

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет информационной безопасности

Кафедра инфокоммуникационных технологий

ВЕБ-ТЕХНОЛОГИИ В ИНФОКОММУНИКАЦИЯХ

Разработка веб-ресурса



Минск 2022

Содержание

Контрольная работа Разработка веб-ресурса	3
Что такое Bootstrap?	3
Адаптивный веб-дизайн	3
Что вам понадобится.....	3
Пошаговая разработка веб-ресурса с использованием Bootstrap.....	3
Правила веб-типографики	13
Шрифт Awesome.....	17
Favicon сайта	19
Сервисы подбора цвета для веб-дизайна	19
Анализ ключевых слов	21
Яндекс.Вордстат	21
Задание для контрольной работы.....	24

Контрольная работа Разработка веб-ресурса

Что такое Bootstrap?

Bootstrap – это бесплатная интерфейсная платформа для быстрой и удобной разработки веб-сайтов.

Bootstrap включает в себя HTML и CSS на основе шаблонов дизайна для типографии, формы, кнопки, таблицы, навигация, модальные, изображения карусели и многие другие, а также дополнительные плагины JavaScript.

Bootstrap также дает возможность легко создавать адаптивные дизайны.

Всю информацию вы найдете на [русскоязычном ресурсе](#) посвященном Bootstrap 4.

Адаптивный веб-дизайн

Адаптивный веб-дизайн – это отзывчивый веб-дизайн сайта, который позволяет автоматически настроить себя для всех устройств, от небольших телефонов до больших настольных компьютеров.



Что вам понадобится

Для разработки необходим [Bootstrap](#) и редактор кода, с работающим плагином emmet.

Пошаговая разработка веб-ресурса с использованием Bootstrap

Шаг 1. Создание базовой HTML страницы

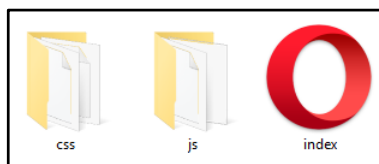
Открываем редактор кода, создаем пустую страничку, со следующим содержимым и сохраняем (файл – сохранить как), в `index.html`.

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <title>MEG</title>
</head>
<body>
  <!-- Будущий код -->
</body>
</html>
```

Шаг 2. Установка Bootstrap 4

Есть несколько способов установки и внедрения Bootstrap 4.

1. Переходим на сайт <https://getbootstrap.com> и скачиваем актуальную версию. Далее распаковываем архив в тот каталог где сохранен `index.html`.



Подключаем основные файлы.

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <title>MEG</title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="css/bootstrap.min.css">
</head>
<body>
  <!-- Будущий код -->
  <script type="text/javascript" src="js/jquery-3.6.0.min.js"></script>
  <script type="text/javascript" src="js/bootstrap.min.js"></script>
</body>
</html>
```

В этом коде используется элемент, отвечающий за адаптивность. Подключены основной `.css`, `.js` файлы, и библиотека [jQuery](https://jquery.com) (jquery.com), которую тоже необходимо скачать.

2. Используем CDN (сеть доставки контента).

Bootstrap можно подключить к странице с использованием сети доставки контента. Для этого необходимо добавить следующую ссылку в элемент `<head>`.

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <title>MEG</title>
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/css/bootstrap.min.css" integrity="sha384-B0vP5xmATw1+K9KRQJQERJvTumQW0nPEzvF6L/
Z6nnonJ3oU0FUFpCjEUQouq2+1" crossorigin="anonymous">
</head>
```

Ссылки на актуальные версии CDN можно получить на этом [ресурсе](#).

Перед закрывающимся элементом `</body>` подключаем библиотеку `jQuery` и `bootstrap.min.js`.

```
<body>
  <!-- Будущий код -->
  <script type="text/javascript" src="js/jquery-3.6.0.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5Ib027qvyjSMFhJOMlkFfuWVxZxUPnCA712mCWNIP69mGCD8wGNIcPD7Txa"
crossorigin="anonymous"></script>
</body>
</html>
```

Чтобы проверить работу фреймворка, можно прописать код какого-нибудь `bootstrap`-элемента, например, кнопки.

```
<body>
  <!-- Будущий код -->
  <button class="btn btn-warning" type="button">Кнопка</button>
  <script type="text/javascript" src="js/jquery-3.6.0.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-Tc5Ib027qvyjSMFhJOMlkFfuWVxZxUPnCA712mCWNIP69mGCD8wGNIcPD7Txa"
crossorigin="anonymous"></script>
</body>
</html>
```

Кнопка

Если кнопка появилась на странице, значит Bootstrap успешно подключен.

Шаг 3. Создание навигации

Навигационные панели в Bootstrap создают с использованием класса `.navbar`. Это так называемая обертка, в которую помещают элементы, формирующие навигационную панель. Она располагается в верхней части страницы и не исчезает при использовании прокрутки.

Добавим в `index.html` элемент `<nav>` с классом `.navbar`, внутри которого, с использованием других классов, таких как `.navbar-brand`, `.navbar-toggler` и `.nav-item`, создадим специальные элементы и структуру системы навигации по

сайту. Использование класса `.fixed-top` необходимо для фиксирования навигационной панели в верхней части страницы.

```
<!-- navbar -->
<nav class="navbar navbar-expand-lg fixed-top">
  <a class="navbar-brand" href="#">Главная</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false"
    aria-label="Toggle navigation"><span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-4">
      <li class="nav-item"><a class="nav-link" href="#">Обо мне</a></li>
      <li class="nav-item"><a class="nav-link" href="#">Статьи</a></li>
      <li class="nav-item"><a class="nav-link" href="#">Контакты</a></li>
    </ul>
  </div>
</nav>
```

Затем создадим файл каскадной таблицы стилей `main.css` поместив его в папку `/css`. Добавим в файл `main.css` селекторы и правила, задающие цветовое оформление навигационной панели.

```
.navbar{
  background:#669999;
}
.nav-link, .navbar-brand{
  color: #f4f4f4;
  cursor: pointer;
}
.nav-link{
  margin-right: 1em !important;
}
.nav-link:hover{
  background: #f4f4f4;
  color: #669999;
}
.navbar-collapse{
  justify-content: flex-end;
}
.navbar-toggler{
  background:#fff !important;
}
.navbar-collapse{
  justify-content: flex-end;
}
.bg-dark{
  background-color:#343a40!important
}
.bg-primary{
  background-color:#343a40!important
}
```

Подключим `main.css` к странице, поместив в элемент `<head>` файла `index.html` следующий код.

```
<link rel="stylesheet" type="text/css" href="css/main.css">
```

Сетка Bootstrap построена на основе `flexbox`, поэтому для выравнивания содержимого нужно использовать его свойства. Например, для того, чтобы поместить меню навигационной панели справа, нужно использовать свойство `justify-content` и установить его значение во `flex-end`. Поместим этот код в `main.css`.

```
.navbar-collapse{
  justify-content: flex-end;
}
```

Для настройки цвета фона навигационной панели воспользуемся классами `.bg-dark` (темный фон) и `.bg-primary` (основной цвет фона). Поместим это код в `main.css`.

```
.bg-dark{
  background-color:#343a40!important
}
.bg-primary{
  background-color:#343a40!important
}
```

Шаг 4. Создание шапки страницы

Для создания шапки страницы необходимо использовать элемент `<header>`. Вставим этот код в файл `index.html`.

```
<header class="header">
  <!-- Будущий код -->
</header>
```

Поместим в шапку страницы какое-нибудь фоновое изображение. Подберем нужное изображение и сохраним его в папке `/images`. Добавим следующий код в файл каскадной таблицы стилей `main.css`.

```
/*header style*/
.header{
  background-image: url('../images/header.jpg');
  background-attachment: fixed;
  background-size: cover;
  background-position: center;
}
```

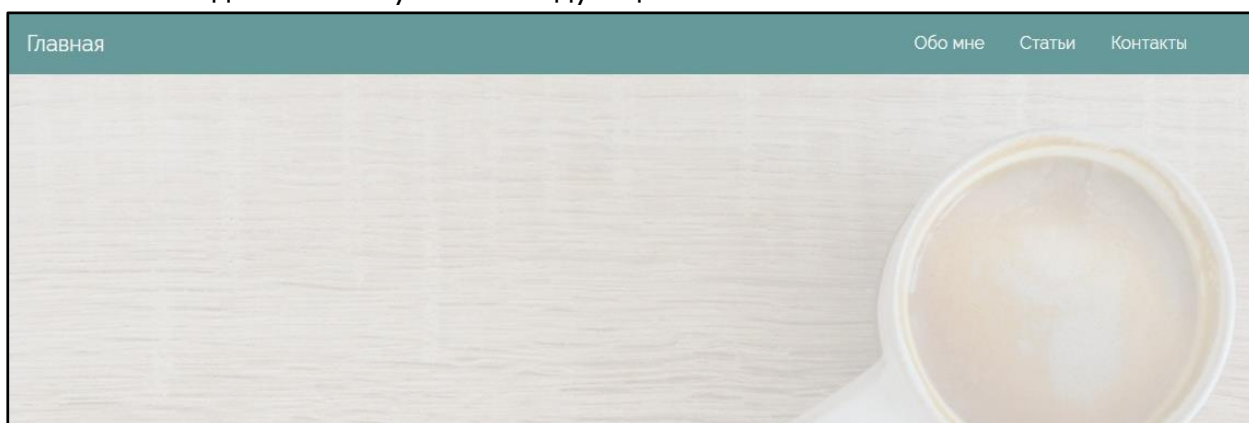
Необходимо чтобы шапка занимала всю высоту окна, поэтому тут воспользуемся возможностями jQuery. Создадим файл `main.js` в имеющейся папке `/js`. Подключим его к `index.html` перед закрывающим элементом `<body>`.

