

Введение

Итоговая оценка разбита на две составляющие: базовая и дополнительная.

Базовая оценка предполагает перепроверку некоторых важных критериев допуска. При условии, что они выполнены, выставляется базовая оценка и наставник переходит к дополнительной оценке.

Дополнительная оценка выставляется по дополнительным критериям. Дополнительные критерии оценивают проект с точки зрения шлифовки его качества и оптимизации, и выстроены по принципу перфекционизма. Мы не требуем сделать проект идеально, но набрать максимальное количество баллов возможно.

Примечание: «Барбершоп» нельзя сдавать в качестве личного макета.

Базовые критерии

Разметка

Б1. Выполнена HTML-разметка всех страниц и всех элементов на страницах.

Критерий требует, чтобы все страницы личного макета и все элементы на этих страницах были свёрстаны.

Б2. К страницам подключён один стилевой файл (с учётом `normalize.css` к каждой странице могут быть подключены два файла).

Критерий говорит о том, что все стили страниц должны находиться в едином файле. Исключением является файл `normalize.css`. Он может быть подключён отдельно, перед основным стилевым файлом.

Верно: к документу подключаются два файла — `normalize.css` и основной стилевой файл.

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

Неверно: к документу подключается несколько стилевых файлов. Один из них содержит общие стили, второй стили конкретной страницы.

```
<head>
  <link rel="stylesheet" href="css/normalize.css">
  <link rel="stylesheet" href="css/all.css">
  <link rel="stylesheet" href="css/main-page.css">
</head>
```

Б3. Стилевой файл подключён внутри `<head>`.

Подключение стилевых файлов должно производиться внутри `<head>`.

Верно: стилевой файл подключён в секции `<head>`.

```
<!DOCTYPE html>
<html>
  <head>
    ...
    <link rel="stylesheet" href="css/style.css">
  </head>
  <body>
    ...
  </body>
</html>
```

Неверно: стилевой файл подключён в секции `<body>`

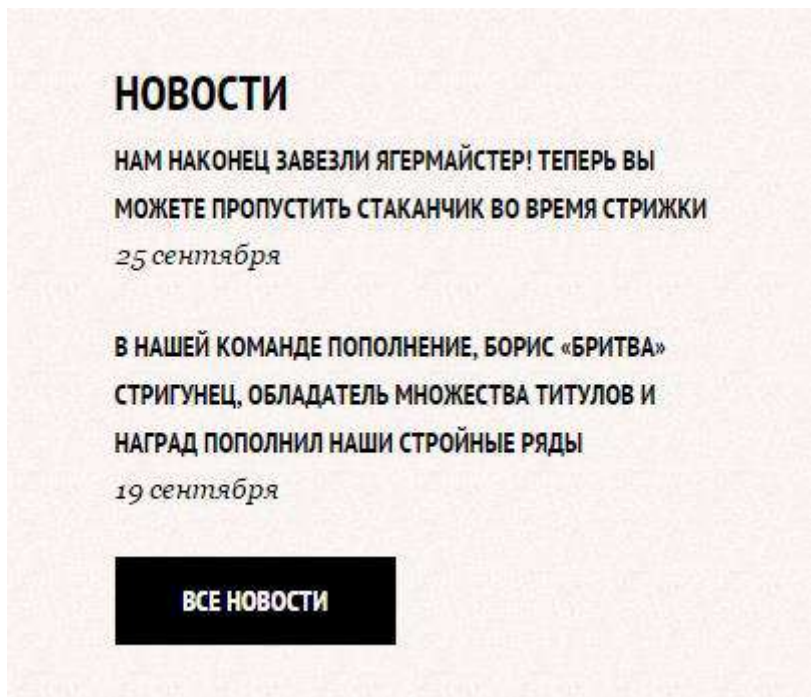
```
<!DOCTYPE html>
<html>
  <head>
    ...
  </head>
  <body>
    <link rel="stylesheet" href="css/style.css">
    ...
  </body>
</html>
```

Б4. Грубые ошибки в разметке отсутствуют.

Грубые ошибки:

1. Ссылки сделаны не тегом `<a>`, а другими тегами.

Для создания определённых элементов на странице необходимо использовать соответствующие теги. Для примера рассмотрим блок с новостями на главной странице макета Барбершоп. В этом случае, ссылка на все новости хоть и выглядит как кнопка, но на самом деле является ссылкой.



2. Использование строчных элементов для создания крупных (сеточных) блоков. Использование строчных тегов для построения сетки недопустимо. Для построения сетки должны использоваться блочные элементы.

Верно: сетка построена с использованием блочных элементов.

```
<div class="features">
  <div class="features-item">
    ...
  </div>
  <div class="features-item">
    ...
  </div>
  <div class="features-item">
    ...
  </div>
</div>
```

Неверно: сетка построена с использованием строчных элементов, которые впоследствии будут изменены на блочные средствами CSS.

```
<div class="features">
  <span class="features-item">
    ...
  </span>
  <span class="features-item">
    ...
  </span>
  <span class="features-item">
    ...
  </span>
</div>
```

3. Абзацы сделаны не тегами `<p>`, а `

`.

Если в макете текст разделен на несколько абзацев, то в разметке каждый такой абзац должен быть обернут в соответствующий тег. Использовать для разбиения на абзацы `
` недопустимо.

Верно: каждый абзац в своём теге `<p>`

```
<p>Наша парикмахерская занимается исключительно мужскими стрижками.</p>
<p>Стрижка каждого клиента для нас - это уникальная и продуманная до мелочей работа.</p>
<p>Наша мастерская расположена в центре города, поэтому сделать стрижку можно в любое время.</p>
```

Неверно: для разбиения на абзацы используется `
`.

```
Наша парикмахерская занимается исключительно мужскими стрижками.
<br><br>
Стрижка каждого клиента для нас - это уникальная и продуманная до мелочей работа.
<br><br>
Наша мастерская расположена в центре города, поэтому сделать стрижку можно в любое время.
Не путайте эту ситуацию с использованием <br> внутри абзацев для разбиения текста на строки, например, внутри адресов.
```

Верно: весь адрес внутри `<p>` разбит на строки с помощью `
`.

```
<p>191186,<br>
Санкт-Петербург,<br>
Большая Конюшенная, 19/8
</p>
```

Также разрешено использовать `
` внутри абзацев, если нет возможности сделать переносы как в макете путём ограничения ширины контейнера с текстом.

В этом блоке текст под заголовком имеет явные переносы, которые нельзя сделать простым ограничением ширины.

[Здесь](#) можно посмотреть, в каких местах допустимо использование тега `br` в каждом из проектов.

ДОСТАВКА

Мы с удовольствием доставим ваш товар прямо
к вашему подъезду совершенно бесплатно!

Ведь мы неплохо заработаем,
поднимая его на ваш этаж!

4. Теги `<input>` использованы без тега `<form>`.

Не считается ошибкой использование тега `<input>` при вёрстке слайдера без тега `<form>`.

