

## Содержание:

# ВВЕДЕНИЕ

Прошло несколько десятков лет как Теодор Нельсон [\[1\]](#) впервые использовал термин «гипертекст», и концепция гипертекста, как способа нелинейной организации, получила существенное развитие.

Концепция организации информации, при которой линейность повествования заменяется возможностью перехода по разным ссылкам, первоначально использовавшейся в печатных изданиях.

До появления компьютеров основной формой гипертекста были энциклопедии, в которых уже не было линейного принципа, то есть положение статьи не играло никакой роли.

С появлением компьютеров стало возможным легко создавать системы страниц, связанных между собой ссылками, выполнять поиск по ключевым словам среди них, постоянно редактировать их содержимое. Примерами гипертекстовых систем являются Интернет и любой веб-сайт.

Теодор Нельсон впервые использовал термин «гипертекст» в докладе «Файловая структура для сложного, меняющегося и окончательно неопределимого» [\[2\]](#).

Со временем было предложено немало моделей гипертекста, некоторые из них, такие как ZOG, HyperCard, Intermedia или HyperTIES, сыграли важную роль в становлении нынешней концепции гипертекста, но сами стали, скорее, достоянием истории. Немногие из моделей продолжают развиваться и ныне, в их числе Xanadu – проект Т. Нильсена.

Наибольшее распространение, безусловно, получила модель гипертекста, основанная на языке HTML, и положенная в основу Интернета.

В начале 90-х годов XX века Тим-Бернерс Ли создал, а его последователи в дальнейшем усовершенствовали, основные инструменты, которые позволили создать интернет на основе гипертекста в его современном виде. Одним из этих инструментов был и есть язык гипертекстовой разметки HTML для верстки содержания веб-страниц сайтов в нужном формате.

Актуальность выбранной темы обусловлена тем, что и по сегодняшний день языки гипертекстовой разметки являются универсальным языком разметки, т.к. не требует интерпретации исходного кода.

Цель работы - изучить сущность и характеристики языков гипертекстовой разметки, представить пример практического применения языка гипертекстовой разметки.

Задачи работы:

- рассмотреть и изучить сущность языка гипертекстовой разметки;
- рассмотреть и изучить структуру документа HTML;
- дать сравнительную характеристику языкам гипертекстовой разметки;
- привести пример использования HTML.

Объект исследования - языки гипертекстовой разметки.

Предметом исследования является алгоритм верстки веб-страницы на языке разметки гипертекста HTML.

Для выполнения поставленных задач были изучены материалы научных статей и пособий; для выполнения практической части работы использовались материалы по структуре, основным тегам и алгоритму написания на языке разметки гипертекста HTML, размещенные на ресурсах Консорциума Всемирной паутины (World Wide Web Consortium, W3C).

## **1. Язык гипертекстовой разметки**

### **1.1 Язык гипертекстовой разметки: основные понятия и сущность**

Для начала, определимся, что такое – гипертекст.

Современная электронно-опосредованная коммуникация предлагает человеку всё новые тексты в разных сферах знания, в разных утилитарных сферах. При этом большинство таких текстов созданы по принципу гипертекста, который

представляет собой многоуровневую структуру, совокупность микротекстов, интегрированных с помощью ссылок в единый массив. Гипертекст характеризуется целостностью, внутренней осмысленностью, информативностью, а также интеграцией вербальных и невербальных средств[3].

Наступление новой эры информационных технологий предполагает перенос сфер человеческой деятельности в электронную среду, в связи с чем мы можем наблюдать интенсивное развитие и активное использование гипертекстовых систем. В современном мире феномен гипертекста стал объектом многих дискуссий, конференций и научно-исследовательских работ по психологии, лингвистике и информатике. Сегодня большое количество организаций в мире проводят масштабные исследования, которые базируются на концепции ассоциативной навигации. Гипертекстовые системы с успехом используются в технике, образовании, экономике и других сферах деятельности. Представляется целесообразным обратиться к истории возникновения термина «гипертекст».

Понятие «гипертекст» было введено в употребление в 1965 году программистом и математиком Теодором Нельсоном[4], который отмечал, что в отличие от традиционного текста с его последовательностью и развернутыми, тесными линейными связями, гипертекст представляет собой нелинейную организацию идей, разветвлённую, взаимосвязанную документацию, а также наличие развитой системы переходов. Читатель обладает возможностью изучать заложенную в данной документации информацию в той очерёдности, какую он выберет самостоятельно. Гипертекст состоит из фрагментов, которые не построены в однозначно заданном порядке. Следовательно, читатель вправе прокладывать разные пути по документации за счёт гиперссылок и в зависимости от своих предпочтений.

Согласно одному из энциклопедических определений гипертекст представляется «в виде графа, в узлах которого находятся традиционные тексты или их фрагменты, изображения, таблицы, видеоролики и т. д.».

Тед Нельсон даёт следующую трактовку гипертексту: «гипертекст - это не последовательное сочинение, а текст, который разветвляется и позволяет читателю выбирать, т.е. это ряд кусков текста, соединённых линками, предлагающими читателю различные пути»[5].

Тим Бернерс-Ли даёт следующее определение: «гипертекст — это соединение смысловой структуры, структуры внутренних связей некоего содержания, и

технической среды, технических средств, дающих возможность человеку осваивать структуру смысловых связей, осуществлять переходы между взаимосвязанными элементами»[\[6\]](#).

Под гипертекстом также понимают надтекст, единицу информации, частями которой являются тексты, части которого имеют «сверхсвязи», то есть соединены не линейным отношением в одномерном пространстве (как в обычном тексте естественного языка), а множеством различных отношений, представляемых в многомерном пространстве.

Гипертекст дает возможность связать текстовую информацию, аудио, фотографии, чертежи, карты, динамичные картинки и иные формы предоставления информации в единое осмысленное целое, доступ к которому возможен посредством системы индексации, которая ориентирована на конкретные идеи, а не слова в тексте.

Таким образом, под гипертекстом стоит понимать совокупность обычных фрагментов текста, связь между которыми реализуется посредством ссылок, встроенных в информационную систему, что позволяет пользователям осуществлять переход от одного текстового фрагмента к другому по их выбору.

Для гипертекста характерным является интерфейсная модальность, проявляющаяся в наличии пиктограмм, икон связей, кнопок, которые пользователь выбирает произвольно[\[7\]](#).

Рассмотрим ключевые черты гипертекста:

- отсутствие непрерывности и наличие прыжка. Данная черта исключает необходимость делить гипертекст на тему и рему, выделять в нём эксплицитные и имплицитные слои по сравнению с обычным линейным текстом;
- множественность автора, то есть изменение гипертекста влечёт за собой изменение коллектива авторов. Отсюда следует, что в гипертексте представлена не одна точка зрения в противоположность классическому тексту. Данное качество гипертекста позволяет судить о нем как о многогранном тексте, поэтому гипертекст является более объективным и толерантным к читателю, чем традиционный текст;
- отсутствие автора;
- нечеткие, размытые границы между автором и читателем;

- объективность, многосторонность;
- интерактивность;
- незаконченность, открытость, обусловленные масштабом электронной сети;
- неоднородность;
- диалогичность (заложенная в тексте информация требует от читателя отклика);
- виртуальность (гипертекст функционирует в электронном коде);
- дисперсность структуры (информация в тексте представлена в виде «гнёзд», а войти в гипертекстовую сеть можно с любого её звена);
- дистанционность и доступность (автор текста и читатель могут обращаться к гипертексту в любое время, находясь при этом на большом расстоянии друг от друга);
- мультимедийность (применение технических средств воздействия на реципиента);
- композиционная нестабильность (альтернативность маршрутов навигации; возможность создавать собственную читательскую модель).

Для построения такого контента и используются языки гипертекстовой разметки.

Прорыв в применении электронной версии гипертекста имел место во время зарождения интернета и связан с именем Тима-Бернерса Ли, который в начале 90-х годов XX века создал, а его последователи в дальнейшем усовершенствовали, основные инструменты, позволяющие создать интернет на основе гипертекста в его современном виде[8].

Следует отметить, что для демонстрации обширного текста, имеющего много информации, т.е. для организации эффективного гипертекста необходимо, чтобы вводимые ссылки были точными, оправданными и соответствовали по уровню сложности, чтобы структура текста была наглядной, а навигация по нему не доставляла бы неудобств.

Гипертекстовая технология представляет собой создание, поддержку, расширение и просмотр текста, представленного в виде сети. Поддерживающие эту технологию программы базируются на следующих функциях гипертекста:

- ссылки – являются самой важной функцией, благодаря им имеется возможность использовать связи информационной сети, в образе которой представлен гипертекст, через которые можно обращаться к связанным на уровне ассоциаций узлам. Так текст можно пролистывать в диалоговом режиме посредством использования компьютерной мыши, а можно использовать запросы, автоматически фильтрующие данные.

- замещение - позволяет заменять при просмотре текста любую часть информации рисунком или иным фрагментом текста;

- заметки - стандартные заметки на полях, но гораздо более выразительные (видеофайлы, цветная графика, звуковое сопровождение и др.). Следует отметить, что данная функция является обратной, т.е. информация связывается с выделенным фрагментом, а не наоборот;

- запросы – помогают проводить анализ текста с определенных пользователем позиций. Так можно использовать данные, не имеющие отношение к некоторому фрагменту текста, но при этом с ним связанные. Поиск можно производить по ключевым словам при сопоставлении с каким-либо образцом или по разделителям на основании фильтрации необходимых знаний из гипертекста[9].

