Федеральное государственное автономное

Образовательное учреждение

Высшего профессионального образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Управления Бизнес-Процессами и Экономики

Кафедра Бизнес-Информатики

**КУРСОВАЯ РАБОТА**

**Дисциплина: «**Web-программирование**»**

**Тема: «**Создание одностраничного сайта с применением современных web-технологий по тематике Хоккей России**»**

Преподаватель \_\_\_\_\_\_\_\_\_ Пупков А.Н.

Студент УБ 15-09, 431515957 \_\_\_\_\_\_\_\_\_ Попова В.Д.

Красноярск 2018 г.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc512368713)

[1 Теоритическая часть 4](#_Toc512368714)

[1.1 Язык гипертекстовой разметки HTML 4](#_Toc512368715)

[1.2 Каскадные таблицы стилей CSS 5](#_Toc512368716)

[1.3 Front-end разработка 7](#_Toc512368717)

[2 Практическая часть 9](#_Toc512368718)

[2.1 Бриф 9](#_Toc512368719)

[2.2 Процесс верстки 10](#_Toc512368720)

[ЗАКЛЮЧЕНИЕ 17](#_Toc512368721)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 18](#_Toc512368722)

[ПРИЛОЖЕНИЕ А 19](#_Toc512368723)

[ПРИЛОЖЕНИЕ Б 25](#_Toc512368724)

# ВВЕДЕНИЕ

Интернет стал наиболее эффективным средством рекламы и продвижения и является одним из важных элементов современной цивилизации. Интернет может удовлетворить все потребности современного человека: это покупки, заключение деловых отношений, поиск клиентов и так далее. Но многие заходят в Интернет читать, развлекаться, общаться, узнавать что-то новое для себя. Поэтому создание сайтов стало таким популярным.

Сайт - набор веб-страниц, объединенных друг с другом общей темой, оформлением и ссылками. Это информационная единица, представляющая компанию или отдельно взятого человека, доступная из любого уголка мира, а также одно из современных средств передачи информации и рекламный продукт, дающий большие возможности в области поиска и привлечения клиентов.  
Создание сайта в Интернете – это один из эффективных методов рекламы своих товаров и услуг, он заслужил признание, как малого предпринимательства, так и крупных международных корпораций. В настоящий момент пользователями сети являются более четверти населения Земли, что является действительно внушительной цифрой. Поэтому сегодня огромное множество успешных компаний используют Всемирную Сеть как рекламу тем или иным способом.

Главное предназначение сайта - это предоставление информации. И несмотря на то, что в мире существует масса других способов передачи и предоставления информации, огромное количество людей сегодня выбирает именно этот - наиболее эффективный.

Цель данной курсовой работы – создать одностраничный сайт, а также получение теоретических знаний и практических навыков в сфере веб-программирования путем создания верстки сайта.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Получить необходимые знания о языке HTML и CSS.
2. Определить требования к сайту.
3. Разработать структуру сайта.
4. Разработать дизайн сайта.
5. Выполнить верстку сайта.

# 1 Теоритическая часть

## 1.1 Язык гипертекстовой разметки HTML

HTML (HyperText Markup Language – язык гипертекстовой разметки) – стандартизированный язык разметки документов во Всемирной паутине. HTML представляет собой достаточно простой набор кодов, которые описывают структуру документа. HTML позволяет выделить в тексте отдельные логические части (заголовки, абзацы, списки и т.д.), поместить на Web-страницу подготовленную фотографию или картинку, организовать на странице ссылки для связи с другими документами.

Основное преимущество HTML заключается в том, что документ может быть просмотрен на веб-браузерах различных типов и на различных платформах.

HTML – это теговый язык разметки документов, то есть любой документ на языке HTML представляет собой набор элементов, причём начало и конец каждого элемента обозначается специальными пометками, называемыми тегами. Под тегом понимается зарезервированные последовательности символов, начинающиеся с < (знака меньше) и заканчивающиеся > (знаком больше). Закрытие тега отличается от открытия только наличием символа «/». Регистр, в котором набрано имя тега, в HTML значения не имеет. Элементы могут быть пустыми, то есть не содержащими никакого текста и других данных (например, тег перевода строки <br>). В этом случае обычно не указывается закрывающий тег. Кроме того, элементы могут иметь атрибуты, определяющие какие-либо их свойства (например, размер шрифта для тега <font>).

В настоящее время язык HTML5 является эволюционным развитием HTML4.01 с сохранением обратной совместимости и добавлением новых возможностей.

Цель разработки HTML5 – улучшение уровня поддержки мультимедиа-технологий с одновременным сохранением обратной совместимости, удобочитаемости кода для человека и простоты анализа для парсеров.

Язык HTML5 реализует подход к проектированию Web-сайтов, основанный на отображении информации, поскольку он воплощает саму суть отображения информации – разделение и маркирование информации для упрощения ее использования и понимания. Именно в этом состоит огромная семантическая и эстетическая ценность HTML5. HTML5 предоставляет дизайнерам и разработчикам всех уровней возможности для предоставления в публичный доступ буквально любого контента – от простых текстов до мультимедийно насыщенных интерактивных материалов.