Негрубые ошибки:

5. Отсутствие семантических тегов `<header>`, `<footer>`, `<section>` и других;

Критерий говорит о том, что отсутствие семантических тегов не является ошибкой. Оба приведённых ниже примера являются допустимыми.

Разметка выполнена с использованием семантических тегов.

```
<header class="main-header">
  <div class="container">
    <nav class="main-navigation">
      ...
    </nav>
  </div>
</header>
```

Разметка выполнена без использования семантических тегов.

```
<div class="main-header">
  <div class="container">
    <div class="main-navigation">
      ...
    </div>
  </div>
</div>
```

```
</div>
</div>
</div>
```

Б5. Документ проходит проверку на валидность <http://validator.w3.org/nu/>.

Стилизация

Б6. Вся собственная стилизация выполнена в одном стилевом файле.

Все стили всех страниц должны находиться в едином файле. Нет необходимости разносить различающиеся файлы разных страниц в отдельные файлы и подключать их по необходимости.

Исключением могут являться файлы с подключением шрифтов, если они грузятся со стороннего сервиса, например, Google Fonts или WebFont.ru. В таком случае эти файлы могут идти отдельно и не являться частью основного файла стилей.

Б7. Раскладка блоков на странице сделана с помощью флексбоксов.

Использование тега `<table>` и блоков с абсолютным позиционированием недопустимо. `<table>` является устаревшим методом построения сеток, а элементы с абсолютным позиционированием вырываются из общего потока. Допускается использование этих методов для создания декоративных элементов и модальных окон.

Использовать табличную раскладку блоков с применением свойств `display: table`, `display: table-row`, `display: table-cell` и так далее не запрещено.

Верно: основная сетка сделана с использованием `flex` элементов.

```
<div class="features">
  <div class="features-item">
    ...
  </div>
  <div class="features-item">
    ...
  </div>
  <div class="features-item">
    ...
  </div>
</div>
.features {
  display: flex;
  justify-content: space-between;
}
.features-item {
```

```
width: 300px;
}
```

Неверно: основная сетка сделана с использованием таблиц.

```
<table class="features">
  <tr>
    <td class="features-item">
      ...
    </td>
    <td class="features-item">
      ...
    </td>
    <td class="features-item">
      ...
    </td>
  </tr>
</table>
```

Неверно: основная сетка сделана с использованием абсолютного позиционирования.

```
<div class="features">
  <div class="features-item features-1">
    ...
  </div>
  <div class="features-item features-2">
    ...
  </div>
  <div class="features-item features-3">
    ...
  </div>
</div>

.features {
  position: relative;
}
.features-item {
  position: absolute;
  top: 0;

  width: 300px;
}
.features-1 {
  left: 0;
}
.features-item {
  left: 300px;
}
.features-item {
  right: 0;
}
```

Б8. В CSS отсутствует !important.

Допускается использование `!important` при обосновании его необходимости в комментарии.

В большинстве случаев, необходимость использования `!important` в CSS возникает из-за неправильного обращения к элементам страницы через каскад. Однако, в редких случаях, без использования `!important` не обойтись. Если без использования `!important` не обойтись или использование осознанно, то это не ошибка.

Подробнее в статье: <http://meiert.com/en/blog/20150310/important/>.

Неверно: в коде CSS используется `!important`.

```
.text-page p {  
  font-size: 14px !important;  
  line-height: 20px !important;  
}
```

Б9. Подключены правильные шрифты, их размеры, высота строк, цвет и толщина равны соответствующим параметрам в макетах и техническом задании.

Необходимо, чтобы в личном проекте использовались именно те шрифты, которые указаны в макете. При этом их размеры, высота строк, цвета, и жирность должны быть равны соответствующим параметрам в макетах и техническом задании.

При подключении шрифтов с Google Fonts, если в проекте используются несколько разных шрифтов, их нужно подключать с помощью одной ссылки.

Б10. Указаны альтернативные варианты шрифта и тип семейства в конце перечисления `font-family`.

Альтернативный веб-безопасный шрифт и тип семейства необходимо указывать для того, чтобы в случае отсутствия основного шрифта изменения внешнего вида шрифтов на странице были минимальны.

Порядок шрифтов следующий:

1. основной шрифт;
2. веб-безопасный;
3. тип шрифта.

Список веб-безопасных шрифтов можно посмотреть здесь: <http://www.cssfontstack.com/>.

Важно: альтернативный веб-безопасный шрифт должен быть такого же типа, что и основной, но подбирать максимально похожий шрифт не нужно.

Верно: указан альтернативный веб-безопасный шрифт и его тип семейства.

```
body {  
  font-family: "PT Sans Narrow", "Arial", sans-serif;  
}  
  
/* Кому-то нравится Arial, кому-то Tahoma */  
body {  
  font-family: "PT Sans Narrow", "Tahoma", sans-serif;  
}
```

Неверно: указан только основной шрифт.

```
body {  
  font-family: "PT Sans Narrow";  
}
```

Неверно: указан только основной шрифт и тип семейства, альтернативный веб-безопасный шрифт отсутствует.

```
body {  
  font-family: "PT Sans Narrow", sans-serif;  
}
```

Неверно: Times New Roman — шрифт с засечками, а основной шрифт — без засечек.

```
body {  
  font-family: "PT Sans Narrow", "Times New Roman", sans-serif;  
}
```

Б11. При наполнении контентом (как в макете) элементы каждой страницы соответствуют макету.

Допускаются:

Различия в 5 пикселей по высоте (при расстояниях более 30 пикселей) и 2 пикселя по ширине;

Различия в отображении шрифтов, связанные со сглаживанием на различных платформах.

Рекомендуем проверять с помощью инструмента [PerfectPixel](#) (работает в Chrome, Opera, Firefox).

В случае резиновой вёрстки основной шириной является большая, меньшая ширина является справочной. Соответствие проверяется на большей ширине.

Тестирование

Б12. Вёрстка идентично отображается в последних версиях браузеров Chrome, Opera, Firefox, Safari, Edge, а также в Internet Explorer 11+.

При проверке этого критерия, необходимо смотреть на размеры и расположение блоков, внешнее сходство с макетом. Проверять работу анимации, если такая имеется. Допускаются небольшие отличия в отображениях шрифтов.

