

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Локтионова Оксана Геннадьевна  
Должность: проректор по учебной работе  
Дата подписания: 12.02.2021 16:02:46  
Уникальный программный ключ:  
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

## МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное  
учреждение высшего образования

«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра информационной безопасности

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Локтионова  
« 1 » 02 2018 г.



### ИЗУЧЕНИЕ АДАПТИВНОЙ ВЁРСТКИ

Методические указания по выполнению лабораторной работы по дисциплине «Веб-программирование» для студентов специальности 10.05.02 и 10.03.01.

Курск 2018

УДК 004.056.55

Составитель А.Л. Марухленко

Рецензент

Кандидат технических наук, доцент *И.В. Калуцкий*

**Изучение адаптивной вёрстки:** методические указания по выполнению лабораторных работ по дисциплине «Веб-программирование» / Юго-Зап. гос. ун-т; сост. А.Л. Марухленко. Курск, 2018. – 25 с.

Рассматриваются основы адаптивной вёрстки с использованием фреймворка Bootstrap. Указывается порядок выполнения лабораторной работы и содержание отчета.

Методические указания соответствуют требованиям программы, утвержденной учебно-методическим объединением по образованию в области информационной безопасности (УМО ИБ).

Предназначены для студентов специальности 10.05.02 и 10.03.01.

Текст печатается в авторской редакции

Подписано в печать *10.2.18*. Формат 60x84 1/16.  
Усл.печ. л.1,45. Уч.-изд.л.1,57. Тираж 30 экз. Заказ. Бесплатно. *244*  
Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## **СОДЕРЖАНИЕ**

**Ошибка! Закладка не определена.**

**ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ****Ошибка! Закладка не определена.**

**ЗАДАНИЕ И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ****Ошибка! Закладка не определена.****13**

**БИБЛИОГРАФИЧЕСКИЙ СПИСОК****Ошибка! Закладка не определена.****25**

## ЦЕЛЬ РАБОТЫ

Целью выполнения лабораторной работы является формирование у студентов навыков и умений по использованию фреймворка Bootstrap.

## ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Общие сведения об фреймворке Bootstrap

Фреймворк – это **стандартизированный набор концепций, практик и критериев для работы с общим типом проблем**, который может быть использован как руководство по помощи и решения новых проблем схожей природы.

Большинство вебсайтов имеют схожую (а иногда и одинаковую) структуры. Целью фреймворка является обеспечение общей структуры, чтобы разработчикам не приходилось делать всё сначала и можно было использовать разработанные куски кода. Таким образом фреймворки позволяют уменьшить объём работы и затрачиваемое время.

Практически любая веб-страница содержит множество похожих компонентов, которые встречаются и на других сайтах. Это меню, навигация, элементы форм, заголовки и др. Не говоря уже про многоколоночную вёрстку, без которой вообще сложно превратить картинку макета в готовую веб-страницу. Чтобы меньше писать кода и по максимуму задействовать уже готовые решения применяются специализированные библиотеки, их часто называют фреймворками. Одним из таких фреймворков для вёрстки является Bootstrap.

Bootstrap разработали Марк Отто и Якоб Торнтон, сотрудники Twitter, именно поэтому в названии фигурирует имя компании. Их цель понятна любому разработчику – создать единый стандартный набор инструментов для сотрудников компании, ускоряющий их работу.

На сегодняшний день Bootstrap давно перерос рамки одной компании, это открытый продукт применяемый веб-разработчиками для вёрстки сайтов во всём мире.

Фактически Bootstrap представляет собой конструктор, фрагменты которого вы включаете в свой проект при необходимости. Это уменьшает время разработки, потому что не требуется придумывать и писать их самостоятельно.

Bootstrap направлен на создание макета под разные устройства – ноутбуки, планшеты, смартфоны. При этом код пишется один, а масштабирование в зависимости от ширины устройства берёт на себя фреймворк.

Компоненты библиотеки написаны и протестированы с учётом работы разных браузеров. Это гарантирует, что макет будет выглядеть одинаково независимо от выбранного браузера.

Чтобы использовать библиотеку в своей работе, требуется обладать минимальными знаниями по HTML, CSS и JavaScript. Это позволяет создавать эффектные сайты даже начинающим разработчикам.

Bootstrap не просто вставляет какие-то элементы на страницу, но сразу же устанавливает их оформление и взаимодействие с пользователем через JavaScript. Вы получаете полностью работающий набор компонент, который достаточно добавить и настроить под себя.

Недостатки, конечно же, тоже имеются и они вытекают в основном из универсальности системы.

Во-первых, файлы библиотеки, даже сжатые, занимают довольно много места и увеличивают нагрузку на сервер. С повышением опыта нужно загружать и устанавливать только требуемые компоненты Bootstrap, сокращая тем самым объём файлов.

Во-вторых, предлагаемые в библиотеке стили могут не подходить под дизайн разрабатываемого сайта и придётся много переделывать. В таких случаях, как известно, проще написать всё самому с нуля.

В общем, Bootstrap годится для типовых сайтов, дизайн которых ориентирован на библиотеку. А сайтов, которые предлагают темы и шаблоны, в том числе бесплатные, в последнее время родилось довольно много. Достаточно поискать по ключевым словам «BootstrapTheme».

Для начала необходимо скачать Bootstrap с сайта разработчика по следующему адресу:

<http://getbootstrap.com/getting-started/#download>

Это полная версия, содержащая все необходимые скрипты и стили. Внутри находится три папки: `css`, `fonts` и `js`.

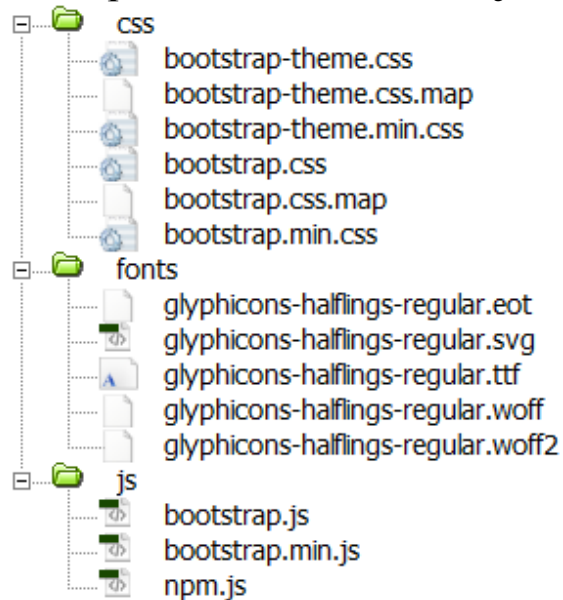


Рисунок 1 – Файлы содержащиеся в Bootstrap.

В папках со стилями и скриптами приложено две версии файлов – исходная и компактная (в имени содержится `min`). Компактная отличается лишь размером файла и снижением читаемости кода. Лучше всего на рабочий сайт добавлять именно эту версию, так мы чуть ускорим загрузку веб-страниц.

Копируем все папки в наш проект и в корне создаём `index.html`. В итоге структура нашего проекта будет выглядеть следующим образом.

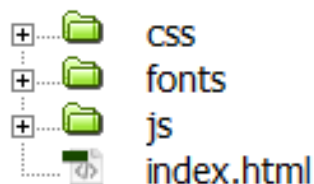


Рисунок 2 – Структура проекта.