HTML5 предоставляет эффективные инструменты для управления данными, для рисования, для воспроизведения видео- и аудиоконтента. HTML5 облегчает разработку кросс-браузерных Web-приложений, а также приложений для мобильных устройств. HTML5 относится к числу технологий, которые стимулируют развитие мобильных сервисов на основе облачных вычислений. Кроме того, HTML5 способствует повышению гибкости – благодаря возможности создания интерактивных Web-сайтов. И, наконец, HTML5 предлагает новые теги и усовершенствования, в числе которых следующие: элегантная структура, органы управления формами, API-интерфейсы, мультимедийные функции, поддержка баз данных, существенно увеличенная скорость обработки.

Фактически HTML5 является синонимом непрерывных инноваций: новые теги, новые методики и общая инфраструктура разработки, базирующаяся на взаимодействии технологии HTML5 с родственными технологиями CSS3 и JavaScript. Это создает основу для функционирования приложений, ориентированных на клиентов. Помимо широкого распространения средств и методик технологии HTML5 на настольных системах, она может быть реализована в функционально насыщенных Web-браузерах для мобильных телефонов. Это растущий рынок, характерными представителями которого являются популярные и вездесущие платформы Apple iPhone, Google Android и Palm webOS.

## 1.2 Каскадные таблицы стилей CSS

CSS (Cascading Style Sheets – каскадные таблицы стилей) – формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

CSS позволяют задавать цвета текста и фона любых элементов, создавать рамки и увеличивать или уменьшать отступы вокруг элементов. Благодаря им можно сделать так, чтобы текст отображался прописными буквами, и добавить дополнительные элементы оформления (например, подчеркивание), разбивки и даже управлять тем, будет ли он отображаться вообще, а также они дают возможность реализовать многие другие эффекты.

Таблицы стилей описывают, как документы представлены на экранах, при печати или, возможно, как они произносятся. W3C активно пропагандирует использование таблиц стилей в вебе с того времени как Консорциум был основан в 1994 году. Деятельность по стилям принесла несколько Рекомендаций W3C (CSS1, CSS2, XPath, XSLT). CSS особенно широко реализован в браузерах.

С помощью добавления таблиц стилей к структурированным документам в вебе (например, HTML), разработчики и пользователи могут влиять на вид документов без ущерба независимости устройства или добавления новых HTML тегов.

Правило CSS - это структурная единица таблицы стилей, которая содержит описание стилей для определенного элемента. Правило состоит из селектора, который всегда располагается слева, и блока объявления стилей, который заключается в фигурные скобки и следует непосредственно за селектором (рисунок). Каждое объявление в свою очередь состоит из свойства и его значения. Именно свойство и определяет вид стиля, который будет применяться к элементу. Также указывается, является ли это свойство наследуемым, т.е. будет ли оно применяться для элементов-потомков. Указываются устройства вывода, к которым применяются эти свойства. Селектор служит для того, чтобы в HTML-документе однозначно определить тот элемент, к которому следует применить данное CSS-правило.

Любая таблица стилей может иметь три источника возникновения.

1) Автор-разработчик Web-страницы. Он может написать таблицы стилей и использовать их согласно правилам, т.е. помещать внутрь документа или подключать из внешнего файла.

2) Пользователь или посетитель Web-страницы. Он может самостоя-тельно создать таблицу стилей для конкретного документа, и если браузер поддерживает такую возможность, для просмотра документа пользователь может использовать собственную таблицу стилей.

3) Браузер. В соответствии со спецификацией, браузеры должны сначала применить к HTML-документу собственную таблицу стилей, используемую по умолчанию, и только потом применять остальные таблицы стилей.

Области действия таблиц, созданных любым из трех субъектов, могут пересекаться, например, в каждой таблице может существовать правило, определяющее внешний вид текста для абзаца. В этом случае выбор приоритетного CSS-правила осуществляется в соответствии с принципами каскадирования. При наличии нескольких правил для одного и того же элемента преимущество имеет правило с наибольшим приоритетом.

По умолчанию правила в таблицах стилей, созданных разработчиками страницы, имеют больший приоритет, чем правила, созданные пользователем. И все правила разработчика и пользователя имеют больший приоритет, чем правила в таблице стилей браузера, используемой по умолчанию. Таким образом, наименее приоритетна - таблица стилей браузера, затем идет таблица стилей пользователя, и самой приоритетной является таблица стилей, созданная разработчиком.

Схема обработки CSS браузером определяется только на уровне понятий, так что ее отдельные реализации в разных браузерах могут отличаться друг от друга. Приведенное ниже описание позволит понять работу браузера по отображению HTML-документа, в котором для оформления элементов используются каскадные таблицы стилей.

На первом этапе осуществляется грамматический анализ исходного до-кумента, и для него создается дерево элементов.

