## Практическая работа 2. Верстка одностраничного сайта

*Цель работы:* сверстать одностраничный сайт по графическому макету.

*Задачи работы:*

1. Осуществить макроверстку и микроверстку с помощью flexbox.
2. Использовать разные значения свойства position для верстки элементов сайта.
3. Научиться извлекать информацию о макете из графического редактора Figma.
4. Освоить инструментальное средство для проверки шаблона на соответствие графическому макету.
5. Сверстать одностраничный сайт.

*Инструменты:*Visual Studio Code, браузер Google Chrome, надстройка PerfectPixel.

**Задание 1. Работа с графическим макетом**

1. Откройте макет, представленный по ссылке <https://www.figma.com/file/lRvtIxgrx6UAS0YHPreOwI/task6> (рис. 103), его также можно найти на сайте <https://figma.info/> в разделе Шаблоны.
2. Сделайте себе дубликат этого макета и далее работайте с ним

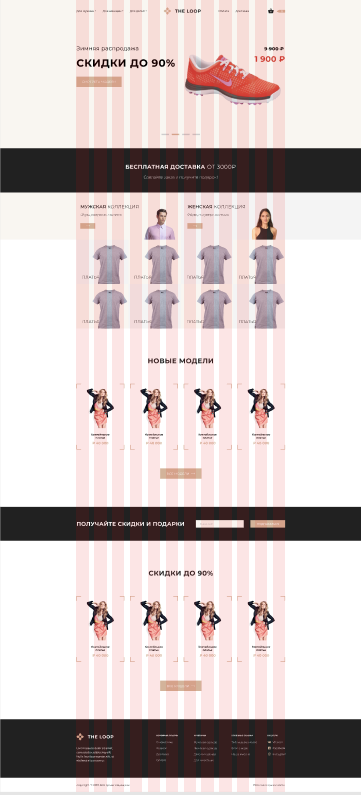


Рисунок 103 – Графический макет сайта

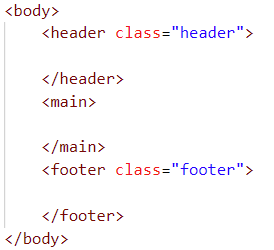
1. Как видно макет построен на основе 12-колоночной сетки. Проанализируйте макет.
2. Чтобы узнать параметры элементов, например, цвета и параметры шрифтом, необходимо:

* щелкнуть на элементе ЛКМ или найти его в палитре слоев (находится слева от рабочей области).
* перейти на вкладку Inspect (палитра справа) и просмотреть его свойства, в том числе и в виде CSS-свойств в разделе Code.

1. Определите базовые цвета макета и параметры основного текста.

**Задание 2. Верстка общей структуры макета**

1. Создайте новую папку проекта – loop.local-Фамилия Имя. В ней стандартные папки css и img.
2. Откройте папку loop.local в редакторе кода.
3. Создайте файл index.html и задайте его структуру.
4. Создайте файл style.css в папке css.
5. Подключите стилевой файл на html-страницу.
6. Создайте три основных блока – header, main и footer:



1. Создайте основные секции сайта. Всего будет 10 секций (рис. 104). Для каждой секции создайте обертку и сам контейнер. Первые две секции находятся внутри header, последние две – в footer, остальные – в main.