Содержимое `index.html` включает в себя ссылку на стилевой файл `bootstrap.css` и внизу страницы мы вызываем `bootstrap.js`. Больше пока ничего не нужно.

```
<!doctype html>
```

```

<html>
<head>
<meta charset="utf-8">
<title>Bootstrap</title>
<link href="css/bootstrap.css" rel="stylesheet">
</head>
<body>
<h1>Привет, мир!</h1>
<scriptsrc="js/bootstrap.js"></script>
</body>
</html>

```

Если в процессе работы потребуется переопределить стили каких-то элементов, то вы можете подключить ещё один собственный стилевой файл и в нём задать необходимые свойства

### **Адаптивная вёрстка с помощью фреймворка Bootstrap**

Адаптивная вёрстка в фреймворке Bootstrap начинается с подключения файлов Bootstrap к нашему index.html.

Неотъемлемой частью адаптивной вёрстки является использование стиля container для общего блока div.

```
<div class="container">...</div>
```

Container отвечает за выравнивание блока по центру с отступами слева и справа по 15 px. Контейнер будет иметь фиксированную ширину, изменение которой происходит в следствии изменения ширины окна браузера (таблица 4).

Таблица 4.

Ширина контейнера	Ширина окна веб-клиента
1170px	Свыше или равна 1200px
970px	Свыше или равна 992px и меньше 1200px
750px	Свыше или равна 768px и меньше 992px

Динамическая ширина (ширина равна ширине рабочей области окна веб-клиента)	Меньше 768px
--	--------------

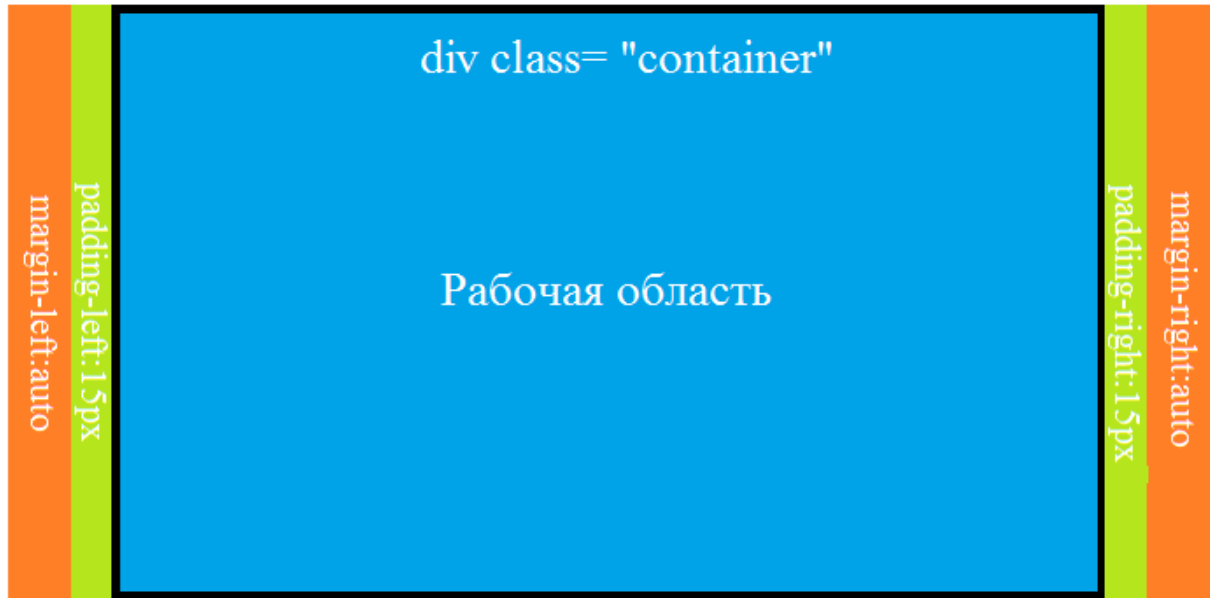


Рисунок 3 – Наглядное представление стиля container.

Или же можно использовать стиль `container-fluid`, который не имеет фиксированной ширины и из свойств имеет только отступы с лева и права по 15px.

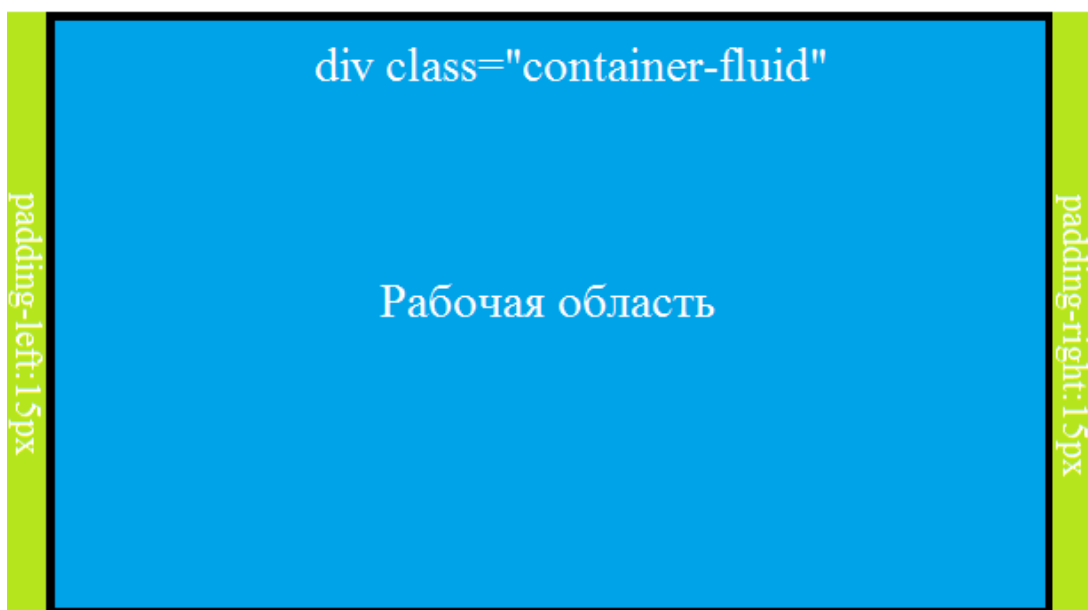


Рисунок 4 – Наглядное представление стиля container-fluid.



В Bootstrap для адаптивной вёрстки используется разделение рабочей области на 12 колонок. Для их правильного функционирования нужно обязательно использовать стиль row.

```
<div class="row">...</div>
```

Ряд принимает такую же ширину, как и контейнер, но имеет отрицательный отступ слева и справа в 15 px.

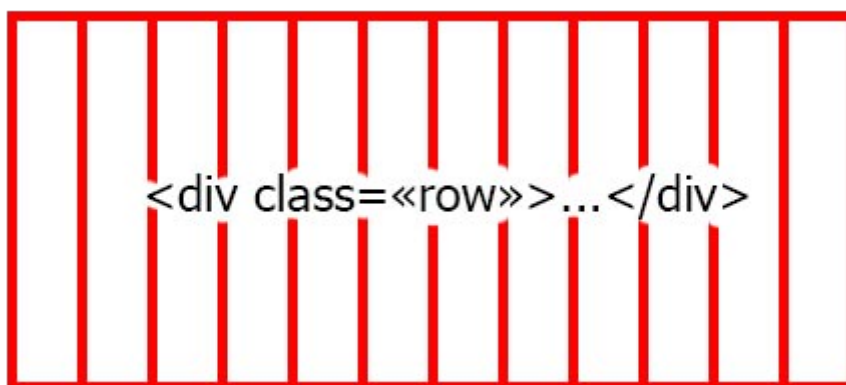


Рисунок 5 – Блоки в одном ряду.

Внутри ряда будут размещаться блоки. В общем виде это выглядит так:

```
<div class="row">  
  <div class="col-*-*">...</div>  
  <div class="col-*-*">...</div>  
  <div class="col-*-*">...</div>  
</div>
```

Ширина каждого блока указывается в относительном формате посредством выбора необходимого числа колонок. То есть минимальная ширина блока равняется одной колонке, а максимальная ширина – двенадцати колонкам. Например, если нам нужно в одном ряду иметь три блока, из которых первый занимает половину всей области, а второй и третий блок равны друг другу и вместе занимают оставшуюся половину, то необходимо присвоить перво-

му блоку ширину в шесть колонок, а второму и третьему по три колонке каждому. В сумме выходит  $6+3+3=12$  колонок.