```
<script type="text/javascript" src="js/main.js"></script>
</body>
</html>
```

В файл `main.js` добавим следующий код.

```
$(document).ready(function(){
  $('.header').height($(window).height());
})
```

В итоге должно получиться следующее.



Добавим в `<header>` элемент `<div>`, назначив ему класс `.overlay`, для создания блока, который будет расположен поверх фонового изображения шапки.

```
<header class="header">
  <div class="overlay"></div>
</header>
```

В файл `main.css`, добавим следующие свойства `.overlay`.

```
.overlay{
  position: absolute;
  min-height: 100%;
  min-width: 100%;
  left: 0;
  top: 0;
  background: rgba(244, 244, 244, 0.79);
}
```

Поместим в `<header>` новый элемент `<div>` с классом `.container`. Это – вспомогательный класс фреймворка Bootstrap, предназначенный для размещения содержимого с учетом возможностей отзывчивого макета.

```
<header class="header">
  <div class="overlay"></div>
  <div class="container"></div>
</header>
```

Сюда же добавим еще один элемент `<div>`, которому назначим класс `.description`.

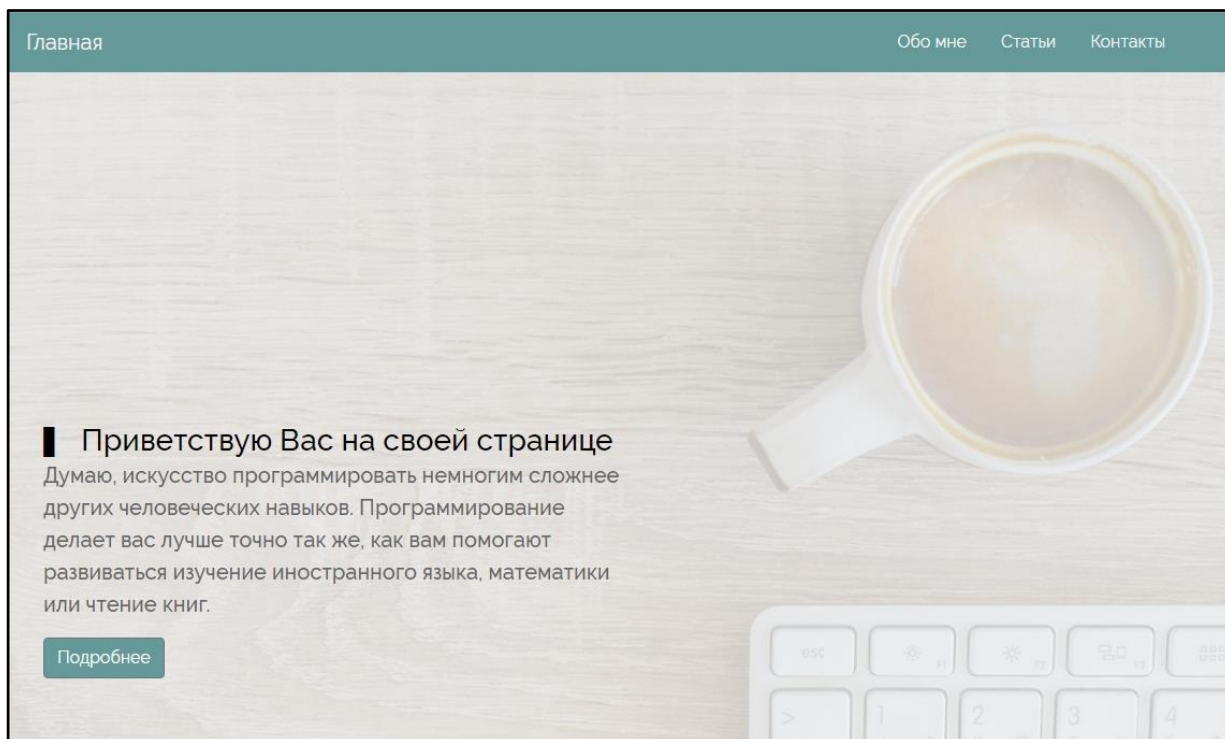
```
<header class="header">
  <div class="overlay"></div>
  <div class="container"></div>
  <div class="description">
    <h3>
      <font color="#000000">■ Приветствую Вас на своей странице</font>
      Думаю, искусство программировать немногим сложнее других человеческих навыков. Программирование делает вас лучше точно так же, как вам помогают развиваться
      изучение иностранного языка, математики или чтение книг.</p>
      <button class="btn btn-outline-secondary">Подробнее</button>
    </h3>
  </div>
</header>
```

В Bootstrap имеется множество классов, предназначенных для кнопок. К элементу `<button>` были добавлены классы `.btn` и `.btn-outline-secondary`.

Настроим стили для класса `.description` поместив код в `main.css`.

```
.description{
  position: absolute;
  top: 30%;
  margin: auto;
  padding: 2em;
}
.description h1{
  color:#F97300 ;
}
.description p{
  color:#666;
  font-size: 20px;
  width: 50%;
  line-height: 1.5;
}
.description button{
  border:1px solid #669999;
  background:#669999;
  color:#fff;
}
```

Вот как будет выглядеть шапка страницы после выполнения вышеописанных действий.



Шаг 4. Создание раздела «Обо мне»

Для создания раздела со сведениями будем использовать возможности сетки Bootstrap для того, чтобы сформировать макет раздела, состоящий из двух частей.



Добавив к родительскому элементу `<div>` класс `.row`.

Первая часть макета расположена слева, и содержит изображение. Вторая часть, расположена справа, в ней размещено описание.

```
<!-- about -->
<div class="about">
  <h1 class="text-center">Обо мне</h1>
  <div class="container">
    <div class="row">
      <div class="col-lg-4 col-md-4 col-sm-12">
        
        <span class="text-justify">Ст. преподаватель <br> кафедры ИКТ БГУИР</span>
      </div>
      <div class="col-lg-8 col-md-8 col-sm-12 desc">
        <h3><font color="#000000">Е.Г. Макейчик</font></h3>
        <p>Читаемые учебные дисциплины:</p>
        <ul>
          <li>Веб-технологии в инфокоммуникациях</li>
          <li>Объектно-ориентированное программирование</li>
          <li>Программная обработка мультимедийной информации</li>
        </ul>
        <p>Область профессиональных интересов:</p>
        <ul>
          <li>Помехоустойчивое кодирование</li>
          <li>Программная обработка информации</li>
          <li>Методики преподавания технических дисциплин</li>
        </ul>
      </div>
    </div>
  </div>
</div>
```

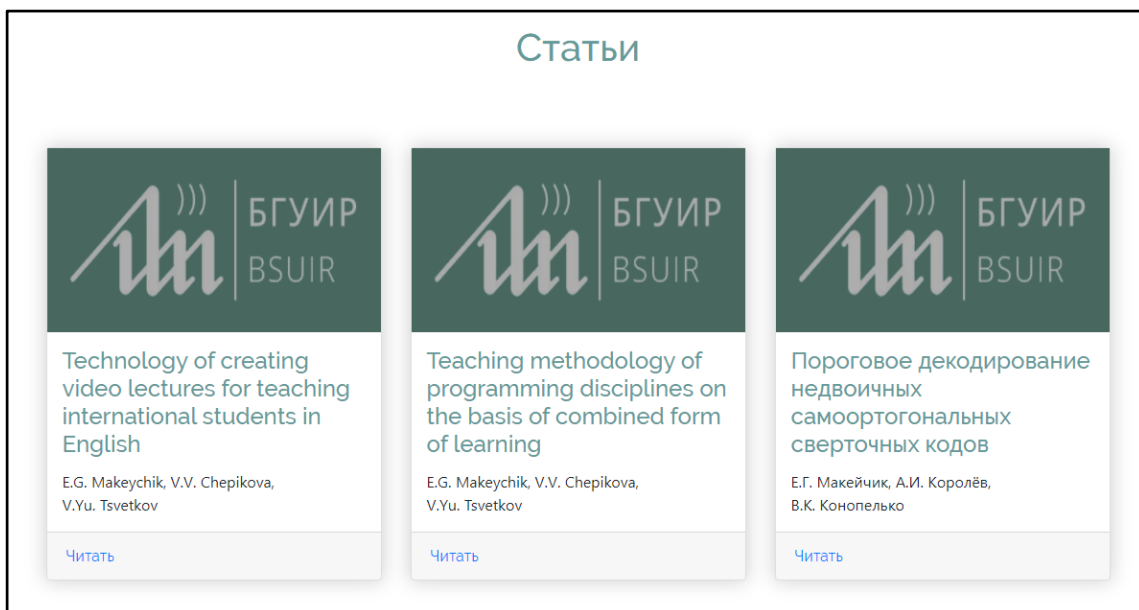
Здесь использована Грид-система ([Grid System](#)) макетов Bootstrap. Она основана на 12 колонках и имеет несколько уровней для медиазапросов.

Настроим стили классов поместив код в `main.css`.