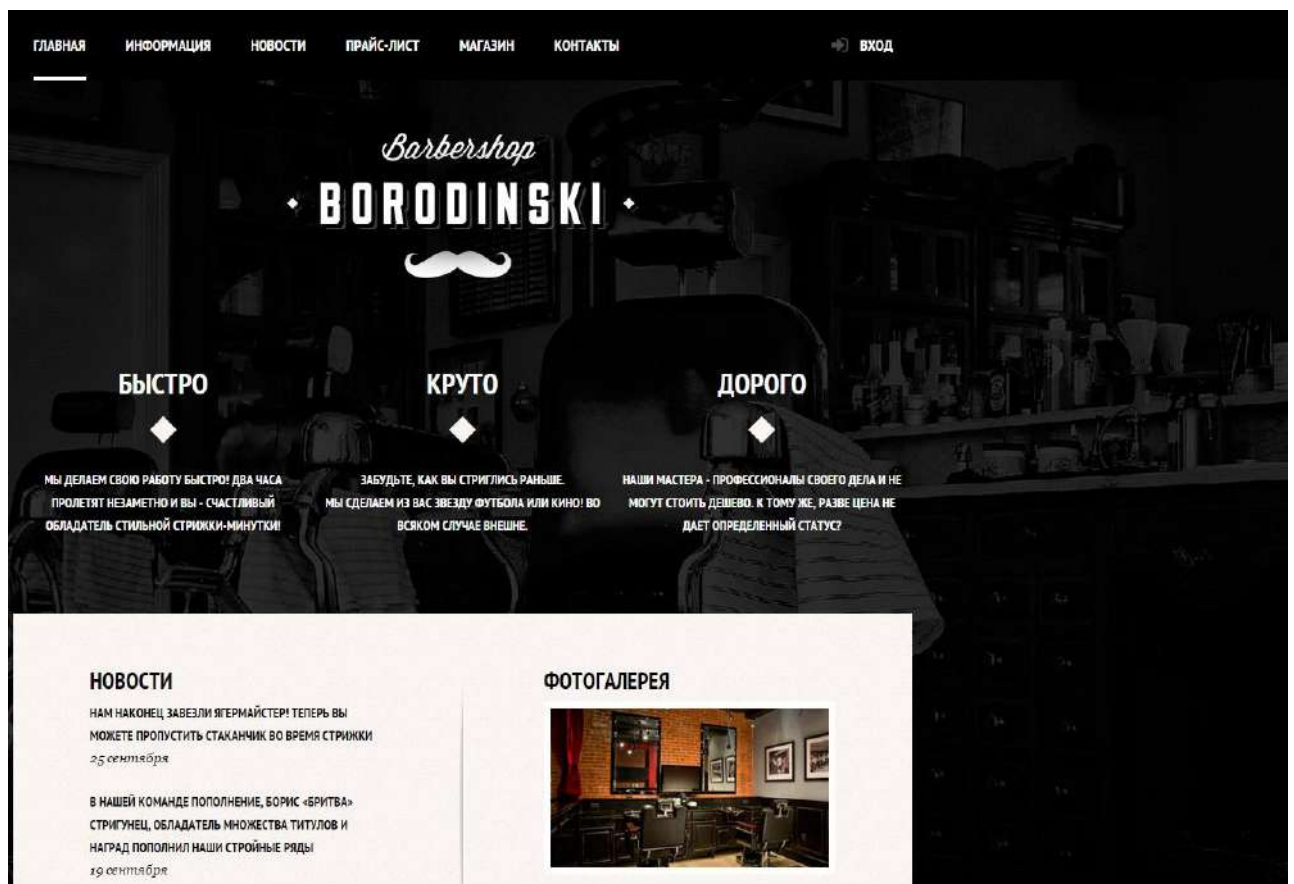
Б13. Сайт нормально смотрится на минимальном для макета ширине.

- Нёрдс — 1160px (1200px с учётом padding);
- Техномарт — 940px (960px с учётом padding);
- Седона — 768px (для фиксированной нужно использовать версию 1200px);
- Глейси — 900px (для фиксированной нужно использовать версию 1200px);
- Девайс — 1160px (1200px с учётом padding).

padding по краям контейнера указывается для того, чтобы контент не прижимался к краям экрана на минимальном разрешении. При большем размере экрана макет должен оставаться по центру и не иметь горизонтального скролла. На разрешениях экрана меньше ширины контейнера вёрстка не должна менять свою структуру.

Критерий рассматривает поведение страницы при размере экрана большим или меньшим, чем ширина основного контейнера страницы.

Неверно: при большем размере экрана страница остаётся слева, а не выравнивается по центру.



Неверно: при меньшей ширине экрана объекты страницы начинают менять свой размер и ломать сетку.

БЫСТРО



КРУТО



ДОРОГО



МЫ ДЕЛАЕМ СВОЮ РАБОТУ
БЫСТРО! ДВА ЧАСА ПРОЛЕТЯТ
НЕЗАМЕТНО И ВЫ - СЧАСТЛИВЫЙ
ОБЛАДАТЕЛЬ СТИЛЬНОЙ
СТРИЖКИ-МИНУТКИ!

ЗАБУДЬТЕ, КАК ВЫ СТРИГЛИСЬ
РАНЬШЕ.
МЫ СДЕЛАЕМ ИЗ ВАС ЗВЕЗДУ
ФУТБОЛА ИЛИ КИНО! ВО ВСЯКОМ
СЛУЧАЕ ВНЕШНЕ.

НАШИ МАСТЕРА -
ПРОФЕССИОНАЛЫ СВОЕГО ДЕЛА И
НЕ МОГУТ СТОИТЬ ДЕШЕВО. К
ТОМУ ЖЕ, РАЗВЕ ЦЕНА НЕ ДАЕТ
ОПРЕДЕЛЕННЫЙ СТАТУС?

НОВОСТИ

НАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ ВЫ
МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ СТРИЖКИ

