

ВЫПИСКА ИЗ ОСНОВНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ ОСНОВНОГО  
ОБЩЕГО ОБРАЗОВАНИЯ

ПРИНЯТА

Решением педагогического совета  
БОУ г. Омска «Гимназия № 85»  
Протокол № 1 от 29 августа 2023 года

УТВЕРЖДЕНА

Директором  
БОУ г. Омска «Гимназия № 85»  
О.Н. Клещенко

**РАБОЧАЯ ПРОГРАММА**  
*курса внеурочной деятельности*  
*общинтеллектуального направления*  
*для обучающихся, осваивающих*  
*программу основного общего образования*  
*«Web- дизайн и разработка»*

Уровень освоения: продвинутый уровень (34 часа)

Разработана:  
Шлеер Александра Григорьевна,  
учитель информатики

г. Омск  
2023

## СОДЕРЖАНИЕ

<b>№ п/п</b>	<b>Наименование раздела</b>	<b>стр.</b>
1.	Глоссарий	3
2.	Пояснительная записка	4
3.	Планируемые результаты	6
4.	Содержание программы курса внеурочной деятельности	7
5.	Тематическое планирование	10
6.	Учебно-методическое и информационное обеспечение образовательной деятельности	13
7.	Список литературы	14
8.	Приложения	15

## Глоссарий

1. **Веб- дизайн** - вид графического дизайна, направленный на разработку и оформление объектов информационной среды Интернета, призванный обеспечить им высокие потребительские свойства и эстетические качества. Подобная трактовка отделяет веб-дизайн от веб-программирования, подчеркивает специфику предметной деятельности веб-дизайнера, позиционирует веб-дизайн как вид графического дизайна.
2. **Веб-приложение** - клиент-серверное приложение, в котором клиент взаимодействует с веб-сервером при помощи браузера.
3. **Интернет** (англ. Internet) - всемирная система объединенных компьютерных сетей для хранения и передачи информации.
4. **Пакет «Microsoft Office»** - офисный пакет приложений, созданных корпорацией Microsoft для операционных систем Windows, Windows Phone, Android, macOS, iOS, iPadOS.
5. **Редактор HTML** - это программа для редактирования HTML-разметки веб-страницы.
6. **Microsoft Windows** - группа семейств коммерческих проприетарных операционных систем корпорации Microsoft, ориентированных на управление с помощью графического интерфейса.
7. **CSS** — формальный язык декорирования и описания внешнего вида документа (веб-страницы), написанного с использованием языка разметки (чаще всего HTML или XHTML).

## Пояснительная записка

Рабочая программа курса внеурочной деятельности – «Web- дизайн и разработка», составлена в соответствии с требованиями Федерального закона от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации», Федеральными государственными образовательными стандартами среднего общего образования, Методическими рекомендациями по реализации проекта «Билет в будущее» по профессиональной ориентации обучающихся 6-11 классов образовательных организаций Российской Федерации, реализующих образовательные программы основного общего и среднего общего образования, Методическими рекомендациями по реализации профориентационного минимума для образовательных организаций Российской Федерации, реализующих образовательные программы основного общего и среднего общего образования, материалами конкурсной документации Регионального чемпионата профессионального мастерства «Профессионалы» Омской области, Концепция преподавания предметной области «Технология» в образовательных организациях Российской Федерации, реализующих основные общеобразовательные программы.

В пояснительной записке раскрываются цель и задачи программы внеурочной деятельности в соответствии с выбранным направлением внеурочной деятельности.

Направление деятельности: программа внеурочной деятельности разработана на основе методических рекомендаций по реализации профминимума для образовательных организаций Российской Федерации, реализующих образовательные программы основного общего среднего общего образования.

Представить нашу современную жизнь без электронной коммерции, видеостриминга, социальных сетей, почты, интернет-магазинов, игр, невозможно. Сфера веб-технологий очень широка. Сейчас прослеживается четкая тенденция к переходу всего программного обеспечения в сеть Интернет, сложно представить предприятие без личного веб-сайта. Создают все это веб-разработчики.

На чемпионатах, конкурсах в данной компетенции ребята демонстрируют свои способности в области веб-технологий, включая, веб-дизайн, верстку, программирование на стороне сервера и клиента, а также разработку сайта с использованием систем управления контентом. Они выполняют в короткий срок задачи пяти должностей.

В наши дни каждый может попробовать свои силы в веб-разработке, оказывая все большее влияние на разработчиков-профессионалов. Чтобы пробудить интерес у посетителей сайта, последние обязаны изучать новые техники и технологии производства сайтов и использовать их при решении оригинальных задач. Данная программа «Web- дизайн и разработка» направлена на ознакомление учащихся с основами профессии веб - разработчик, востребованной на рынке труда. Рассчитана на учащихся в возрасте 14-16 лет. Может быть реализована в рамках предмета

«Информатика» как вариативная составляющая, а также в рамках организации профессиональных проб.

**Целью программы** курса внеурочной деятельности является формирование у учащихся целостного представления о возможности решения распространенных задач веб-дизайна и разработки кода для создания собственных информационных ресурсов.

**Основные задачи программы:**

- сформировать у учащихся единую систему понятий, связанных с созданием, получением, обработкой, интерпретацией и хранением информации;
- показать основные приемы эффективного использования информационных ресурсов Интернет;
- сформировать логические связи с другими предметами, входящими в курс среднего образования.

Работа с информацией стала отдельной специальностью, остро востребованной на рынке труда. Данный курс включает в себя элементы общей информатики, элементы WEB-дизайна и элементы электронного делопроизводства.

На занятиях учащиеся получают возможность приобрести умения работы на современных профессиональных ПК и программных средствах, включая оптические диски, сканеры, модемы, текстовый редактор “Блокнот”, редактор HTML, графические редакторы Photoshop. Получают практический опыт в использовании специальных готовых технических решений при разработке веб-приложений; выполнении разработки и проектирования информационных систем; модернизации веб-приложений с учетом правил и норм подготовки информации для поисковых систем; реализации мероприятий по продвижению веб-приложений в сети Интернет.