Запросы в совокупности с замещением позволяют получать срезы исходного текста, являющиеся функционально-ориентированными, т.е. структурировать гипертекст, исходя из потребностей каждого отдельного пользователя.

Следует подчеркнуть, что электронная книга также обладает опцией поддержки связи между ссылками, т.е. смысловые связи в информации можно использовать и без компьютера. Однако связи, поддерживаемые техникой, позволяют вывести уровень чтения на более высокий качественный уровень.

Таким образом, идея гипертекстовой системы, зародившаяся еще в середине прошлого столетия, получила развитие, и была реализована на практике как основа Интернета, с которой сегодня сталкивается каждый ежедневно.

Основное преимущество гипертекста состоит в нелинейности повествования, которая позволяет создавать сразу несколько линейных последовательностей в изучении материала: захотел — перешел по одной ссылке, захотел — по другой либо продолжил чтение. При этом такие действия можно совершать не только в пределах одной веб-страницы какого-либо сайта, но по всей сети, при этом, не изучая всю предоставленную информацию, а, только выбирая нужную, что

позволяет сэкономить много времени[10].

Именно принцип разрыва линейности текста стал основополагающим при создании Всемирной паутины. Так Тимом Бернесом-Ли в целях реализации принципа гипертекста и гиперссылок и его последователями были созданы следующие основные инструменты, которые позволили создать интернет на основе гипертекста в его современном виде:

- язык разметки гипертекста HTML – кодирует содержание документов (веб-страниц сайтов) в нужный формат;
- протокол передачи гипертекста HTTP (HTTPS) - передает по сети информацию в электронном виде;
- система адресации URI и URL - предусматривает указание протокола передачи данных в составе адреса того или иного документа в сети;
- браузеры - позволяют реализовать все перечисленные выше возможности, а именно: отобразить текст, закодированный с помощью средств HTML, в удобном для чтения виде; осуществить прямой переход на нужную страницу любого веб-сайта путем ввода URL в адресную строку браузера; следовать по созданным гиперссылкам, находящимся в составе контента веб-страницы.

Таким образом, под гипертекстом стоит понимать совокупность обычных фрагментов текста, связь между которыми реализуется посредством ссылок, встроенных в информационную систему, что позволяет пользователям осуществлять переход от одного текстового фрагмента к другому по их выбору. Основное преимущество гипертекста состоит в нелинейности повествования, которая позволяет создавать сразу несколько линейных последовательностей в изучении материала, что позволяет, не изучая всю предоставленную информацию, а только выбирать нужную, что позволяет сэкономить много времени[11].

## **1.2. Характеристика и сравнение языков разметки гипертекста**

Языки разметки позволяют создавать гибкую форму представления информации. С их помощью стало возможным демонстрировать информацию в удобном человеку виде, что активно используется во всех сферах жизни. Хотя основные задачи этих

языков заключаются в структурировании представления информации и облегчении ее восприятия, сами же языки разметки обладают рядом проблем, которые затрудняют восприятие их самих как человеком, так и программными средствами.

Языки разметки используются для структурирования и корректного представления любого вида информации[12]. Являясь инструментарием для выражения информации в удобной человеку форме, языки разметки отличаются строгой формализованностью, как и любые другие языки, отвечающие за взаимодействие компьютера и человека. И хотя наполнение семантикой тезауруса семейства языков разметки, произошедших от SGML, упрощает понимание исходного кода для человека, их синтаксис заставляет мысленно представлять структуру документов в виде дерева визуализации. Человеку же более удобно воспринимать естественные языки. Отсюда возникает необходимость пересмотра представления разметки и создание нового подхода, который был бы ориентирован в первую очередь на восприятие языка человеком.

Так в 1986г. Международная организация по стандартизации (ISO) приняла первый стандарт ISO-8879, именуемый «Standard Generalized Markup Language (SGML)». Этот стандарт был посвящен описанию языка SGML - обобщенного метаязыка, позволяющего строить системы логической и структурной разметки любых разновидностей текстов. При этом следует подчеркнуть, что в такой разметке управляющие коды, вносимые в текст, не несут информации о внешнем виде документа, а только задают его логическую структуру, т.е. указывают границы и соподчинение его составных частей[13].

Следует отметить, что SGML является наследником разработанного в 1960 году в IBM языка GML (Generalized Markup Language - обобщенный язык разметки), который представлял собой набор макросов для реализации разметки и использовал теги для оформления текста, используя процедурную разметку SCRIPT, определяющую, чем текст является с точки зрения параграфов, заголовков, списков и таблиц, что позволяло осуществить автоматическое форматирование текста для различных устройств, необходимо было только указать разновидность устройства.

SGML широко использовался в печатной и издательской сфере, но ввиду его сложности широкого распространения для повседневного использования он не получил.

SGML документ состоит из следующих основных частей:



- SGML декларация – для определения символов и ограничителей, которые могут появляться в приложении;
- Document Type Definition (DTD) – тип документа, для определения синтаксиса конструкций разметки;
- спецификация семантики – для определения ограничений синтаксиса, которые не могут быть выражены внутри DTD;
- содержимое SGML-документа - должен быть хотя бы корневой элемент.

Отличительными характеристиками SGML являются:

- упор на описательную, а не на процедурную разметку. Так система описательной разметки использует коды разметки лишь предоставляющие названия для классификации частей документа, а инструкции, необходимые для обработки документа собираются вне документа в отдельных процедурах или программах. Так один и тот же документ может быть обработан различными программами, применяющими различные правила обработки к определенным частям документам в зависимости от степени их важности;
- концепция типа документа. Так тип документа определяется набором частей документа и их структурой. Ввиду того, что документ может быть отнесен к какому-либо известному типу, по определенным атрибутам программа анализатор проверяет, что все элементы, требуемые типом документа, на самом деле присутствуют и расположены в правильном порядке;
- независимость от конкретной системы в представлении текста – позволяющая обеспечить транспортабельность закодированных документов из одной аппаратной и программной среды в другую без потери информации на уровне строк байтов (символов), из которых составляется документ.

На сегодняшний день семейство языков, произошедших от SGML (Standart Generalized Markup Language), занимает доминирующие позиции в задачах представления информации. Наиболее часто используемыми языками этого семейства являются XML, HTML и XHTML, семантика и синтаксис которых были разработаны в конце прошлого столетия[14]. В числе ключевых особенностей данных языков можно выделить:

- 1) относительно компактный базовый тезаурус;

2) возможность добавления в тезаурус пользовательских литеральных последовательностей;

3) строго формализованная модель синтаксического анализа исходных кодов.

Несмотря на синтаксические различия, языки данного семейства основаны на базовом семантическом принципе: блочная структура кода использует модель контейнеров данных, каждый из которых помечается тегом, определяющим способ интерпретации данных.

Фундаментальным недостатком такой модели является низкая человекочитаемость (мера способности оператора-человека анализировать содержимое документа), обусловленная:

1) сложной структурой кода;

2) большим количеством специальных символов.

Таким образом, SGML решал проблему стандартизации обмена данными между разнородными программами и платформами. Однако являлся очень сложным. Кроме того, SGML-документ было трудно интерпретировать без определений языка разметки, который хранится в определении типа документа, который необходимо было посылать вместе с самим SGML-документом, чтобы распознать теги, созданные пользователем. Поэтому язык SGML не получил масштабного распространения, однако лег в основу языка HTML (Hyper Text Markup Language - язык разметки гипертекста).

По сути, HTML представляет собой набор предписаний SGML, сформулированных в виде DTD. В случае языка HTML DTD хранится в браузере и этот DTD во много раз меньше, чем DTD SGML.

HTML - это технология представления данных, которая не несет содержательной информации, помимо той, которая заключена в тегах, при этом значимость тегов здесь не определена, поэтому нельзя представить иерархию данных[\[15\]](#).

Язык HTML интерпретируется браузерами - полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства.

Существует девять версий языка HTML.

HTML 2.0 — опубликован IETF как RFC 1866 в статусе Proposed Standard (24 ноября 1995 года);

HTML 3.0 — 28 марта 1995 года — IETF Internet Draft (до 28 сентября 1995 года);

HTML 3.2 — 14 января 1997 года;

HTML 4.0 — 18 декабря 1997 года;

HTML 4.01 — 24 декабря 1999 года;

ISO/IEC 15445:2000 (так называемый ISO HTML, основан на HTML 4.01 Strict) — 15 мая 2000 года;

HTML5 — 28 октября 2014 года;

HTML 5.1 начал разрабатываться 17 декабря 2012 года. Рекомендован к применению с 1 ноября 2016 года.

HTML 5.2 был представлен 14 декабря 2017 года.

Официальной спецификации HTML 1.0 не существует. До 1995 года существовало множество неофициальных стандартов HTML. Чтобы стандартная версия отличалась от них, ей сразу присвоили второй номер.