Второй этап определяет тип конечного носителя, на который будет выводиться информация (монитор, принтер или другие).

На третьем этапе браузер выбирает все таблицы стилей, связанные с документом и ассоциированные с установленным на предыдущем этапе устройством вывода. Ненужные таблицы стилей игнорируются.

На четвертом этапе браузер помечает каждый элемент дерева документа, присваивая каждому метки значения специфичности, как было описано выше. Назначение этих меток в том, чтобы помочь определить, какой стиль оформления является приоритетным, если оформление для элемента задавалось несколькими способами.

На пятом этапе создается структура форматирования на основании помеченного дерева документа, которая сопоставляет с элементом заданный стиль оформления, который и будет, в конечном счете, к нему применен. Очень часто структура форматирования очень похожа на дерево документа, но она также может отличаться от него, если использовались псевдоэлементы или автоматически генерируемое содержимое.

На последнем этапе браузер передает структуру форматирования на конечное устройство (выполняется печать результатов, отображение их на экране, звуковое воспроизведение и т.д.). Пользователь получает необходимый ему результат.

Таким образом, CSS – замечательный инструмент для любого разработчика или пользователя, поскольку обладает возможностями импортирования, каскадирования и реализации разнообразных эффектов.

## 1.3 Front-end разработка

Front-end разработка – это работа по созданию публичной части сайта, с которой непосредственно контактирует пользователь и функционала который обычно обыгрывается на клиентской стороне (в браузере).

К front-end разработке можно отнести CSS верстку, создание макета и шаблонов для сайта, привязка к пользовательскому интерфейсу скриптов, которые обеспечивают визуализацию и анимацию страниц сайта, а также обеспечение необходимого уровня пользовательского интерфейса (UI – User Interface) и опыта взаимодействия (UX – Uzer Experience).

Фронт-энд разработчик (с англ. «front-end developer») – это программист, занимающийся разработкой пользовательского интерфейса, то есть внешней публичной части сайта в браузере. Главная задача фронт-энд разработчика – сделать максимально удобным взаимодействие пользователей с сайтом или веб-приложением, изучить привычки пользователей и изменение дизайна с помощью тестирования, а также следить за тем, чтобы был сайт доступен для всех пользователей, и реагирует на все представления – для мобильных телефонов и ПК.

В первую очередь фронтенд-разработчик должен отлично владеть базовыми технологиями – HTML, CSS и JavaScript.

JavaScript – главный язык фронт-энд разработчика в веб, на котором он осуществляет программирование пользовательского интерфейса. Основная задача фронтендера - вёрстка сайтов: написание HTML-кода и CSS-стилей для представления информации в понятном всем браузерам виде и внешнему соответствию сайта макету дизайнера. Часто бывает, что макеты не отражают всех деталей общей идеи. В данном случае фронт-энд разработчик включает свои дизайнерские способности, чтобы учесть все технические детали и нюансы. Есть много дополнительных инструментов, которые автоматизируют рутинную часть работы: Sass/SCSS, jQuery, LESS, AngularJS, Bootstrap, Prototype, Ember.js, Backbone, React.js, Grunt Gulp. А появление Chrome Dev Tools и Firebug позволило повысить производительность труда.

Фронт-энд разработчик в веб отвечает за то, чтобы страницы сайта выглядели во всех браузерах одинаково и соответствовали макету дизайнера. С появлением новых средств разработки и более современных браузеров эта задача решается проще. Но интерфейсы современных сайтов или интернет-сервисов становятся более сложными, а многообразие устройств, с которых пользователи выходят в сеть, постоянно расширяется — ноутбуки, планшеты, смартфоны. Поэтому фронт-энд разработчику очень важно, как и любому программисту, постоянно находиться в курсе всех новшеств, чтобы успешно адаптировать сайты под новые устройства.

# 2 Практическая часть

## 2.1 Бриф

Название сайта: «Хоккей России».

Цель сайта: Развитие и популяризация хоккея, повышение его роли во всестороннем и гармоничном развитии личности, укреплении здоровья, формировании здорового образа жизни у населения.

Задачи сайта:

1. Предоставление пользователю информации о событиях русского Хоккея.
2. Предоставление контактной информации в удобном виде.
3. Круглосуточный доступ с любой точки мира.

Целевая аудитория:

1. Выбор сегмента рынка:

* мужчины, женщины, дети любого возраста.

1. Портрет целевой аудитории:

* географические характеристики: жители всего мира, которые имею доступ в Интернет, люди, начиная от среднего класса и выше;
* демографические: в приоритете мужской пол, в возрасте от 15 до 35(преимущественно), независимо от того женат или нет. Национальность, профессия, образование не имеет значения;
* экономические: занятость (лица, у которых есть свободное время) со средним уровнем дохода;
* психологические: любой стиль жизни;
* поведенческие: всегда следит за новостями спорта.

Тип сайта: сайт визитка.

Информационные и функциональные блоки сайта:

* главная;
* преимущества;
* работа;
* события;
* информация;
* контакты.