```
.about{
  margin: 4em 0;
  padding: 1em;
  position: relative;
}
.about h1{
  color:#669999;
  margin: 2em;
}
.about img{
  height: 100%;
  width: 100%;
  border-radius: 50%
}
.about span{
  display: block;
  color: #888;
  position: absolute;
  left: 115px;
}
.about .desc{
  padding: 2em;
  border-left:4px solid #10828C;
}
.about .desc h3{
  color: #10828C;
}
.about .desc p{
  line-height:2;
  color:#888;
}
```


Шаг 5. Создание раздела «Статьи» и работа с карточками

Для создания этого раздела необходимо использовать так называемые карточки (cards в терминологии Bootstrap).



Для того чтобы создать карточку, нужно включить в макет элемент `<div>` и добавить к нему класс `.card`. Для настройки различных элементов карточки можно использовать следующие классы:

- `.card-header` – шапка;
- `.card-body` – основное содержимое;
- `.card-title` – заголовок;
- `.card-footer` – подвал;
- `.card-image` – изображение.

```
<!-- Cards section -->
<div class="blog">
  <div class="container">
    <h1 class="text-center">Статьи</h1>
    <div class="row">
      <div class="col-md-4 col-lg-4 col-sm-12">
        <div class="card">
          <div class="card-img"></div>
          <div class="card-body">
            <h4 class="card-title">Technology of creating video lectures for teaching international students in English</h4>
            <p class="card-text">E.G. Makeychik, V.V. Chepikova, <br> V.Yu. Tsvetkov</p>
          </div>
          <div class="card-footer">
            <a href="" class="card-link">Читать</a>
          </div>
        </div>
      </div>
      <div class="col-md-4 col-lg-4 col-sm-12">
        <div class="card">
          <div class="card-img"></div>
          <div class="card-body">
            <h4 class="card-title">Teaching methodology of programming disciplines on the basis of combined form of learning</h4>
            <p class="card-text">E.G. Makeychik, V.V. Chepikova, <br> V.Yu. Tsvetkov</p>
          </div>
          <div class="card-footer">
            <a href="" class="card-link">Читать</a>
          </div>
        </div>
      </div>
      <div class="col-md-4 col-lg-4 col-sm-12">
        <div class="card">
          <div class="card-img">
            
          </div>
          <div class="card-body">
            <h4 class="card-title">Пороговое декодирование не двоичных самоортогональных сверточных кодов</h4>
            <p class="card-text">Е.Г. Макейчик, А.И. Королёв, <br> В.К. Конопелько</p>
          </div>
          <div class="card-footer">
            <a href="" class="card-link">Читать</a>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

Добавим стили для карточек поместив код в `main.css`.

```
.blog{
  margin: 4em 0;
  position: relative;
}
.blog h1{
  color:#669999;
  margin: 2em;
}
.blog .card{
  box-shadow: 0 0 20px #ccc;
}
.blog .card img{
  width: 100%;
  height: 12em;
}
.blog .card-title{
  color:#669999;
}
.blog .card-body{
  padding: 1em;
}
```

Шаг 6. Создание формы обратной связи

В данном разделе страницы будет находиться форма, с помощью которой посетители сайта могут отправлять сообщения.

Связаться со мной

Для того, чтобы стилизовать элементы и обеспечить их отзывчивость, воспользуемся возможностями Bootstrap.

Bootstrap используется класс `.form-control` для полей ввода, `.input-group-prepend` для значков и меток. Подробности об этом можно найти в документации к Bootstrap. Поместим каждое поле ввода в элемент `<div>`, которому назначен класс `.form-group`.

```
<!-- Contact form -->
<div class="contact-form">
  <div class="container">
    <form>
      <div class="row">
        <div class="col-lg-4 col-md-4 col-sm-12">
          <h1>Связаться со мной</h1>
        </div>
        <div class="col-lg-8 col-md-8 col-sm-12 right">
          <div class="form-group">
            <input type="text" class="form-control form-control-lg" placeholder="Ваше имя" name="">
          </div>
          <div class="form-group">
            <input type="email" class="form-control form-control-lg" placeholder="makeichik@bsuir.by" name="email">
          </div>
          <div class="form-group">
            <textarea class="form-control form-control-lg"></textarea>
          </div>
          <input type="submit" class="btn btn-secondary btn-block" value="Отправить" name="">
        </div>
      </div>
    </form>
  </div>
</div>
```

Добавим стили для формы обратной связи в `main.css`.

```
.contact-form{
  margin: 6em 0;
  position: relative;
}
.contact-form h1{
  padding: 2em 1px;
  color: #669999;
}
.contact-form .right{
  max-width: 600px;
}
.contact-form .right .btn-secondary{
  background: #669999;
  color: #fff;
  border: 0;
}
.contact-form .right .form-control::placeholder{
  color: #888;
  font-size: 16px;
}
```

Шаг 7. Добавление шрифта

Воспользовавшись возможностями Google Font API, изменим стандартный шрифт на **Raleway**. Для импорта шрифта добавим в файл **main.css** следующую директиву.

```
@import url('https://fonts.googleapis.com/css?family=Raleway');
```

Зададим глобальные стили для HTML-элементов, поместив код в **main.css**.

```
html, h1, h2, h3, h4, h5, h6, a{
  font-family: "Raleway";
}
```

Более подробно об правилах веб-типографики можно прочесть в [соответствующем разделе](#).

Шаг 8. Создание прокрутки страницы

Для того, чтобы страница, при клике по ссылкам навигационной панели, плавно прокручивалась к нужному разделу, воспользуемся возможностями jQuery. Добавим код в уже имеющийся файл **main.js**.

```
$(".navbar a").click(function(){
  $("body,html").animate({
    scrollTop:$( "#" + $(this).data('value')).offset().top
  },1000)
})
```

Теперь необходимо добавить атрибут **data-value** к каждой из ссылок в навигационной панели и атрибут **id** к основному элементу **<div>** каждого из разделов страницы.

```
<!-- navbar -->
<nav class="navbar navbar-expand-lg fixed-top">
  <a class="navbar-brand" href="#">Главная</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation"><span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarSupportedContent">
  <ul class="navbar-nav mr-4">
    <li class="nav-item"><a class="nav-link" data-value="about" href="#">Обо мне</a></li>
    <li class="nav-item"><a class="nav-link" data-value="blog" href="#">Статьи</a></li>
    <li class="nav-item"><a class="nav-link" data-value="contact" href="#">Контакты</a></li>
  </ul>
</div>
</nav>
```

```
<!-- about -->
<div class="about" id="about">

<!-- Cards section -->
<div class="blog" id="blog">

<!-- Contact form -->
<div class="contact-form" id="contact">
```

Шаг 9. Создание нижнего колонтитула с авторским правом

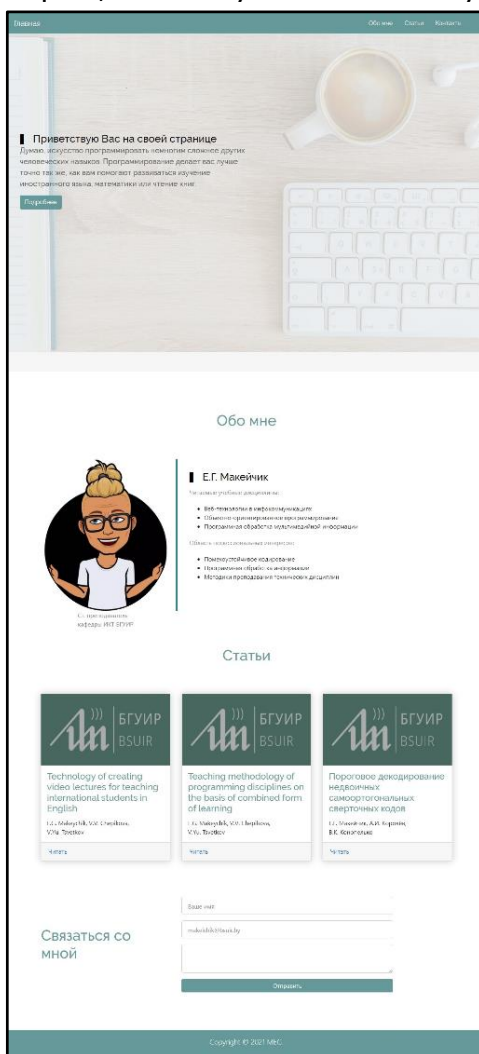
Используем класс **.footer mt-auto py-3** для создания нижнего колонтитула. Для автоматической установки текущего года необходимо получить доступ через

объектную модель документа ([DOM](#)) к методу `getFullYear()`, который возвращает текущий год. Добавим следующий код в `index.html`.

```
<footer class="footer mt-auto py-3" style="background:#669999;">
  <div class="container">
    <p class="text-center text-white lead">Copyright &copy; <script type="text/javascript">
      document.write(new Date().getFullYear());
    </script> MEG.</p>
  </div>
</footer>
```

Copyright © 2021 MEG.