Для изучения курса предусмотрено использование мультимедийных технологий, таких как презентации, OnLine-тестирование и т.п. Часть материала предлагается в виде теоретических занятий. Занятия по использованию ресурсов Интернет должны происходить в режиме OnLine. При отсутствии стабильного доступа к Интернет, рекомендуется использовать режим OffLine с заранее подготовленным информационным материалом.

Текущий контроль усвоения материала может осуществляться путем устного/письменного опроса и в основном при выполнении практических заданий.

По окончании обучения по данному курсу внеурочной деятельности, учащиеся должны уметь создавать, размещать в сети Интернет и поддерживать WEB-сайт, спроектированный и созданный ими на занятиях. Разрабатывать программный код клиентской и серверной части веб-приложений; осуществлять оптимизацию веб-приложения с целью повышения его рейтинга в сети Интернет; разрабатывать и проектировать информационные системы.

На учебных и практических занятиях обращается внимание учащихся на соблюдение требований безопасности труда, пожарной безопасности, производственной санитарии и личной гигиены.

## **Особенности реализации программы**

### **Формы организации занятий**

Формы организации занятий:

- деловые игры, познавательные игры и беседы;
- социальная проба.

### **Виды деятельности**

В курсе внеурочной деятельности применяются такие виды деятельности как:

- Игровая;
- Познавательная;
- Проблемно-целостное общение;
- Досугово- развлекательная деятельность (досуговое общение)
- Социальное творчество
- Техническое творчество

**Общая трудоемкость курса составляет: 68 часов.**

**Количество часов в неделю: 1 час.**

## **Планируемые результаты освоения курса внеурочной деятельности**

Программа курса внеурочной деятельности рассчитана на определенный уровень подготовки учащихся:

- базовые знания по информатике;
- владение основными приемами работы в операционной среде Microsoft Windows.
- оптимизация веб-приложений и соответствующие ему общие компетенции
- владение пакетом «Microsoft Office».

### **Личностные результаты**

готовность и способность к самостоятельному обучению на основе учебно-познавательной мотивации, в том числе готовность к проектированию личного информационного пространства с учётом устойчивых познавательных интересов.

### **Метапредметные результаты**

*регулятивные универсальные учебные действия:*

определять действия в соответствии с учебной и познавательной задачей, планировать свои действия в соответствии с поставленной задачей и условиями её решения, осуществлять пошаговый контроль своей познавательной деятельности, определять потенциальные затруднения при

решении практической задачи и находить средства для их устранения, осознавать качество и уровень усвоения материала по модулям;

*познавательные универсальные учебные действия:*

строить рассуждение от общих закономерностей к частным явлениям и от частных явлений к общим закономерностям; строить рассуждение на основе сравнения предметов и явлений, выделяя при этом общие признаки;

*коммуникативные универсальные учебные действия*

полученные знания помогут социализации, вовлекут в проектную и научно-исследовательскую деятельность.

### **Предметные результаты**

- Учащийся получит знания о/об:
  - основных правил и инструкции по технике безопасности при работе с ПК;
  - наборе необходимых инструментов для создания Web-страницы/сайта;
  - основных принципах использования языка HTML;
  - принципах работы с Web-редакторами;
  - основных средствах редактирования Web-страниц;
  - типичной структуре WEB-страницы и принципы проектирования сайта;
  - основные принципы WEB-дизайна;
  - языках программирования и разметки для разработки клиентской и серверной части веб-приложений;
  - принципах функционирования поисковых сервисов и особенности оптимизации веб-приложений под них;
  - научится готовить свой сайт к публикации в Интернет.
- Учащийся получит углублённые знания о способах разработки личного информационного пространства; проектировании сайтов; проектирования его дизайна с использованием растровых, векторных и 3D-редакторов, предназначенных для компьютерной обработки изображений; разработки флешфильмов и веб-дизайна.
- Учащийся научится самостоятельно проектировать сайты и их дизайн, а также размещать их в сети Интернет; получит возможность научиться основам web-дизайна и проектированию личного информационного пространства.

### **Оценка результатов деятельности**

- представление коллективного результата группы учащихся в рамках модуля;
- индивидуальная оценка результатов деятельности каждого учащегося.

### **Содержание курса внеурочной деятельности**

**1 год обучения**

## **Раздел 1. Подготовка рабочего места**

Ознакомление с рабочим местом, настройка ПК, программ для создания web- сайтов. Работа с атрибутами HTML-тегов. Технологические особенности WEB-дизайна. Профессиональные приемы создания страниц. Графические эффекты. Профессиональные ресурсы Интернет. Современные технологии разработки веб-приложений. Устройство и функционирование современных информационных ресурсов.

## **Раздел 2. Основы World Wide Web**

Основы World Wide Web (WWW), HTML, HTML-теги их практическое использование. Работа с атрибутами HTML-тегов. История возникновения и стандарты языка HTML. Синтаксис HTML-документов. Тег <IMG> и его параметры. Создание таблиц в HTML-документе. Основы создания WEB-страниц. Основы языка гипертекстовой разметки HTML.

Практическая работа № 1 «Основные теги HTML»

## **Раздел 3. Каскадная таблица стилей CSS**

Что такое CSS. Общий синтаксис таблиц стилей. Свойства CSS, каскадные таблицы стилей. Практическое использование CSS. Функции сопровождения сайта: отслеживание работоспособности, добавление/изменение информации, создание резервной копии, восстановление бэкапов, мониторинг ресурса на вирусы и т.д. Система контроля версий Git. Многомерное тестирование. Перенос веб-приложения на хостинг, программное обеспечение для переноса сайта на хостинг (Filezilla и т.п). Программное обеспечение для решения практических задач по созданию резервных копий.

Практическая работа № 2 «Использование CSS»

## **2 год обучения**

**Раздел 1. HTML** – 8 часов Технологические особенности WEB-дизайна. Профессиональные приемы создания страниц. Графические эффекты. Профессиональные ресурсы Интернет. Профессия WEBдизайнера. Интернет – новая сфера бизнеса.