Технические требования: кросс-браузерная адаптивная верстка для устройств с любым разрешением. Без использования сторонних CMS. С использованием framework bootstrap 3. Верстка с использованием технолгий HTML5 и CSS3.

Цветовая гамма оформления: комбинация светлых и темных оттенков с ярким акцентом, представленным красным цветом. Дизайн сайта имеет легкую для восприятия типографику, четкую сетку блоков, простую структуру.

## 2.2 Процесс верстки

Для написания кода воспользуемся текстовым редактором Sublime Text Build 3103. А также Bootstrap – свободный набор инструментов для создания сайтов и веб-приложений. Style.css – для создания кроссбраузерной анимации. Modernizr.js – для определения возможностей браузера клиента. Для добавления графических элементов, необходимо скачать с сайта Font Awesome Icons файлы с расширением css и шрифты.

Далее приступаем к написанию кода в файле index.html. Прописываем элемент <!DOCTYPE>, который предназначен для указания типа текущего документа. Записываем стандартную разметку <html>, внутри <head> и <body>. Далее в <head> указываем метатег charset, в котором укажем кодировку UTF-8, и другой метатег <meta name="viewport" content="width=device-width, initial-scale=1.0"> для адекватного отображения на мобильных устройствах. Прописываем <title>Сайт</title>.Прописываем подключение стилей <link rel="stylesheet" href="css/bootstrap.css">, <link rel="stylesheet" href="css/font-awesome.min.css"> и <link rel="stylesheet" href="css/font-awesome.min.css">

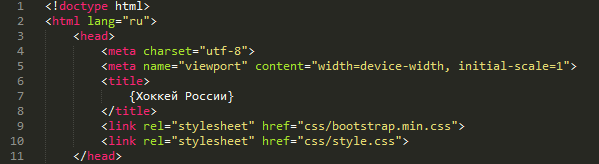


Рисунок 1 – Тег <head>

Прописываем шапку сайта. Строка <nav class="navbar navbar-inverse"> предполагает что navbar – навигация, navbar-inverse – темная расцветка для navbar'а. Далее создаем класс контейнер <div class="container- fluid ">, внутри него прописываем <div class="navbar-header"> – заголовок шапки. Создаем ссылку на логотип-иконку <a class="navbar-brand" href="#"><img src="Img/logo.png"</a>.

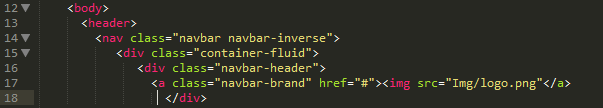


Рисунок 2 – Шапка сайта

Добавляем следующий блок <ul class="nav navbar-nav">, который устанавливает список,в нашем случае из 8 элементов <li><a href="#">ФХР</a></li>, аналогично добавляем остальные названия разделов в меню.

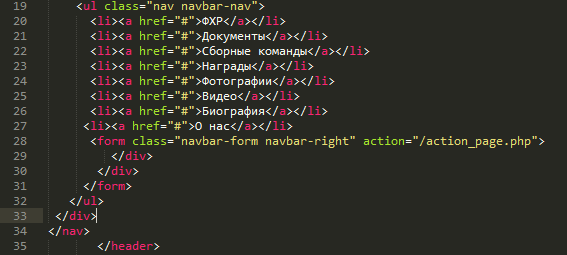


Рисунок 3 – Блок «Меню»

Далее устанавливаем цвет фона нашей страницы <main> <style> body

{ background: url("img/fon.jpg"); }</style>.



Рисунок 4 − Фон

Создаем секцию <section id="osnova">, в ней создаем еще блок с классом контейнер <div class="container">. В контейнере создаем <div class="row text-center">. Добавляем заголовки <h1>Олимпийские чемпионы</h1> и изображение <img src="img/baner.jpg" alt="">.

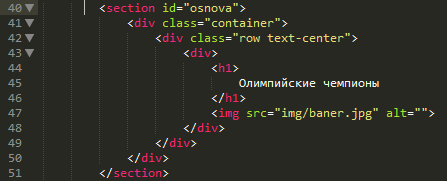


Рисунок 5 – Основной блок

Аналогично прописываем второй блок нашего сайта.

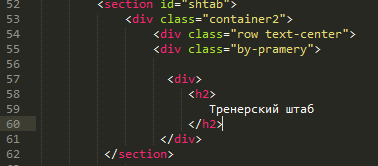


Рисунок 6 – Блок «Тренерский штаб»

Третий блок создаем аналогично 1 и 2 блокам. Делаем отступ и прописываем <div class="col-md-3">, добавляем заголовок <h3>Олег Знарок </h3>, добавляем изображение <img src="img/rash.jpg" width="100" height="111" alt="">. Таким образом добавляем еще 3 заголовка и изображения.