В версии HTML 3.0 в целях разрешения противоречия между идеологией структурной разметки и потребностями пользователей в гибких и разнообразных возможностях визуального представления информации, ввели иерархические стилевые спецификации (Cascading Style Sheets - CSS). Система CSS формально независима от HTML, имеет совершенно иной синтаксис, не наследует никаких идеологических ограничений и позволяет уже совершенно в иных терминах задавать параметры визуального представления для любого тега HTML.

В HTML.3.2 и HTML.4.0 разработчики попытались объединить большинство нововведений языка, появившихся после выхода версии 3.0, с учетом современных концепций развития компьютерных технологий, в частности Мультимедиа.

В поддержку языка HTML была создана целая серия программных языков, таких, как Java, JavaScript, Perl, PHP и т.д. Тем самым открылась возможность для создания динамических Web-страниц, включающих формы, таблицы, фреймы и другие элементы.

Следует подчеркнуть, что HTML до его 5-ой версии определялся как приложение SGML. Спецификации HTML5 формулируются в терминах DOM (объектной модели документа).

Достоинства и недостатки HTML представлены в Таблице 1.1. Приложения 1.

В 1998г. члены рабочей группы Консорциума World Wide Web, упростив SGML для использования в Web, предложили расширяемый язык разметки XML (eXtensible Markup Language - расширяемый язык разметки).

XML представляет собой подмножество SGML, причем любой действительный документ XML является действительным документом SGML. Как и SGML, XML - это метаязык, определяющий другие языки разметки для специфических целей. Например, язык синхронизированной интеграции мультимедиа (Synchronized Multimedia Integration Language - SMIL) базируется на XML[\[16\]](#).

XML предназначен для хранения структурированных данных (взамен существующих файлов баз данных), для обмена информацией между программами

Целью создания XML было обеспечение совместимости при передаче структурированных данных между разными системами обработки информации, особенно при передаче таких данных через Интернет. Словари, основанные на XML (например, RDF, RSS, MathML, XHTML, SVG), сами по себе формально описаны, что позволяет программно изменять и проверять документы на основе этих словарей, не зная их смыслового значения.

XML используется для разметки стандартных документов во многом так же, как HTML. Однако XML ориентирован на работу со структурированными данными, такими, как результаты запроса, метаинформация о Web-узле или элементы и тип схемы.

Документ XML внешне похож на HTML. Он также состоит из текстовых фрагментов, аннотированных заключенными в угловые скобки тегами. Однако, в отличие от HTML, смысл тега зависит от регистра, а каждый открывающий тег должен во всех случаях иметь парный закрывающий тег.

В Таблице 1.2. Приложения 1 приведены основные преимущества и недостатки языка XML.

Язык XML целесообразно использовать разработчиками сложных информационных систем, с большим количеством приложений, связанных потоками информации

самой различной структурой. В этом случае XML - документы выполняют роль универсального формата для обмена информацией между отдельными компонентами большой программы.

XML является базовым стандартом для нового языка описания ресурсов, RDF, позволяющего упростить многие проблемы в Web, связанные с поиском нужной информации, обеспечением контроля за содержимым сетевых ресурсов, создания электронных библиотек и т.д.[\[17\]](#) Он также позволяет описывать данные произвольного типа и используется для представления специализированной информации, например химических, математических, физических формул, медицинских рецептов, нотных записей, и т.д. Следовательно, XML может служить мощным дополнением к HTML для распространения в Web "нестандартной" информации.

Вся информация, располагающаяся между начальным и конечными тэгами, рассматривается в XML как данные, и поэтому учитываются все символы форматирования (в HTML они игнорируются).

Содержимое XML документа представляет собой набор элементов, секций CDATA, директив анализатора, комментариев, спецсимволов, текстовых данных.

Первой строкой документа XML является декларация XML. Эта необязательная часть файла определяет его как файл XML (например, `<?xml version="1.0" encoding="utf-8"?>`). Начальный и замыкающий теги корневого элемента окружают весь текст XML-документа. В файле должен присутствовать только один корневой элемент (например, `<recipe>` (рецепт)), и это необходимая "обложка" для него. По мере создания документа в него помещается текст и дополнительные теги между `<recipe>` и `</recipe>`. При создании XML регистры начального и конечного тегов должны совпадать. В XML сначала выбираются имена элементов, а затем на основе этих имен определяется соответствующее описание DTD или схема. Пробелы в именах элементов не допускаются, при этом имена должны начинаться с буквы.

Язык XML устраняет существенные недостатки, присущие HTML. Но в то же время XML остается гораздо более простым языком, чем SGML, сохраняя при этом большинство из его возможностей. Упрощение XML по отношению к SGML было достигнуто в основном за счет облегчения запутанных синтаксических правил определения новых классов документов в DTD. Более того, простые документы XML могут вообще не иметь определения типа (DTD). В XML не были перенесены многие редко используемые возможности SGML, что облегчает написание программ

обработки. Другими словами можно сказать, что упрощение XML было произведено за счет потери обобщенности, присущей SGML - язык XML изначально ориентирован на Web-технологии.

XHTML представляет собой язык разметки гипертекста, основанный на XML и максимально приближенный к стандартам HTML. XHTML отличается от HTML строгостью написания кода. Если HTML позволял писать практически любые конструкции и браузер их корректно распознавал, то теперь, с появлением XHTML, это стало невозможным. XHTML требует строгого соблюдения всех правил, предъявляемых W3C.

XHTML создавался с учетом общей совместимости пользовательских агентов. С помощью нового механизма профилирования пользовательских агентов и документов серверы, прокси и пользовательские агенты смогут преобразовывать содержимое наилучшим образом. В конечном счете, возможна разработка соответствующего XHTML содержимого, пригодного для любого соответствующего XHTML пользовательского агента.

XHTML совместим с HTML при соблюдении некоторых правил, т.е. даже самые старые браузеры, которые понимают HTML, могут работать и с XHTML.

Правильность написания XHTML-кода осуществляют программы-валидаторы, а пользовательские агенты, поддерживающие XHTML, сигнализируют о синтаксических ошибках во время написания кода.

При написании кода на XHTML необходимо выполнение следующих основных требований:

- в начале документа должен указываться один из возможных DTD;
- в теле документа должны обязательно присутствовать тэги: html, head, title и body;
- обязательно наличие атрибута xmlns в элементе html;
- имена тегов и атрибутов должны записываться в нижнем регистре;
- все значения атрибутов должны заключаться в "кавычки";
- все тэги должны закрываться;
- необходимо соблюдать корректную вложенность тэгов.

У XHTML есть свой собственный MIME-тип: `<application/xhtml+xml>` - специальный набор расширений, указывающий программам, как обрабатывать входящую информацию. Поэтому XHTML-данные правильно отдавать клиенту именно в формате `<application/xhtml+xml>`, поскольку все преимущества, помимо кросс-браузерности (увеличение скорости анализа кода процессором XML, сообщение об ошибках самим браузером и пр.), могут достигаться только в случае, если пользовательский агент поддерживает XHTML и ему сообщается о том, что входящие данные - это XHTML-код[\[18\]](#).

WML (Wireless Markup Language) - язык разметки для работы в Интернет беспроводных устройствах (WAP), основанный на XML. WML используется для описание контента и пользовательского интерфейса для особого класса узкополосных устройств, типа Palm Pilot, сотовых телефонов и пейджеров. Следует подчеркнуть, что размер WML-страниц не должен превышать 4 килобайт.

WML специально ориентирован ограничения подобный WAP-устройств, поэтому в его основе лежит идея хорошо структурированных данных - WML наследует синтаксис XML. В WML ориентируются на такие механизмы событий, как голос, либо еще какие-то варианты, кроме кнопочных.

В WML предъявляются жесткие требования к парности тегов, использованию регистра и обрамлению атрибутов кавычками. Используемые компоненты синтаксиса в WML - это символьные сущности, элементы, атрибуты, комментарии, переменные и секции CDATA.

В WML набор поддерживаемых тегов невелик.

По структуре WML напоминает несколько упрощенный HTML, при этом WML ориентирован на устройства, не обладающие возможностями персональных компьютеров (малоформатный экран, узкая полоса пропускания канала связи, малая собственная память, ограничения на "вычислительные" способности). Поэтому вся информация в WML содержится в так называемых «колодах» или «деках» (англ. deck) - минимальных блоках данных, которые могут быть переданы сервером.

В деках находятся «карточки» (card), каждая из которых ограничена тегами `<card>` и `</card>`. В одной деке всегда должна быть, по меньшей мере, одна карточка. При этом на экране устройства в каждый момент времени отображается только одна карточка, а пользователь может переключаться между ними, переходя по ссылкам.

Так, если для обычного сайта единицей является Web-page, то единица WML – это deck, состоящая из одной или нескольких card, составляющих вместе WML-документ (wml-файл). Как только wml-файл загружен в устройство с сервера, пользователь может листать экраны, переходить к следующему или возвращаться к предыдущему, не ожидая загрузки с сайта, что, конечно, выгодно по скорости.

Поскольку WML представляет собой ветвь XML, а любой XML-документ сначала должен пройти проверку на соответствие своему DTD, в любом wml-файле, необходимо указывать полный путь к WML DTD.

Строительные блоки и функциональные средства WML можно условно разделить на четыре категории: форматирование, навигация, ввод данных и управление действиями/событиями.

