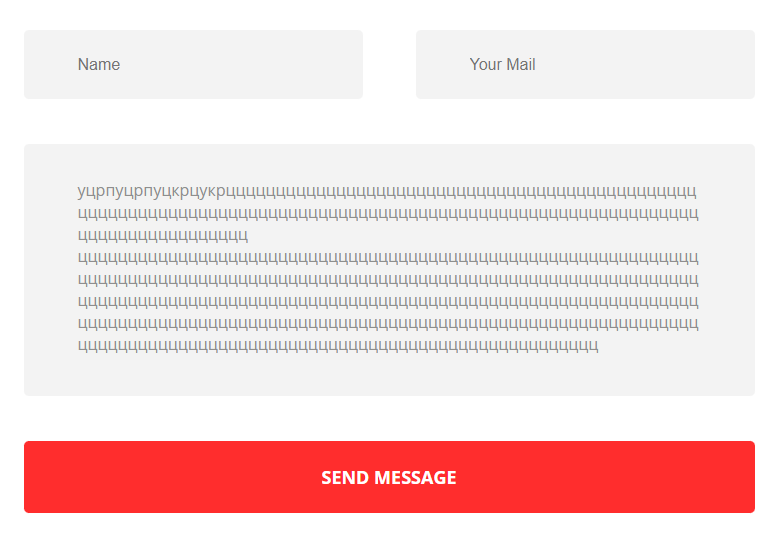
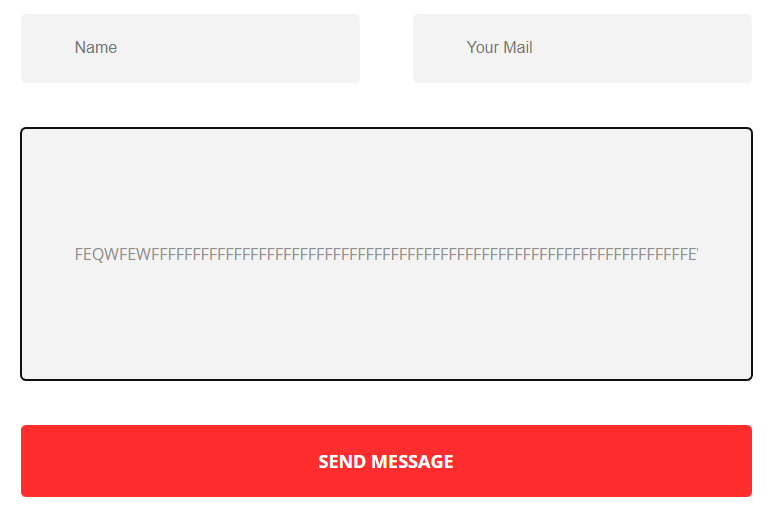
**A5.1 Содержание модуля**

Библиотека **Bootstrap 4.** Но сналача о прошлом. Они в форме вместо **input** использовали **textarea.** А в чем разница? Вот



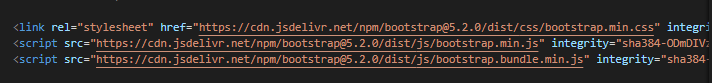
**A5.3 Знакомимся с CDN, подключаем CSS, шаблон-заготовка**

CDN – это распределенная сеть серверов, которые размещают в открытом доступе. Много библиотек можно так найти. Итак **Bootstrap 4.**

Bootstrap - это CSS-фреймфорк, позволяющий создавать сложные и при этом легко конфигурируемые интерфейсы с помощью классов. Сервис популярный и заметно ускоряет скорость верстки сайта.

Скачать его можно здесь. <https://www.bootstrapcdn.com/>

Но буду ли я устанавливать старый 4-ый? Лучше поставлю 5-ый. Думаю там лишь больше фич.