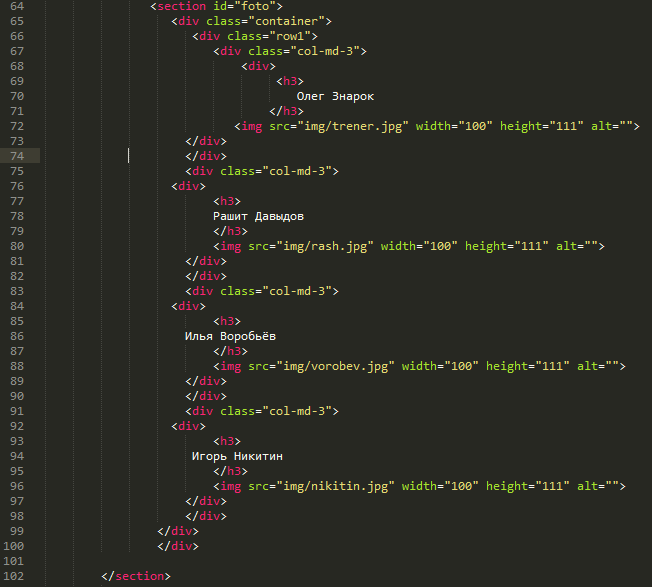


Рисунок 7 – Блок с фотографиями

Создаем новую секцию <section id="sobutiya"> , в ней создаем блок <div class="container"> , затем <div class="row"> <div class="panel-primary"> <div class="col-md-8">. Вставляем заголовок нашей таблицы <h1>Cобытия </h1>. Прописываем параметры нашей таблицы <table border="1" width="100%" cellpadding="5">. Заполняем таблицу 2\*5 <tr> <th>07:30 Динамо-Казань-2 — Водник-2 5:4</th> <th>13:30 Волга-2 — Енисей-2 3:3</th </tr>.

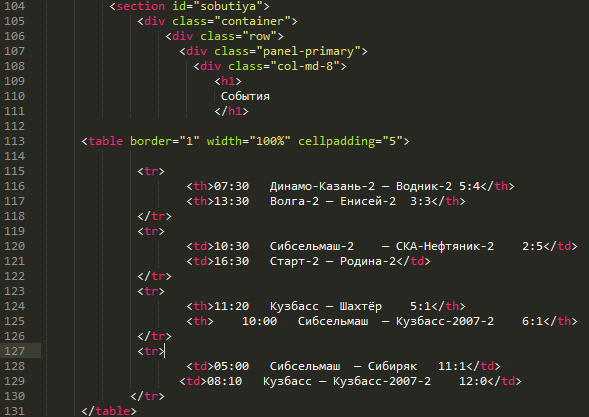


Рисунок 8 – Блок «Таблица»

Добавляем новый блок <div class="text">. Вставляем заголовок <h3>Интересные факты</h3>. Далее прописываем текс <p class="line"> Большинству неизвестно, что первая шайба для хоккея имела форму квадрата.</p>, таких строк у нас 10.

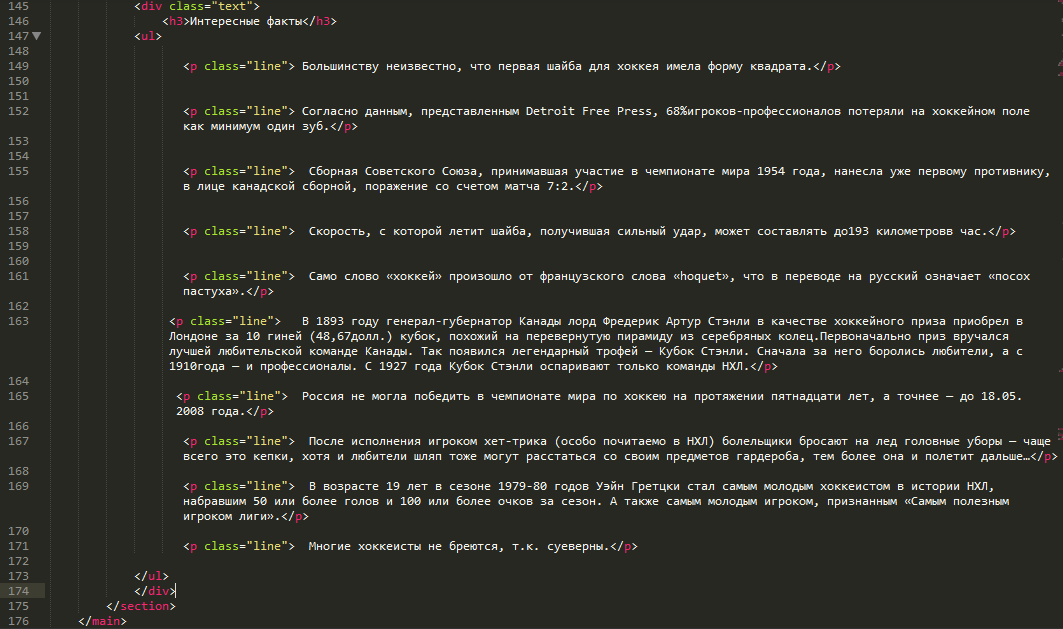


Рисунок 9 – Блок «Текст»

Переходим к созданию подвала сайта, для этого прописываем <div class="container"><div class="row"> <div class="col-lg-12"> <ul class="list-inline">. Далее пишем название <li> <a href="#">Главная</a> </li> <li class="footer-menu-divider">&sdot;</li> и таких блоков у нас 4. Так же прописываем копирайтер <p class="copyright text-muted small">Copyright &copy; Your Company 2018. Попова Виктория</p>. Подключаем скрипты jquery.min.js и bootstrap.min.js.