Так поскольку на экран выводится по одной карте, то в элементе card предусмотрен атрибут id - внутренний идентификатор для переходов, а также видимый на экране заголовок title. Для форматирования текста, почти аналогично элементы, аналогичные HTML: p – для параграфов, em и I – для выделения курсивом, big – для текста большего размера и др.

Для навигации и установки гиперссылок предусмотрены теги anchor и a.

Для ввода данных предусмотрены такие теги, как input, select/option (поле выбора из заранее заданного списка), postfield.

Для управления действиями используются tasks, do. Типовые "исполнительные" теги действий – это go, prev, refresh (перечитать текущую карту) или noop (ничего не делать). Действия на уровне конкретной карты описываются в тегах do, а на уровне всей колоды – в элементе template.

В WML есть также возможность назначать клавиши навигации.

Тег `<onevent type="button">действие</onevent>` позволяет приписать к одной из кнопок телефона пользовательское событие. В качестве действия обычно используется `<go href="url"/>`. Так, добавив к карте строку `<onevent type="GO"><go href="#card2"/></onevent>`, можно назначить кнопке GO переход непосредственно в меню.

Итак, подытожив материал первой главы работы можно сделать следующие выводы:



- под гипертекстом стоит понимать совокупность обычных фрагментов текста, связь между которыми реализуется посредством ссылок, встроенных в информационную систему, что позволяет пользователям осуществлять переход от одного текстового фрагмента к другому по их выбору. Основное преимущество гипертекста состоит в нелинейности повествования, которая позволяет создавать сразу несколько линейных последовательностей в изучении материала, что позволяет, не изучая всю предоставленную информацию, выбирать нужную, что позволяет сэкономить много времени;

- язык разметки документов – это набор специальных инструкций, называемых тегами, предназначенных для формирования в документах какой-либо структуры и определения отношений между различными элементами этой структуры.

- SGML - обобщенный метаязык, позволяющего строить системы логической и структурной разметки любых разновидностей текстов и не несущий информации о внешнем виде документа. Основным недостатком SGML является его сложность. Соответственно, и написание программ-обработчиков, поддерживающих SGML в полной мере, является очень трудоемким занятием. Более того, многие возможности SGML используются очень редко или не используются совсем, так что затраты на написание обработчиков, позволяющих построить крупные информационные системы на основе SGML, не оправдывают себя.

- HTML – широко используемый стандартизированный язык разметки документов в Интернете. Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора. Существует восемь версий языка HTML. HTML до его 5-ой версии определялся как приложение SGML. Спецификации HTML5 формулируются в терминах DOM. Один из основных недостатков HTML - его ограниченность. Это обратная сторона той самой простоты, которая позволила HTML стать основным форматом данных WWW в самом начале развития Web-технологии. В HTML DTD определен фиксированный набор допустимых элементов разметки и правил их расстановки. Расширение этого набора возможно только при принятии очередной версии стандарта, а это - чрезвычайно долгий и трудный процесс, осложняющийся постоянными разногласиями крупнейших разработчиков html-браузеров (программ просмотра) по поводу полной или частичной поддержки нового стандарта, или введения собственных расширений языка, идущих вразрез со стандартизированными соглашениями. Таким образом, чем более специализированной становится область применения HTML (например, публикация инженерных расчетов, изобилующих сложными формулами), тем сильнее ощущаются ограничения HTML.

Другой недостаток HTML - определенная степень безразличия к структуре документа. Тэги HTML предназначены в первую очередь для определения способа отображения информации на экране, но они никак не связаны с отображаемыми данными по смыслу. Иными словами, тэги HTML указывают только как следует отобразить элемент на экране монитора, но они не указывают, какое значение имеет отображаемый элемент, к какой категории, классу он относится. Это не позволяет использовать тэги для ускорения поиска информации в документе, поиск таким образом приходится вести в сплошном тексте, а не в некоторой совокупности структурированных элементов. Разумеется, это еще более сужает область использования HTML в сегодняшних условиях.

- XML - это метаязык, определяющий другие языки разметки для специфических целей, предназначен для хранения структурированных данных и для обмена информацией между программами;

- XHTML - язык разметки гипертекста, основанный на XML и максимально приближенный к стандартам HTML. XHTML отличается от HTML строгостью написания кода;

- WML – язык разметки для работы в Интернет беспроводных устройствах (WAP), основанный на XML. WML используется для описания контента и пользовательского интерфейса для особого класса узкополосных устройств, типа Palm Pilot, сотовых телефонов и пейджеров.

В Таблице 1.3 Приложения 1 дана сравнительная характеристика языков гипертекстовой разметки.

### **1.3. Структура HTML-документа**

HTML сегодня является самым универсальным языком разметки гипертекста и стандартизированным языком разметки документов во Всемирной паутине.

На данный момент разработано большое количество различных пособий и самоучителей, в том числе электронных, построенных на базе языка разметки гипертекста HTML (Hyper Text Markup Language). Именно на базе данного языка отформатированы все web-документы, а также гиперссылки, которые позволяют легко перемещаться со страницы на страницу[19].

Большинство документов имеют стандартные элементы, такие, как заголовок, параграфы или списки[20]. Используя тэги html, можно обозначать данные элементы, обеспечивая браузеры минимальной информацией для отображения данных элементов, сохраняя в целом общую структуру и информационную полноту документов. Все что необходимо, чтобы прочитать html - документ - это браузер, который интерпретирует тэги html и воспроизводит на экране документ в виде, который ему придает его автор. Html - документы могут быть созданы при помощи любого текстового редактора или специализированных html - редакторов и конвертеров.

Все тэги html начинаются с «<» (левой угловой скобки) и заканчиваются символом «>» (правой угловой скобки). Как правило, существует стартовый тэг и завершающий тэг. Завершающий тэг выглядит так же, как стартовый, и отличается от него слэшем «/» перед текстом внутри угловых скобок. Также существуют теги, которые не требуют завершающего тэга, но его использование придает исходному тексту документа улучшенную читаемость и структурируемость. HTML не реагирует на регистр символов, описывающих тэг.

Любой HTML документ содержит определенную структуру, без которой он будет просто обычным текстовым документом. Т.е., набор элементов, начало и конец каждого из которых обозначается специальными пометками - тегам, с помощью которых можно выделять заголовки документа, изменять цвет, размер и стиль букв, вставлять графические изображения и таблицы, а также специальных конструкций языка HTML, которые позволяют щелчком мыши перейти к просмотру другого документа[21].

HTML-документ состоит из собственно текста - данных, составляющих содержимое документа, и тегов — специальных конструкций языка HTML, используемых для разметки документа и управляющих его отображением. Графическая и звуковая информация, имеющаяся в HTML-документе, хранится в отдельных файлах[22].

Самый первый тэг, который встречается в документе, должен быть тэгом <html>. Тэг <!DOCTYPE html> сообщает браузеру, что документ написан с использованием html, и показывает, на какую версию html ориентироваться браузеру.

Основными тегам, используемыми HTML, являются html, head, body.

Тегом html определяется все содержание документа. С него документ начинается <html> и им заканчивается </html>. Эти теги сигнализируют о том, что находящиеся между ними строки являются единым гипертекстовым документом.

Наличие этих тегов дает возможность программе просмотра идентифицировать формат документа и правильно его интерпретировать[23].

Основная, содержательная часть html документа заключается между тегами <body> и </body>, эта часть является «телом» документа. Тэги тела документа идентифицируют отображаемые в окне компоненты HTML документа. Тело документа может содержать ссылки на другие документы, текст и другую форматированную информацию.

Уровни заголовков <Hx>, где x=1,2...6. Когда пишется HTML-документ, текст делится на просто текст, заголовки частей текста, заголовки более высокого уровня и т.д. Первый уровень заголовков (самый большой) обозначается цифрой 1, следующий - 2, и т.д.

В отличие от большинства текстовых процессоров, в html-документе обычно игнорируются символы возврата каретки. Физический разрыв абзаца может находиться в любом месте исходного текста документа (для удобства его читаемости). Тэг <br> извещает браузер о разрыве строки. Однако браузер разделяет абзацы только при наличии тэга <p>. Ниже перечислены дополнительные параметры выравнивания содержимого тэга <p>[24].

<p align="left/ center/ right /justufy"> - позволяют выравнивать абзац по левому краю, центру, по правому краю и по ширине соответственно.

Тег <a> является одним из важных элементов HTML и предназначен для создания ссылок. В зависимости от присутствия атрибутов name или href тег <a> устанавливает ссылку или якорь (закладка внутри страницы, которую можно указать в качестве цели ссылки). При использовании ссылки, которая указывает на якорь, происходит переход к закладке внутри веб - страницы.

Элемент тега a обозначает "якорь/anchor", а атрибут href (от "hypertext reference/гипертекстовая ссылка") - место, на которое выполняется переход по данной ссылке, при этом "http://" всегда должно входить в состав URL.

При ссылках между страницами на одном сайте, полный адрес документа ("http://") указывать необязательно.

Для создания ссылки-перехода внутри самой страницы используется атрибут id и символ "#". Так сначала посредством тега <h1 id="heading1">heading 1</h1> маркируется элемент, на который необходимо сделать переход. А посредством

тега `<a href="#heading1">Ссылка на heading 1</a>` создана ссылка на этот элемент[25].