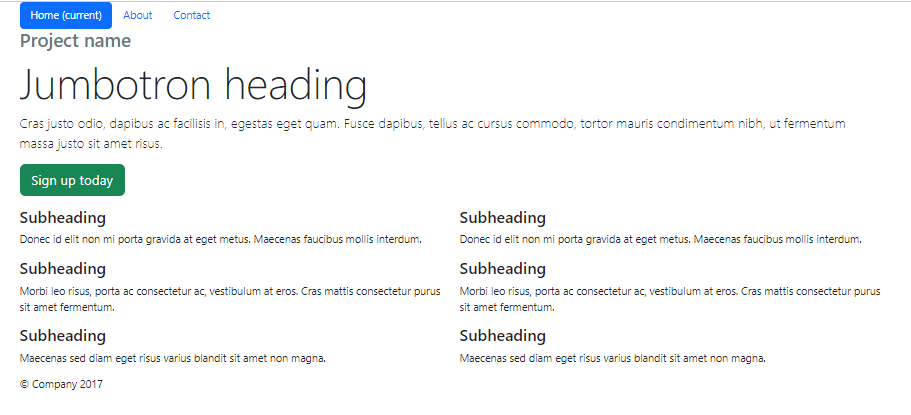


Поставил **js, css** и какой-то **bundle**

****

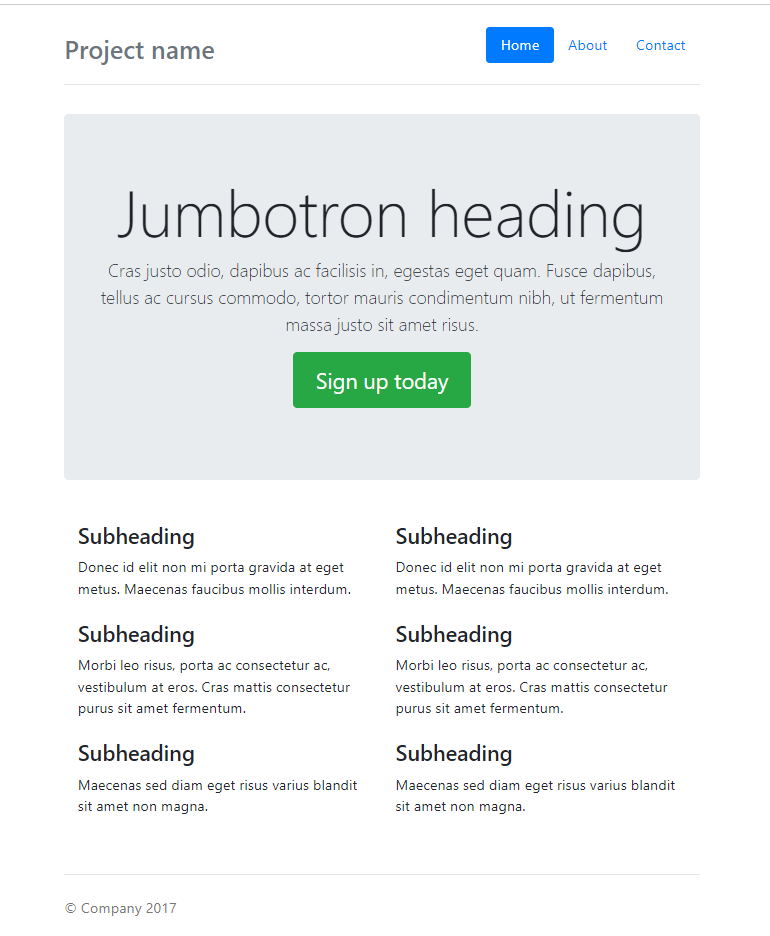
**OK**

Так вот, я скопировал код с другой страницы, где используется **bootstrap и…**

****

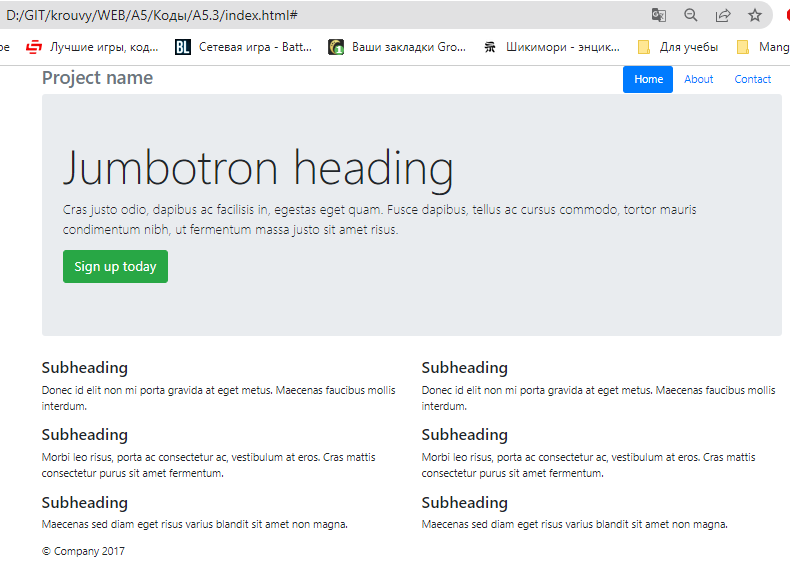
Получил это.