Остановимся и посмотрим, что получилось на текущий момент. [Пояснение](#).



Шаг 10. Установка логотипа и фавикона

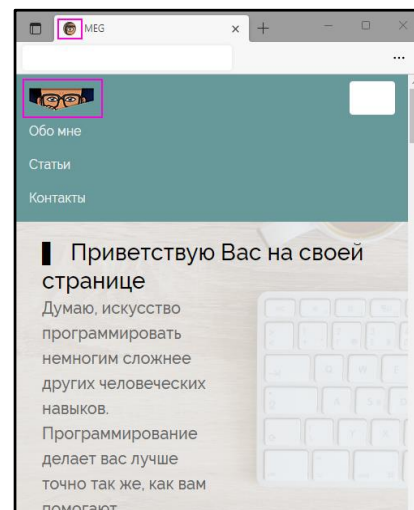
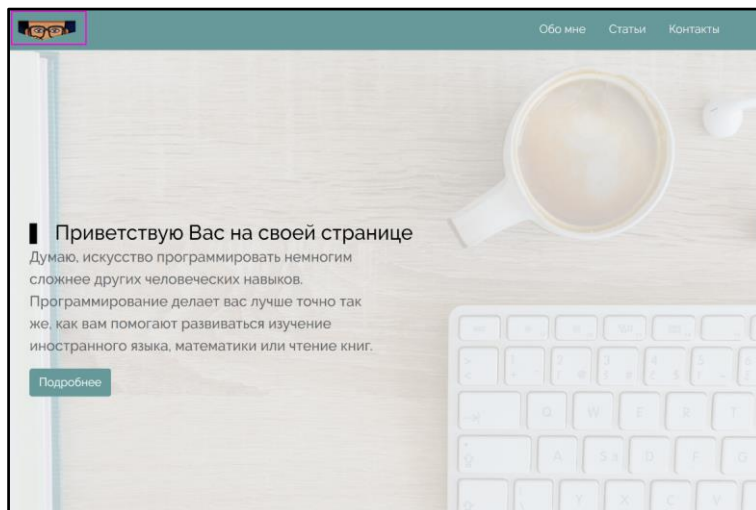
Заменяем стандартную кнопку «Главная» на логотип. Готовую иконку логотипа поместим в папку где находятся все изображения `/images`. Добавим следующий код в `index.html`.

```
<!-- navbar -->
<nav class="navbar navbar-expand-lg fixed-top">
  <a class="navbar-brand" href="#"></a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation"><span class="navbar-toggler-icon"></span>
</button>
  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-4">
      <li class="nav-item"><a class="nav-link" data-value="about" href="#">Обо мне</a></li>
      <li class="nav-item"><a class="nav-link" data-value="blog" href="#">Статьи</a></li>
      <li class="nav-item"><a class="nav-link" data-value="contact" href="#">Контакты</a></li>
    </ul>
  </div>
</nav>
```

Поместим в папку с проектом разработанный фавикон `favicon.ico`. И добавим его на страницу поместив в элемент `<head>` следующий код.

```
<!-- Favicon -->
<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">
```

Более подробно о разработке фавикона можно прочесть [в соответствующем разделе](#).



Шаг 11. Добавление социальных закладок

Используя [Font Awesome](#) создадим кнопки социальных закладок в нижнем колонтитуле. Для импорта иконочного шрифта добавим в элемент `<head>` следующий код.

```
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.1/css/all.css" integrity="sha384-gfdkjb58dAXd+lj+gudLWI+BXq44uLW5IT+brZEFzSLFm++aCM1F1V92rMkPaX4PP" crossorigin="anonymous">
```

Поместим в элемент `<footer>` ссылки на социальные сети добавив классы иконочного шрифта `.fab fa-vk`, `.fab fa-instagram`, `.fab fa-telegram`.

```
<footer class="footer mt-auto py-3" style="background:#669999;">
  <div class="container">
    <p class="text-center text-white lead">Copyright &copy; <script type="text/javascript">
      document.write(new Date().getFullYear());
    </script> MEG.
    <a class="btn btn-outline-light" href="#">
      <i class="fab fa-vk"></i>
    </a>
    <a class="btn btn-outline-light" href="#">
      <i class="fab fa-instagram"></i>
    </a>
    <a class="btn btn-outline-light" href="#">
      <i class="fab fa-telegram"></i>
    </a>
  </p>
</div>
</footer>
```



Более подробно об использовании **Font Awesome** можно прочесть в [соответствующем разделе](#).

Шаг 12. Добавление мета-тегов

Мета-теги располагаются в заголовке html-страницы, между элементами `<head>` и `</head>`, и никак не отображаются браузером (за исключением содержимого элемента `<title>`). Добавим мета-теги `keywords` и `description` в элемент `<head>` поместив их перед `<title>`.

```
<meta name="keywords" content="Макейчик, Макейчик Е.Г., БГУИР, Белорусский государственный университет информатики и радиоэлектроники, Инфокоммуникационные технологии, ИКТ">
<meta name="description" content="Персональная страница Макейчик Екатерины Геннадьевны, старшего преподавателя кафедры ИКТ БГУИР!">
<title>MEG</title>
```

Более подробно о подборе ключевых слов можно прочесть в [соответствующем разделе](#).

Правила веб-типографики

Типографика играет ключевую роль в веб-дизайне, по статистике 95% содержимого сайтов составляет текстовый контент. Шрифтовое оформление управляет настроением и создает определенную атмосферу при прочтении текстового содержимого веб-страниц.

Современная веб-типографика базируется на CSS-стилях. Меняя значения стилей браузера по умолчанию, можно сделать текстовое содержимое более привлекательным.

Семейство шрифтов (свойство font-family)

Подбор шрифта начинается с выбора гарнитуры шрифта. **Гарнитура** – шрифт или несколько шрифтов, имеющих стилистическое единство начертания. Состоит из набора знаков (обычно – цифры, буквы, знаки пунктуации, спецсимволы, но может состоять так же исключительно из неалфавитных знаков). Например, гарнитура «Times New Roman» состоит из обычного, курсивного, полужирного и множества других шрифтов этого семейства.

Гарнитуры можно разделить на две основные категории: с засечками (антиква и брусковые шрифты) и без засечек (гротески).

Шрифты без засечек имеют простой и четкий внешний вид. Шрифты с засечками, напротив, придают более серьезный и официальный тон.

В процессе чтения глаза привыкают к основному шрифту, и они утомляются, если заголовки, оглавление и второстепенный текст набраны шрифтами разной гарнитуры, не гармонирующими с основным шрифтом. Поэтому, при подборе шрифтов достаточно остановиться на одном-двух шрифтах, а акценты расставлять за счет размера, цвета, начертания и т.п.

В каких случаях можно использовать несколько шрифтов:

- в гарнитуре нет специальных начертаний (жирного, полужирного, курсивного);
- необходимо добиться сходства с определенной эпохой;
- для визуального разделения текстов нескольких типов (например, комментарии в тексте, строки кода, формулы, тексты на другом языке);
- для эстетических целей.

[Сервис TypeTester для подбора шрифта.](#)

Сочетание шрифтов с засечками и без засечек

В рекомендациях по сочетанию шрифтов преобладает негласное правило – для заголовков выбирается шрифт без засечек, а основной текст формируется шрифтом с засечками. Однако такой подход не настолько популярен, как может показаться.

На самом деле это правило касается в первую очередь печатных изданий, книгопечатания. Связано это с тем, что шрифт с засечками (небольшими росчерками на концах основных штрихов) плавно выстраивается в одну линию, облегчая восприятие текста и делая его более читабельным.

Иначе обстоят дела с отображением текста дисплеями различных устройств. Возникает проблема сглаживания и неровности засечек. Поэтому для основного текста лучше подходит шрифт без засечек с несколько увеличенной высотой строчных знаков.