Если необходимо сделать ссылку на e-mail адрес, используется следующий тег:

```
<a href="mailto:nobody@html.net"> </a>
```

В результате при щелчке по ссылке открывается программа по умолчанию для работы с электронной почтой с адресом, уже записанным в строке адреса.

Также, при создании списка будут полезны следующие тэги. Существует три основных вида списков в html-документе:

1. Пронумерованный список создаётся с помощью парного тега `<ol></ol>`. Каждый пункт списка также создаётся с помощью элемента `<li>`.
2. Маркированный список представляет собой неупорядоченный список. Создаётся с помощью парного тега `<ul></ul>`.
3. Списки определений создаются с помощью тега `<dl></dl>`. Для добавления термина применяется тег `<dt></dt>`, а для вставки определения — тег `<dd></dd>`.

Одна из наиболее привлекательных черт web - возможность включения ссылок на графические и иные типы данных в html - документ. Делается это при помощи тэга `<img...>`. Использование данного тэга позволяет значительно улучшить внешний вид и функциональность документов[26].

Существует несколько способов использования графики в html-документах. Основной - это внедрение графических образов в документ, что позволяет пользователю видеть изображения непосредственно в контексте других элементов документа. Это наиболее используемая техника при проектировании документов, называемая иногда «inline image». Синтаксис тэга:

```
<img src = "url" alt = "text" height = n1 width = n2 align=top /middle /bottom /texttop>
```

Для того, чтобы поместить на страницу сайта изображение, необходимо использовать тэг `img src`, где `src` ("source") указывает размещение изображения. Следует отметить, что тэг `img` не требует наличия закрывающего тэга.

Чтобы загрузить файлы изображений, необходима программа редактирования файлов изображений (Paint Shop Pro, PhotoShop, Macromedia Fireworks, Irfan View

(бесплатная)).

Можно также вставить изображения, используя ссылки на другие сайты.

При вставке изображений, как правило, используют атрибуты `width` и `height` для установки ширины и высоты изображения, при этом значение указывается в пикселах. Например: ``

Язык HTML поддерживает два вида форматирования документа [\[27\]](#):

- логическое, указывающее на назначение определенного фрагмента текста, при котором выделяются части текста в соответствии со структурой документа (заголовки шести уровней);
- физическое, задающее внешний вид текста (тип и размер шрифта). Например, тег `<b>` выделяет текст полужирным начертанием, `<u>` - подчеркивание текста.

В целом тэг представляет собой метку, указывающую браузеру, как нужно представлять сайт. В некоторых тэгах можно вводить дополнительную информацию - атрибут, после которого следует знак равенства и детали атрибута, заключенные в двойные кавычки. Точка с запятой после атрибута служит для разделения команд разных стилей.

Например, тег `<body style="background-color:#ff0000;">` сигнализирует браузеру о выводе страницы красного цвета, при этом использован шестнадцатиричном формате (HEX), где каждому цвету соответствует свое 16-ричное число. Но можно использовать и английские названия цветов, например, `<body style="background-color: red;">`.

Для вставки таблиц используются 3 базовых тэга:

- `<table>` - начало и конец таблицы;
- `<tr>` - "table row/ряд таблицы", начинает и заканчивает горизонтальный ряд ячеек;
- `<td>` - "table data/табличные данные" - начинает и заканчивает каждую ячейку ряда таблицы.

Тег `<div></div>` представляет собой блочный элемент, на котором можно построить весь сайт. Использование блоков `<div>` является универсальным способом вставки изображения, делает код более читабельным. Применяя блоки

<div> можно достичь максимально комфортных условий для показа контента.

Следует отметить, что если задавать стили блокам текста непосредственно в самом коде каждого <div>, то код разрастается. В целях упрощения работы были разработаны каскадные таблицы стилей CSS (Cascading Style Sheets), представляющие собой формальный язык описания внешнего вида документа, написанного с использованием языка разметки. Так создается файл CSS, который является единым файлом стилей для всего сайта и подключается ко всему сайту, а, следовательно, и задает свойства стилей для всех блоков и всего шаблона[28].

Таким образом, язык HTML прост и логичен. Браузер читает HTML так, как читает текст человек: сверху вниз и слева направо. HTML-документ начинается и заканчивается тем, чем должен начинаться и заканчиваться текст. Создать веб-сайт посредством языка HTML, зная структуру HTML-документа, а также синтаксис языка гипертекстовой разметки, как видится, не сложная задача.

## **2. Пример работы с HTML: верстка вебстраницы**

Чтобы показать на примере действие языка гипертекстовой разметки создадим на HTML небольшую по содержанию условную веб-страничку. Т.е., покажем как при помощи языка HTML разместить выдаваемую в странице браузера текстовую, графическую и ссылочную информацию именно в том виде, как нам нужно – например, создадим часть, внешне похожей на оригинальную страничку, странички личного кабинета студента МФПУ, например – мою.

Вообще, создание веб-страниц (сайтов) возможно несколькими способами: от ручного, например, в обычном блокноте (что не продуктивно), до полу-автоматического (например, визуальный HTML-редактор Dreamweaver и т.п.) и полностью автоматического (веб-конструкторы типа uCoz, WIX и т.п.)

Т.к., наша страничка не будет увязана ни с CMS и с базами данных, то структура каталогов для размещения файлов будет проста и условна: корневой каталог webpage, в нем будет храниться наша страничка .html; в корневом каталоге вложен каталог img для файлов изображений. Таблицы CSS отсутствуют, файлы скриптов отсутствуют. Использована табличная верстка.

Для создания HTML-странички используем текстовый редактор Notepad, просмотр результата осуществляем в Mozilla Firefox ESR 52.5.0.

Для упрощения, будет использована табличная верстка.

Первым тэгом определим тип документа как HTML[29]:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Тэгом <head> создадим контейнер, в котором мы разместим мета-тэг о типе и кодировке страницы, тэг <title> устанавливающий заголовок окна веб-страницы, и вместо подключения CSS-документа, некоторые стилевые оформления текста (класс .whitetext, цвет белый, начертание полужирное); шрифт ссылок белого цвета, без декора; цвет фона страницы светло-серый:

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

```
<title>Пример работы с HTML</title>
```

```
<style type="text/css">
```

```
.whitetext {
```

```
color: #FFF;
```

```
font-weight: bold;
```

```
}
```

```
a:link {
```

```
text-decoration: none;
```

```
color: #FFF;
```

```
}
```

```
a:visited {
```

```
text-decoration: none;
```

```
color: #FFF;
```

```
}
```



```
a:hover {  
text-decoration: none;  
color: #FFF;  
}  
a:active {  
text-decoration: none;  
color: #FFF;  
}  
body {  
background-color: #f3f2ef;  
}  
</style>  
</head>
```

Теперь, в контейнере `body`, задаваемом одноименным тэгом начнем размещать контент.

Под размещение контента будут создаваться таблицы с шириной 940px (отталкиваясь от ширины оригинального баннера «`student_discount.jpg`», размещенного на оригинальной страничке).

Создадим первую таблицу, в которой разместим логотипы МФПУ; окна обращения в деканат и персональных данных(оба окна статичны, т.к. для их отображения и функционирования требуются скрипты, css и подключение к базам, что не является основной темой работы):

Зададим ширину 940px (`table width="940"`), толщину границы ячеек 0 (`border="0"`) и выравнивание по центру (`align="center"`)

```
<body>  
  
<table width="940" border="0" align="center">
```

Далее, сделаем разбивку на ячейки (по количеству элементов к размещению) – на 3 ячейки и начнем размещать контент:

В этой ячейке мы разместили логотип МФПУ и задали гиперссылку с логотипа на главную страницу Мегакампуса. Параметры ячейки подобраны под параметры картинки-логотипа:

```
<tr>
```

```
<td colspan="4"></td>
```

В этой ячейке мы разместили статичную картинку, внешне воспроизводящую вид оригинала – формы отправки сообщений в деканат. Параметры ячейки подобраны под параметры картинки-скриншота формы:

```
<td width="346" align="right"></td>
```

В этой ячейке мы разместили статичную картинку, внешне воспроизводящую вид оригинала – формы подменю перехода к личной информации. Параметры ячейки подобраны под параметры картинки-скриншота оригинального элемента:

```
<td width="199" align="right"></td>
```

```
</tr>
```

```
</table>
```

В следующей таблице мы разместим меню с присвоенными гиперссылками на внешние ресурсы:

```
<table width="940" border="0" align="center">
```

```
<tr class="whitetext">
```

```
<td align="center" bgcolor="#FF0000"><a href="http://my.megacampus.ru/announce">Объявления</a></td>
```

```
<td align="center" bgcolor="#FF0000"><a href="http://my.megacampus.ru/student/up">Обучение</a></td>
```

```
<td align="center" bgcolor="#FF0000"><a  
href="http://my.megacampus.ru/schedule/academ">Расписание</a></td>
```

```
<td align="center" bgcolor="#FF0000"><a  
href="http://my.megacampus.ru/finance">Платежи</a></td>
```

```
<td align="center" bgcolor="#FF0000"><a  
href="http://my.megacampus.ru/docfiles/student">Документы</a></td>
```

```
<td align="center" bgcolor="#FF0000">Библиотека</td>
```

```
</tr>
```