А должен был это:

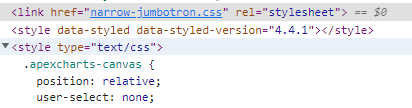
****

Ладно. А если bootsrap 4?

**Ладно с bootstrap 4 пашет также.** Буду следовать инструкциям.

****

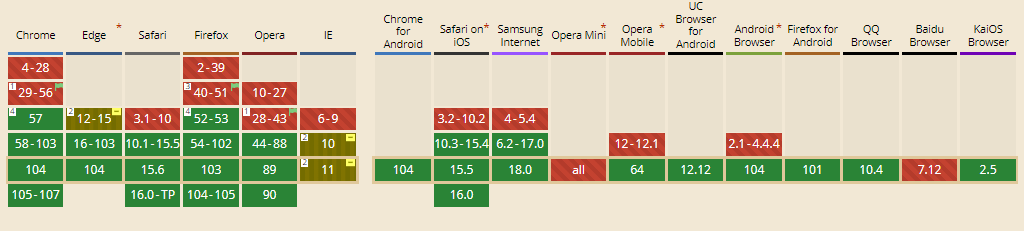
Даже так есть местами отличия, но не страшно. Дело в том, что на той странице есть еще дополнительные ссылки на стили.



Главное, что видно. Bootstrap подключен.

**A5.4 Сетки: построение, использование**

Назначение сеток. Это полезный инструмент при разработке интерфейсов. В новом стандарте **CSS** появилась новая спецификация **Grid Layout –** набор правил для построения интерфейсов в нативной и более удобной форме. Стандарт поддерживается современными браузерами.

****

Научимся же использовать этот грид. А то флексы уже приелись наверно. Основная цель, облегчить создание сложных макетов без сторонних библиотек. Bootstap не единственный фрейморк, поддерживающий сетки. Его аналони это:

**Foundation**

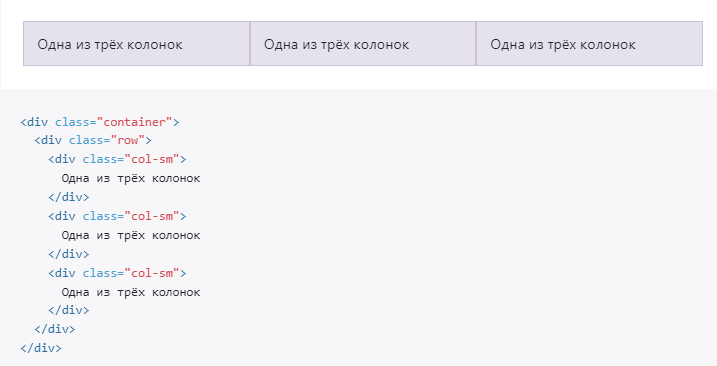
**Bulma**

**Skeleton**

**Bootstrap Grid** нужна для разметки страницы. Если конкретно, то для адаптивных макетов.

Сетка – структура похожая на таблицу. У нее есть общий контейнер обертка(**container**), в нем располагаются ряды**(row),** каждый из которых может содержать от 1 до 12 столбцов(**col**). Колонка тоже может быть контейнером и содержать в себе еще дочерние элементы.

Бутстрап реализован с помощью флексбокса и полностью адаптивен. Оформляется это с помощью классов и дивов.