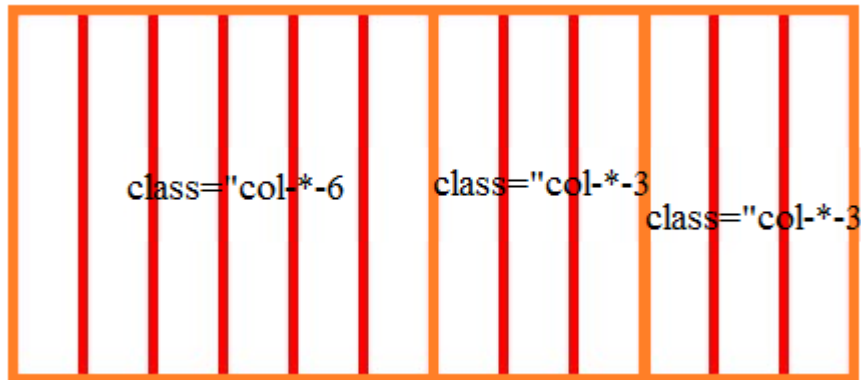


Рисунок 6 – Пример 3 блоков в одном ряду

Теперь разберёмся со звёздочками после col. Так как существует большое количество устройств с различным размером экранов, в Bootstrap они были разделены на 4 группы. Собственно вместо звёздочки нужно будет указывать необходимую группу. Рассмотрим эти группы:

- col-xs-\* – применяется для создание сетки с маленькими экранами, то есть телефоны;
- col-sm-\* – для устройств чуть большей ширины экрана, чем у телефона, например планшет;
- col-md-\* – средних размеры экранов, например, нетбуки;
- col-lg-\* – для больших экранов.

Таблица 5 – данные по этим группам.

Вид системы сеток	Маленький экран (<768px)	Небольшой экран (≥768px и <992)	Средний экран (≥992px и <1200px)	Большой экран (≥1200px)
Фиксированный макет (class="container")	Соответствует 100% ширине веб-клиента	750px	970px	1170px
Резиновый макет (class="container-fluid")	Соответствует 100% ширине экрана веб-клиента			
Префикс класса	class="col-xs-*"	class="col-sm-*"	class="col-md-*"	class="col-lg-*"
Максимальная ширина колонки Bootstrap для фиксированного макет (class="container")	Ширина веб клиента делится на 12	62px (750px / 12)	81px (970px / 12)	97px (1170px / 12)
Максимальная ширина колонки для резинового макета (class="container-fluid")	Ширина веб клиента делится на 12			
Внутренние отступы	15px по краям			

Важная особенность, если использовать класс class="col-sm-\*", то он будет применяться не только к небольшим экранам, но и к средним и большим экранам, но это действует в том случае, если не было указано "col-md-\*" и "col-lg-\*". То есть если в ряду будет один блок для всех экранов, то достаточно написать <div class="col-xs-12"> ... </div>, а не <div class="col-xs-12 col-sm-12 col-md-12 col-lg-12"> ... </div>.

Так же бывает необходимость убирать некоторые блоки, когда окно браузера становится меньше необходимой величины. Для этого используется класс `hidden-*`, где вместо звёздочки пишется префикс класса.

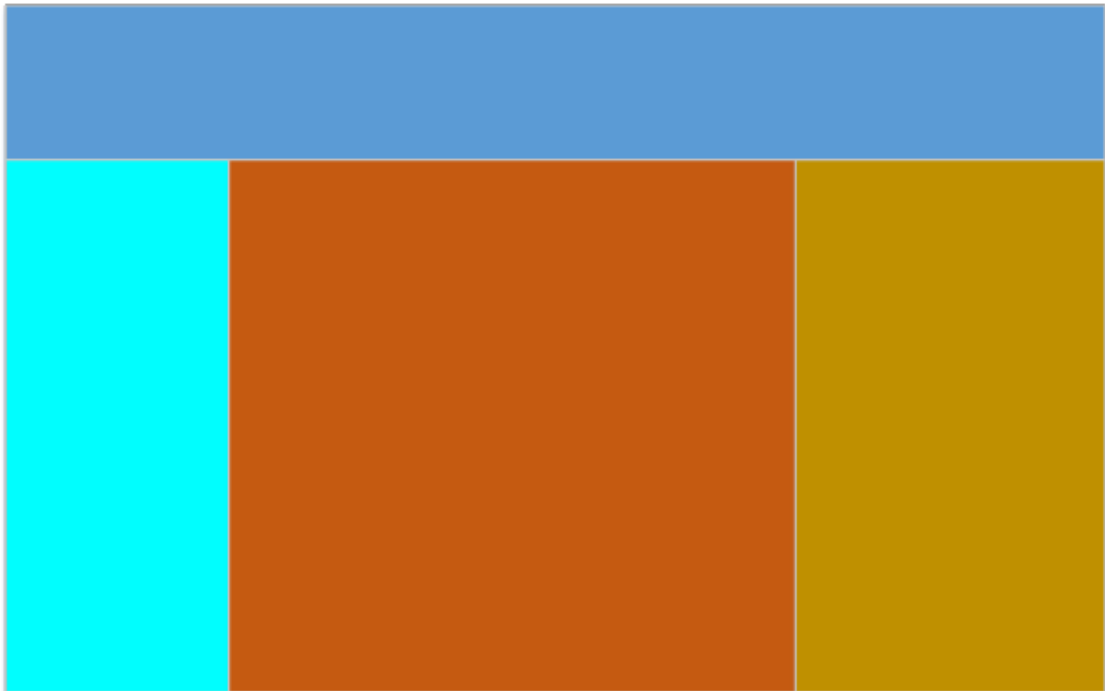
## ЗАДАНИЕ И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Сделать адаптивную вёрстку для трёх размеров экрана:

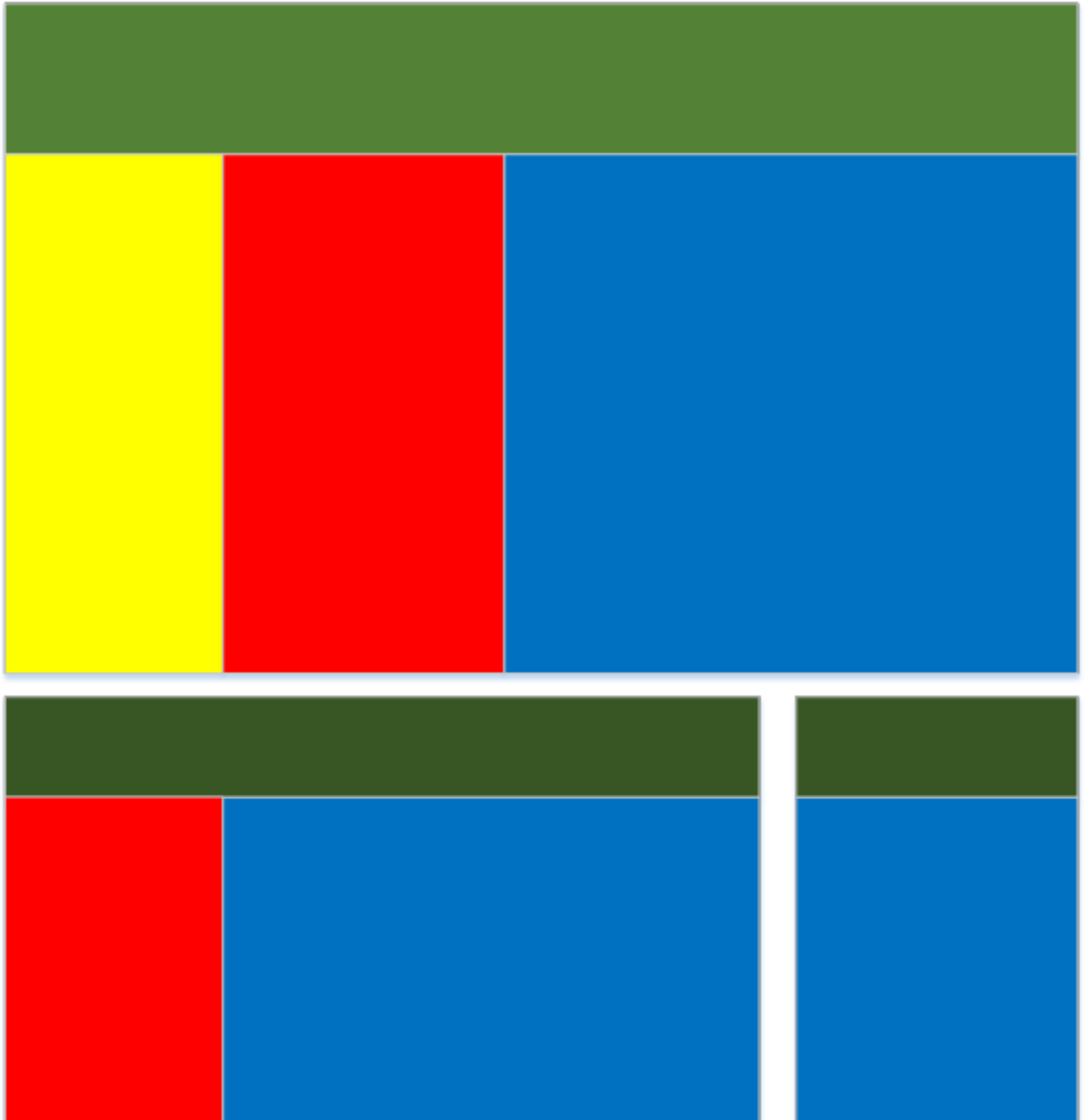
- средний и большой экраны
- небольшой экран
- маленький экран

Варианты заданий представлены ниже:

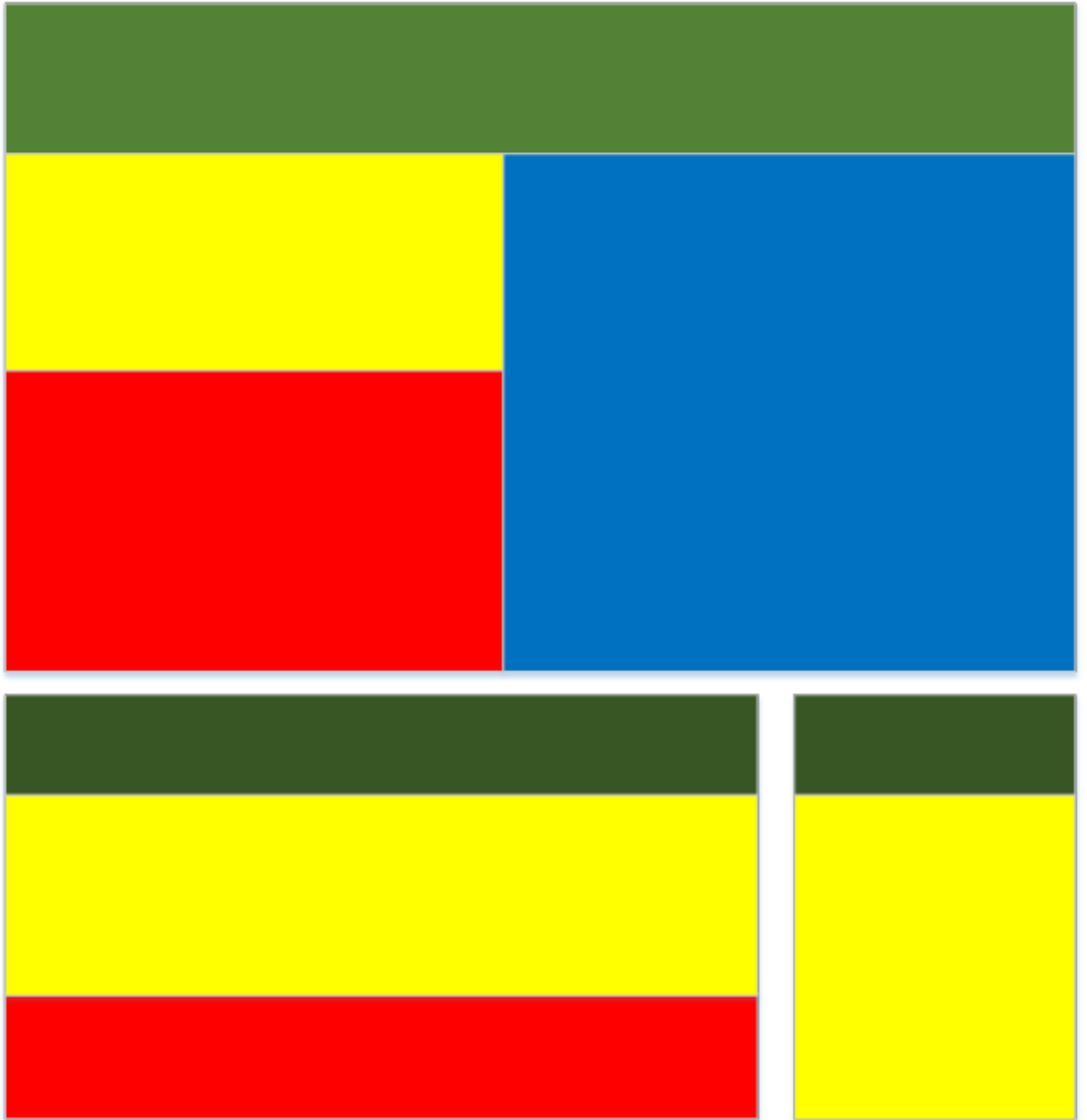
1 вариант



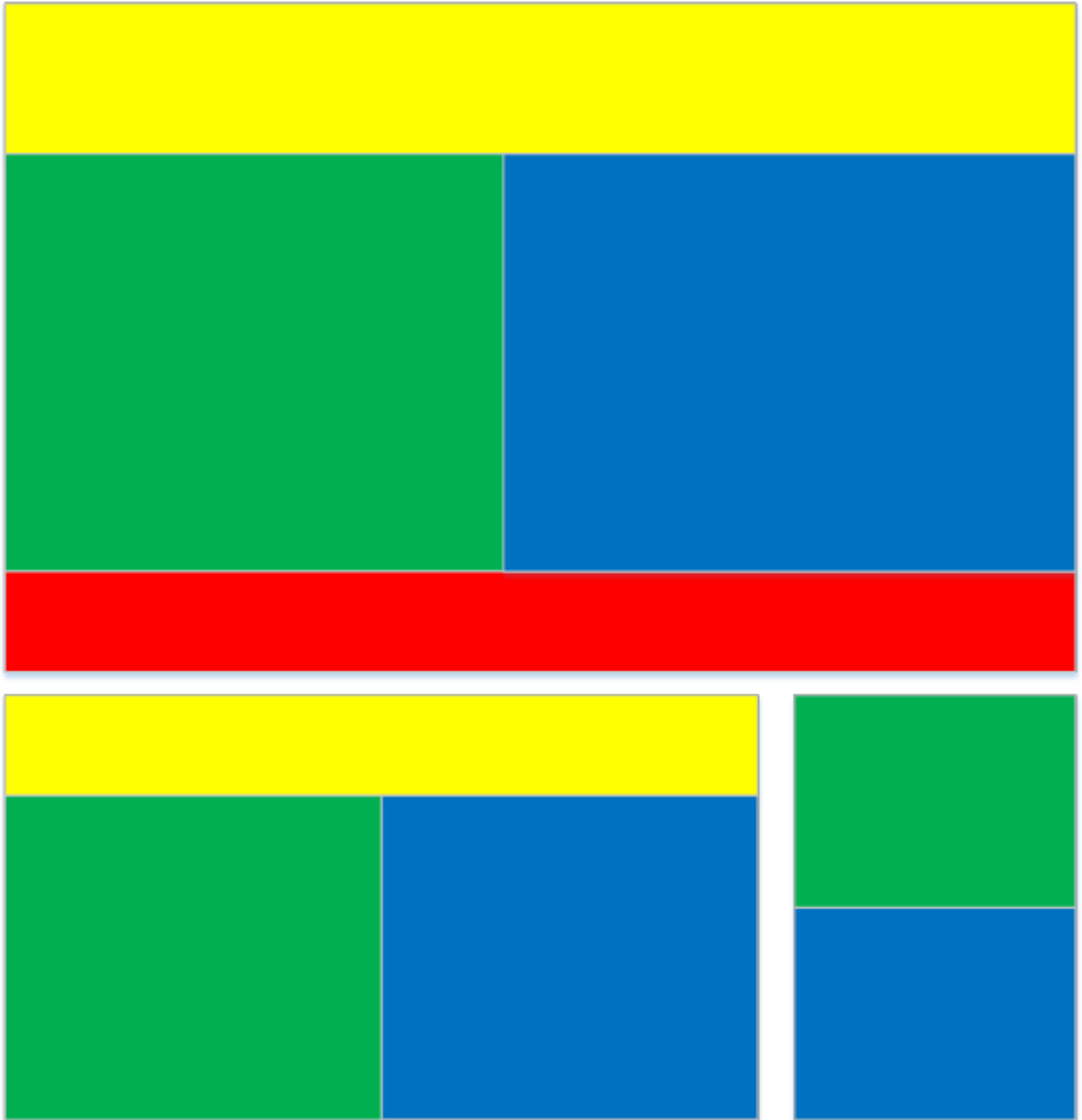
2 вариант



3 вариант

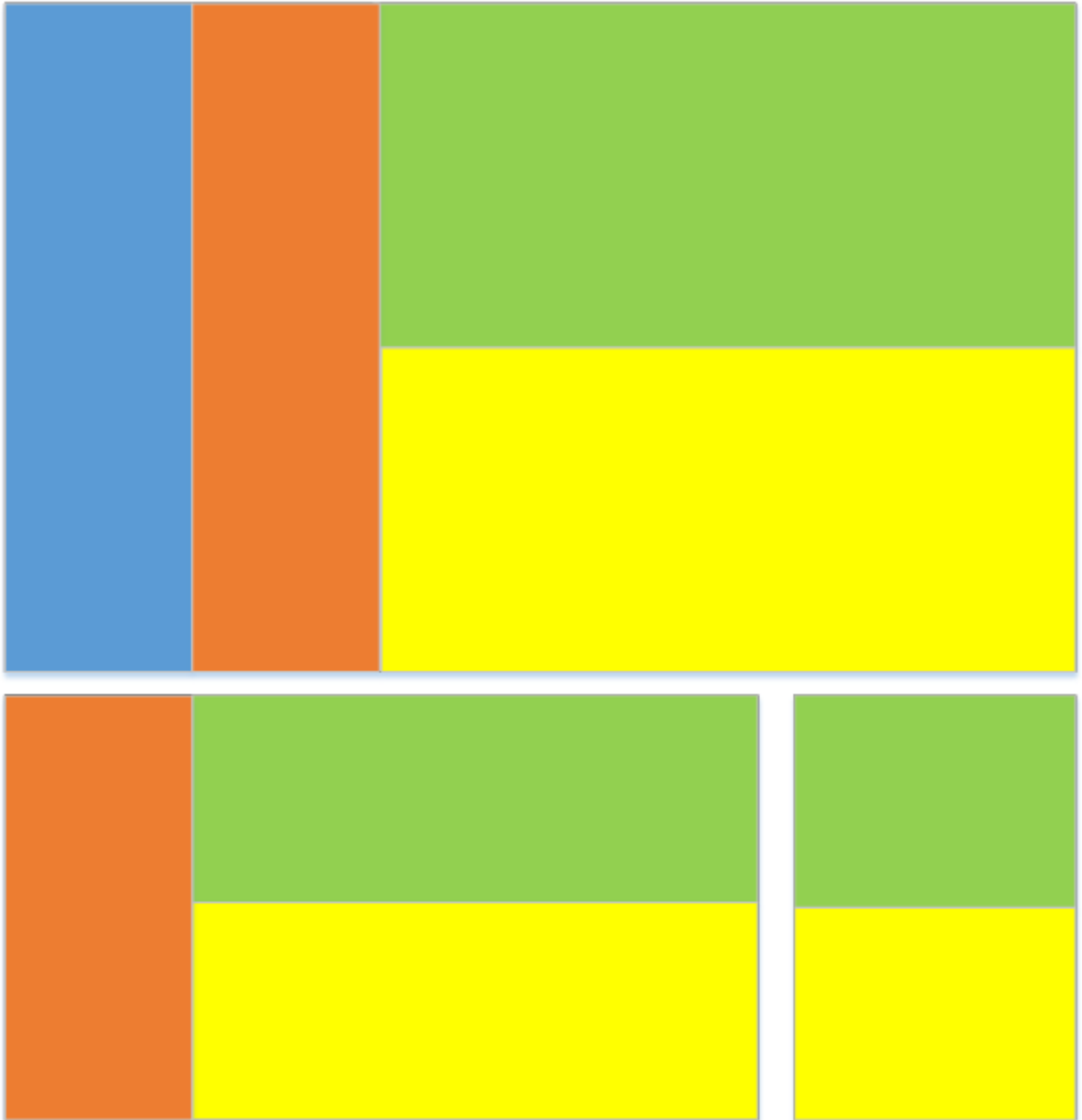


4 вариант

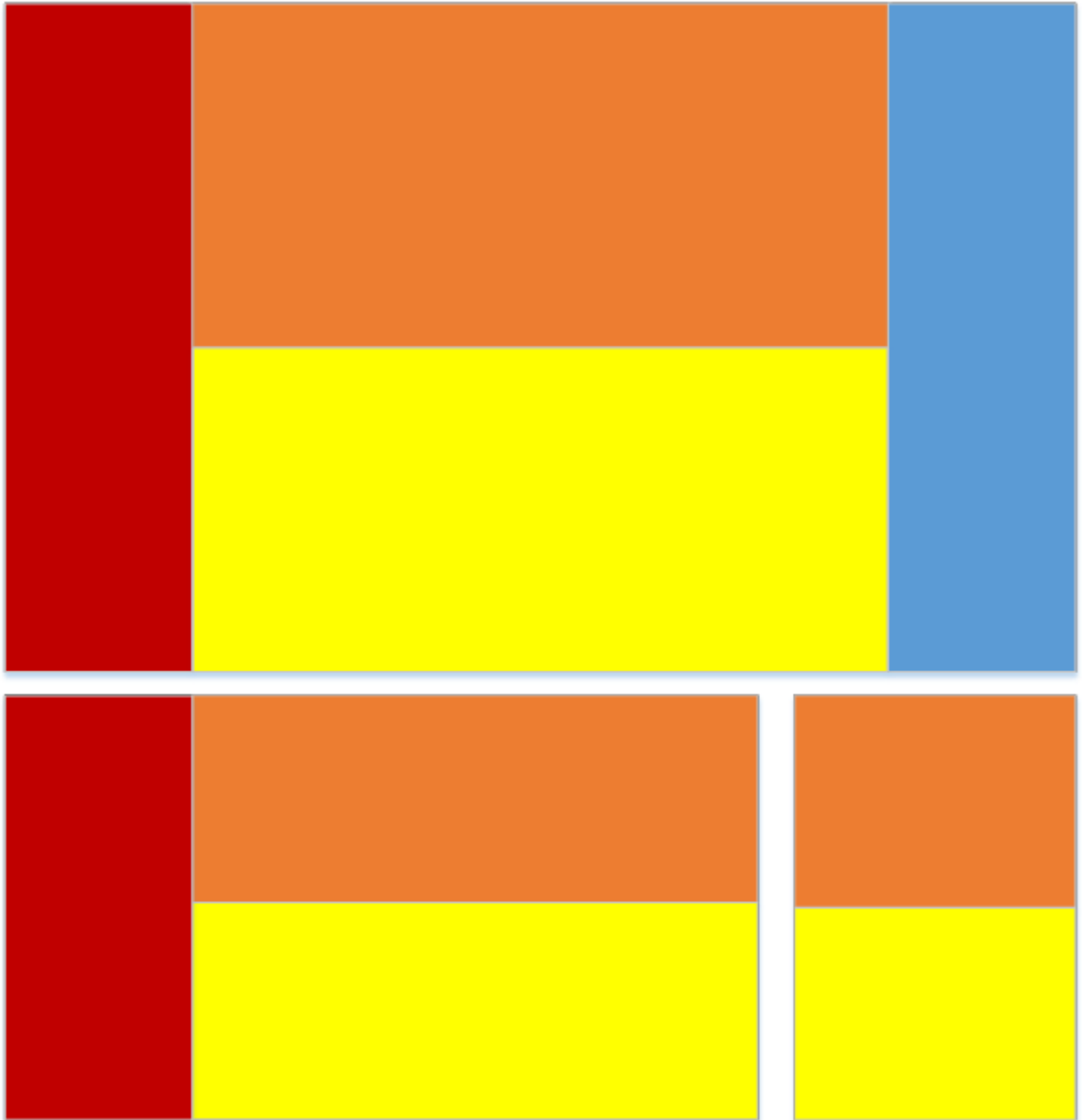




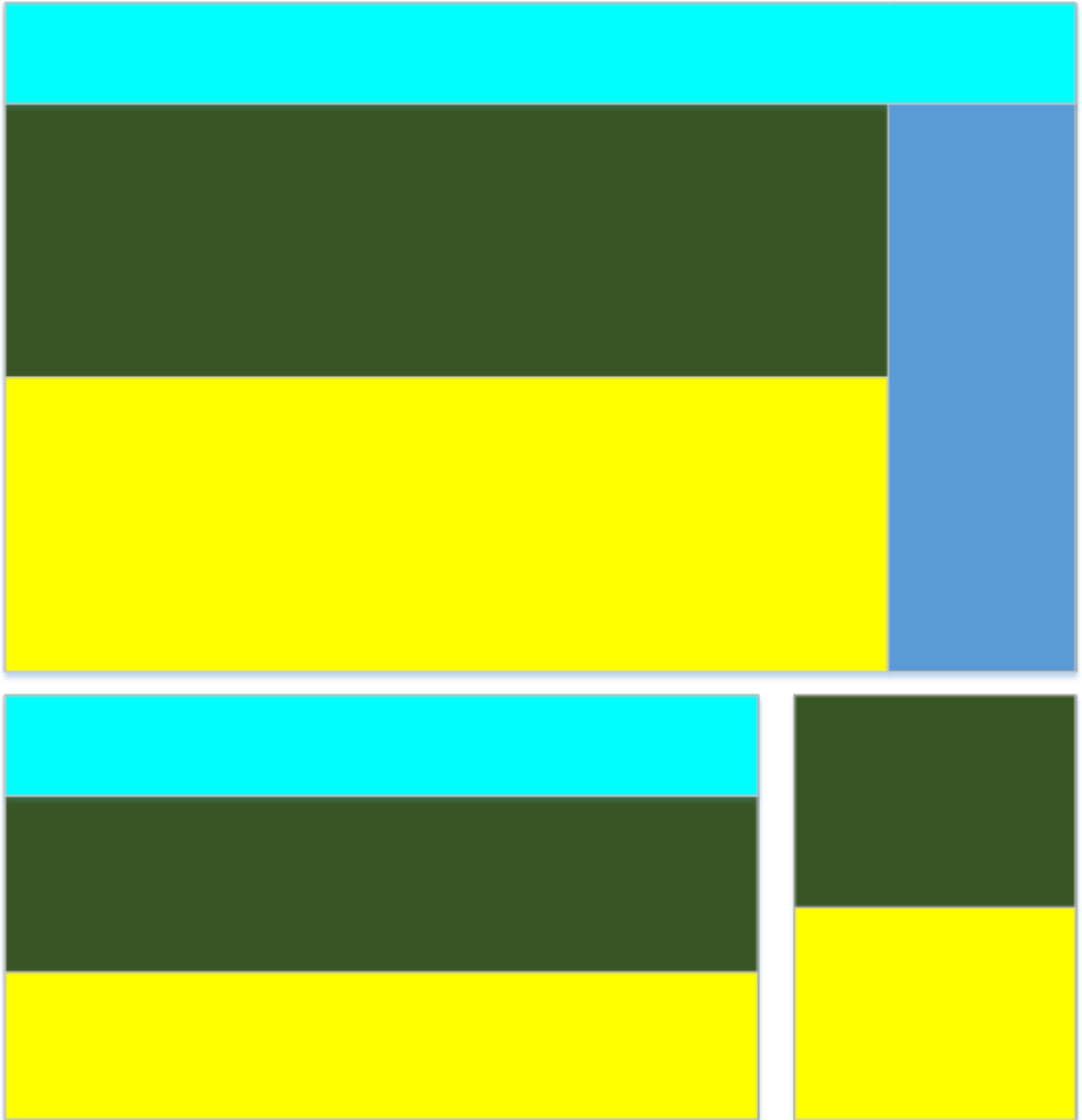
5 вариант



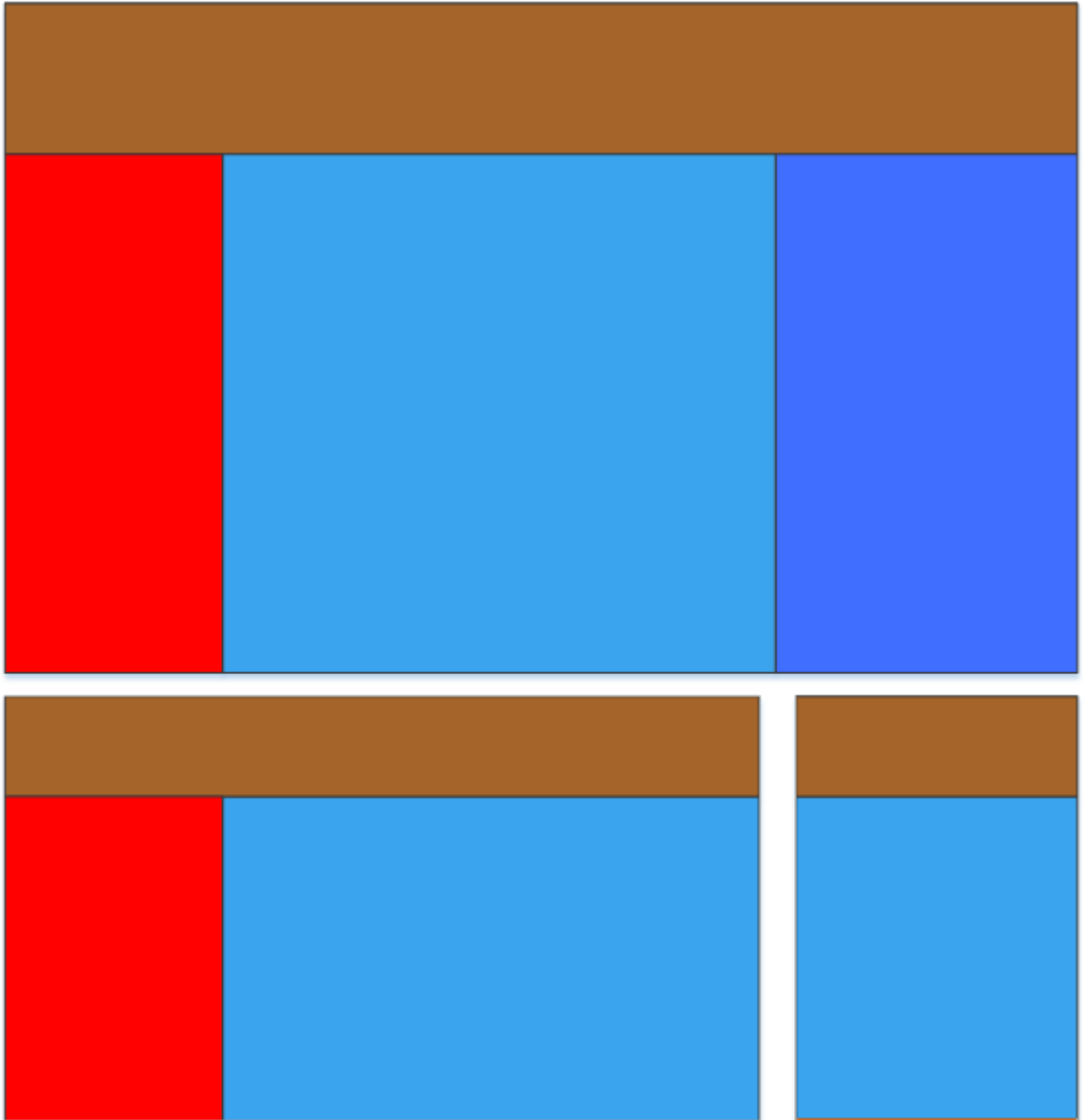
6 вариант



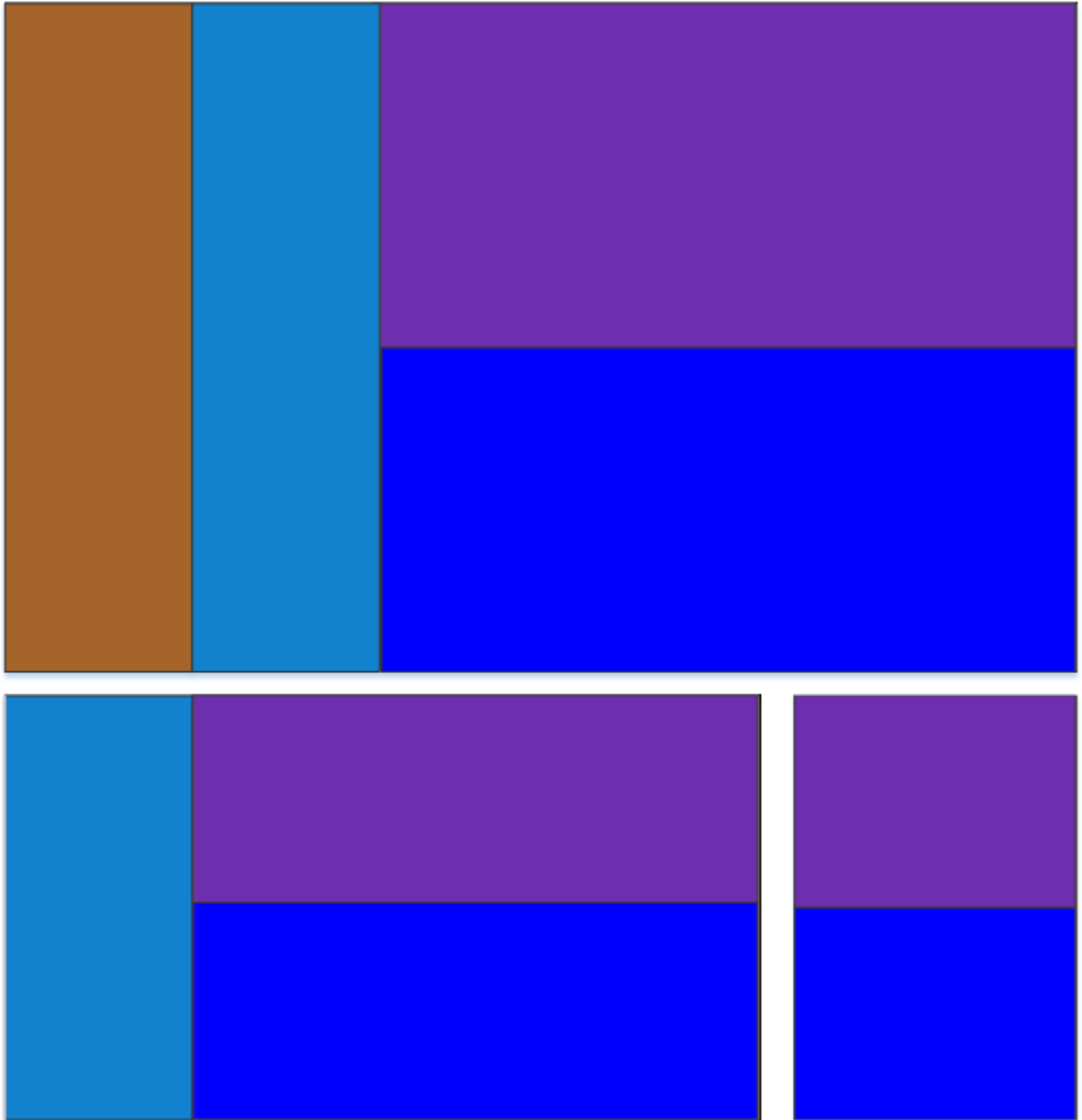
7 вариант



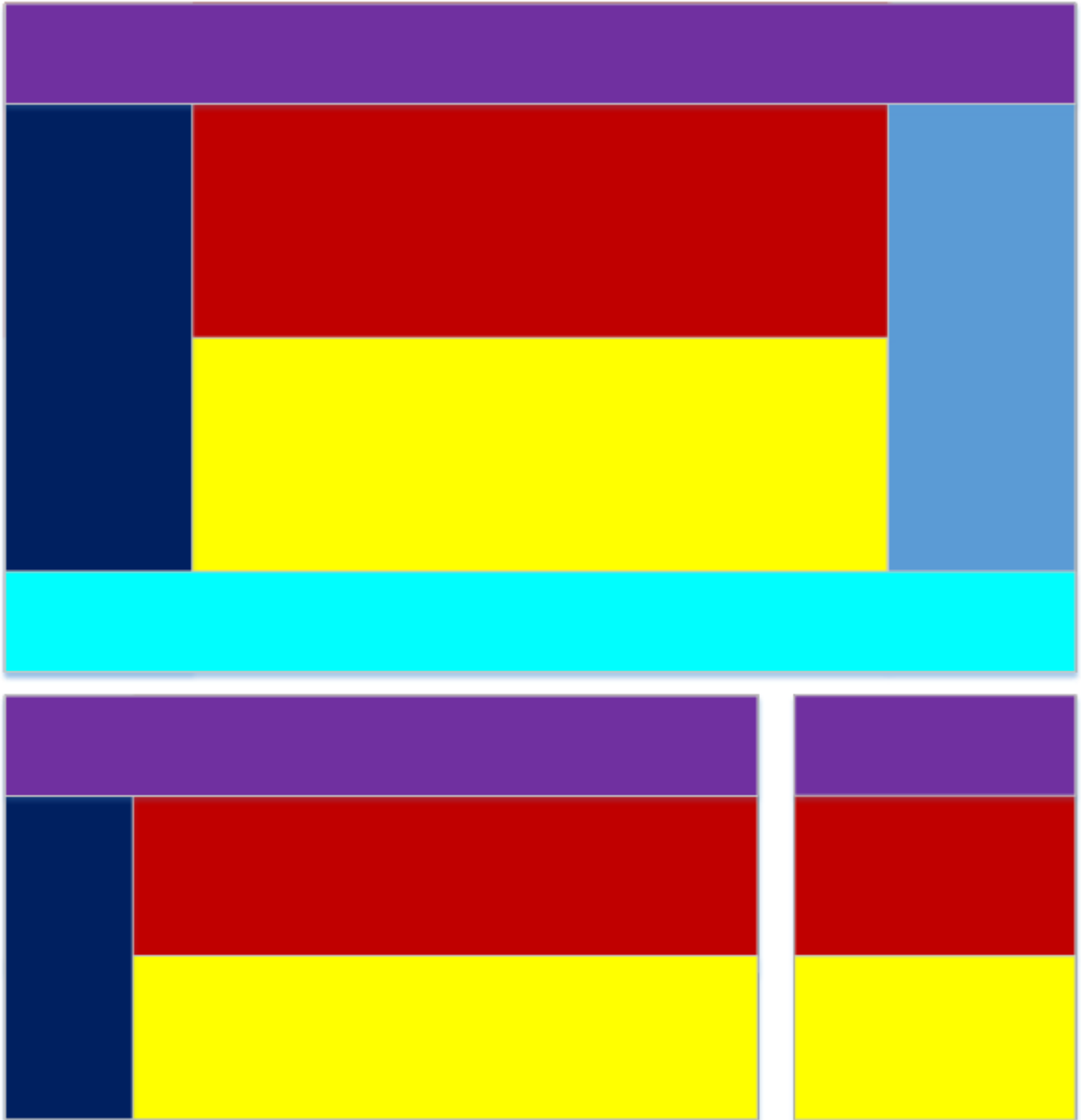
8 вариант



9 вариант



## 10 вариант



Рассмотрим на простом примере, как работает адаптивная вёрстка. Для этого сначала определимся, из каких блоков будет состоять