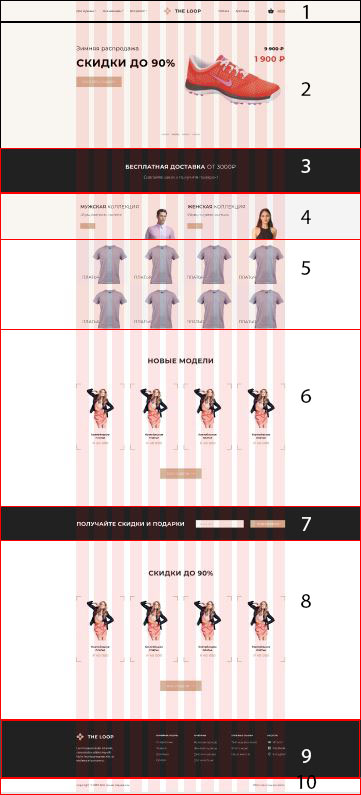
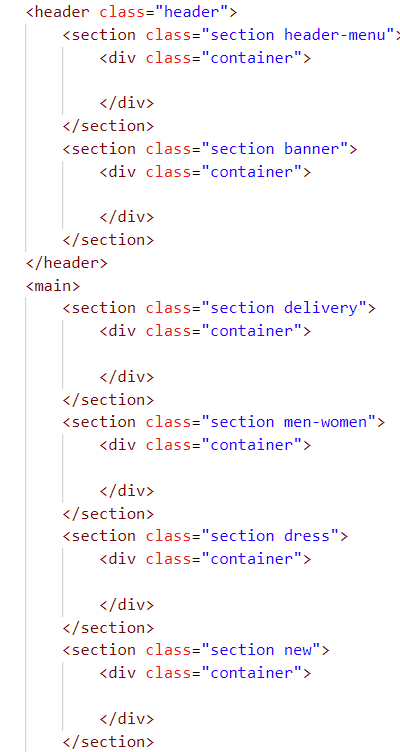
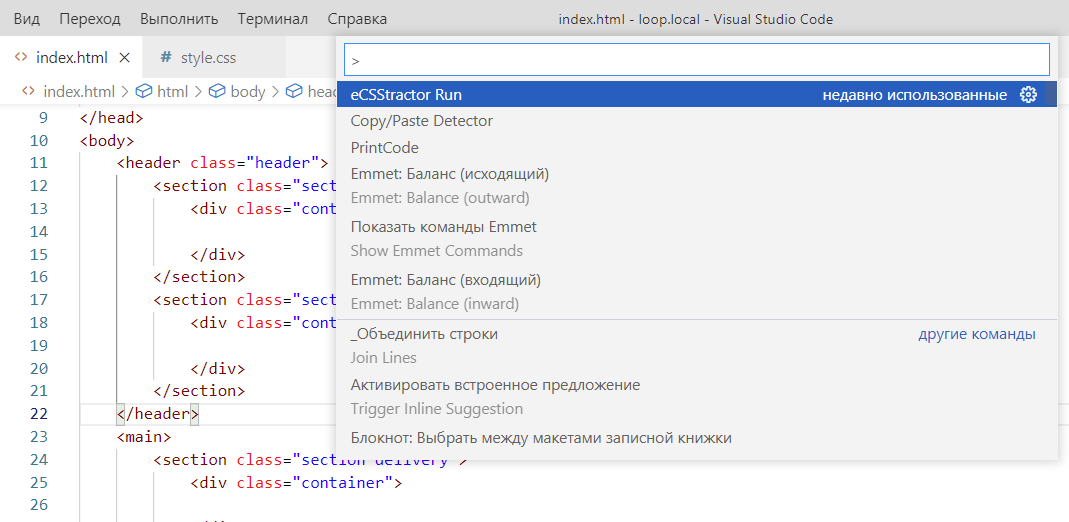


Рисунок 104 – Разделы сайта



…

1. Создайте остальные секции по аналогии.
2. Создайте классы в CSS-файле автоматически. Для этого:
   1. Установите расширение eCSStractor for VSCode. Данное расширение позволяет извлечь селекторы из HTML и создать их в файл CSS.
   2. Выделите весь HTML-файл.
   3. Запустите расширение с помощью Палитры команд (CTRL+SHIFT+P). При этом откроется палитра (рис. 105). Выберите eCSStractor Run.



* 1. Также можно найти эту команду в контекстном меню.

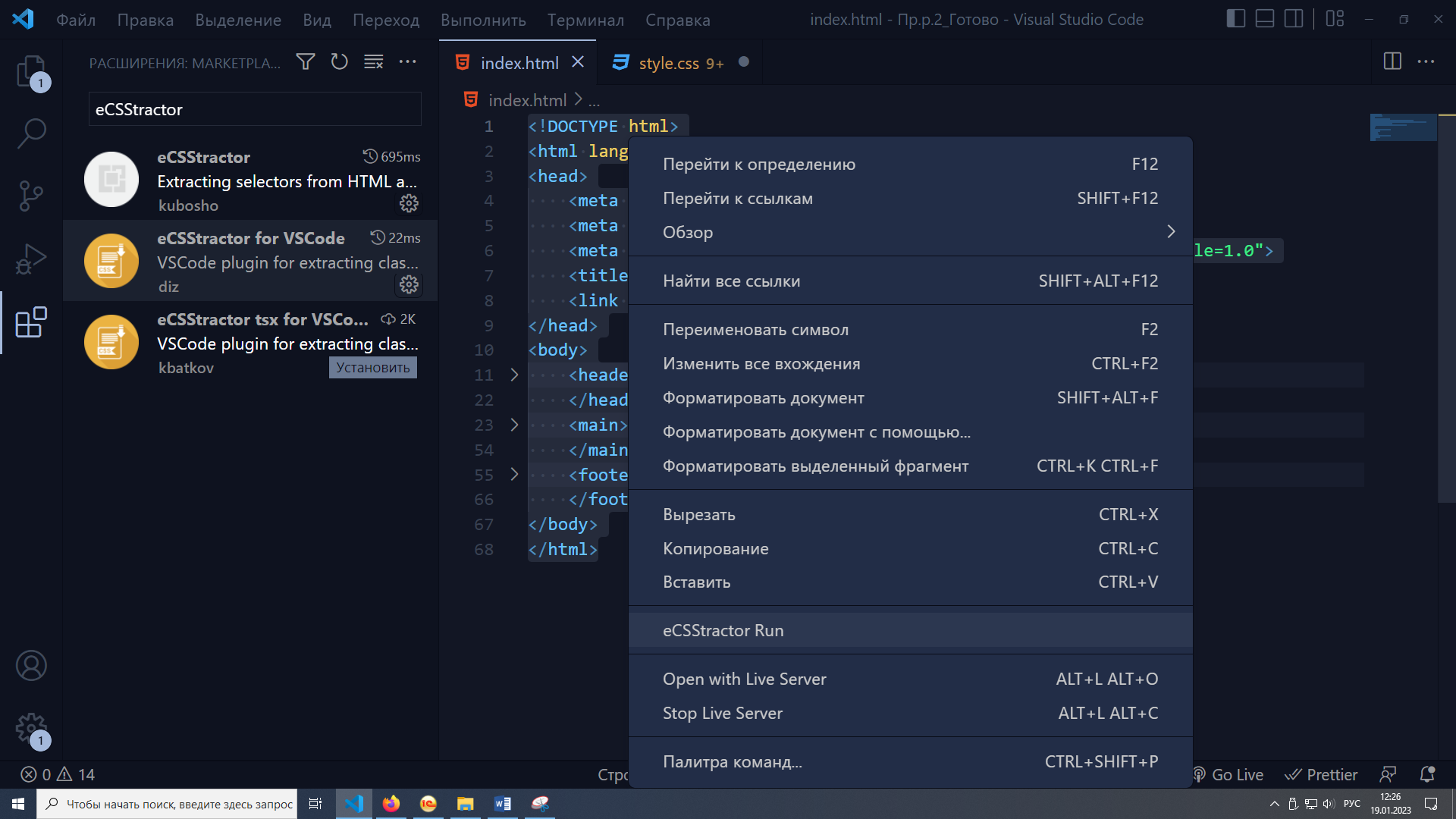
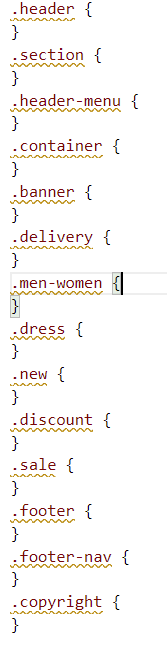