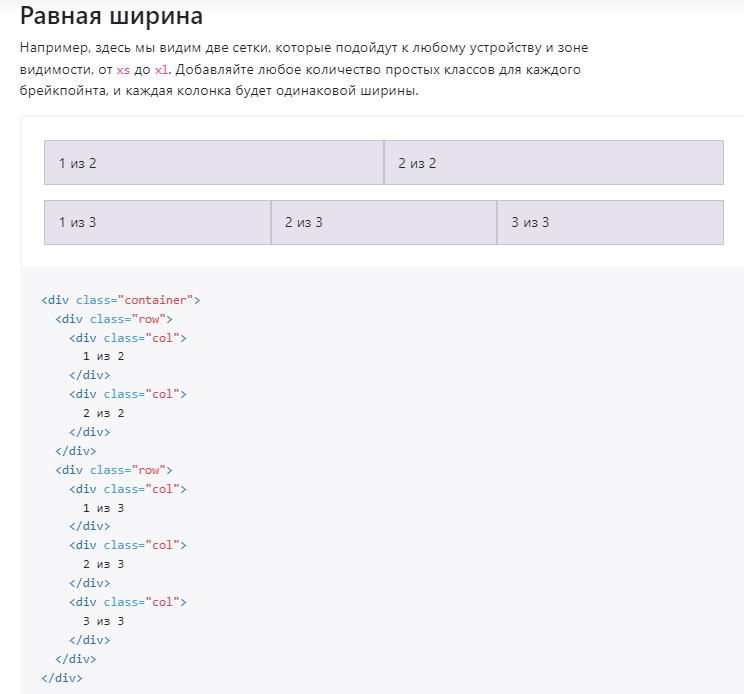
Создана строка с тремя колонками равного размера.

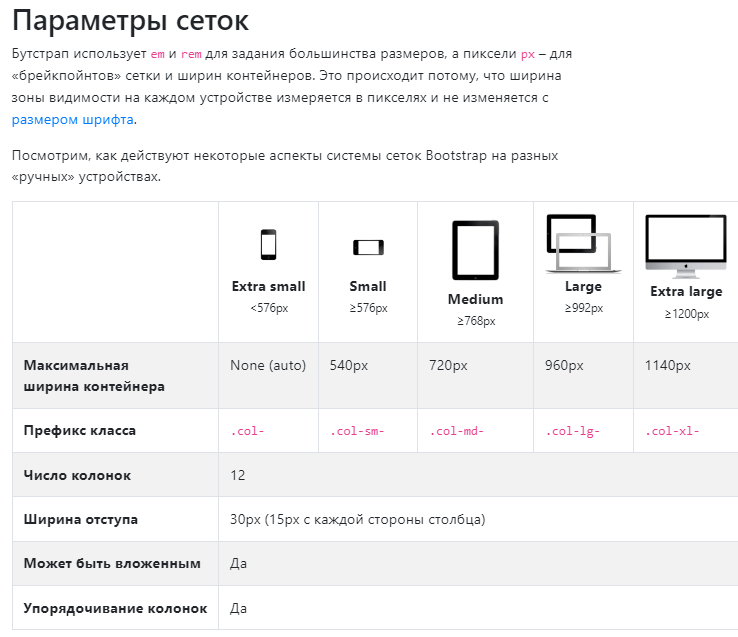
**.container** для фиксированной ширины

**.container-fluid** для 100% ширины

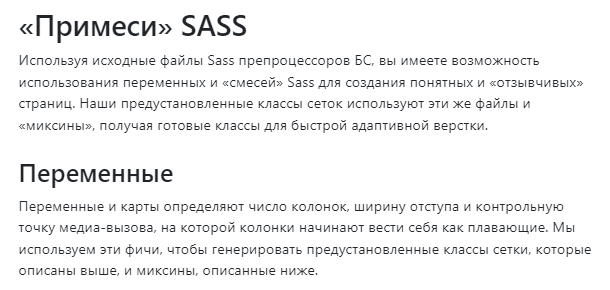
**Содержимое должно быть расположено в колонках, и только колонки могут быть расположены в рядах.**

**Используйте классы колонок со специальными контрольными точками (например, .col-sm-6) для легкого расположения колонок без использования явно обозначенных номеров классов.**

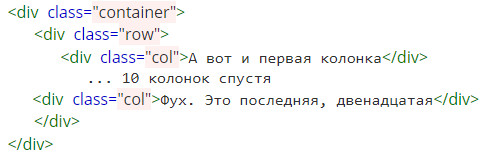
****

****

Короче. Дохера документации. Курс оплачен. Учите. А не ссылки кидайте.