сайт. Будем использовать четыре блока различной ширины. Для средних и больших экранов дизайн будет одинаковым.

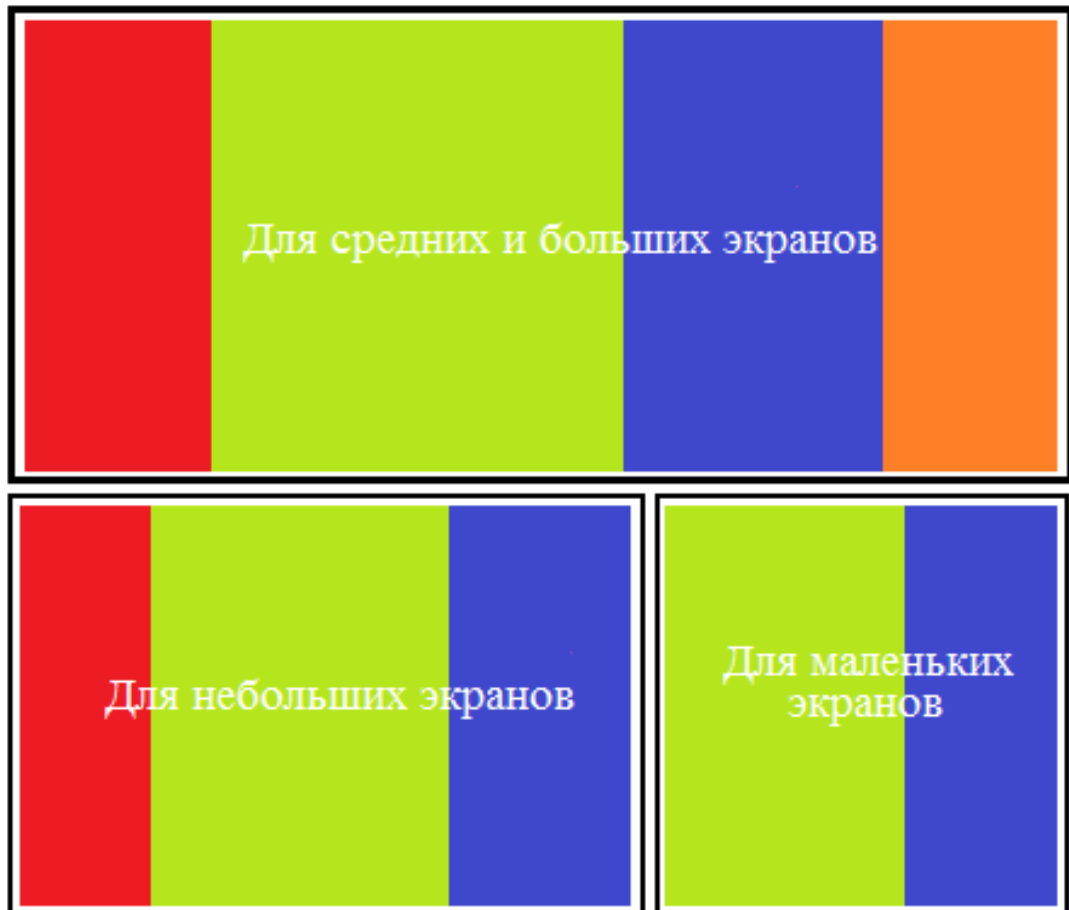


Рисунок 7 – Макет адаптивной вёрстки для разных устройств

Код выполнения данного примера для index.html представлен ниже:

```

<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>Bootstrap</title>
  <link href="css/bootstrap.css" rel="stylesheet">
  <link href="css/style.css" rel="stylesheet">
  <script src="js/bootstrap.js"></script>
</head>
<body>
<div class="container-fluid">
  <div class="row">
    <div class="col-md-2 col-sm-2 hidden-xs red"></div>