Цвет шрифта (свойство color)

Цвет придает тексту четкость и выразительность. Цветные заголовки и небольшие акценты в тексте способны привлечь больше внимания, чем текст черного цвета.

Тем не менее, не стоит забывать, что любой цвет несет с собой собственное настроение, а у каждого человека есть личные ощущения, которые вызывает у них тот или иной цвет.

Теплый цвет активно привлекает внимание к тексту, делая его визуально большим по размеру, чем шрифт аналогичного размера холодных оттенков. Для небольших по размеру элементов текста подходят более яркие цвета, боковое содержимое страницы можно выделить при помощи цвета, который на 20–30% светлее цвета текста основного содержимого веб-страницы.

При выборе количества цветов текста предпочтительно ограничиться двумя достаточно контрастными цветами (не считая черного и белого цвета). Черный текст на белом фоне – это классика, достаточно контрастная.

Размер шрифта (свойство `font-size`)

Размер базового шрифта в браузере равен `16px`, а размер заголовков устанавливается пропорционально размеру базового шрифта (`h1` – `2em`, `h2` – `1.5em`, `h3` – `1.17em` и т.д.).

Варьируя размером шрифта, можно придать тексту визуальную значимость и привлечь внимание посетителей к наиболее важным фрагментам текста. Как правило, чем крупнее элемент, тем он важнее.

На размер текста в окне браузера оказывает влияние разрешение монитора пользователя: текст одного и того же размера на мониторе с большим разрешением кажется меньше, чем текст такого же размера на мониторе с более низким разрешением.

Шрифты различных семейств одинакового размера также могут иметь разный фактический размер.

С ростом разрешения экранов и размеров мониторов необходимо пересмотреть привычный размер текста в `12–14px`. Для обычного текста уже повсеместно применяется шрифт размером `14–18px`. Задавая размер шрифта, нужно не забывать про адаптивность, т.е. размер шрифта должен изменяться в зависимости от размера экрана.

Выравнивание текста (свойство `text-align`)

Выводимый по ширине текст хорошо выглядит на печатной странице за счет равномерного распределения слов по строкам. Подобное форматирование веб-страниц средствами CSS невозможно и выравнивание текста по ширине образует большие неприятные промежутки между словами. Поэтому, в веб-типографике обычно используется выравнивание по левой стороне.

Межбуквенный и межсимвольный интервал (свойства `word-spacing` и `letter-spacing`)

Межбуквенный интервал оказывает влияние на читабельность текста. Варьирование плотности текста (уплотненный и разряженный текст) позволит разнообразить темп восприятия, добившись баланса текстовой композиции сайта.

Шрифты с засечками смотрятся более выразительно при уменьшенном значении `letter-spacing`.

Чем меньше шрифт, тем дальше друг от друга должны стоять буквы, чем крупнее – тем плотнее должны быть расположены буквы.

Межстрочный интервал (свойство `line-height`)

Оптимальное значение межстрочного интервала в 1,4–1,6 раза больше размера шрифта.

Также, не делайте больших отступов между заголовками и связанным с ним абзацем.

Длина строки

В строке для сплошного чтения должно помещаться от 30 до 75 знаков (приблизительно 7–10 слов на одной строке). Чем шире строка текста, тем больше должен быть межстрочный интервал. Межстрочное расстояние не должно быть меньше пробела между словами.

Начертание шрифта (свойство font-style)

Можно создавать красивую типографику, управляя начертанием шрифта. Например, курсив придает тексту некую торжественность. Жирное начертание плюс увеличенный размер шрифта позволяет выделить нужное содержимое, но только в том случае, если такой текст будет выделяться на фоне расположенных рядом объектов.

Свойство `font-variant: small-caps;` придает тексту типографскую изысканность, преобразуя текст таким образом, что все буквы отражаются малыми прописными. Данный прием подходит для небольших фрагментов текста.

Структура текста и визуальная иерархия

В структуре веб-страницы выделяют следующие объекты:

- Логотип / название сайта;
- Названия / заголовки;
- Заголовки второго плана;
- Основной текст;
- Навигация;
- Гипертекстовые ссылки;
- Длинные цитаты;
- Боковые панели;
- Подписи / надписи в таблицах, рисунках.

Каждый элемент списка представляет собой фундаментальную часть сайта и это оправдывает необходимость его выделения.

В HTML существует шесть уровней заголовков, начиная с более важного `<h1>` и заканчивая менее значимым `<h6>`. Заголовок `<h1>` должен быть первым в структуре документа, а заголовки низших уровней должны идти за ним и детализировать информацию. Для выделения заголовков можно воспользоваться приемом выделения цветом части заголовка.

Для более легкого усвоения текст нужно разбивать на части и для каждого раздела выбрать заголовок соответствующего уровня. Чем выше уровень заголовка, тем значимее по содержанию должен быть раздел.

В общей структуре текста особое внимание нужно уделять ссылкам, они должны с легкостью выделяться из окружающего их контента.

Визуальная иерархия устанавливает на странице правильную структуру, благодаря чему текст становится прост для восприятия и чтения. В большинстве текстов присутствует деление текста по смыслу. Существует несколько способов, самый простой – разбивка текста на абзацы.

- Капитель;
- Курсив;
- Полужирное/жирное начертание;
- Размер;
- Цвет;
- Смена гарнитуры;

- Изменение положения знаков на полосе текста (верхний и нижний регистр, отступы);
- Выделение знаков при помощи графических элементов – указателей, рамок, иконок и т.п.

Слишком сильное выделение не только чрезмерно акцентирует внимание на каком-либо фрагменте, но и затрудняет чтение.

Шрифт Awesome

Шрифт Awesome – это коллекция масштабируемых векторных иконок. Иконки можно форматировать с помощью CSS-свойств, устанавливать для них цвет, размер, тень и многое другое.

Как установить шрифт Awesome

- Способ 1

Использовать версию файла `font-awesome.css`, размещенную на ресурсе CDNJS. Для этого нужно добавить следующий код в раздел `<head>`:

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/fontawesome.min.css">
```

После того, как шрифт подключен, можно использовать иконки на своем сайте.

- Способ 2

Скачать шрифт с сайта <https://fontawesome.com>.

Можно использовать полную или минимизированную версию файла:

```
<link rel="stylesheet" href="http://ваш_сайт/css/font-awesome.css">
<link rel="stylesheet" href="http://ваш_сайт/css/font-awesome.min.css">
```

Как использовать иконки Font Awesome

Иконки можно добавить на веб-страницу двумя способами: задать соответствующие классы для элементов `<i>` и `` или добавить к нужному элементу с помощью псевдоэлементов `::before`, `::after` и соответствующего значения свойства `content`.


```
<i class="fa fa-home" aria-hidden="true"></i>

li::before {
  content: "\f015"; /* добавляем иконку дом */
  font-family: FontAwesome;
  color: #aaaaaa;
  margin-right: 10px;
}
```

Стандартные иконки

Font Awesome предназначен для использования со строчными элементами. Для того, чтобы добавить иконки, сначала нужно задать класс `fa` для элемента `<i>` или ``.

Чтобы добавить выбранную иконку перед элементом или после него, к элементу добавляется пустой элемент `<i></i>` или ``, которому назначен класс иконки, а также дополнительный класс, расширяющий стилевое оформление.

```
 fa-camera-retro <i class="fa fa-camera-retro" aria-hidden="true"></i>
```

Большие иконки

Чтобы увеличить размер иконки относительно ее контейнера, используются классы `fa-lg` (33% увеличение), `fa-2x`, `fa-3x`, `fa-4x` или `fa-5x`.

 fa-lg

 fa-2x

 fa-3x





 fa-4x

 fa-5x

```
<i class="fa fa-camera-retro fa-lg"></i>
<i class="fa fa-camera-retro fa-2x"></i>
<i class="fa fa-camera-retro fa-3x"></i>
<i class="fa fa-camera-retro fa-4x"></i>
<i class="fa fa-camera-retro fa-5x"></i>
```

Иконки с фиксированной шириной



Используя класс `fa-fw`, можно зафиксировать ширину иконки. Это может пригодиться для оформления навигации или списков на сайте.

 Главная
 Библиотека
 Приложения
 Настройки

```
<ul>
<li><a href="#"><i class="fa fa-home fa-fw"></i> Главная</a></li>
<li><a href="#"><i class="fa fa-book fa-fw"></i> Библиотека</a></li>
<li><a href="#"><i class="fa fa-pencil fa-fw"></i> Приложения</a></li>
<li><a href="#"><i class="fa fa-cog fa-fw"></i> Настройки</a></li>
</ul>
```

Иконки для маркированного списка

Используя классы `fa-ul` и `fa-li`, можно заменить маркеры по умолчанию в маркированном списке `...`.

☒ элемент списка
 элемент списка
 элемент списка

```
<ul class="fa-ul">
<li><i class="fa-li fa fa-check-square"></i>элемент списка</li>
<li><i class="fa-li fa fa-spinner fa-spin"></i>элемент списка</li>
<li><i class="fa-li fa fa-square"></i>элемент списка</li>
</ul>
```

Иконки в рамке и кавычки

Можно воспользоваться классом `fa-border`, чтобы установить рамку для иконки. Классы `pull-right` и `pull-left` добавят кавычки для текста.