**Раздел 2. CSS** – 5 часов История возникновения и стандарты языка HTML. Синтаксис HTML-документов. Структура HTML-документов. Элементы оформления текста. Абзацы, заголовки, разделители, параграфы, выравнивание, цветовое и шрифтовой оформление. Создание списков в HTML-документе. Создание гиперссылок. Графическое оформление (фоновые изображения, линии, кнопки). Тег IMG и его параметры. Создание таблиц в HTMLдокументе. Параметры, определяющие оформление таблиц. Управление размещением содержимого документа на экране при помощи таблиц. Понятие кадра-фрейма. Разбиение экрана на части. Параметры, определяющие взаиморасположение фреймов на экране и их внешний вид. Гипертекстовые переходы внутри и между фреймами. Контрольная работа.

**Раздел 3. Интерактивность: HTML + CSS** – 10 часов Обзор программных средств для создания WEB-сайтов. Основы работы в DreamWeaver. Рабочее пространство редактора. Создание WEB – узлов и

документов. Подготовка документов. Редактирование HTML – кода. Режимы работы объектов.

**Раздел 4. Интерактивность: JavaScript** – 9 часов Обзор основных видов интерактивных документов (формы, гостевые книги, анкеты, голосования, счетчики и пр.), и графических эффектов, использующихся при создании WEB-сайтов. Размещение готовых активных форм CGI и JavaScript в HTML-документе. Поиск в Интернет готовых активных формы CGI и JavaScript и размещение их в HTMLдокументе.

**Раздел 5. Практикум** – 2 часа Выработка контента (содержимого) WEB-сайта и разработка его структуры. Создание WEB-сайта с использованием редактора DreamWeaver. Размещение WEB-сайтов Интернет. Контрольная работа.

**Учебно-тематическое планирование**

**1 год обучения**

№ п/п	Наименование разделов, темы занятия	Количество часов			Форма аттестации (контроля)
		Всего	Теория	Практика	
<b>Раздел 1. Подготовка рабочего места – 5 часа</b>					
1	Введение. Ознакомление с рабочим местом, настройка ПК, программ для создания web-сайтов. Технологические особенности WEB-дизайна. Современные технологии разработки веб-приложений. Устройство и функционирование современных информационных ресурсов. Подготовка рабочего места	2	1	1	Опрос
2	Профессиональные приемы создания страниц. Графические эффекты. Профессиональные ресурсы Интернет. Современные технологии разработки веб-приложений. Устройство и функционирование современных информационных ресурсов.	3	1		Результаты деловой игры
<b>Раздел 2. Основы World Wide Web – 10 часов</b>					
1	Основы World Wide Web (WWW)	2	1	1	
2	Основы HTML, HTML-теги	2	1	1	
3	Практическое использование HTML-тегов	2	1	1	
4	Работа с атрибутами HTML-тегов	4	2	2	Практическая работа № 1
<b>Раздел 3. Каскадная таблица стилей CSS- 18 часов</b>					
1	Что такое CSS.	2	1	1	
2	Синтаксис таблиц стилей.	2	1	1	
3	Использование CSS в веб-страницах.	2	1	1	
4	Свойства CSS	2	1	1	
5	Каскадные таблицы стилей.	4	2	2	
6	Практическое использование CSS.	6	3	3	Практическая работа № 2
<b>Итоговое занятие (1 час)</b>					

1	Ролевая игра «Разработчик сайта»			1	Результаты ролевой игры
	<b>ИТОГО</b>	<b>34</b>	<b>16</b>	<b>16</b>	

## 2 год обучения

№ п/ п	Наименование разделов, темы занятия	Количество часов			Форма аттестации (контроля)
		Всего	Теория	Практика	
<b>Раздел 1. HTML – 8 часов</b>					
1	ТБ. Организация рабочего места. Знакомство с технологиями веб-разработки	1	1		
2	Структура HTML-документа. Содержание блока заголовков	1	1		
3	Форматирование текста. Физическое и логическое форматирование. Списки	1		1	
4	Гиперссылки	1		1	
5	Изображения	1		1	
6	Изображение. Видео	1		1	
7	Таблицы	1	1		
8	Цвета	1	1		
<b>Раздел 2. CSS – 5 часов</b>					
1	Основы CSS	1	1		
2	Селекторы. Приоритеты стилей. Каскадность	1		1	
3	Стилевые свойства текста	1	1		
4	Стилевые свойства графики	1		1	
5	Блоки-контейнеры. Блочная модель в CSS	1	1		
<b>Раздел 3. Интерактивность : HTML + CSS- 10 часов</b>					
1	Знакомство с технологиями интерактивности	1	1		

2	HTML: форма. Общая структура. Типы полей: txt, email, password, submit	1		1	
3	HTML: формы. Типы полей: number, date, time	1		1	
4	HTML: формы. Типы полей: color, range, checkbox, radio, file и другие. Пример реакции веб-страницы на изменение значения в форме.	1		1	
5	HTML: формы. Атрибуты тега input. Теги select и textarea	1	1		
6	CSS: навигация. Псевдоклассы ссылок	1	1		
7	CSS: трансформация. Функция scale()	1		1	
8	CSS: трансформация. Функции rotate(), skew(), translate()	1	1		
9	CSS: анимация. Свойство transition	1		1	
10	CSS: применение. Панели навигации и фотогалерея. Слои	1		1	
<b>Раздел 4. Интерактивность: JavaScript- 9 часов</b>					
1	JavaScript: основные сведения. Объектная модель (DOM)	1	1		
2	JavaScript. Внедрение в документ. Изменение текста объекта	1		1	
3	JavaScript. Переменные. Математические функции. Чередование контента	1		1	
4	JavaScript. Строки. Дата и время. Отображение даты на странице	1		1	
5	JavaScript. Условный оператор. Изменение контента по дате	1		1	
6	JavaScript. Функции. Форма авторизации. Калькулятор	1	1		
7	JavaScript. Массивы. Слайдер на веб-странице	1		1	
8	JavaScript. Циклы. Динамическое формирование контента. Обработка форм на устройстве пользователя	1	1		