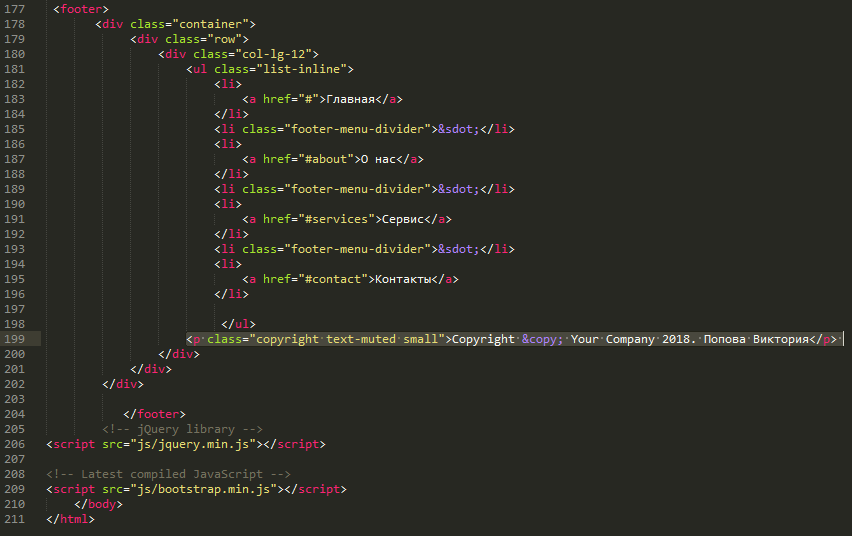


Рисунок 10 − Подвал

Приступаем к написанию кода в файле style.css. Прописываем класс . .text-center, где text-align: center обозначает выравнивание текста по центру.



Рисунок 11 – Текст по центру

Для nav li, что является панелью меню мы прописываем list-style-position: inside, что определяет, как будет размещаться маркер относительно текста, устанавливаем шрифт и размер, отобразили список горизонтально.

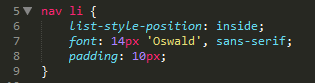


Рисунок 12 – Стили для меню

Для класса .text прописываем размер стиль шрифта font-style: italic; font-size:110%. Для класса .line прописываем border-left:3px groove;margin-left:20px;

padding-left:10px, что значит ширина границы, внутренний отступ и внешний отступ для левой границы.

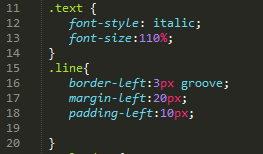


Рисунок – Текст

Для класса изображений .col-md-3 img задаем такие параметры, как ширину, высоту. Также при наведении на изображение делаем такой эффект, как увеличение. Приписываем границы увеличения и время.

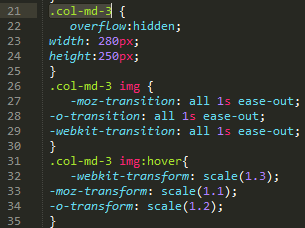


Рисунок 14 – Изображения

Для заголовка второго контейнера устанавливаем цвет lightgrey.



Рисунок 15 – Цвет заголовка

Для тега img применяем свойство border: 2px solid DimGrey. Для взятия изображений в рамку.

Для подвала устанавливаем цвет и положения копирайтера background-color:black; color:white, text-align: right.

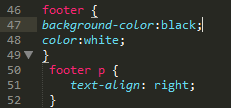


Рисунок 16 – Стили для «подвала»

Таким образом, текст копирайтера будет располагаться справа, написан будет белым цветом на черном фоне.

# ЗАКЛЮЧЕНИЕ

Итак, поняв принцип построения Web-страницы, изучив возможности соединения в ней различных видов информации, мы можем смело сказать, что Web-страницы, с их потенциалом могут применятся для различных целей. Web-страница – это лицо той фирмы, того учреждения, человека, который разместил ее в WWW. Именно поэтому сегодня Web-дизайну уделяется такое огромное внимание, ибо от него на прямую зависит популярность того или иного информационного ресурса Сети. Недаром сейчас профессия Web-дизайнера является одной из самых высокооплачиваемых. Человек, создающий Web-страницу, соединяет свои знания и навыки со своим творческим потенциалом. Умение творить – вот что отличает настоящего Web-дизайнера. Для того чтобы создать Web-страницу, которая бы радовала глаз, нужно сочетать в себе качества художника и программиста.