```
<p><i class="fa fa-quote-left fa-3x pull-left fa-border"></i>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec purus congue, posuere libero in, porta sapien. In sed elit lectus. Sed lacus elit, semper vitae felis id, sodales congue sapien.</p>
```



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec purus congue, posuere libero in, porta sapien. In sed elit lectus. Sed lacus elit, semper vitae felis id, sodales congue sapien.



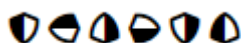
Анимированные иконки

Классы `fa-spin`, `fa-pulse`, `fa-spinner`, `fa-refresh` или `fa-cog`, предоставляют установку вращающихся иконок.

```
<i class="fa fa-spinner fa-spin"></i>
<i class="fa fa-circle-o-notch fa-spin"></i>
<i class="fa fa-refresh fa-spin"></i>
<i class="fa fa-cog fa-spin"></i>
<i class="fa fa-spinner fa-pulse"></i>
```

Трансформированные иконки

Чтобы повернуть иконки или отобразить их зеркально, можно использовать классы `fa-rotate-*` и `fa-flip-*`.



```
<i class="fa fa-shield"></i>
<i class="fa fa-shield fa-rotate-90"></i>
<i class="fa fa-shield fa-rotate-180"></i>
<i class="fa fa-shield fa-rotate-270"></i>
<i class="fa fa-shield fa-flip-horizontal"></i>
<i class="fa fa-shield fa-flip-vertical"></i>
```

Комбинированные иконки

Можно комбинировать иконки, накладывая одну на другую. Класс `fa-stack` используется для родительской иконки, класс `fa-stack-1x` – для стандартного размера и `fa-stack-2x` для увеличенного размера.



```
<span class="fa-stack fa-lg">
  <i class="fa fa-square-o fa-stack-2x"></i>
  <i class="fa fa-twitter fa-stack-1x"></i>
</span>

<span class="fa-stack fa-lg">
  <i class="fa fa-circle fa-stack-2x"></i>
  <i class="fa fa-flag fa-stack-1x fa-inverse"></i>
</span>

<span class="fa-stack fa-lg">
  <i class="fa fa-square fa-stack-2x"></i>
  <i class="fa fa-terminal fa-stack-1x fa-inverse"></i>
</span>