В следующей таблице разместим графические элементы, имитирующие элементы навигации страницы-оригинала:

```
<tr>
```

```
<td colspan="6"></td>
```

```
</tr>
```

```
<tr>
```

В следующей таблице разместим рекламный баннер как на странице-оригинале, и, также присвоим ему гиперссылку на внешний ресурс:

```
<tr>
```

```
<td colspan="6"><a  
href="http://synergy.ru/lp/zao/new4/?version=vvo&partner=yivanova&utm_source=r  
src="img/student_discount.jpg.png" width="940" height="317" /></a></td>
```

```
</tr>
```

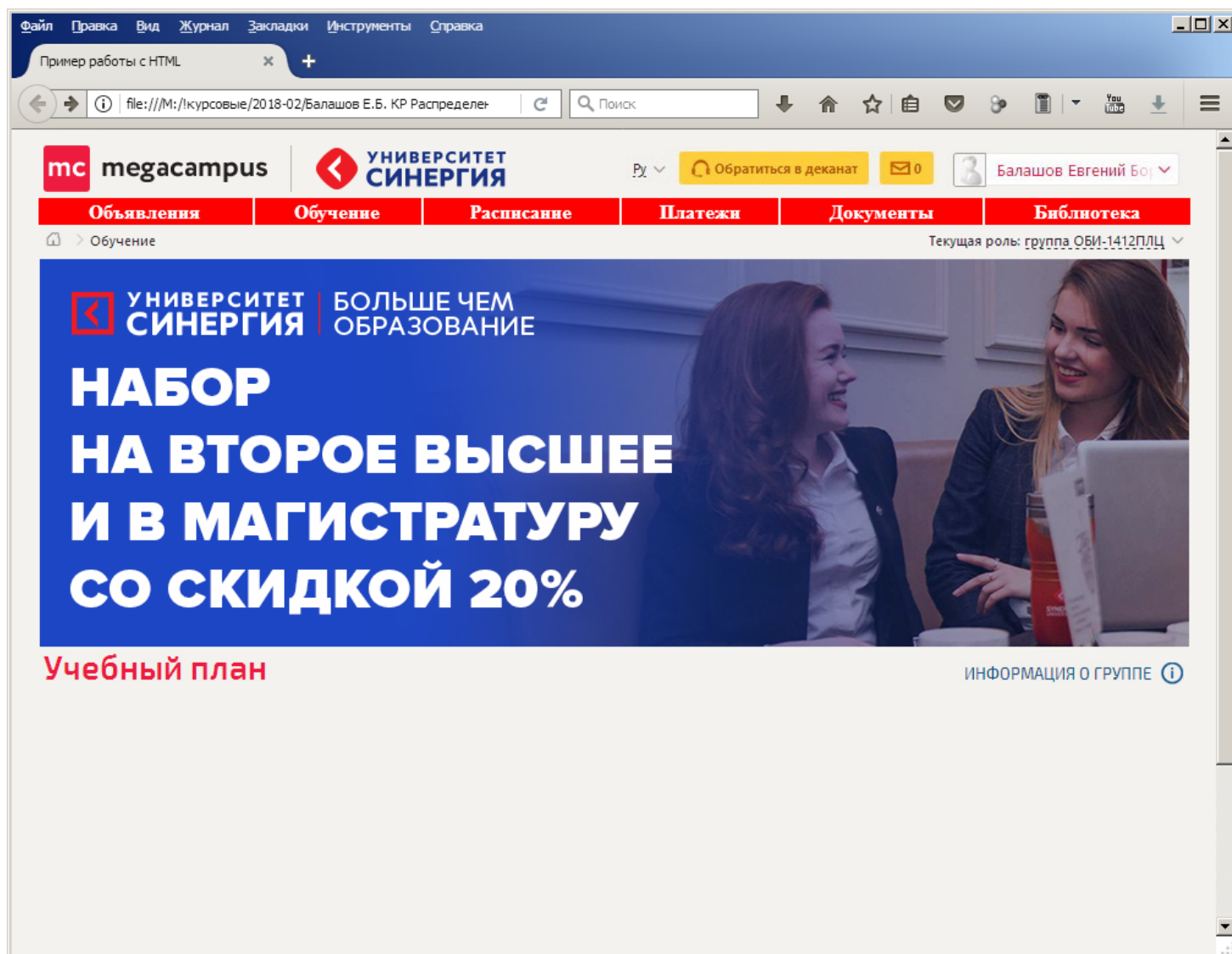
```
<tr>
```

```
<td colspan="6"></td>
```

```
</tr>
```

Уже теперь, (если проставим закрывающие тэги </table>, </body> и </html>) при просмотре через браузер, мы получим следующий внешний вид, напоминающий

оригинал – Рисунок 2.1:



**Рисунок 2.1**

Можно продолжить заполнение страницы и разместить весь контент идентичный странице-оригиналу, но это не требуется. Основная цель – показать на примере действие разметки, и мы её достигли: мы разместили и текстовую, и гипертекстовую (ссылки), и графическую информацию на странице именно так и именно в том виде как должен видеть её пользователь (как оригинал страницы ЛК на Мегакампусе).

В Приложении 2 размещён полный листинг получившейся страницы.

В Приложении 3 размещён архив «webpage.rar», в котором приведена страница, созданная для примера. Необходимо распаковать архив и запустить файл

«primer.html»

Итак, подытожим:

- HTML-документ состоит из собственно текста - данных, составляющих содержимое документа, и тегов — специальных конструкций языка HTML, используемых для разметки документа и управляющих его отображением. Графическая и звуковая информация, имеющаяся в HTML-документе, хранится в отдельных файлах.

- как правило, теги используются парами, состоящими из открывающего `<имя_тега>` и закрывающего `</имя_тега>` тегов, которые иначе называют контейнером. Основными тегами, используемыми HTML, являются `html`, `head`, `body`. Тегом `html` определяется все содержание документа. Внутри `head` размещается служебная информация, которая не видна на открытой веб-странице, необходимая браузеру для корректной интерпретации `html` документа и поиске браузерами. Вся информация, которая отображается на веб-странице, помещается между закрывающим и открывающим тегами `body`;

- для создания HTML-страницы подойдет или простой текстовый редактор типа блокнота, или его продвинутые версии, или специальные HTML-редакторы, или программы-конструкторы.

В данной работе использован обычный блокнот и браузер Mozilla Firefox ESR 52.5.0 для просмотра полученного результата.

## **ЗАКЛЮЧЕНИЕ**

Итак, языки разметки предоставляют людям возможность работать с информацией в удобном и структурированном виде, однако сами не имеют удобного представления.

Язык гипертекстовой разметки – это инструмент посредством которого электронный контент разбивают на блоки смысловой информации (заголовки, параграфы, таблицы, рисунки и т.п.). Состоит из набора специальных инструкций, называемых тегами, предназначенных для формирования в документах какой-либо структуры и определения отношений между различными элементами этой структуры.

Гипертекст – это совокупность обычных фрагментов текста, связь между которыми реализуется посредством ссылок, встроенных в информационную систему, что позволяет пользователям осуществлять переход от одного текстового фрагмента к другому по их выбору. Основное преимущество гипертекста состоит в нелинейности повествования, которая позволяет создавать сразу несколько линейных последовательностей в изучении материала, что позволяет, не изучая всю предоставленную информацию, выбирать нужную, что позволяет сэкономить много времени.

Наиболее широко используемый стандартизированный язык разметки документов в Интернете – это HTML.

Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора.

Существует девять версий языка HTML:

HTML 2.0 — опубликован IETF как RFC 1866 в статусе Proposed Standard (24 ноября 1995 года);

HTML 3.0 — 28 марта 1995 года — IETF Internet Draft (до 28 сентября 1995 года);

HTML 3.2 — 14 января 1997 года;

HTML 4.0 — 18 декабря 1997 года;

HTML 4.01 — 24 декабря 1999 года;

ISO/IEC 15445:2000 (так называемый ISO HTML, основан на HTML 4.01 Strict) — 15 мая 2000 года;

HTML5 — 28 октября 2014 года;

HTML 5.1 начал разрабатываться 17 декабря 2012 года. Рекомендован к применению с 1 ноября 2016 года.

HTML 5.2 был представлен 14 декабря 2017 года.

Официальной спецификации HTML 1.0 не существует. До 1995 года существовало множество неофициальных стандартов HTML. Чтобы стандартная версия отличалась от них, ей сразу присвоили второй номер.

HTML до его 5-ой версии определялся как приложение SGML. Спецификации HTML5 формулируются в терминах DOM.

SGML - обобщенный метаязык, позволяющего строить системы логической и структурной разметки любых разновидностей текстов и не несущий информации о внешнем виде документа;

XML - это метаязык, определяющий другие языки разметки для специфических целей, предназначен для хранения структурированных данных и для обмена информацией между программами;

XHTML - язык разметки гипертекста, основанный на XML и максимально приближенный к стандартам HTML. XHTML отличается от HTML строгостью написания кода;

WML - язык разметки для работы в Интернет беспроводных устройствах (WAP), основанный на XML. WML используется для описания контента и пользовательского интерфейса для особого класса узкополосных устройств.