25 сентября

В НАШЕЙ КОМАНДЕ ПОПОЛНЕНИЕ, БОРИС «БРИТВА»
СТРИГУНЕЦ, ОБЛАДАТЕЛЬ МНОЖЕСТВА ТИТУЛОВ И
НАГРАД ПОПОЛНИЛ НАШИ СТРОЙНЫЕ РЯДЫ

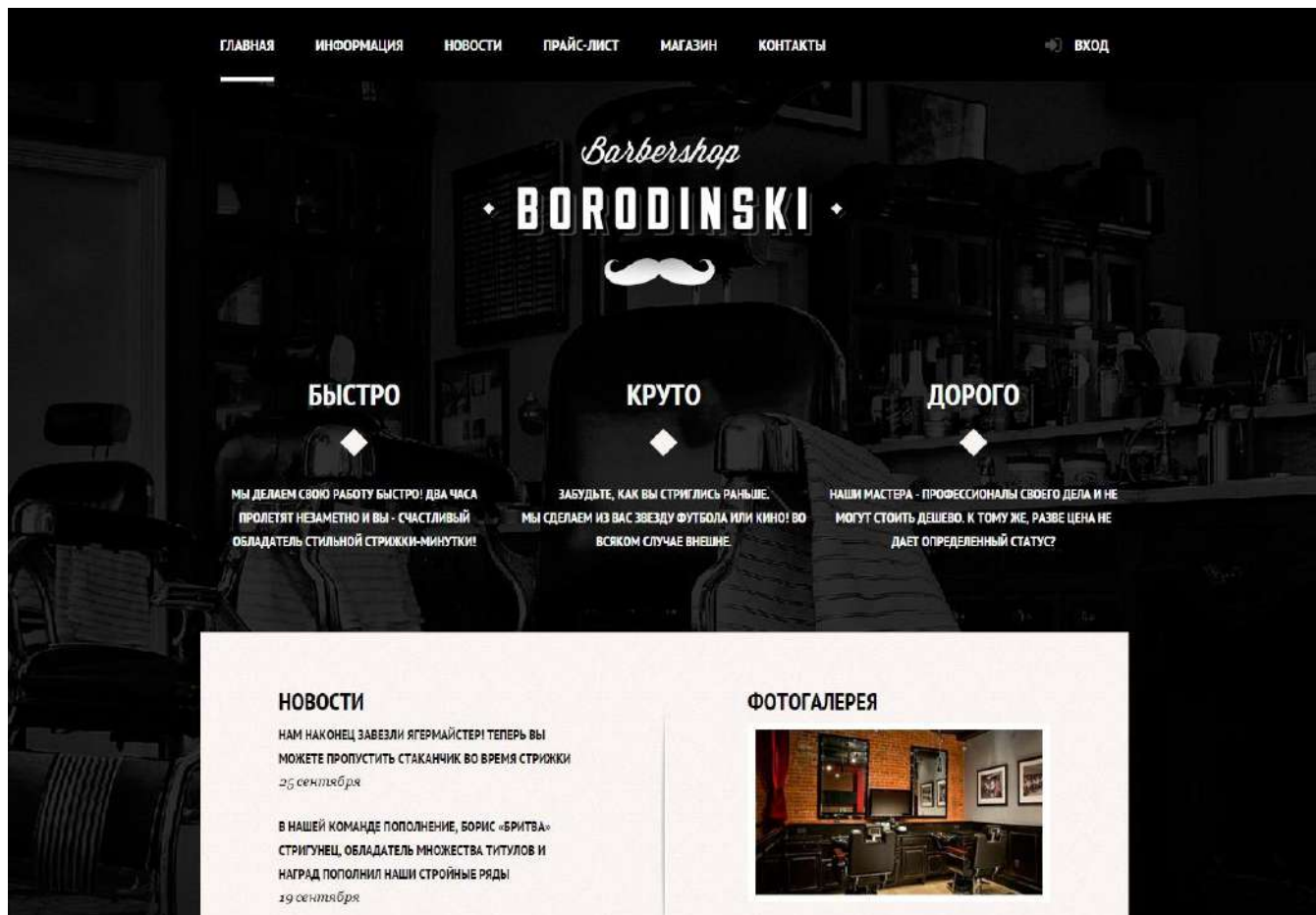
19 сентября

[ВСЕ НОВОСТИ](#)

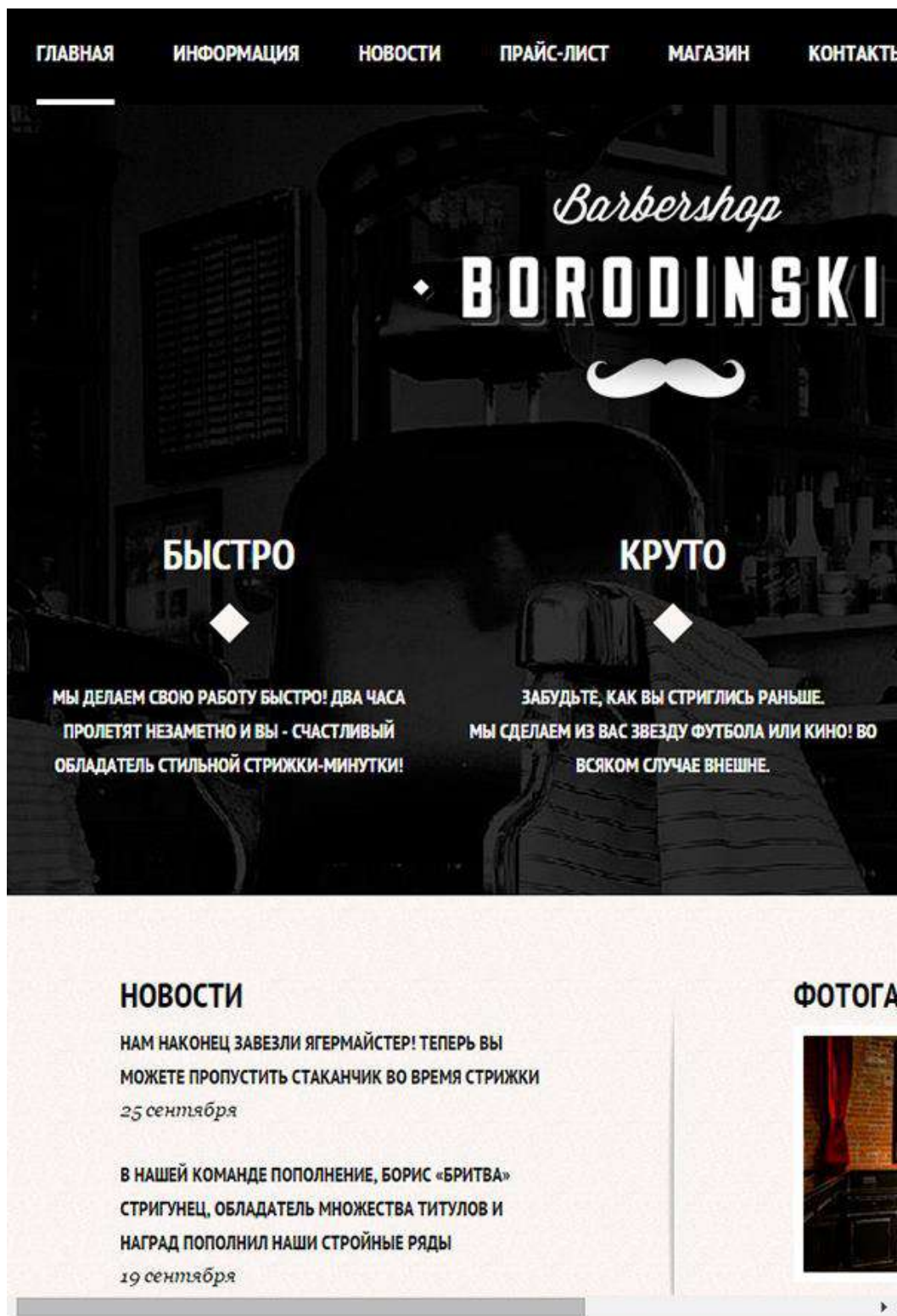
ФОТОГАЛЕРЕЯ



Верно: при большем размере экрана страница остаётся по центру экрана и имеет фиксированную ширину.



Верно: при меньшей ширине экрана страница не меняет свой размер, а внизу окна браузера появляется горизонтальный скролл для её прокрутки.

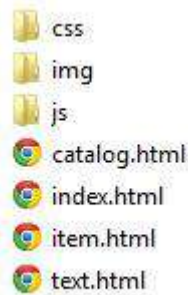


Разное

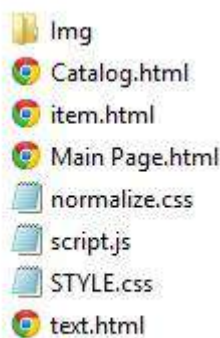
Б14. В корне документа имеются папки **css**, **img**, **js** или аналогичные. Главная страница имеет название **index.html**. В названиях и расширениях файлов нет заглавных букв и пробелов, использованы только латинские символы.