<span class="fa-stack fa-lg">
  <i class="fa fa-camera fa-stack-1x"></i>
  <i class="fa fa-ban fa-stack-2x text-danger"></i>
</span>
```

Favicon сайта

Фавикон (Favicon) – это иконка, которая отображается рядом с названием страницы во вкладке браузера. Также данный значок отображается в сниппете сайта на странице результатов поиска в некоторых поисковых системах, например Яндекс.

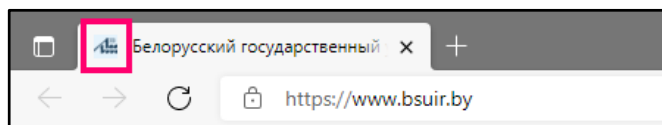
Фавикон повышает узнаваемость сайта во вкладке браузера, повышает кликабельность в результатах выдачи Яндекса, также он может способствовать уровню запоминаемости ресурса. В сети Интернет достаточно сервисов, предназначенных для генерации фавикон.

Как установить фавикон

Для установки фавикона нужно иметь подходящее изображение. Оно должно быть размером 16×16 пикселей и иметь формат `.ico`. Фавикон можно создать самостоятельно или выбрать из существующих на специальных [сайтах](#).

Данную иконку необходимо загрузить на сайт. После чего указать на нее ссылку между тегами `<head>...</head>`. В атрибуте `href` указывается адрес соответствующего файла.

После выполнения данных действий, если все сделано правильно, фавикон должен появиться во вкладке браузера на сайте.



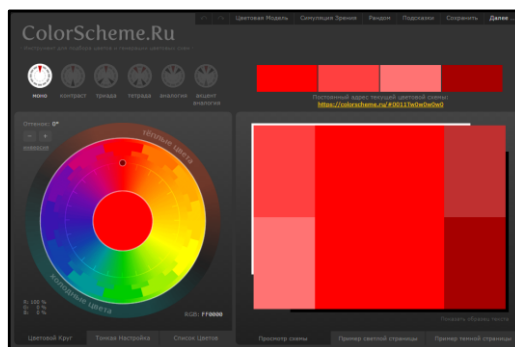
Сервисы подбора цвета для веб-дизайна

Цвета для веб-дизайна играют огромную роль. Чтобы грамотно подобрать цветовую схему для сайта, существуют специальные сервисы.

colorscheme.ru

Удобный инструмент для подбора цветов. У него много дополнительных возможностей. Например, можно посмотреть пример светлой и темной страницы с выбранными цветами.

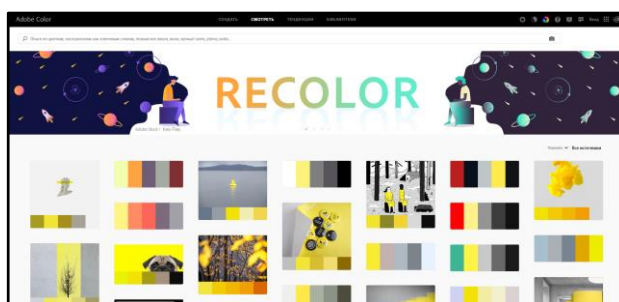
Есть возможность оценить, как будут видеть цветовую гамму люди с дальтонизмом и другими особенностями зрения. Можно выбрать безопасные веб-цвета.



color.adobe.com

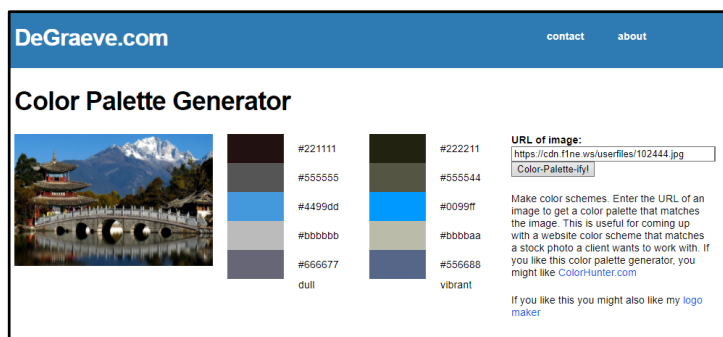
Тут можно не только создавать цветовые схемы, но и посмотреть и использовать схемы, созданные другими людьми.

Для этого надо нажать кнопку в верхнем меню «Смотреть».



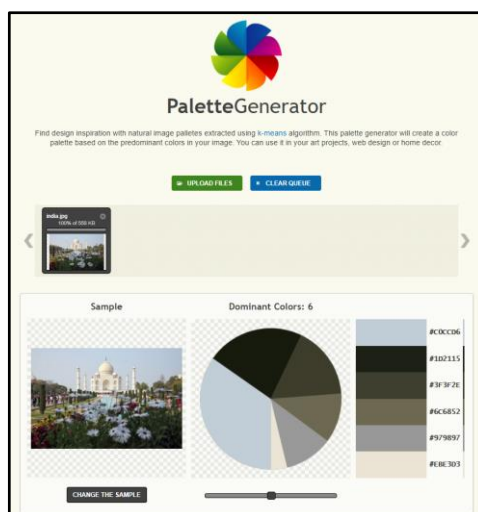
[Color Palette Generator](https://colorpalettegenerator.com)

Сайт генерирует палитру из выбранного изображения. Необходимо указать ссылку на изображение.



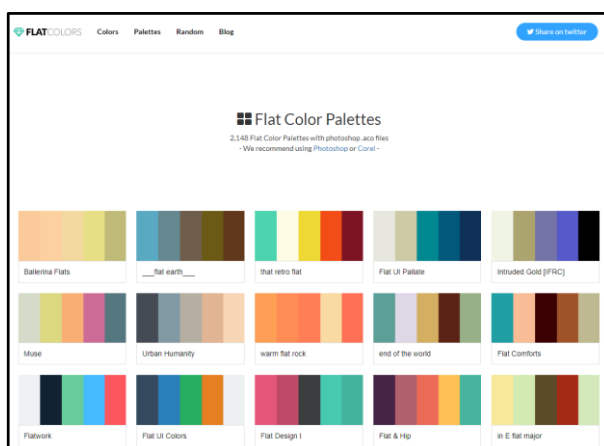
palettegenerator.com

Сайт генерирует палитру из выбранного изображения. Необходимо загрузить изображение со своего компьютера.



flatcolors.net

Как видно из названия, здесь можно подобрать цвета для плоского дизайна.



Анализ ключевых слов

Анализ ключевых слов – очень важная процедура для продвижения своего ресурса или своих услуги в Интернете, так как от выбора слов в значительной степени зависит и эффективность интернет-рекламы, и результативность оптимизации веб-ресурсов.

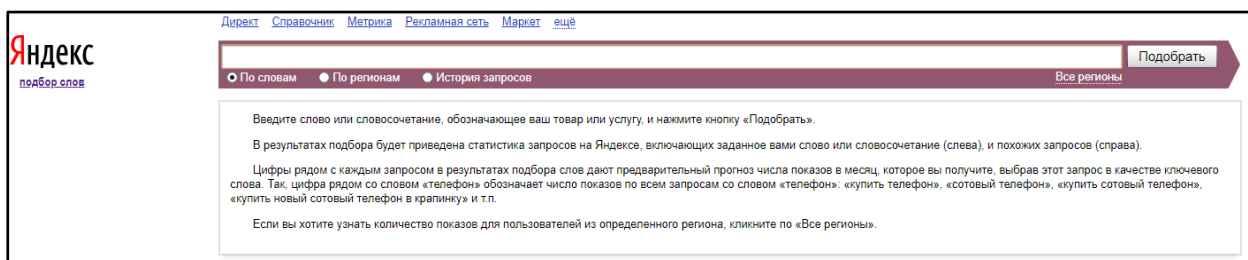
Существует несколько путей подбора ключевых слов.

- Во-первых, можно сформулировать ключевые запросы самостоятельно, представив, что именно и с какими формулировками будут искать пользователи. Но подобрать таким образом большой массив ключевых слов невозможно, как и быть уверенным в том, что они окажутся эффективными.
- Во-вторых, слова можно подобрать путем анализа ключевых слов сайта. Анализ может осуществляться либо визуально (просматривая контент сайта и выбирая наиболее часто встречающиеся слова), либо при помощи специальных инструментов, которые «просканируют» содержимое сайта и выдадут список релевантных для поисковой системы запросов.
- В-третьих, ключевые слова можно подобрать при помощи инструмента Яндекс.Вордстат (Yandex Wordstat).

Яндекс.Вордстат предоставляет обширную статистику поисковых запросов, с возможностью отслеживать результаты по регионам, по месяцам или по языку. В **Яндекс.Вордстат** помимо информации о количестве запросов, содержащих интересующие вас слова, имеется функция «подсказки». Под заголовком «Ищут также» отображаются похожие словосочетания, синонимы и близкие по темам запросы.

Яндекс.Вордстат

Главная страница сервиса [Яндекс.Вордстат](http://yandex.ru/wordstat) состоит из поисковой строки с небольшим набором настроек и кнопку «Подобрать».



Здесь же отображается краткая справка по работе с сервисом и в общих чертах описывается логика отображаемых данных.

Первый запрос в левой колонке содержит в себе все последующие. То есть расширенные ключевые слова под основным – это не дополнительные запросы, а вложенные. Это значит, что «семиструнная электрогитара» с 466 показами включает в себя «семиструнная гитара купить» с 154 показами из вышеназванных 466.

Не нужно складывать «ключи» друг с другом, так как получится некорректный расчет. Сами показатели условны. Это не точная статистика, а лишь прогноз на количество показов по выбранному ключевому слову.

[Главная](#) [Справочник](#) [Метрики](#) [Расширенная сеть](#) [Макеты](#) еще

смартфон iPhone

По словам По регионам История запросов Все регионы Последнее обновление: 21.09.2022

Выбор: Десктопы Мобильные Только телефоны Только планшеты

Что искали со словом «смартфон iPhone» — 48 355 показов в месяц

Систематика по словам	Показы в месяц
смартфон iphone	46 355
смартфон apple iphone	39 177
смартфон iphone 12	13 794
смартфон iphone 11	13 063
смартфон apple iphone 12	12 622
смартфон apple iphone 11	12 064
смартфон apple iphone 128gb	10 451