    <div class="col-md-5 col-sm-6 col-xs-8 green"></div>
  
```

```
<div class="col-md-3 col-sm-4 col-xs-4 blue"></div>
  <div class="col-md-2 hidden-sm hidden-xs orange"></div>
</div>
</div>
</body>
</html>
```

Код style.css

```
.red {
  background-color: red;
  height: 100vh;
}
.green {
  background-color: greenyellow;
  height: 100vh;
}
.blue {
  background-color: blue;
  height: 100vh; }
.orange {
  background-color: orange;
  height: 100vh;
```



**БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

- 1) Начало работы с Bootstrap [Электронный ресурс] \ Bootstrap Internet – <http://mybootstrap.ru/get-started>.
- 2) Основы Bootstrap [Электронный ресурс] \ Bootstrap Internet – <http://bootstrap-v4.ru/getting-started/introduction/>.
- 3) Изучение технологии Bootstrap [Электронный ресурс] \ Айти прогер Internet – <https://itproger.com/course/bootstrap>.
- 4) Буч Г. Объектно-ориентированный анализ и проектирование. – М.: Вильямс, 2008.
- 5) Леоненков А.В. Самоучитель языка UML. – СПб.: БХВ-Петербург, 2004.
- 6) Розенберг Д., Скотт К. Применение объектного моделирования с использованием UML и анализ прецедентов. – М.: ДМК Пресс, 2002.