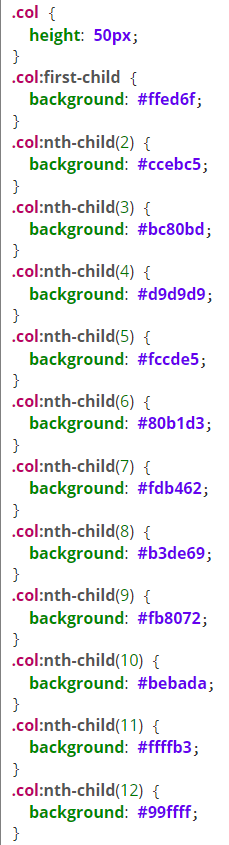


Вот как строится сетка:

****

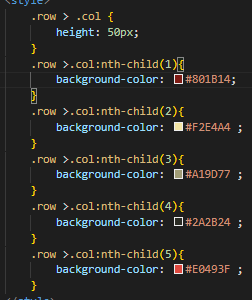
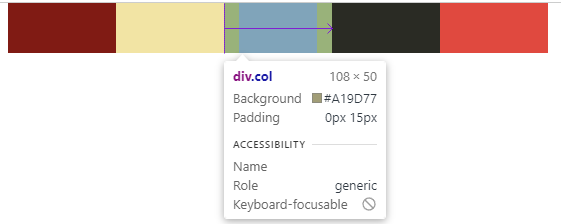
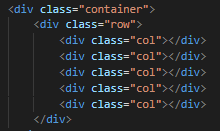
И что, нужно прямо все колонки прописать так? А не.. Ряд может содержать от одной, до 12 колонок.

Допустим дадим колонкам разные цвета. И будет

ЛГБТ со всеми 12 гендерами крч.

Попробую сам.



Нормально. А если вместо **container** использовать **container-fluid,** то сетка будет растянута на всю ширину экрана

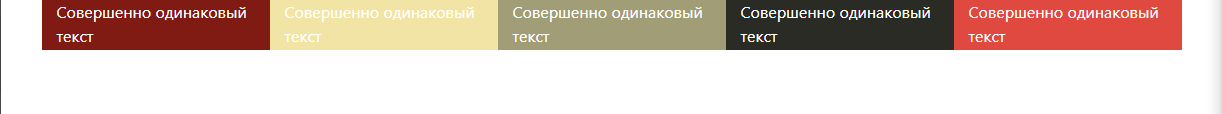


А если вставить текст и взглянуть на адаптивность?

25%



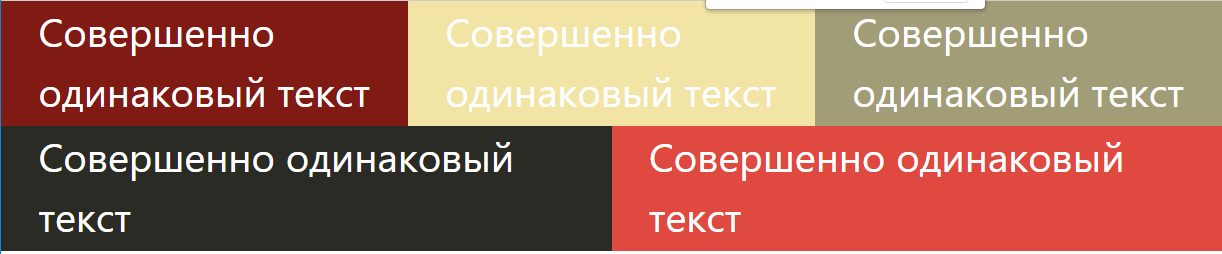
100%



175%



200%

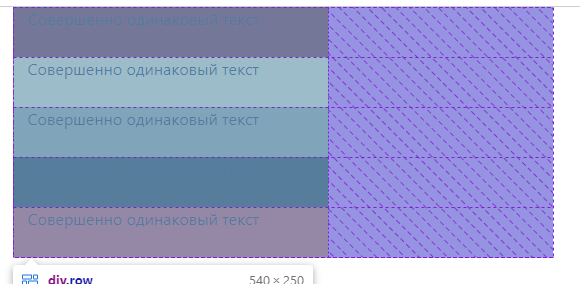
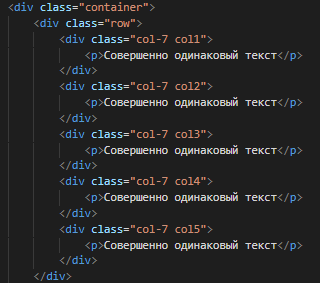


Ну в целом не плохо. Но не идеально. Где-то слово текст съелось.