Критерий рассматривает файловую организацию проекта. Все файлы должны лежать в соответствующих папках и иметь читаемые и единообразные написанные названия.

Верно: все файлы находятся в своих папках. Названия этих папок отражает тип файлов, которые в ней содержатся. Главная страница имеет название `index.html`. Все названия написаны маленькими буквами и не имеют пробелов.



Неверно: файлы смешаны в одну кучу. В именах используются заглавные буквы и пробелы. Главная страница имеет неверное название.



Б15. Единообразное написание и форматирование кода в HTML, CSS и JavaScript.

Критерий рассматривает единообразие в написании кода:

- используется один тип кавычек: в одном языке (HTML, CSS или JS) должны использоваться только кавычки определённого типа (двойные или одинарные). Однако между языками тип кавычек может отличаться;
- если размер отступа в два пробела, то таким он должен быть везде;
- если для отступов используются табы или пробелы, то для всех отступов должны быть либо табы, либо пробелы;
- названия классов должны быть оформлены единообразно.

Важно: этот критерий учитывает именно единообразие, а не стиль написания и форматирования кода.

Верно: единообразное написание кода разметки. Везде используется одинаковое количество пробелов, одинаковые скобки, единообразное написание названий классов.

```
<header class="main-header">
  <div class="container clearfix">
    <nav class="main-navigation">
      <ul>
```

```

        <li class="active">
            <a href="index.html">Главная</a>
        </li>
        <li>
            <a href="#">Информация</a>
        </li>
        <li>
            <a href="#">Новости</a>
        </li>
        <li>
            <a href="text.html">Прайс-лист</a>
        </li>
        <li>
            <a href="catalog.html">Магазин</a>
        </li>
        <li>
            <a href="#">Контакты</a>
        </li>
    </ul>
</nav>
<div class="log-in">
    <a href="#" class="log-in-button">Вход</a>
</div>
</div>
</header>

```

Верно: единообразное написание кода стилей. Используется одинаковое количество пробелов, одинаковое положение скобок, единообразная последовательность написания свойств.

```

.btn {
    width: 100%;
    padding: 10px 15px;

    font-size: 14px;
    line-height: 24px;
    text-align: center;
    color: #ffffff;

    cursor: pointer;
}

.main-navigation a {
    display: block;
    padding: 30px 0 23px;

    font-size: 16px;
    line-height: 20px;
    text-decoration: none;
    color: #ffffff;

    border-bottom: 4px solid transparent;
}

```

Неверно: используются разные кавычки и разное количество пробелов в табуляции. В именах классов нет единообразия.

```
<header class="main-header">
```

```

<div class="container clearfix">
  <nav class='mainNavigation'>
    <ul><li class="active">
      <a href="index.html">Главная</a>
    </li>
    <li>
      <a href="#">Информация</a>
    </li>
    <li><a href=' '>Новости</a></li>
    <li>
      <a href="text.html">Прайс-лист</a>
    </li>
    <li>
      <a href="catalog.html">Магазин</a>
    </li>
    <li>
      <a href="#">Контакты</a>
    </li>
    </ul>
  </nav>
  <div class='logIn'>
    <a href="#" class="log-in-button">Вход</a>
  </div>
</div>
</header>

```

Неверно: используется разное количество пробелов в табуляции, скобки в разных местах.

```

.btn{
width: 100%;
  font-size: 14px;
line-height: 24px;
padding: 10px 15px;

cursor: pointer;
text-align: center;
color: #fff;
border: none;
  outline: none;
background: #000000;
  text-decoration: none;
  text-transform: uppercase}

.main-navigation a
{
  display: block;
padding: 30px 0 23px;
  font-size: 16px;
line-height: 20px;
text-decoration: none;

color: #ffffff;
border-bottom: 4px solid transparent;
}

```


Б16. Выбран подходящий формат изображений.

Например:

- JPEG для фотографий;
- SVG для векторных изображений;
- PNG для всех прочих.

Если изображение в макете сделано вектором, то при вёрстке для него надо использовать формат SVG.

Б17. Проект соответствует техническому заданию

Техническое задание каждого проекта можно посмотреть или сохранить на [странице](#) в интерфейсе интенсива, ссылка на которую есть в верхнем меню.

Должны быть выполнены все пункты из раздела «Обязательные требования».

Обратите внимание, что:

- должна быть сделана вёрстка всех модальных окон, реализация открытия необязательна;
- должна быть сделана разметка всех слайдов, оживление слайдеров необязательно;
- если проекты Седона и Глейси сделаны с фиксированной шириной, то требования про процентную ширину не учитываются.

При конфликте технического задания и критериев, считать техническое задание более приоритетным.

Дополнительные критерии

Разметка

Д1. У всех изображений в теге `` прописан размер.

Должны быть указаны действительные размеры картинок без указания единиц измерения.

Верно: размеры картинке заданы.

```
<div class="logo-area">
  
</div>
```

Неверно: размеры картинке заданы с размерностью в `px`.

```
<div class="logo-area">
  
```

```
</div>
```

Неверно: размеры картинке не заданы.

```
<div class="logo-area">
  
</div>
```

Д2. Использовано минимально возможное количество HTML-элементов (нет лишних элементов).

В разметке должно быть использовано минимально возможное количество элементов. Не должно быть лишних обёрток и блоков, которые используются для оформления и могут быть заменены на псевдоэлементы.

Верно: не используется лишняя обёртка, а для декоративного элемента нет дополнительного блока в разметке.

```
<div class="features clearfix">
  <div class="features-item">
    <h2>Быстро</h2>
    <p>Мы делаем свою работу быстро!</p>
  </div>
  ...
</div>
```

Неверно: используется лишняя обёртка, которую можно сократить, а для декоративного элемента использован тег ``, который можно заменить на псевдоэлемент.

```
<div class="features">
  <div class="clearfix">
    <div class="features-item">
      <h2>Быстро</h2>
      <span class="triangle"></span>
      <p>Мы делаем свою работу быстро!</p>
    </div>
  </div>
  ...
</div>
```

Д3. Все скрипты подключены непосредственно перед `</body>`.

Подключение скриптов в самом низу страницы необходимо для того, чтобы загружающийся скрипт не блокировал содержимое страницы.

Верно: скрипт подключается перед закрывающим тегом `<body>`.

```
<!DOCTYPE html>
<html>
  <head>
    ...
```

```
</head>
<body>
  ...
  <script src="js/script.js"></script>
</body>
</html>
```

Неверно: скрипт подключается в секции `<head>`.

```
<!DOCTYPE html>
<html>
  <head>
    ...
    <script src="js/script.js"></script>
  </head>
  <body>
    ...
  </body>
  • </html>
```

Д4. Названия полей форм привязаны к своим полям с помощью `<label>`.

Для улучшения взаимодействия пользователя с элементами форм должна быть реализована активация поля при нажатии на его описание. Для этого необходимо связывать элемент формы с его описанием, используя тег `</label>` и идентификаторы.