HTML-документ состоит из, собственно, текста - данных, составляющих содержимое документа, и тегов — специальных конструкций языка HTML, используемых для разметки документа и управляющих его отображением. Графическая и звуковая информация, имеющаяся в HTML-документе, хранится в отдельных файлах.

Как правило, теги используются парами, состоящими из открывающего `<имя_тега>` и закрывающего `</имя_тега>` тегов, которые иначе называют контейнером. Основными тегами, используемыми HTML, являются `html`, `head`, `body`. Тегом `html` определяется все содержание документа. Внутри `head` размещается служебная информация, которая не видна на открытой веб-странице, необходимая браузеру для корректной интерпретации `html` документа и поиске браузерами. Вся информация, которая отображается на веб-странице, помещается между закрывающим и открывающим тегами `body`.

Для создания HTML-сайта необходимы браузер и текстовый редактор. В данной работе использован браузер Google Chrome и текстовый редактор Notepad, посредством которого создан сайт из семи страниц под названием «Языки гипертекстовой разметки». Для его создания использована табличная верстка. Все документы, связанные с созданием сайта, сохранены в папке "htdocs". Две страницы, на которые имеются ссылки на главной странице сайта, созданы

самостоятельно, остальные ссылки на сайте перенаправляют пользователя на созданные сайты иных разработчиков, посвященные соответственно SGML, XML, HTML, XHTML и WML.

На сегодняшний день семейство языков, произошедших от SGML (Standart Generalized Markup Language), занимает доминирующие позиции в задачах представления информации. Наиболее часто используемыми языками этого семейства являются XML, HTML и XHTML.

## **СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. Dave Raggett, Getting started with HTML / Dave Raggett // URL: <https://www.w3.org/MarkUp/Guide/> (Дата обращения: 03.02.2018)
2. HTML Current Status / World Wide Web Consortium (W3C) // URL: [https://www.w3.org/standards/techs/html#w3c\\_all](https://www.w3.org/standards/techs/html#w3c_all) (Дата обращения: 03.02.2018)
3. Steven Pemberton, XML Events for HTML Authors / Steven Pemberton // URL: <https://www.w3.org/MarkUp/2004/xmlevents-for-html-authors> (Дата обращения: 03.02.2018)
4. Ted Nelson, A File Structure for the Complex, the Changing, and the Indeterminate / Ted Nelson // URL: <https://archive.org/details/nelson-file-structure> (Дата обращения: 06.02.2018)
5. XHTML Modules and Markup Languages / // URL: <https://www.w3.org/MarkUp/Guide/xhtml1-m12n-tutorial/> (Дата обращения: 03.02.2018)
6. Антониоу Григорис, Семантический веб. Руководство / Антониоу Григорис, Гросс Паул, Фрэнк ван Хармелен, Хоекстра Ринке. – М.: ДМК пресс, 2016г. – 240с.
7. Асхатова Л.И. Развитие языков разметки / Л.И. Асхатова, Э.Р. Галимов, И.М. Габдуллин // APRIORI. Серия: естественные и технические науки, № 2, 2015, с.1 (Дата обращения: 02.02.2018)
8. Барнаш В.Г. HTML - развитие языка стандарта отображения страниц / В.Г. Барнаш, Т.А.Коваленко // Вестник научных конференций, № 2-6 (18), 2017, с.14-15 (Дата обращения: 02.02.2018)
9. Белякова Д.Г. Язык разметки гипертекста HTML как инструментальное средство разработки электронных учебных пособий / Д.Г. Белякова, М.С.



- Герасимова, А.А. Салмин // наука, техника и образование, № 6 (24), 2016, с.54-56 (Дата обращения: 02.02.2018)
10. Демин И.С. Многослойная модель гипертекста / И.С. Демин // Вестник тамбовского государственного технического университета, т.9, №2, 2003, с.199-203 (Дата обращения: 03.02.2018)
  11. Дунаев В.В. (X)HTML, скрипты и стили. Самое необходимое / В.В. Дунаев. – СПб.: БХВ-Петербург, 2013 г. - 496 с.
  12. Коваленко Т.А. Анализ возможностей языка HTML / Т.А. Коваленко, В.Г. Барнаш // Наука и мир, т.1, №4 (44), 2017, с.64-65 (Дата обращения: 03.02.2018)
  13. Рязанцева Т.И. Гипертекст и электронная коммуникация / Т.И. Рязанцева. – М.: URSS, 2010 г. – 256с.
  14. Степанов А.В. Языки разметки / А.В. Степанов // Компьютерные инструменты в образовании, №1, 2008, с.11-18 (Дата обращения: 02.02.2018)
  15. Степанов А.В. Языки разметки. Часть 2: Основные средства форматирования / А.В. Степанов // Компьютерные инструменты в образовании, №2, 2008, с.12-19 (Дата обращения: 02.02.2018)
  16. Степанов А.В. Языки разметки. Часть 3: Верстка таблиц / А.В. Степанов // Компьютерные инструменты в образовании, №3, 2008, с.3-12 (Дата обращения: 02.02.2018)
  17. Сухов К. Контуры HTML 5: больше тегов, хороших и разных. Обзор предварительной спецификации нового стандарта языка гипертекстовой разметки / К. Сухов // Системный администратор, № 2 (63), 2008, с.70-75 (Дата обращения: 03.02.2018)
  18. Сухов К. HTML 5 - путеводитель по технологии / К. Сухов. – М.: ДМК Пресс, 2013 г. - 352 с.
  19. Теодор Нельсон создаёт технологию гипертекста / URL: <http://vikent.ru/enc/676/> (Дата обращения 03.02.2018)
  20. Тим Бернерс-Ли. Да пребудет с нами Веб! / Тим Бернерс-Ли // В мире науки, №2, 2011. – С.78-95
  21. Учебник HTML / URL: <http://ru.html.net/tutorials/html/> (Дата обращения 03 - 18.02.2018)
  22. Хрусталева А.А. HTML5 + CSS3. Основы современного WEB-дизайна / А.А. Хрусталева, А.В. Кириченко. – М.: Наука и Техника, 2018г. - 352с.
  23. Ципоркина Т.Л. Web-конструирование / Т.Л. Ципоркина, А.В. Греков // Научные исследования: от теории к практике, № 1 (7), 2016, с.13-14 (Дата обращения: 04.02.2018)

24. Шапошников И. Web-сервисы Microsoft. NET / И. Шапошников. – СПб.: БХВ-Петербург, 2002г. – 336с.

## ПРИЛОЖЕНИЕ 1

### Таблица 1.1

Достоинства и недостатки HTML

	<b>Достоинства</b>	<b>Недостатки</b>
<b>HTML</b>	Легковесный и простой язык для построения документов	Нестандартность языка приводила к путанице в конструкциях сайтов и неправильному отображению страницы на разных браузерах. Нет поддержки воспроизведения аудио и видео, анимации. Парсер предшествующего языка несовместим с этим стандартом.
<b>HTML2</b>	Кросс-браузерность Избавление от диалектов языка	Нет поддержки воспроизведения аудио и видео, анимации. Версии браузеров, вышедшие до этого стандарта, не совместимы с новыми функциями.

<p>Создание таблиц</p> <p>Обтекание изображений текстом</p>	<p>Трудная реализация для браузеров того времени.</p>
<p><b>HTML3</b> Отображение сложных математических формул</p>	<p>Без дополнительных плагинов нет возможности воспроизведения видео и аудио.</p>
<p>Поддержка gif формата</p> <p>Поддержка таких технологий как Java и Flash Player</p>	<p>Версии браузеров, вышедшие до этого стандарта, не совместимы с новыми функциями.</p>
<p>Упрощение языка</p>	
<p>Введение технологии каскадных таблиц стилей CSS</p>	
<p><b>HTML4</b> Улучшена совместимость с предыдущими версиями HTML</p> <p>JavaScript и CSS дали возможность делать анимацию объектов страницы без дополнительных плагинов</p>	<p>Без дополнительных плагинов нет возможности воспроизведения видео и аудио.</p> <p>Версии браузеров, вышедшие до этого стандарта, не совместимы с новыми функциями.</p>
<p>Поддержка динамических объектов</p>	

	Упрощение конструкции сайта	
	Функция рисования	
	Функция проигрывания аудио и видео	
<b>HTML5</b>	Редактирование документов в браузере	Версии браузеров, вышедшие до этого стандарта, не совместимы с новыми функциями.
	Технология отображения трехмерной графики	
	Замещение функций плагинов встроенными функциями	
	Эффект анимации объектов страницы без JavaScript	

## Таблица 1.2

Преимущества и недостатки языка XML

**Преимущества**

**Недостатки**

- язык разметки, который позволяет стандартизировать вид файлов-данных в виде текста, который будет понятен человеку;
- поддержка Юникод;
- возможность описания записей, списков, деревьев, форматированного текста;
- описание структур и имен полей, как описание значения полей, иными словами – это самодокументируемый формат;
- строго определенные требования к анализу и синтаксис, позволяющие быть простым, непротиворечивым и эффективным;
- основан на стандартах международного уровня;
- иерархическая структура подходит для описания любых документов, кроме видео и аудио файлов, растровых изображений, двоичных данных и сетевых структур данных;
- простой текст, свободный от всяких ограничений и лицензирования;
- подмножество SGML;
- свободен от всяких требований на порядок расположения атрибутов в элементах;
- реализация парсеров для всех языков программирования;
- XSLT стандартный механизм преобразования;
- поддерживается на низком программном уровне.
- избыточный синтаксис;
- размер документа больше бинарного представления данных;
- размер документа больше, чем документа в других форматах передачи текстовых данных;
- возрастание стоимости хранения, передачи и обработки данных;
- содержит метаданные, и одновременно является языком взаимодействия открытых систем;
- для большого количества задач можно использовать более простые решения;
- отсутствие общепринятой методологии;
- неоднозначность моделирования;
- не содержит язык поддержки для типов данных;
- ограниченность модели данных;
- сложное использование пространств имен.