Как происходит управление размерами колонок? Если не давать размеры, то они будут автоматическими. Нужно больше гибкости. К примеру, чтобы шапка занимала всю ширину, но блоки ниже нет.

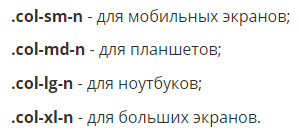
Для задания ширины используются специальные классы вида **.col-x-n,** где вместо **x** ставится буквенное **обозначение размера экрана.** Вместо **n –** число от 1 до 12. Обозначающие сколько место займет **данная колонка.** Если опустить **x.** То размер будет применяться на всех экранах.

Если **n** будет 6. То эта колонка будет шириной в половину от всего ряда. Хм… А если все сделать 7?

Как я и думал.

**Обозначение размера экрана такие:**

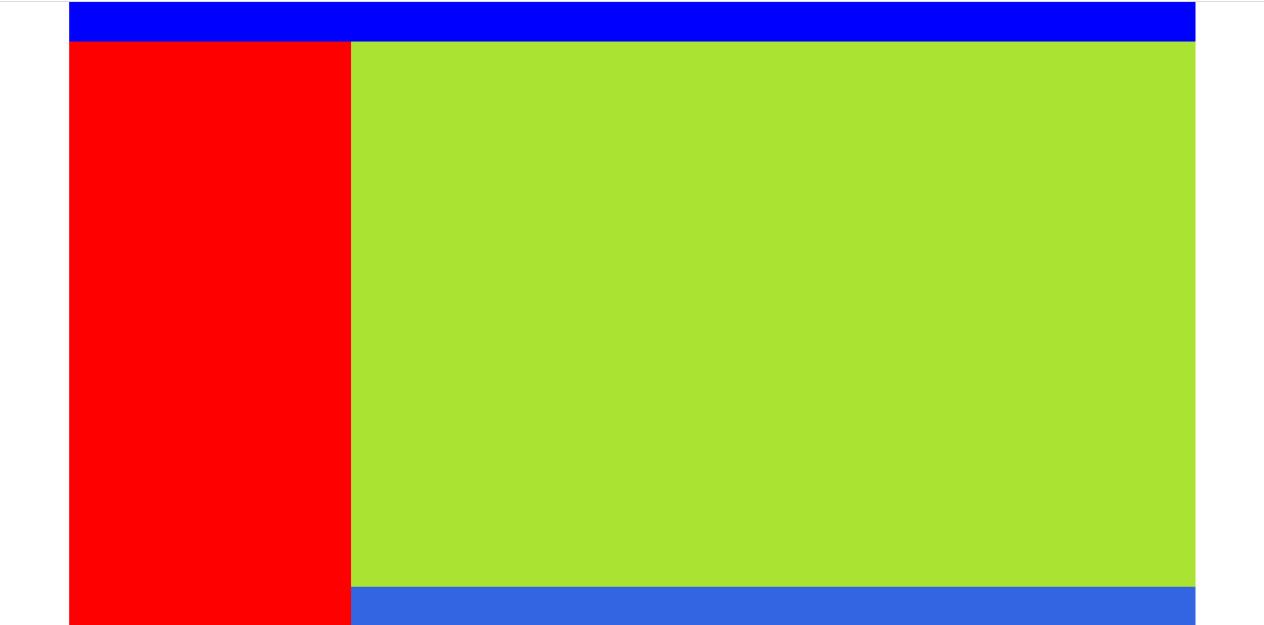


Не обязательно указывать все экраны. У ноутбуков и больших например они могут совпадать.