9	JavaScript. Внедрение интерактивных элементов. Скрытие и отображение элементов на веб-странице	1		1	
<b>Раздел 5. Практикум- 2 часа</b>					
1	Верстка одностраничного сайта. Адаптивная верстка	2		2	
	<b>ИТОГО</b>	<b>34</b>	14	20	

## **Учебно- методическое и информационное обеспечение образовательной деятельности**

### **Требования к минимальному материально-техническому обеспечению.**

Оборудование кабинета:

- доска – 1 шт.,
- компьютерные столы – 6 шт.,
- компьютерный стол преподавателя – 1 шт.;
- столы – 6 шт.;
- стулья – 6 шт.,

### **Технические средства обучения:**

- АРМ преподавателя (системный блок, монитор, принтер, сканер, клавиатура, мышь) – 1 шт.;
- Акустическая система – 1 шт.;
- Проектор – 1 шт.;
- Интерактивная доска – 1 шт.,
- АРМ ученика (системный блок, монитор, клавиатура, мышь) – 6 шт.;
- Коммуникационные устройства – 1 шт.

### **Программное обеспечение обучения:**

- Офис профессиональный (MS Word, MS Excel, MS Access, MS Power Point, MS Visio);
- Браузеры;
- Среды для веб-разработки (VScode, Sublime Text 3, PHPStorm, Notepad++);
- Доступ в Интернет.

1. Колисниченко Д.Н. PHP и MySQL: разработка web-приложений / Д.Н. Колисниченко. – СПб: БХВ–Петербург, 2015. – 592 с.
2. Котеров, Д.В. PHP 7 / Д.В. Котеров, И.В.Симдянов – СПб: БХВ–Петербург, 2016. – 1088 с.
3. Бенкен, Е.С. PHP, MySQL, XML: программирование для интернета / Е.С. Бенкен. – СПб: БХВ–Петербург, 2014. – 336 с.
4. Бенкен, Е.С. PHP, MySQL, XML: программирование для интернета / Е.С. Бенкен. – СПб: БХВ–Петербург, 2014. – 336 с.: ил.+CD–ROM
5. Ляпин, Д.А. PHP это просто. Начинаем с видеоуроков / Д.А. Ляпин, А.В. Никитин. – СПб: БХВ–Петербург, 2013. – 176 с.: ил.+CD–ROM.
6. Дунаев, В. Самоучитель JavaScript / В. Дунаев. 2–е изд. – СПб. : Питер, 2012. – 400 с.
7. Кузнецов, М.В. PHP 5. Практика разработки Web-сайтов / М.В. Кузнецов, И.В. Симдянов, С. В. Голышев. – СПб: БХВ-Петербург, 2012. – 960 с.: ил.
8. Котеров, Д. PHP 5 в подлиннике / Д. Котеров, А. Костарев. – СПб: Символ – Плюс, 2014. – 1120 с., ил.
9. Методы и средства инженерии программного обеспечения: Учебник. Автор/создатель Лавришева Е.М., Петрухин В.А. Единое окно доступа к образовательным ресурсам.  
<http://window.edu.ru/catalog/pdf2txt/699/41699/18857> (дата обращения 25.04.2021)
10. Современный учебник JavaScript [Электронный ресурс]. Режим доступа: <https://learn.javascript.ru/> (дата обращения 27.04.2021)
11. Прогрессивный JavaScript-фреймворк [Электронный ресурс]. Режим доступа: <https://ru.vuejs.org/v2/guide/> (дата обращения 22.04.2021)

Приложение 1

Практическая работа 1 «Основные тэги HTML»

Назначение	Формат	Значения аргументов
<b>Структура Web-страницы</b>		
Начало и конец страницы	<HTML> </HTML>	
Описание страницы, в том числе ее имя	<HEAD> </HEAD>	
Имя страницы	<TITLE> </TITLE>	
Содержание страницы	<BODY> </BODY>	
<b>Форматирование текста</b>		
Заголовок (уровни от 1 до 6)	<H?> </H?>	
Заголовок с выравниванием	<H? ALIGN="*"> </H?>	left, center, right
Абзац	<P> </P>	
Абзац с выравниванием	<P ALIGN="*"> </P>	left, center, right
Перевод строки	 	
Горизонтальный разделитель	<HR>	
Выравнивание по центру	<CENTER> </CENTER>	
Адрес автора	<ADDRESS> </ADDRESS>	
<b>Форматирование шрифта</b>		
Жирный	<B> </B>	
Курсив	<I> </I>	
Подчеркивание	<U> </U>	
Верхний индекс	<SUB> </SUB>	
Нижний индекс	<SUP> </SUP>	
Размер шрифта (от 1 до 7)	<FONT SIZE=?> </FONT>	

Цвет шрифта (задается названием цвета или его 16-ричным кодом)	<FONT COLOR="IRRGGBB"> </FONT>	red , blue #FFFFFFH и др
Гарнитура шрифта	<FONT FACE="*"> </FONT>	Arial TimesET и др.
<b>Вставка изображений</b>		
Вставка изображения	<IMG SRC="URL">	
Выравнивание текста около изображения	<IMG SRC="URL" ALIGN="*">	top bottom middle left right
Вывод текста вместо изображения	<IMG SRC="URL" ALT="*">	текст
<b>Цвет фона, текста и ссылок</b>		
Фоновое изображение	<BODY BACKGROUND="URL">	
Цвет фона	<BODY BGCOLOR="#RRGGBB">	Red, blue #FFFFFFH и др
Цвет текста	<BODY TEXT="#RRGGBB">	red blue IFFFFFFF и др.
Цвет ссылки	<BODY LINK="#RRGGBB">	
Цвет пройденной ссылки	<BODY VLINK="#RRGGBB">	
Цвет активной ссылки	<BODY ALINK="IRRGGBB">	
<b>Вставка гиперссылок</b>		
Ссылка на другую страницу	<A HREF="URL"> указатель ссылки </A>	
Ссылка на закладку в другом документе	<A rel="nofollow ugc" target="_blank" href="URL#*"> указатель ссылки </A>	
Ссылка на закладку в том же документе	<A HREF=" # * " указатель ссылки </A>	
Определение закладки	<A NAME="*"></A>	
<b>Списки</b>		
Ненумерованный	<UL> <LI> </UL>	
Тип метки	<UL TYPE="*">	disk