Верно: элемент формы `radio` связан с его описанием через идентификатор.

```
<input type="radio" id="spb">
<label for="spb">Санкт-Петербург</label>
```

Верно: элемент формы `radio` и подпись обёрнуты в `<label>`

```
<label>
  <input type="radio"> Санкт-Петербург
</label>
```

Неверно: описание никак не связано с элементом формы.

```
<input type="radio" id="spb"> Санкт-Петербург
```

Стилизация

Д5. Использован `normalize.css`.

Для выполнения этого критерия необходимо, чтобы был использован файл `normalize.css`.

Верно: файл `normalize.css` подключается отдельным файлом перед основным файлом стилей.

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

Неверно: файл сброса стилей не подключается.

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

По желанию содержимое `normalize.css` можно включить в начало основного стилевого файла.

Д6. Для стилизации не использованы `#id`.

Для стилизации объектов лучше использовать селекторы по классам. Использовать `id` для стилизации недопустимо.

Важно: использовать атрибут `id` в HTML-разметке можно и нужно. Например, для привязки полей к подписям в формах.

Верно: стилизация элементов через классы.

```
.feedback {}
.feedback p {}
.main-header {}
```

Неверно: стилизация элементов и каскад строится через идентификаторы.

```
#feedback {}
#feedback p {}
#main-header {}
```

Д7. Нет вложенности селекторов больше двух.

Длинные цепочки вложенных селекторов усложняют код и его поддержку. Хорошим подходом считается использовать вложенность двух или трёх уровней. Селекторы с вложенностью не более двух — это наш маленький, но важный каприз.

Псевдоэлементы или псевдоклассы не увеличивают уровень вложенности.

Важно: допускается использовать селекторы с вложенностью больше двух при оживлении слайдеров на чистом CSS.

Верно: максимальная глубина селекторов не превышает два элемента.

```
.catalog {}
.catalog-item img {}
.catalog-item .item-description {}
.catalog-item h3 {}
.catalog-item:nth-child(odd):hover .item-description::after {} /* это тоже
2 уровня вложенности */
```

Неверно: глубина селекторов больше двух, но при этом их можно сократить.

```
.catalog{}
.catalog .catalog-item img {}
.catalog .catalog-item .item-description {}
.catalog .catalog-item h3 {}
```

Д8. Явно указано подходящее **vertical-align** для потоковых блоков с **display: inline-block**.

По умолчанию элементы **inline-block** имеют выравнивание **baseline**, которое может приводить к нежелательному поведению элементов, например, когда внутри них разное количество строк.

Поэтому нужно явно и осознанно задавать выравнивание таким элементам.

Потоковыми элементами стоит считать те элементы, которые образуют крупные сеточные блоки. При переполнении таких блоков, можно заметить ошибку, при которой блоки будут выравниваться по **baseline**, а не по верхнему краю, как это должно быть. В таком случае задавать блокам **vertical-align** необходимо. Потоковыми элементами могут считаться колонки, карточки товаров, кнопки.

В случае, если **inline-block** используется для создания элементов, для которых значение по умолчанию приемлемо, например элементы имеют фиксированную высоту, вертикальное выравнивание можно не задавать.

Верно: вертикально выравнивание задано явно.

```
.list-item {
  display: inline-block;

  vertical-align: top;
}
```

Неверно: вертикально выравнивание не задано.

```
.list-item {
  display: inline-block;
}
```

Д9. Для CSS-свойств с префиксом указан его вариант без префикса, и это указание идёт последним.

Вендорные префиксы обеспечивают кроссбраузерную работу некоторых свойств CSS. Какие именно свойства необходимо использовать с префиксами, можно посмотреть [здесь](#).

Свойство без префиксов обязательно должно идти после всех свойств с префиксами.

Верно:

```
.class {  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
  box-sizing: border-box;  
}
```

Неверно:

```
.class {  
  box-sizing: border-box;  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
}
```

Д10. Для блока, у которого есть фоновое изображение, прописан фоновый цвет, который соответствует преобладающему цвету изображения (пока изображение не загружено, страница выглядит похоже на макет).

Такой метод используется для перестраховки, как и в случае со шрифтом. Только в этом случае, если фоновое изображение не загрузится, на заднем фоне останется преобладающий цвет.

Верно: указан цвет фона перед основным изображением.

```
body {  
  background: #000000 url("../img/background-body.jpg") no-repeat center top;  
}
```

Результат: картинка не загрузилась, но при этом цвет фона остался тёмным.

Barbershop
BORODINSKI
✦ ✦
✦ ✦

БЫСТРО



МЫ ДЕЛАЕМ СВОЮ РАБОТУ БЫСТРО! ДВА ЧАСА
ПРОЛЕТЯТ НЕЗАМЕТНО И ВЫ - СЧАСТЛИВЫЙ
ОБЛАДАТЕЛЬ СТИЛЬНОЙ СТРИЖКИ-МИНУТКИ!

КРУТО



ЗАБУДЬТЕ, КАК ВЫ СТРИГЛИСЬ РАНЬШЕ.
МЫ СДЕЛАЕМ ИЗ ВАС ЗВЕЗДУ ФУТБОЛА ИЛИ КИНО! ВО
ВСЯКОМ СЛУЧАЕ ВНЕШНЕ.

ДОРОГО



НАШИ МАСТЕРА - ПРОФЕССИОНАЛЫ СВОЕГО ДЕЛА И НЕ
МОГУТ СТОИТЬ ДЕШЕВО. К ТОМУ ЖЕ, РАЗВЕ ЦЕНА НЕ
ДАЕТ ОПРЕДЕЛЕННЫЙ СТАТУС?

НОВОСТИ

НАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ ВЫ
МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ СТРИЖКИ
25 сентября

В НАШЕЙ КОМАНДЕ ПОПОЛНЕНИЕ, БОРИС «БРИТВА»
СТРИГУНЕЦ, ОБЛАДАТЕЛЬ МНОЖЕСТВА ТИТУЛОВ И
НАГРАД ПОПОЛНИЛ НАШИ СТРОЙНЫЕ РЯДЫ
19 сентября

ФОТОГАЛЕРЕЯ



Неверно: указано только основное изображение.

```
body {  
  background: url("../img/background-body.jpg") no-repeat center top;  
}
```

Результат: картинка не загрузилась, а цвет фона стал белым.

BORODINSKI

НОВОСТИ

НАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ ВЫ
МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ СТРИЖКИ
25 сентября

В НАШЕЙ КОМАНДЕ ПОПОЛНЕНИЕ, БОРИС «БРИТВА»
СТРИГУНЕЦ, ОБЛАДАТЕЛЬ МНОЖЕСТВА ТИТУЛОВ И
НАГРАД ПОПОЛНИЛ НАШИ СТРОЙНЫЕ РЯДЫ
19 сентября

ФОТОГАЛЕРЕЯ



Для блока с картой также следует указывать фоновый цвет.