купить смартфон iphone	8 073
смартфон apple iphone 64gb	7 794
купить смартфон apple iphone	7 419
смартфон iphone 12 pro	6 155
смартфон apple iphone 12 pro	5 685
смартфон iphone 12 128gb	4 810
смартфон apple iphone 12 128gb	4 662
смартфон apple iphone max	4 497
смартфон apple iphone 256gb	4 361
смартфон iphone 11 128gb	3 889
смартфон apple iphone 11 128gb	3 743
смартфон iphone se	3 674
смартфон iphone 11 64gb	3 410
смартфон iphone xr	3 376
смартфон apple iphone 11 64gb	3 317
смартфон apple iphone se	3 316
смартфон apple iphone xr	3 172
смартфон iphone 11 pro	3 130
смартфон iphone 12 pro max	3 032
купить смартфон iphone 11	2 904

Запросы, похожие на «смартфон iPhone»

Систематика по словам	Показы в месяц
купить сенсорный телефон	37 431
купить сенсорный телефон	37 431
айфон не работает	15 942
айфон rsc s	2 968
купить айфон 2	3 601
айфон se(2020)	2 838
ростер айфон что это	2 471
новый айфона	14 707
хороший мобильный телефон	16 599
айфон эс э	1 590
айфон 5s	102 266
mobile phone	20 277
айфон 16	7 229
айфон11	492
мобайл телефон	35 194
telfon	54 639
a1586 какой айфон	156
купить выносной телевизор	15 103
переломка ланчех тс айфона тиа айфон	14 499
интерактив сенсорных телефонов	7 140

Запросы в Яндекс.Вордстате можно дополнительно настроить с помощью специальных символов. Их ставят перед словом или фразой для выполнения какого-то условия. Соответственно, у каждого символа есть свое предназначение и выполняемое условие.

- **+** – этот символ обязывает систему Wordstat учитывать слово при поиске. По умолчанию некоторые слова русского языка игнорируются. Это касается предлогов и следующих союзов: в, на, от, для, как, из, и, от.
- **-** – этот символ позволяет более адекватно оценивать перспективность выбранной ниши за счет исключения из ключевой фразы всех лишних составляющих, способных повлиять на результат.
- **!** – этот символ запрещает Яндекс.Вордстату корректировать словоформу. То есть поиск будет учитывать исключительно ключевые слова с выбранным окончанием, числом, родом и т.п. Его рекомендуют использовать оптовикам, чтобы точно учитывать запросы на покупку большого количества товаров.
- **«»** – кавычки выводят статистику по выделенному слову или фразе в отдельное окно, чтобы можно было оценить количество запросов без вложенных «ключей». То есть увидеть запрос «купить электрогитару» без «купить электрогитару Ibanez» и других вариаций, которые учитываются в первом значении левого столбца Яндекс.Вордстата.
- **[]** – этим символом можно зафиксировать используемый в запросе порядок слов. Этим пользуются туристические фирмы и авиакомпании, чтобы предлагать клиентам билеты в точных (а не в похожих) направлениях.
- **(|)** – в скобки можно занести 2 или больше похожих слов. К примеру, если вы продаете товары сразу из двух стран, можно занести их в скобки и посмотреть статистику сразу по двум категориям запросов.

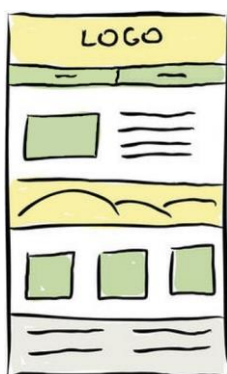
Задание для контрольной работы

Выполненный проект необходимо разместить в репозитории [GitHub](#) и выслать ссылку преподавателю.

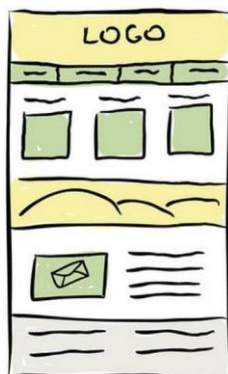
Условие: Необходимо сверстать посадочную страницу сайта или лендинг используя фреймворк Bootstrap.

1. Выбрать прототип согласно своему варианту (номер по списку в группе).
2. Осуществить выбор цветовой палитры для веб-дизайна используя сервисы подбора цветовой схемы сайта.
3. Подобрать иконочный шрифт используя Font Awesome.
4. Заполнить страницу своим уникальным контентом с общим смысловым содержанием.
5. Разработать и установить фавикон.
6. Подобрать и заполнить метатеги (description, keywords), используя Яндекс.Вордстат.
7. Выполнить адаптивную верстку под разные экраны устройств.
8. Осуществить проверку файлов на валидность.

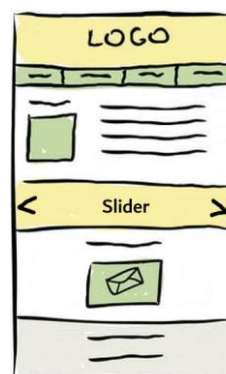
Вариант 1



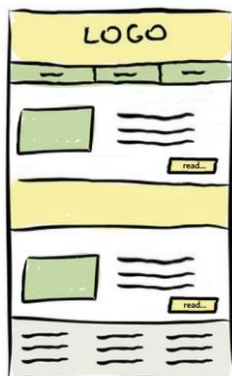
Вариант 2



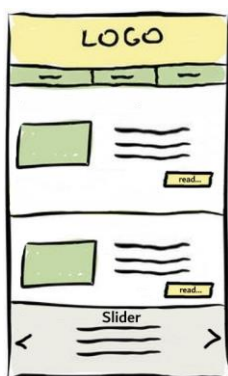
Вариант 3



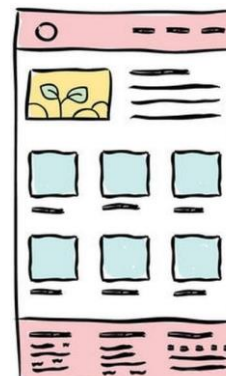
Вариант 4



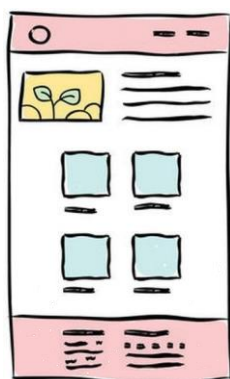
Вариант 5



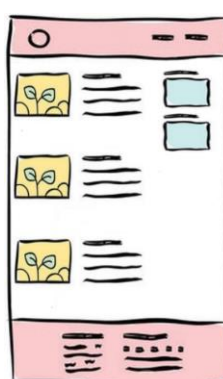
Вариант 6



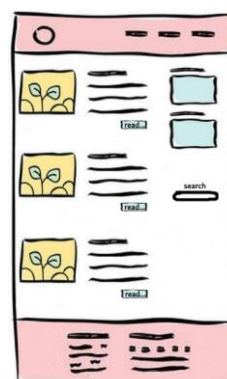
Вариант 7



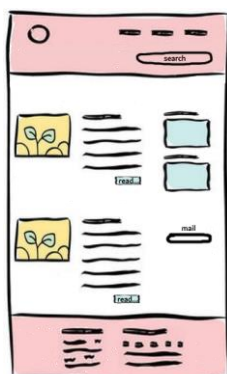
Вариант 8



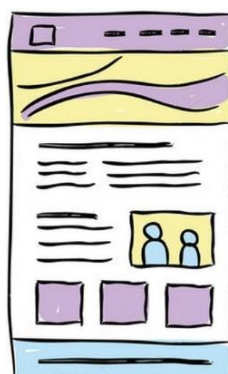
Вариант 9



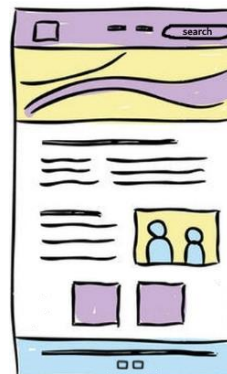
Вариант 10



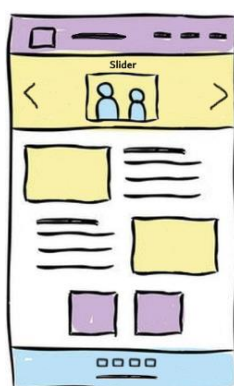
Вариант 11



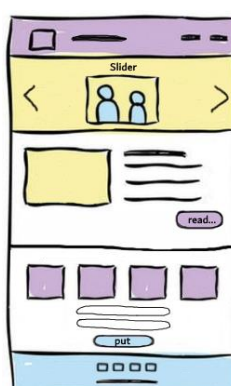
Вариант 12



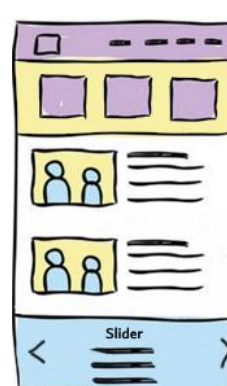
Вариант 13



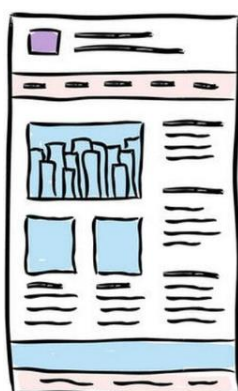
Вариант 14



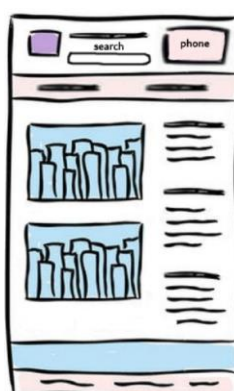
Вариант 15



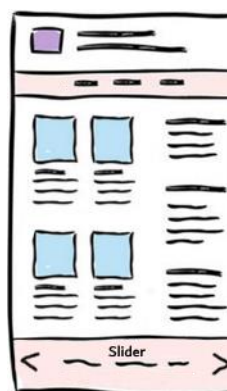
Вариант 16



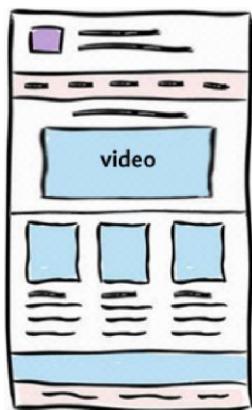
Вариант 17



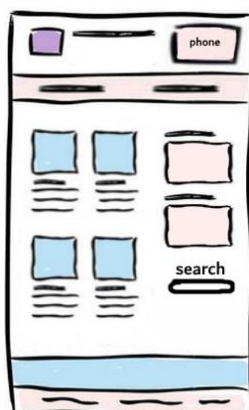
Вариант 18



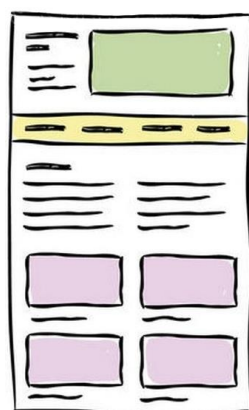
Вариант 19



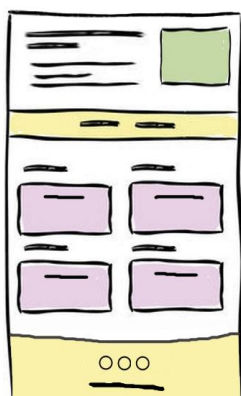
Вариант 20



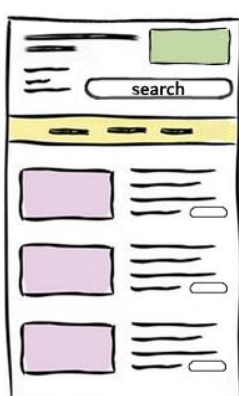
Вариант 21



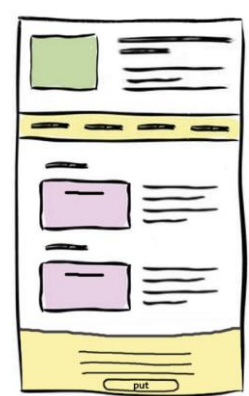
Вариант 22



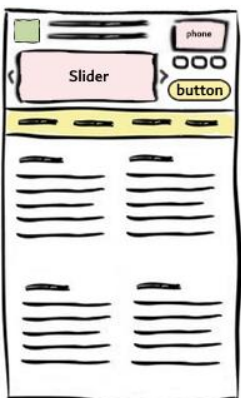
Вариант 23



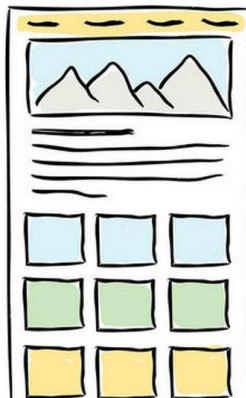
Вариант 24



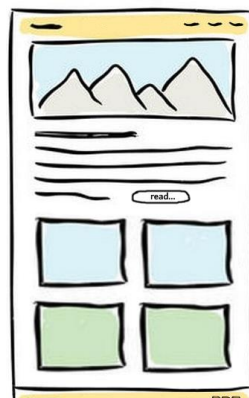
Вариант 25



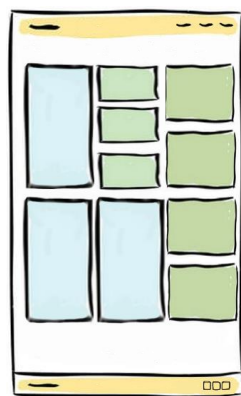
Вариант 26



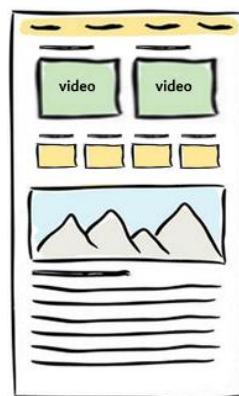
Вариант 27



Вариант 28



Вариант 29



Вариант 30