В процессе создания web-сайта я познакомился с методом программирования HTML, научился создавать web-страницы и использовать базовые теги. Работать с файлами разных форматов и расширений для корректного отображения в среде интернета.

Подводя итог всему выше сказанному, хочется отметить, что HTML стал тем форматом передачи данных, который наиболее полно и качественно удовлетворяет запросы современного общества. Несомненным фактом является и то, что будущее именно за HTML.

# СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Свободная энциклопедия «Википедия». HTML [Электронный ресурс]: Режим доступа: https://ru.wikipedia.org/wiki/HTML.
2. Основы HTML [Электронный ресурс]: Режим доступа: http://html-exp.narod.ru/base.htm.
3. Mart Комолова Н. В.,Яковлева Е. С. HTML: Самоучитель. 2-е изд. – СПб.: Питер, 2015. – 288 с.
4. Ляпин Д .А. PHP - это просто. – СПб.: БХВ-Петербург, 2016. – 576 с.
5. Thomas S. So you want to be a front end developer. / Пер. с англ. Шкребко А., 2016.
6. Основы CSS 3. [Электронный ресурс]: Режим доступа: http://skoolsite.ru/css-3/.
7. Свободная энциклопедия «Википедия». CSS [Электронный ресурс]: Режим доступа: https://ru.wikipedia.org/wiki/CSS.
8. Веб таблицы стилей. [Электронный ресурс]: Режим доступа: https://www.w3.org/Style/Overview.ru.html.
9. Эрик А. Мейер. CSS – каскадные таблицы стилей. 3-е изд./ Пер. с англ. – СПб.: Символ-Плюс, 2014. – 576 с.
10. Основы CSS [Электронный ресурс]: Режим доступа: http://www.css-style.su/.
11. Навыки и знания фронтенд-веб-разработчика. [Электронный ресурс]: Режим доступа: http://jetbrains.ru/careers/requirements/frontender/.
12. «Front-end» и «Back-end» разработка. [Электронный ресурс]: Режим доступа: http://hinex.ru/front-end-i-back-end-razrabotka.
13. Фронт-энд разработчик. [Электронный ресурс]: Режим доступа: http://www.profguide.ru/professions/front\_end\_developer.html.
14. Front-end и Back-end разработка. [Электронный ресурс]: Режим доступа: http://juice-health.ru/programming/web-development/504-front-end-i-back-end-razrabotka.
15. Робсон Э., Фримен Э. Изучаем HTML, XHTML и CSS. 2-е изд./ Пер. с англ. В. Черник – СПб.: Питер, 2014. — 720 с.
16. Свободная энциклопедия «Википедия». Элементы HTML [Электронный ресурс]: Режим доступа: https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D1%8B\_HTML.
17. Свободная энциклопедия «Википедия». HTML5 [Электронный ресурс]: Режим доступа: https://ru.wikipedia.org/wiki/HTML5.
18. Основы HTML5. [Электронный ресурс]: Режим доступа: https://www.ibm.com/developerworks/ru/library/wa-html5fundamentals/.
19. Мальчук Е.В. HTML и CSS. – М.: Издательский дом «Вильямс», 2014. – 416 с.
20. Макфарланд Д. CSS. – СПб.: Питер, 2014. – 236 с.

# ПРИЛОЖЕНИЕ А

**Листинг файла index.html**

<!doctype html>

<html lang="ru">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>

{Хоккей России}

</title>

<link rel="stylesheet" href="css/bootstrap.min.css">

<link rel="stylesheet" href="css/style.css">

</head>

<body>

<header>

<nav class="navbar navbar-inverse">

<div class="container-fluid">

<div class="navbar-header">

<a class="navbar-brand" href="#"><img src="Img/logo.png"</a>

</div>

<ul class="nav navbar-nav">

<li><a href="#">ФХР</a></li>

<li><a href="#">Документы</a></li>

<li><a href="#">Сборные команды</a></li>

<li><a href="#">Награды</a></li>

<li><a href="#">Фотографии</a></li>

<li><a href="#">Видео</a></li>

<li><a href="#">Биография</a></li>

<li><a href="#">О нас</a></li>

<form class="navbar-form navbar-right" action="/action\_page.php">

</div>

</div>

</form>

</ul>

</div>

</nav>

</header>

<main>

<style>

body { background: url("img/fon.jpg"); }

</style>

<section id="osnova"