		circle square
Нумерованный	<OL> <LI> </OL>	
Тип нумерации	<OL TYPE="*">	A, a, I, i, 1
Первый номер списка	<OL START=?>	1,2, ...
Список определений	<DL> <DT> термин <DD> определение </DL>	
Меню	<MENU> <LI> </MENU>	
Каталог	<DIR> <LI> </DIR>	
<b>Формы</b>		
Форма	<FORM> </FORM>	
Текстовое поле NAME="name"	<INPUT TYPE="text" NAME="name" SIZE=?>	1,2,3 ...
Группа переключателей NAME="group"	<INPUT TYPE="radio" NAME="group" VALUE="*">	rad1 rad2 rad3
Группа флажков NAME="group"	<INPUT TYPE="checkbox" NAME="group" VALUE="*">	ch1 ch2 ch3
Раскрывающийся список NAME="list"	<SELECT NAME="list"> <OPTION> Первый <OPTION> Второй </SELECT>	
Текстовая область NAME="resume"	<TEXTAREA> NAME="resume" ROWS=? COLS=?> </TEXTAREA>	1,2,3 ...
Кнопка <i>Отправить</i>	<INPUT TYPE="submit" VALUE="Отправить">	
Кнопка <i>Очистить</i>	<INPUT TYPE="reset" VALUE="Очистить">	

### Лабораторная работа №1 Тема: «Основные теги HTML»

Цель работы:

Научить учащихся:

- созданию первичного HTML-документа,

- использованию основных тегов,
- построению простейших HTML-документов.

### 1.1. Подготовка HTML-документа

Документ HTML состоит из основного текста документа и тегов разметки, которые, как мы уже знаем, являются наборами обычных символов. Таким образом, документ HTML — это, по существу, обычный текстовый файл. Для его создания можно использовать любой текстовый редактор, хотя бы и тот простейший редактор Блокнот, который входит в состав Windows 98/2000/XP.

Простейший HTML-документ – это просто текстовый файл с расширением ИМЯ.htm. Можно набрать следующий самый простой HTML-документ в блокноте:

```
<html>
<head> Моя первая страница
<title> Пример 1 </title>
</head>
<body>
<H1>
Привет!
</H1>
<P> Это простейший пример HTML-документа. </P>
<P>
```

Этот \*.htm-файл может быть одновременно открыт и в Notepad, и в Internet Explorer. Сохранив изменения в Notepad, просто нажмите кнопку Обновить в Internet Explorer, чтобы увидеть эти изменения реализованными в HTML-документе.

```
</P>
</body>
</html>
```

Теперь нужно этот текст сохранить на Рабочем столе под именем, например, ПРОБА.HTM, далее его нужно закрыть. Для просмотра этого текста как в Интернете, нужно загрузить с рабочего стола снова файл ПРОБА. HTM. Загрузится этот текст в браузере Internet Explorer.

Результат этого фрагмента HTML-программы на экране:

```
Моя первая страница
Привет!
```

Это простейший пример HTML-документа.

Этот \*.htm-файл может быть одновременно открыт и в Notepad, и в Internet Explorer. Сохранив изменения в Notepad, просто нажмите кнопку

Обновить в Internet Explorer, чтобы увидеть эти изменения реализованными в HTML-документе.

Для изменения исходного текста программы в виде HTML-программы выполните команду Вид – В виде HTML и после каждого изменения нужно выполнить команду Файл – Сохранить.

Далее, чтобы просмотреть эти изменения в браузере нужно перейти в Internet Explorer и выполнить команду Обновить.

Для удобства чтения введены дополнительные отступы, однако в HTML это совсем не обязательно. Более того, браузеры просто игнорируют символы конца строки и множественные пробелы в HTML-файлах. Поэтому наша программа вполне мог бы выглядеть и вот так:

```
<html>   <head>   <title> Пример 1</title> </head>   <body>
<H1>Привет!</H1>
```

```
<P> Это простейший пример HTML-документа.</P>
```

<P> Этот \*.htm-файл может быть одновременно открыт и в Notepad, и в Internet Explorer. Сохранив изменения в Блокнот, просто сохраните как ИМЯ.НТМ . Далее закройте и загрузите документ ИМЯ.НТМ , чтобы увидеть эти изменения реализованными в HTML-документе.</P>

```
</body> </html>
```

Результат работы на экране:

Привет!

Это простейший пример HTML-документа.

Этот \*.htm-файл может быть одновременно открыт и в Notepad, и в Internet Explorer. Сохранив изменения в Блокнот, просто сохраните как ИМЯ.НТМ . Далее закройте и загрузите документ ИМЯ.НТМ, чтобы увидеть эти изменения реализованными в HTML-документе.

## 1.2. Теги определения структуры документа HTML

Все документы HTML имеют строго заданную структуру. Документ должен начинаться с тега <HTML> и заканчиваться соответствующим закрывающим тегом </HTML>. Эта пара тегов сообщает браузеру, что перед ним действительно документ HTML. Тег <HTML> должен открывать документ и, аналогично, метка </html> должна завершать HTML-документ.

Документ HTML состоит из раздела заголовков и тела документа, идущих именно в таком порядке. Раздел заголовков заключен между тегами <HEAD> и </HEAD> и содержит информацию о документе в целом. В частности, этот раздел должен содержать внутри себя теги <TITLE> и </TITLE >, между которыми размещают заголовок документа. Все, что находится между тегами <title> и </title>, толкуется браузером как название

документа. Internet Explorer, например, показывает название текущего документа в заголовке окна и печатает его в левом верхнем углу каждой страницы при выводе на принтер.

Сам текст документа располагается в теле документа. Тело документа располагается между тегами <BODY> и </BODY>.

Четыре перечисленных парных тега определяют основную структуру документа HTML. Они встречаются (или их наличие подразумевается) во всех документах HTML.

