    <TH>Заголовок может быть расположен перед столбца-
ми</TH>
    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
  </TR>
  <TR>
    <TH ROWSPAN=3>Заголовок может объединять стро-
ки</TH>
    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
  </TR>
  <TR>

```

```

    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
</TR>
<TR>
    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
</TR>
</TABLE>

```

Использование заголовков таблицы — тег <CAPTION>

Тег <CAPTION> позволяет создавать заголовки таблицы. По умолчанию заголовки центрируются и размещаются либо над (<CAPTION ALIGN=top>), либо под таблицей (<CAPTION ALIGN=bottom>). Заголовок может состоять из любого текста и изображений. Текст будет разбит на строки, соответствующие ширине таблицы. Иногда тег <CAPTION> используется для подписи под рисунком. Для этого достаточно описать таблицу без границ.

```

<TABLE>
<CAPTION ALIGN=top>Заголовок над таблицей</CAPTION>
<TR>
    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
</TR>
</TABLE>
<TABLE>
<CAPTION ALIGN=bottom>Заголовок под таблицей
</CAPTION>
<TR>
    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
    <TD>Текст или данные</TD>
</TR>
</TABLE>

```

Атрибут NOWRAP

Обычно любой текст, не помещающийся в одну строку ячейки таблицы, переходит на следующую строку. Однако при использовании атрибута NOWRAP с тегами <TH> или <TD> длина ячейки расширяется настолько, чтобы заключенный в ней текст поместился в одну строку.

Атрибут COLSPAN

Теги <TD> и <TH> модифицируются с помощью атрибута COLSPAN (Column Span, соединение столбцов). Если вы хотите сделать какую-нибудь ячейку шире, чем верхняя или нижняя, можно воспользоваться атрибутом COLSPAN, чтобы растянуть ее над любым количеством обычных ячеек.

```
<TABLE BORDER="3">
  <TR>
    <TD>Если вы хотите сделать какую-нибудь
      ячейку шире, чем верхняя или нижняя,
    </TD>
    <TD>можно воспользоваться атрибутом
      COLSPAN=, </TD>
  </TR>
  <TR>
    <TD BGCOLOR=white COLSPAN="2" > чтобы
      растянуть ее над любым количеством
      обычных ячеек.</TD>
  </TR>
</TABLE>
```

Атрибут ROWSPAN

Атрибут ROWSPAN, используемый в тегах <TD> и <TH>, подобен атрибуту COLSPAN=, только он задает число строк, на которые растягивается ячейка. Если вы указали в атрибуте ROWSPAN=s число, большее единицы, то соответствующее количество строк должно находиться под растягиваемой ячейкой. Внизу таблицы ее поместить нельзя.

Атрибут WIDTH

Атрибут WIDTH применяется в двух случаях. Можно поместить его в тег <TABLE>, чтобы дать ширину всей таблицы, а можно использовать в тегах <TR> или <TH>, чтобы задать ширину ячейки или группы ячеек. Ширину можно указывать в пикселах или в процентах. Например, если вы задали в теге <TABLE> WIDTH=250, вы получите таблицу шириной 250 пикселей независимо от размера страницы на мониторе. При задании WIDTH=50% в теге <TABLE> таблица будет занимать половину ширины страницы при любом размере изображения на экране. Так что, указывая ширину таблицы в пикселах, имейте в виду, что если у пользователя узкая область просмотра, ваша страница может выглядеть несколько странно. Если вы пользуетесь пикселями, и таблица оказывается шире области просмотра, внизу появится полоса прокрутки для перемещения вправо и влево по странице. В зависимости от поставленных за-

дач и тот, и другой способ задания ширины таблицы может оказаться полезным.

Атрибут WIDTH

Атрибут WIDTH применяется в двух случаях. Можно поместить его в тег <TABLE>, чтобы дать ширину всей таблицы, а можно использовать в тегах <TR> или <TH>, чтобы задать ширину ячейки или группы ячеек. Ширину можно указывать в пикселах или в процентах. Например, если вы задали в теге <TABLE> WIDTH=250, вы получите таблицу шириной 250 пикселей независимо от размера страницы на мониторе. При задании WIDTH=50% в теге <TABLE> таблица будет занимать половину ширины страницы при любом размере изображения на экране. Так что, указывая ширину таблицы в пикселах, имейте в виду, что если у пользователя узкая область просмотра, ваша страница может выглядеть несколько странно. Если вы пользуетесь пикселями, и таблица оказывается шире области просмотра, внизу появится полоса прокрутки для перемещения вправо и влево по странице. В зависимости от поставленных задач и тот, и другой способ задания ширины таблицы может оказаться полезным.

Атрибут CELLPADDING

Данный атрибут определяет ширину пустого пространства между содержимым ячейки и ее границами, то есть задает поля внутри ячейки.

Атрибут CELLSPACING

Атрибут CELLSPACING определяет ширину промежутков между ячейками в пикселах. Если этот атрибут не указан, по умолчанию задается величина, равная двум пикселям. С помощью атрибута CELLSPACING= можно размещать текст и графику там, где вам нужно. Если вы хотите оставить пустое место, можно вписать в ячейку пробел.

Атрибуты ALIGN и VALIGN

Теги <TR>, <TD> и <TH> можно модифицировать с помощью атрибутов ALIGN и VALIGN.

- Атрибут ALIGN определяет выравнивание текста и графики по горизонтали, то есть по левому или правому краю, либо по центру. Горизонтальное выравнивание может быть задано несколькими способами:
 - ALIGN=bleedleft прижимает содержимое ячейки вплотную к левому краю.