**Продолжение приложения А**

<div class="container">

<div class="row text-center">

<div>

<h1>

Олимпийские чемпионы

</h1>

<img src="img/baner.jpg" alt="">

</div>

</div>

</div>

</section>

<section id="shtab">

<div class="container2">

<div class="row text-center">

<div class="by-pramery">

<div>

<h2>

Тренерский штаб

</h2>

</div>

</section>

<section id="foto">

<div class="container">

<div class="row1">

<div class="col-md-3">

<div>

<h3>

Олег Знарок

</h3>

<img src="img/trener.jpg" width="100" height="111" alt="">

</div>

</div>

<div class="col-md-3">

<div>

<h3>

Рашит Давыдов

</h3>

<img src="img/rash.jpg" width="100" height="111" alt="">

</div>

</div>

**Продолжение приложения А**

<div class="col-md-3">

<div>

<h3>

Илья Воробьёв

</h3>

<img src="img/vorobev.jpg" width="100" height="111" alt="">

</div>

</div>

<div class="col-md-3">

<div>

<h3>

Игорь Никитин

</h3>

<img src="img/nikitin.jpg" width="100" height="111" alt="">

</div>

</div>

</div>

</div>

</section>

<section id="sobutiya">

<div class="container">

<div class="row">

<div class="panel-primary">

<div class="col-md-8">

<h1>

Cобытия

</h1>

<table border="1" width="100%" cellpadding="5">

<tr>

<th>07:30 Динамо-Казань-2 — Водник-2 5:4</th>

<th>13:30 Волга-2 — Енисей-2 3:3</th>

</tr>

<tr>

<td>10:30 Сибсельмаш-2 — СКА-Нефтяник-2 2:5</td>

<td>16:30 Старт-2 — Родина-2</td>

</tr>

<tr>

<th>11:20 Кузбасс — Шахтёр 5:1</th>

**Продолжение приложения А**

<th> 10:00 Сибсельмаш — Кузбасс-2007-2 6:1</th>

</tr>

<tr>

<td>05:00 Сибсельмаш — Сибиряк 11:1</td>

<td>08:10 Кузбасс — Кузбасс-2007-2 12:0</td>

</tr>

</table>

</div>

<div class="row">

<div class="col-md-4">

<div>

<h1>

Трансляции

</h1>

<img src="img/utub.png" width="176" height="110" alt="">

</div>

</div>

</div>

</div>

</div>

<div class="text">

<h3>Интересные факты</h3>

<ul>

<p class="line"> Большинству неизвестно, что первая шайба для хоккея имела форму квадрата.</p>

<p class="line"> Согласно данным, представленным Detroit Free Press, 68%игроков-профессионалов потеряли на хоккейном поле как минимум один зуб.</p>

<p class="line"> Сборная Советского Союза, принимавшая участие в чемпионате мира 1954 года, нанесла уже первому противнику, в лице канадской сборной, поражение со счетом матча 7:2.</p>

<p class="line"> Скорость, с которой летит шайба, получившая сильный удар, может составлять до193 километровв час.</p>

**Продолжение приложения А**

<p class="line"> Само слово «хоккей» произошло от французского слова «hoquet», что в переводе на русский означает «посох пастуха».</p>

<p class="line"> В 1893 году генерал-губернатор Канады лорд Фредерик Артур Стэнли в качестве хоккейного приза приобрел в Лондоне за 10 гиней (48,67долл.) кубок, похожий на перевернутую пирамиду из серебряных колец.Первоначально приз вручался лучшей любительской команде Канады. Так появился легендарный трофей — Кубок Стэнли. Сначала за него боролись любители, а с 1910года — и профессионалы. С 1927 года Кубок Стэнли оспаривают только команды НХЛ.</p>

<p class="line"> Россия не могла победить в чемпионате мира по хоккею на протяжении пятнадцати лет, а точнее — до 18.05. 2008 года.</p>

<p class="line"> После исполнения игроком хет-трика (особо почитаемо в НХЛ) болельщики бросают на лед головные уборы — чаще всего это кепки, хотя и любители шляп тоже могут расстаться со своим предметов гардероба, тем более она и полетит дальше…</p>

<p class="line"> В возрасте 19 лет в сезоне 1979-80 годов Уэйн Гретцки стал самым молодым хоккеистом в истории НХЛ, набравшим 50 или более голов и 100 или более очков за сезон. А также самым молодым игроком, признанным «Самым полезным игроком лиги».</p>

<p class="line"> Многие хоккеисты не бреются, так как суеверны.</p>

</ul>

</div>

</section>

</main>

<footer>

<div class="container">

<div class="row">

<div class="col-lg-12">

<ul class="list-inline">

<li>

<a href="#">Главная</a>

</li>

<li class="footer-menu-divider">&sdot;</li>

<li>

<a href="#about">О нас</a>

**Продолжение приложения А**

</li>

<li class="footer-menu-divider">&sdot;</li>

<li>

<a href="#services">Сервис</a>

</li>

<li class="footer-menu-divider">&sdot;</li>

<li>

<a href="#contact">Контакты</a>

</li>

</ul>

<p class="copyright text-muted small">Copyright &copy; Your Company 2018. Попова Виктория</p>

</div>

</div>

</div>

</footer>

<!-- jQuery library -->

<script src="js/jquery.min.js"></script>

<!-- Latest compiled JavaScript -->

<script src="js/bootstrap.min.js"></script>

</body>

</html>

# ПРИЛОЖЕНИЕ Б

**Макет сайта**