На практике определить местоположение этих основных структурных тегов можно и при их отсутствии. Поэтому, если теги <HTML>, <HEAD> и <BODY>, а также соответствующие им закрывающие теги опущены, то программа-браузер может сама определить то место, где они должны были находиться. Тег <TITLE>, определяющий заголовок документа, считается обязательным, но и его пропуск не вызовет никаких изменений в современных браузерах. Но все-таки при создании Web-страниц опускать все эти теги не рекомендуется, ведь заранее неизвестно, как поведет себя конкретный браузер, установленный на компьютере пользователя.

### 1.3. Основные теги HTML-документа

Язык HTML предназначен для описания функциональных разделов документа. В большинстве документов основными разделами являются заголовки и абзацы.

HTML поддерживает шесть уровней внутренних заголовков документа. Они помечаются тегами от <H1> и </H1> до <H6> и </H6>. Заголовок первого уровня – самый крупный, шестого уровня, естественно – самый мелкий. Реально на экране компьютера все эти заголовки изображаются шрифтами полужирного начертания.

В соответствии с идеологией HTML текст, который действительно является заголовком, следует пометить с помощью одного из этих тегов. В составе языка имеются теги форматирования, изменяющие размер и начертание шрифта, но пользоваться ими в этом случае не рекомендуется.

Для обозначения обычных абзацев в языке HTML используют тег <P> (и соответствующий закрывающий тег </P>). Теги, описывающие абзацы, являются необязательными. Однако при наличии этих тегов браузеры четко отслеживают границы между абзацами.

Теги <H1> ... <H6> и <P> могут содержать дополнительный атрибут ALIGN (читается "элайн", от английского "выравнивать"), <P ALIGN=left|center|right>

позволяют выравнивать абзац по левому краю, центру и правому краю соответственно. Например:

```
<H1 ALIGN=CENTER> Выравнивание заголовка по центру </H1>  
или
```

```
<P ALIGN=RIGHT> Образец абзаца с выравниванием по правому краю  
</P>
```

Вы можете центрировать все элементы документа в окне браузера. Для этого можно использовать тег <CENTER>. Все элементы между тегами <CENTER> и </CENTER> будут находиться в центре окна.

В языках программирования общепринята возможность использования комментариев — текстовых строк, не являющихся частью программы, а служащих для пояснения. Язык HTML тоже имеет такую возможность.

#### Комментарии

Как любой язык, HTML позволяет вставлять в тело документа комментарии, которые сохраняются при передаче документа по сети, но не отображаются браузером. Комментарии в языке HTML начинаются со специального тега <!--. Следует обратить внимание на отсутствие закрывающей угловой скобки. Любой текст, идущий за этим тегом, рассматривается как комментарий и не отображается на экране браузера при просмотре документа. Заканчивают комментарий символами -->. Комментарий может содержать любые символы, кроме символа «больше» (>), и, таким образом, не может включать в себя теги.

Синтаксис комментария:

```
<!-- Это комментарий -->
```

Комментарии могут встречаться в документе где угодно и в любом количестве.

В языке HTML для создания абзаца браузер обычно вводит пустую строку. Важным средством создания разделителей в тексте являются горизонтальные полосы-линейки, визуально отделяющие разные части документа друг от друга. Горизонтальная линейка создается тегом <HR>. Это одиночный тег, не имеющий соответствующего закрывающего тега.

При применении этих тегов важно принять во внимание, что все кратные («лишние») пробелы между словами и переходы на новую строку при воспроизведении документа HTML игнорируются.

Подытожим все, что мы знаем, и поработаем с помощью примера 2:

```
<html>
  <head>
    <title> Пример 2 </title>
  </head>
  <body>
    <H1 ALIGN=CENTER> Привет! </H1> <HR>
    <H2>Это чуть более сложный пример HTML-документа </H2>
    <P> Теперь мы знаем, что абзац можно выравнивать не только влево,
</P>
    <P ALIGN=CENTER>но и по центру</P> <P ALIGN=RIGHT>или по
правому краю.</P>
  </body>
</html>
```

Результат на экране:

С этого момента Вы знаете достаточно, чтобы создавать простые HTML-документы самостоятельно от начала до конца.

В языке HTML нет средств для создания абзацного отступа («красной строки»), поэтому для удобочитаемости текста между абзацами применяют тег <P> или используется тег <PRE>.

Тег преформатирования – <PRE>, позволяет представлять текст со специфическим форматированием на экране. Предварительно сформатированный текст заканчивается завершающим тегом </PRE>. Внутри предварительно сформатированного текста разрешается использовать:

- перевод строки;
- символы табуляции (сдвиг на 8 символов вправо);
- непропорциональный шрифт, устанавливаемый браузером.

Использование тегов, определяющих формат абзаца, таких как <Hx> или <ADDRESS>, будет игнорироваться браузером при помещении их между тегами <PRE> и </PRE>.

Теперь рассмотрим, как можно улучшить наш простой HTML-документ. Если же надо осуществить переход на новую строку без создания абзаца, можно использовать тег <BR>. Он очень удобен при публикации стихов. Следующий пример демонстрирует использование этого тега.

Дополнительный атрибут позволяет расширить возможности тега <BR>.  
<BR CLEAR=left|right|all>

Данный атрибут позволяет выполнить не просто перевод строки, а разместить следующую строку, начиная с чистой левой (left), правой (right) или обеих (all) границ окна браузера. Например, если рядом с текстом слева встречается рисунок, то можно использовать тег <BR> для смещения текста ниже рисунка:

```
<p> Эта моя собачка <BR CLEAR=left>  
  
Ее кличка Актос </p>  
Неразрывная строка <NOBR>
```

Если вы не хотите, чтобы браузер автоматически переносил строку, то вы можете обозначить ее тегами <NOBR> и </NOBR>. В этом случае браузер не будет переносить строку, даже если она выходит за границы экрана; вместо этого браузер позволит горизонтально прокручивать окно. Например:

```
<NOBR> Данная строка является самой длинной строкой документа,  
которая не допускает какого-либо разбиения, где бы то ни было </NOBR>
```

Если же вы хотите все же позволить разбиение данной строки на две, но в строго запланированном месте, то вставьте тег <WBR> в это место. Например:

```
<NOBR> Данная строка является самой длинной строкой  
документа,<WBR> которая не допускает какого-либо разбиения где бы то ни  
было </NOBR>
```

Цитата <BLOCKQUOTE>

Данный тег предназначен для обозначения в документе цитаты из другого источника. Текст, обозначенный тегом <BLOCKQUOTE>, отступает от левого края документа на 8 пробелов. Например:

На первой лекции преподаватель произнес: <P>

<BLOCKQUOTE>

Сегодня один из знаменательных дней для вашей группы. <BR>

Мы изучаем новую технологию, позволяющую Вам открыть свои сайты с помощью простого языка разметки гипертекста – HTML-кода.