Д11. Все состояния элементов (смотрите [styleguide.psd](#)) прописаны в стилевом файле.

В соответствии с файлом [styleguide.psd](#), который присутствует в каждом проекте, все указанные в нём элементы должны иметь соответствующие эффекты при наведении и нажатии. Должны присутствовать все активные состояния, а для чекбоксов и радио кнопок прописаны состояния [disabled](#).

Д12. Стилизация нестандартных элементов форм.

Нестандартные элементы форм (чекбоксы, радиокнопки, выпадающие списки) должны быть реализованы без использования JavaScript.

При реализации нестандартного выпадающего списка стилизуется всё, кроме выпадающего списка. В том числе стилизовать нужно и стрелочку для [select](#).

Д13. Файл стилей представлен в двух вариантах: с форматированием и минимизированный, к странице подключён минимизированный.

Обычный CSS-код.

```
.feedback a {  
  font-size: 22px;  
  line-height: 30px;  
  color: #000000;  
}
```

```
.feedback p {  
  font-size: 14px;  
  line-height: 20px;  
}
```

Сжатый CSS-код.

```
.feedback a{font-size:22px;line-height:30px;color:#000}.feedback p{font-size:14px;line-height:20px}
```

Д25. Нет глобальных стилей тегов.

Для задания стилей используются только селекторы по классам или каскад.

Исключения:

- Normalize.css, который исправляет браузерные умолчания
- Уникальные теги документа: `html`, `body`
- Дополнительная нормализация: `a` и `img`
- Общее правило для `box-sizing` с помощью `inherit` (смотри пример в колонке «Верно»)

Верно:

```
body { margin: 0 }  
img { max-width: 100% }  
a { text-decoration: none }  
  
.feedback { color: black }  
.feedback ul { list-style: none }  
.feedback p { margin-bottom: 0 }  
  
html { box-sizing: border-box; }  
*, *:before, *:after { box-sizing: inherit; }
```

Неверно:

```
ul { list-style: none }  
p { margin: 0 }  
div { font-size: 80% }  
* { box-sizing: border-box }
```

JavaScript

Д14. Собственные скрипты собраны в отдельном JavaScript-файле, который подключён в HTML перед закрывающим тегом `body`.

Если JavaScript-код вставлен напрямую в HTML-файле в теге `script`, то критерий считается невыполненным. Исключением являются скрипты интерактивной карты.

Д15. JavaScript-файл представлен в двух вариантах: с форматированием и минимизированный, к странице подключён минимизированный.

Обычный JS-код.

```
var popup = document.querySelector(".popup");
var open = document.querySelector(".open-btn");
var close = document.querySelector(".close-btn");

open.addEventListener("click", function(event) {
    event.preventDefault(event);
    popup.classList.add("show");
});

close.addEventListener("click", function(event) {
    event.preventDefault(event);
    popup.classList.remove("show");
});
```

Сжатый JS-код.

```
var popup=document.querySelector(".popup"),open=document.querySelector(".open-btn"),close=document.querySelector(".close-btn");open.addEventListener("click",function(a){a.preventDefault(a),popup.classList.add("show")}),close.addEventListener("click",function(a){a.preventDefault(a),popup.classList.remove("show")});
```

Д16. С помощью JavaScript реализовано открытие/закрытие окна с формой (без вспомогательных библиотек).

Д17. С помощью JavaScript добавлена анимация формы (без вспомогательных библиотек).

Д18. Добавлена интерактивная карта (допускается подключение с помощью `iframe`).

Тестирование

Д19. Вёрстка проходит тест на переполнение контентом.

Не ломается при добавлении в элементы большего количества текста;

Верно: текст растягивает блок.

НОВОСТИ

НАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ ВЫ
МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
ВЫ МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
ВЫ МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР!

25 сентября

В НАШЕЙ КОМАНДЕ ПОПОЛНЕНИЕ, БОРИС «БРИТВА»
СТРИГУНЦ, ОБЛАДАТЕЛЬ МНОЖЕСТВА ТИТУЛОВ И
НАГРАД ПОПОЛНИЛ НАШИ СТРОЙНЫЕ РЯДЫ

19 сентября

ВСЕ НОВОСТИ

ФОТОГАЛЕРЕЯ



НАЗАД

ВПЕРЕД

КОНТАКТНАЯ ИНФОРМАЦИЯ

BARBERSHOP «BORODINSKI»

АДРЕС: Г. САНКТ-ПЕТЕРБУРГ, УЛ. СОБАКИ ПАВЛОВА, Д. 13

ТЕЛЕФОН: +7 (495) 666-02-666

ВРЕМЯ РАБОТЫ:

ПН – ПТ: С 10:00 ДО 22:00

СБ – ВС: С 10:00 ДО 19:00

КАК ПРОЕХАТЬ

ОБРАТНАЯ СВЯЗЬ

ЗАПИСАТЬСЯ

УКАЖИТЕ ЖЕЛАЕМУЮ ДАТУ И ВРЕМЯ И МЫ СВЯЖЕМСЯ С
ВАМИ ДЛЯ ПОДТВЕРЖДЕНИЯ БРОНИ

ДД.ММ.ГГ

--:--

ВАШЕ ИМЯ

ТЕЛЕФОН

ОТПРАВИТЬ

Неверно: текст выпадает за пределы блока со светлым фоном.

НОВОСТИ

НАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ ВЫ
МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
ВЫ МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
ВЫ МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
ВЫ МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
ВЫ МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ

ФОТОГАЛЕРЕЯ

[НАЗАД](#)[ВПЕРЕД](#)

СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
ВЫ МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
ВЫ МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ
СТРИЖКИНАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ

25 сентября

В НАШЕЙ КОМАНДЕ ПОПОЛНЕНИЕ, БОРИС «БРИТВА»
СТРИГУНЕЦ, ОБЛАДАТЕЛЬ МНОЖЕСТВА ТИТУЛОВ И
НАГРАД ПОПОЛНИЛ НАШИ СТРОЙНЫЕ РЯДЫ
19 сентября

[ВСЕ НОВОСТИ](#)

КОНТАКТНАЯ ИНФОРМАЦИЯ

BARBERSHOP «BORODINSKI»

АДРЕС: Г. САНКТ-ПЕТЕРБУРГ, УЛ. СОБАКИ ПАВЛОВА, Д. 13

ТЕЛЕФОН: +7 (495) 666-02-666

ВРЕМЯ РАБОТЫ:

ПН — ПТ: С 10:00 ДО 22:00

СБ — ВС: С 10:00 ДО 19:00

[КАК ПРОЕХАТЬ](#)[ОБРАТНАЯ СВЯЗЬ](#)[ЗАПИСАТЬСЯ](#)

- Не ломается при добавлении в кнопки большего количества текста;
- Не ломается при использовании картинок с неподходящими размерами;

Верно: добавление картинки не сломало сетку, она вписалась в соответствующий размер.

ФОТОГАЛЕРЕЯ



НАЗАД

ВПЕРЕД

Неверно: добавление большей картинки сломало сетку.