## Таблица 1.2

Сравнительная характеристика языков гипертекстовой разметки

Язык	HTML	XHTML	WML
Основан на	SGML	XML	XML
Тип разметки	тег	тег	тег
Структурная разметка	да	да	да
Презентационная разметка	да	да	да
Формат использования	текст/html	приложение/xhtml+xml	приложение/wml+xml

## ПРИЛОЖЕНИЕ 2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Пример работы с HTML</title>
<style type="text/css">
.whitetext {
color: #FFF;
font-weight: bold;
```

```
}  
  
a:link {  
  
text-decoration: none;  
  
color: #FFF;  
  
}  
  
a:visited {  
  
text-decoration: none;  
  
color: #FFF;  
  
}  
  
a:hover {  
  
text-decoration: none;  
  
color: #FFF;  
  
}  
  
a:active {  
  
text-decoration: none;  
  
color: #FFF;  
  
}  
  
body {  
  
background-color: #f3f2ef;  
  
}  
  
</style>  
  
</head>  
  
<body>
```

```
<table width="940" border="0" align="center">
<tr>
<td colspan="4"></td>
<td width="346" align="right"></td>
<td width="199" align="right"></td>
</tr>
</table>
```

```
<table width="940" border="0" align="center">
<tr class="whitetext">
<td align="center" bgcolor="#FF0000"><a
href="http://my.megacampus.ru/announce">Объявления</a></td>
<td align="center" bgcolor="#FF0000"><a
href="http://my.megacampus.ru/student/up">Обучение</a></td>
<td align="center" bgcolor="#FF0000"><a
href="http://my.megacampus.ru/schedule/academ">Расписание</a></td>
<td align="center" bgcolor="#FF0000"><a
href="http://my.megacampus.ru/finance">Платежи</a></td>
<td align="center" bgcolor="#FF0000"><a
href="http://my.megacampus.ru/docfiles/student">Документы</a></td>
<td align="center" bgcolor="#FF0000">Библиотека</td>
</tr>
<tr>
<td colspan="6"></td>
```



</tr>

<tr>

<td colspan="6"><a

href="http://synergy.ru/lp/zao/new4/?version=vvo&partner=yivanova&utm\_source=n  
src="img/student\_discount.jpg.png" width="940" height="317" /></a></td>

</tr>

<tr>

<td colspan="6"></td>

</tr>

</table>

</body>

</html>

### ПРИЛОЖЕНИЕ 3

image not found or type unknown



архив «webpage.rar», в котором приведена страница, созданная для примера.  
Необходимо распаковать архив и запустить файл «primer.html»

1. Теодор Холм Нельсон, англ. Ted Nelson, Theodor Holm Nelson; род. 17 июня 1937 года — американский социолог, философ и первооткрыватель в области информационных технологий, изобретатель понятия «гипертекст» и ряда других терминов, таких как теледильдоника, создатель гипертекстовой системы Xanadu. [↑](#)
2. «A File Structure for the Complex, the Changing, and the Indeterminate» by Ted Nelson <https://archive.org/details/nelson-file-structure> (Дата обращения: 06.02.2018) [↑](#)

3. Белякова Д.Г. Язык разметки гипертекста HTML как инструментальное средство разработки электронных учебных пособий / Д.Г. Белякова, М.С. Герасимова, А.А. Салмин // наука, техника и образование, № 6 (24), 2016 (Дата обращения: 02.02.2018) [↑](#)
4. Ted Nelson, A File Structure for the Complex, the Changing, and the Indeterminate / Ted Nelson // URL: <https://archive.org/details/nelson-file-structure> (Дата обращения: 06.02.2018) [↑](#)
5. Теодор Нельсон создаёт технологию гипертекста / URL: <http://vikent.ru/enc/676/> (Дата обращения 03.02.2018) [↑](#)
6. Тим Бернерс-Ли. Да пребудет с нами Веб! / Тим Бернерс-Ли // В мире науки, №2, 2011. [↑](#)
7. Рязанцева Т.И. Гипертекст и электронная коммуникация / Т.И. Рязанцева. – М.: URSS, 2010 г. [↑](#)
8. Коваленко Т.А. Анализ возможностей языка HTML / Т.А. Коваленко, В.Г. Барнаш // Наука и мир, т.1, №4 (44), 2017 (Дата обращения: 03.02.2018) [↑](#)
9. Рязанцева Т.И. Гипертекст и электронная коммуникация / Т.И. Рязанцева. – М.: URSS, 2010 г. [↑](#)
10. Антониоу Григорис, Семантический веб. Руководство / Антониоу Григорис, Гросс Паул, Фрэнк ван Хармелен, Хоекстра Ринке. – М.: ДМК пресс, 2016г [↑](#)
11. Белякова Д.Г. Язык разметки гипертекста HTML как инструментальное средство разработки электронных учебных пособий / Д.Г. Белякова, М.С. Герасимова, А.А. Салмин // наука, техника и образование, № 6 (24), 2016, (Дата обращения: 02.02.2018) [↑](#)
12. Степанов А.В. Языки разметки / А.В. Степанов // Компьютерные инструменты в образовании, №1, 2008, (Дата обращения: 02.02.2018) [↑](#)

13. Асхатова Л.И. Развитие языков разметки / Л.И. Асхатова, Э.Р. Галимов, И.М. Габдуллин // APRIORI. Серия: естественные и технические науки, № 2, 2015, с.1 (Дата обращения: 02.02.2018) [↑](#)
14. Степанов А.В. Языки разметки / А.В. Степанов // Компьютерные инструменты в образовании, №1, 2008, (Дата обращения: 02.02.2018) [↑](#)
15. Барнаш В.Г. HTML - развитие языка стандарта отображения страниц / В.Г. Барнаш, Т.А.Коваленко // Вестник научных конференций, № 2-6 (18), 2017, (Дата обращения: 02.02.2018) [↑](#)
16. Steven Pemberton, XML Events for HTML Authors / Steven Pemberton // URL: <https://www.w3.org/MarkUp/2004/xmlevents-for-html-authors> (Дата обращения: 03.02.2018) [↑](#)
17. Асхатова Л.И. Развитие языков разметки / Л.И. Асхатова, Э.Р. Галимов, И.М. Габдуллин // APRIORI. Серия: естественные и технические науки, № 2, 2015, с.1 (Дата обращения: 02.02.2018) [↑](#)
18. Дунаев В.В. (X)HTML, скрипты и стили. Самое необходимое / В.В. Дунаев. – СПб.: БХВ-Петербург, 2013 г. [↑](#)
19. Сухов К. HTML 5 - путеводитель по технологии / К. Сухов. – М.: ДМК Пресс, 2013 г [↑](#)
20. Сухов К. Контурь HTML 5: больше тегов, хороших и разных. Обзор предварительной спецификации нового стандарта языка гипертекстовой разметки / К. Сухов // Системный администратор, № 2 (63), 2008, (Дата обращения: 03.02.2018) [↑](#)
21. Хрусталеv А.А. HTML5 + CSS3. Основы современного WEB-дизайна / А.А. Хрусталеv, А.В. Кириченко. – М.: Наука и Техника, 2018г [↑](#)

22. Сухов К. HTML 5 - путеводитель по технологии / К. Сухов. – М.: ДМК Пресс, 2013 г [↑](#)
23. Хрусталеv А.А. HTML5 + CSS3. Основы современного WEB-дизайна / А.А. Хрусталеv, А.В. Кириченко. – М.: Наука и Техника, 2018г [↑](#)
24. Учебник HTML / URL: <http://ru.html.net/tutorials/html/> (Дата обращения 03 - 18.02.2018) [↑](#)
25. Учебник HTML / URL: <http://ru.html.net/tutorials/html/> (Дата обращения 03 - 18.02.2018) [↑](#)
26. Коваленко Т.А. Анализ возможностей языка HTML / Т.А. Коваленко, В.Г. Барнаш // Наука и мир, т.1, №4 (44), 2017, (Дата обращения: 03.02.2018) [↑](#)
27. Дунаев В.В. (X)HTML, скрипты и стили. Самое необходимое / В.В. Дунаев. – СПб.: БХВ-Петербург, 2013 [↑](#)
28. Дунаев В.В. (X)HTML, скрипты и стили. Самое необходимое / В.В. Дунаев. – СПб.: БХВ-Петербург, 2013 [↑](#)
29. Учебник HTML / URL: <http://ru.html.net/tutorials/html/> (Дата обращения 03 - 18.02.2018) [↑](#)