</BLOCKQUOTE>

При отображении браузером данный текст будет выглядеть так:

Список базовых тегов HTML

Стартовый

Завершающий

Описание

<HTML>

</HTML>

Обозначение HTML-документа

<HEAD>

</HEAD>

Заголовочная часть документа

<TITLE>

</TITLE>

Заголовок документа

<BODY>

</BODY>

Тело документа

<H1>

</H1>

Заголовок абзаца первого уровня

<H2>

</H2>

Заголовок абзаца второго уровня

<H3>

</H3>

Заголовок абзаца третьего уровня

<H4>

</H4>

Заголовок абзаца четвертого уровня

<H5>

</H5>

Заголовок абзаца пятого уровня

<H6>

</H6>

Заголовок абзаца шестого уровня

<P>

</P>

Абзац

<PRE>

</PRE>

Форматированный текст

<BR>

Перевод строки без конца абзаца

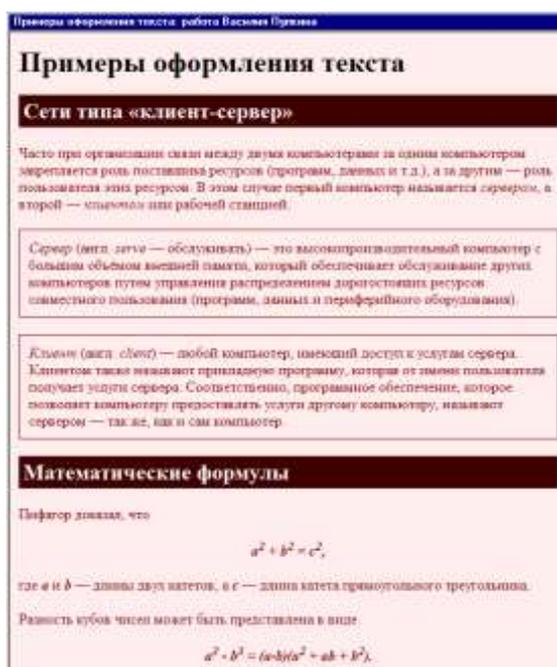
<BLOCKQUOTE>

</BLOCKQUOTE>

Цитата

## Практическая работа № 2 «Использование CSS»

## Задание 1. Использование CSS. Часть 1



1. Откройте **text.htm**, полученный в результате предыдущей работы.

2. Создайте в той же папке стилевой файл **mystyle.css**, подключите его к веб-странице и определите стили оформления, перечисленные в следующих пунктах. Для получения дополнительной справочной информации по CSS используйте Интернет, например, сайт [css.manual.ru](http://css.manual.ru).

3. Задайте свои цвета фона и текста. Текст должен хорошо читаться на выбранном фоне.

4. Определите цвет заголовков (**H1**, **H2**), отличающийся от цвета

основного текста. Заголовки **H2** должны также выделяться фоном, причем фон должен быть темный, а буквы — светлые. Можно добавить внутренний отступ (**padding**) в 5 пикселей.

5. Определите класс **formula**, применимый как к абзацам (**P**), так и к отдельным словам (**SPAN**), и задайте для него *выравнивание по центру*; *жирный шрифт*; *курсив*. Используя этот класс, выделите абзацы с формулами и имена переменных в тексте.

6. Определите класс **definition**, применимый только к абзацам (**P**), и задайте для него: *рамку толщиной в 1 пиксель*; *внутренний отступ в 10 пикселей*.

Используя этот класс, выделите два абзаца с определениями.

## **Задание 2. Использование CSS. Часть 2**

1. Скопируйте на свой компьютер каталог **CSS**. Откройте файл **cats.htm**. Все последующие задания нужно выполнять с помощью CSS.

2. Установите для тела страницы бледно-розовый фон (цвет #FFEEEE) и шрифт *Georgia*, а если его нет – любой шрифт с засечками (*serif*).

3. Для заголовков **H1** и **H2** определите шрифт *Arial*, если его нет – *Helvetica*, а если нет их обоих, то любой рубленый шрифт (*sans-serif*).

4. Для заголовка **H1** задайте фоновый рисунок **cat.png** из подкаталога **images** (без повторения).

5. Для заголовков **H2** определите темно-красный цвет (#800000).

6. Создайте класс оформления **latin** для выделения слов и выражений на латинском языке: курсив, тёмно-красный цвет. Выделите в тексте все латинские слова с помощью этого стиля.

7. Установите для всех абзацев отступ (красную строку) 30 пикселей.

8. Создайте новый стиль оформления абзацев – **author**: курсивный шрифт, без абзацного отступа. Выделите этим стилем имена и фамилии авторов цитат.

9. Создайте стиль оформления абзацев с цитатами с именем **quote**:

- 1) фон – светло-жёлтый (#FEFEE2);
- 2) абзацного отступа нет;
- 3) ширина 50% от ширины окна браузера;
- 4) внешние отступы: сверху – 0, справа и слева по 30 пикселей, снизу – 10 пикселей;
- 5) рамка шириной 1 пиксель, точечная (*dotted*), черного цвета;
- 6) внутренние отступы 10 пикселей.

10. Найдите в Интернете информацию про авторов цитат и сделайте их имена ссылками на соответствующие страницы.

11. Установите для ссылок, которые находятся внутри абзаца стиля **author**, красный цвет при наведении мыши.

## **Задание 3. Вставка рисунков в документ**

1. Скопируйте на свой компьютер каталог **IMG**. Откройте файл **img.htm** в редакторе.