ФОТОГАЛЕРЕЯ



НАЗАД

ВПЕРЕД

- Не ломается при изменении количества потоковых блоков;

Верно: переполнение верхнего блока растягивает его и двигает нижний.

НОВОСТИ

НАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ ВЫ
МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ СТРИЖКИ

25 сентября

НАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ ВЫ
МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ СТРИЖКИ

25 сентября

НАМ НАКОНЕЦ ЗАВЕЗЛИ ЯГЕРМАЙСТЕР! ТЕПЕРЬ ВЫ
МОЖЕТЕ ПРОПУСТИТЬ СТАКАНЧИК ВО ВРЕМЯ СТРИЖКИ

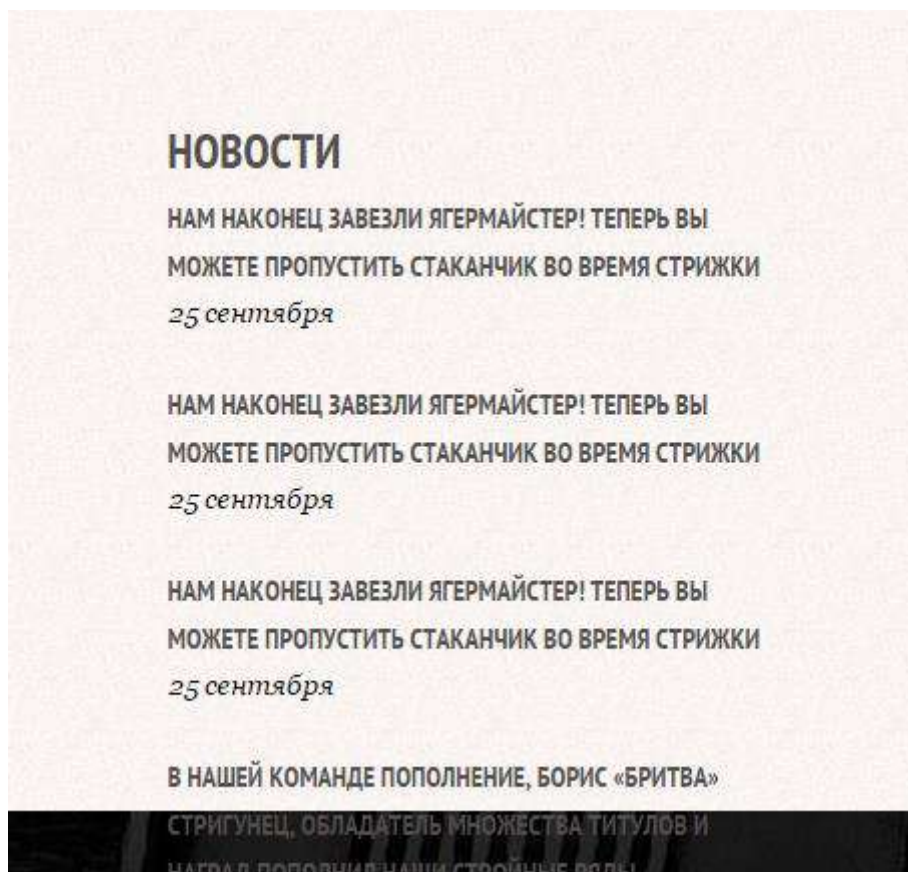
25 сентября

В НАШЕЙ КОМАНДЕ ПОПОЛНЕНИЕ, БОРИС «БРИТВА»
СТРИГУНЕЦ, ОБЛАДАТЕЛЬ МНОЖЕСТВА ТИТУЛОВ И
НАГРАД ПОПОЛНИЛ НАШИ СТРОЙНЫЕ РЯДЫ

19 сентября

[ВСЕ НОВОСТИ](#)

Неверно: переполнение верхнего блока ломает всю сетку документа.



Текст не выпадает из блоков, нижерасположенные блоки не скрываются, смещение блоков в потоке сохраняет логику потока (не приводит к нарушению сетки).

Примеры переполнений по проектам можно посмотреть [здесь](#).

Доступность

Д26. У интерактивных элементов при нажатии или фокусе с клавиатуры есть активное состояние.

Активное состояние интерактивных элементов при нажатии или фокусе с клавиатуры должно оставаться либо встроенным браузерным, либо быть равноценно переназначено. В таком случае по интерактивным элементам сайта можно передвигаться с клавиатуры клавишей **Tab** и видеть каждый текущий элемент в активном состоянии.

Д27. Все интерактивные элементы имеют текстовое описание.

Интерактивные элементы, представленные на макете только графически, без текста, должны содержать текстовое описание. Тогда при чтении или прослушивании интерфейса пользователь будет понимать, что произойдёт при нажатии на элемент.

- Нёрдс — социальные кнопки, стрелочки для сортировки;
- Техномарт — социальные кнопки, стрелочки для сортировки, кнопка для выхода из личного кабинета;

- Седона — социальные кнопки, кнопка для появления календаря, кнопки для уменьшения и увеличения количества взрослых и детей), логотип Академии в подвале, стрелочки для фильтрации, количество звёзд у гостиницы;
- Глейси — социальные кнопки, логотип Академии в подвале;
- Девайс — социальные кнопки, стрелочки для сортировки, логотип Академии в подвале.

Разное

Д20. Критическая функциональность сайта работоспособна без JavaScript (использовано прогрессивное улучшение).

Например:

- Все формы являются работоспособными без JavaScript;
- Элементы, вызывающие появление попапов, являются ссылками, ведущими на отдельные страницы;

Для выполнения этого критерия достаточно указать адрес на страницу (например, `writeus.html`), на которую будет происходить переход в случае неработоспособности JavaScript. При этом верстать саму страницу необязательно. Пример:

```
<a class="btn btn-open-form" href="form.html">Открыть форму</a>
```

- Интерактивная карта без JavaScript показывает статичную картинку с картой.

Д21. Отсутствуют непонятные сокращения и транслит в названиях классов, атрибутах, названиях файлов и так далее.

Для написания классов необходимо использовать английские слова и термины.

Верно: используются только английские слова.

```
.login-button {}
.container {}
.footer {}
main-logo.png
social-btn.png
```

Неверно: транслит с русского языка, непонятные сокращения.

```
.kнопка-vxoda {}
.kontainer {}
.sf_pw {}
1.jpg
b.png
```

Д22. При взаимодействии с элементами (наведение, нажатие) ни сам элемент, ни окружающие его блоки не меняют своего положения (если иное не прописано в техническом задании или **styleguide.psd**).

Д23. Проведена базовая оптимизация: минифицированы стили и скрипты (при наличии).

Д24. Страницы должны ссылаться друг на друга из главного меню.

Для написания классов необходимо использовать английские слова и термины.

Верно: используются только английские слова.

```
.login-button {}  
.container {}  
.footer {}  
main-logo.png  
social-btn.png
```

Неверно: транслит с русского языка, непонятные сокращения.

```
.knopka-vxoda {}  
.kontainer {}  
.sf_pw {}  
1.jpg  
b.png
```