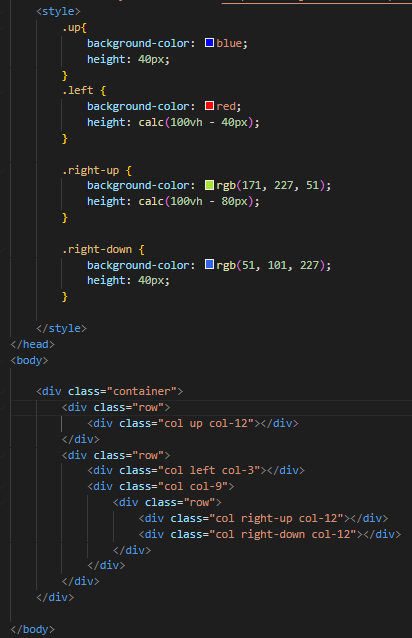
Если у сайта есть 2 раскладки, одна - для всех десктопных устройств, вторая - для мобильных и планшетов, будет достаточно поставить 2 класса: **.col-n** и **.col-lg-n.** Первый класс будет действовать по умолчанию на всех устройствах, где разрешение меньше, чем у ноутбука (по стандартам **Bootstrap** это означает меньше 960px). Как только разрешение будет больше 960px, начнет действовать второй класс.



Пришлось чуть повозится. Но…

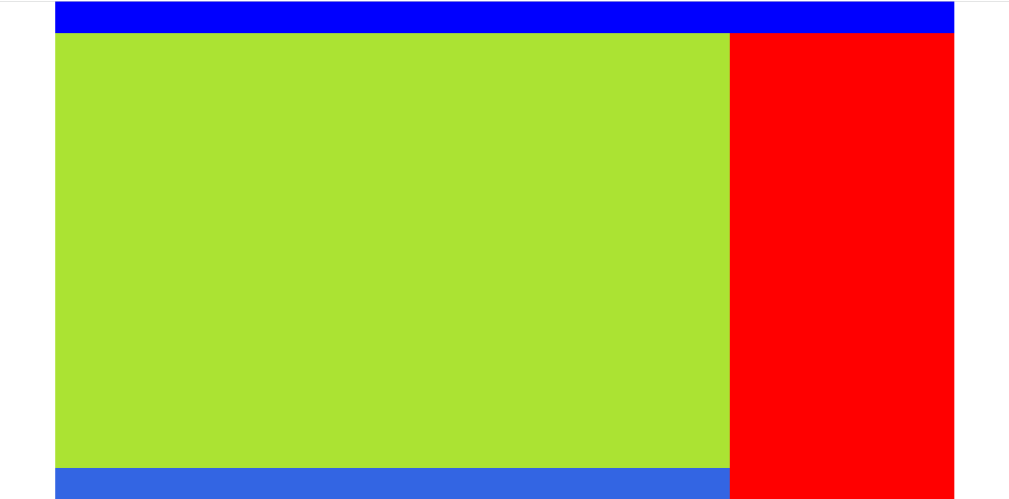


Да… заметно, что шапка и блоки должны быть выше. Но такие параметры были в условии.



Важно отметить. Все **row** имеют в свойства **display:flex;** Так что если нужно как-то изменить позиционирование строки, всегда пожалуйста. Например добавить:

**flex-direction: row-reverse;**

****

**Вот эту статью прочесть:**

[**https://medium.com/@stasonmars/%D1%81%D0%B5%D1%82%D0%BA%D0%B0-%D0%B2-bootstrap-4-%D0%BF%D0%BE%D0%B4%D1%80%D0%BE%D0%B1%D0%BD%D0%BE%D0%B5-%D1%80%D1%83%D0%BA%D0%BE%D0%B2%D0%BE%D0%B4%D1%81%D1%82%D0%B2%D0%BE-bb20ce8a14ad**](https://medium.com/@stasonmars/%D1%81%D0%B5%D1%82%D0%BA%D0%B0-%D0%B2-bootstrap-4-%D0%BF%D0%BE%D0%B4%D1%80%D0%BE%D0%B1%D0%BD%D0%BE%D0%B5-%D1%80%D1%83%D0%BA%D0%BE%D0%B2%D0%BE%D0%B4%D1%81%D1%82%D0%B2%D0%BE-bb20ce8a14ad)

**A5.5 Типографика в Bootstrap**