Ваша задача – оформить веб-страницу про некоторые направления в музыке XX века. Для оформления нужно использовать рисунки из подкаталога **images**.

2. Добавьте заголовок страницы (**TITLE**), укажите там название документа и фамилию автора, например:

*Веб-страница с рисунками: работа Василия Пупкина*

3. Найдите и выделите соответствующими тэгами заголовков документа и заголовки разделов.

4. Разбейте текст на абзацы.

5. Замените верхние кавычки на «ёлочки».

6. Замените везде знаки «минус» на тире, сделайте так, чтобы тире не отрывались от предыдущих слов.

7. Создайте стилевой файл **mystyle.css** и подключите его к веб-странице.

С помощью стилового файла:

1. Установите в качестве фона страницы рисунок **back.jpg**;

2. Выберите соответствующий цвет текста.

3. Создайте класс **name**, определите для него жирный шрифт, курсив и свой цвет, и выделите имена и фамилии, названия ансамблей, фильмов, песен (тэг **SPAN**).

8. Добавьте на страницу рисунки, расположенные в подкаталоге **images**.

9. Установите для рисунков выравнивание (с обтеканием текстом) и всплывающие подсказки. Рисунки лучше ставить в шахматном порядке (один налево, следующий – направо и т.д.)

10. С помощью стилового файла установите для рисунков внешний отступ (**margin**) 10 пикселей.

#### **Задание 4. Вставка звука и видео в документ**

1. Скопируйте на свой компьютер каталог **MEDIA**. Откройте файл **media.htm** в редакторе. Ваша задача – научиться добавлять на веб-страницу звуковые файлы, флэш-анимацию и видеоролики. Для этого нужно использовать мультимедийные данные из подкаталогов **audio**, **flash** и **video**. Места вставки медиафайлов вы легко определите из текста файла **media.htm**.

2. Добавьте на веб-страницу звуковой файл **hcopter.wav** из каталога **audio** (шум вертолёт). Используйте тэг **embed**, отключите автозапуск проигрывания, установите ширину проигрывателя 300 пикселей и высоту 20 пикселей.

3. Добавьте на веб-страницу звуковой файл **nature.mp3** из каталога **audio** (звуки природы). Используйте тэг **embed**, укажите те же атрибуты, что и в предыдущем случае.

4. Добавьте на веб-страницу звуковой файл **nature.ogg** из каталога **audio** (звуки природы). Используйте тэг **audio**, укажите его атрибуты:

**src="адрес файла"** ссылка на файл-источник

**controls="controls"** показывать элементы управления

**width="300"** ширина проигрывателя в пикселях

**height="20"** высота проигрывателя в пикселях

Откройте веб-страницу в браузере и выясните, поддерживает ли ваш браузер тэг **audio**.

5. С помощью тэге **embed** добавьте на страницу флэш-ролик **cube.swf** (вращающийся кубик) из каталога **flash**. Установите размеры 257 на 200 пикселей.

6. С помощью тэге **embed** добавьте на страницу флэш-ролик **lightbot.swf** (игра *Lightbot*) из каталога **flash**. Установите размеры 640 на 480 пикселей. Откройте веб-страницу в браузере и посмотрите, как работает флэш-анимация.

7. Зайдите на сервис **youtube.com**, с помощью поиска по сайту найдите ролик по теме «HTML5» и вставьте на веб-страницу. Для этого нужно щелкнуть по кнопке *Поделиться* под роликом, затем – по строчке *HTML-код*. После этого остается просто скопировать код, появившийся в текстовом окне, на веб-страницу.

8. С помощью тэга **video** добавьте на страницу видеофайл **finish.mp4** из каталога **video** следующим образом:

```
<video width="ширина" height="высота" controls="controls">  
<source src="адрес файла"  
type='video/mp4; codecs="avc1.42E01E, mp4a.40.2"'>  
</video>
```

Установите размеры окна 320 на 240 пикселей.

9. С помощью тэга **video** добавьте видеофайл **finish.webm** из каталога **video** следующим образом:

```
<video width="ширина" height="высота" controls="controls">  
<source src="адрес файла"  
type='video/webm; codecs="vp8, vorbis"'>  
</video>
```

Установите размеры окна 320 на 240 пикселей.

10. Обновите страницу в браузере и выясните, поддерживает ли ваш браузер тэг **video**, и с какими форматами он умеет работать.

\*\*\*Возможен *другой вариант* выполнения Практической работы № 2 «Использование CSS»

1. Скопируйте на свой компьютер каталог CSS. Откройте файл **cats.htm**. Все последующие задания нужно выполнять с помощью CSS.

2. Установите для тела страницы бледно-розовый фон (цвет #FFEEEE) и шрифт Georgia, а если его нет – любой шрифт с засечками (serif).

3. Для заголовков H1 и H2 определите шрифт Arial, если его нет – Helvetica, а если нет их обоих, то любой рубленый шрифт (sans-serif).

4. Для заголовка H1 задайте фоновый рисунок cat.png из подкаталога images (без повторения).

5. Для заголовков H2 определите темно-красный цвет (#800000).

6. Создайте класс оформления latin для выделения слов и выражений на латинском языке: курсив, тёмно-красный цвет. Выделите в тексте все латинские слова с помощью этого стиля.

7. Установите для всех абзацев отступ (красную строку) 30 пикселей.

8. Создайте новый стиль оформления абзацев – author: курсивный шрифт, без абзацного отступа. Выделите этим стилем имена и фамилии авторов цитат.

9. Создайте стиль оформления абзацев с цитатами с именем quote:

а) фон – светло-жёлтый (#FEFEE2);

б) абзацного отступа нет;

в) ширина 50% от ширины окна браузера;

г) внешние отступы: сверху – 0, справа и слева по 30 пикселей, снизу – 10 пикселей;

д) рамка шириной 1 пиксель, точечная (dotted), черного цвета;

е) внутренние отступы 10 пикселей.

10. Найдите в Интернете информацию про авторов цитат и сделайте их имена ссылками на соответствующие страницы.

11. Установите для ссылок, которые находятся внутри абзаца стиля author, красный цвет при наведении мыши.