Рисунок 105 – Палитра команд

* 1. Классы скопированы. Теперь перейдите в CSS-файл и вставьте классы.
  2. Результат работы плагины eCSStractor:



1. Скачайте файл для нормализации стилей normalize.css с сайта <https://github.com/necolas/normalize.css/>.
2. Добавьте этот файл в папку style.
3. Подключите файл normalize.css на HTML-страниц до файла style.css.

**Задание 3. Установка базовых стилей**

1. Согласно спецификации CSS ширина блока складывается из ширины контента (width), значений отступов (margin), полей (padding) и границ (border). Аналогично обстоит и с высотой блока. Свойство box-sizing позволяет изменить этот алгоритм, чтобы свойства width и height задавали размеры не контента, а размеры блока.
2. Поэтому в самом начале файла style.css установите значениеbox-sizing:

html {

  box-sizing: border-box; /\* Ширина блока с полями \*/

}

\*,

\*::before,

\*::after {

  box-sizing: inherit; /\* Наследует значение родителя\*/

}

1. Для селектора body задайте свойства:

min-width: 1100px;

margin: 0;

padding: 0;

Минимальную ширину страницы берем из графического макета.

1. Выбор, оптимизация и подключение шрифтов:
   1. В графическом макете выделите поочередно разные текстовые блоки и в инспекторе свойств (слева) определите параметры шрифтов. Везде используется шрифт Montserrat разной толщины и размера.
   2. Перейдите на сервис Google Fonts найдите нужный шрифт.
   3. Скачайте шрифты (кнопка Download All)
   4. Перейдите на сервис Font Squirrel в раздел Generate - <https://www.fontsquirrel.com/tools/webfont-generator>
   5. Загрузите только нужные типы шрифта для макета (рис. 106).

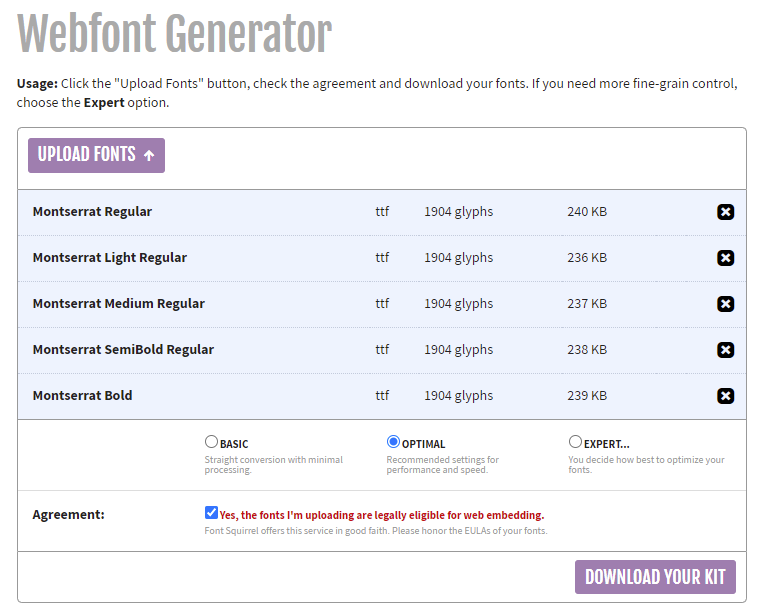


Рисунок 106 – Оптимизация шрифтов на Font Squirrel

* 1. Выберите режим Expert и в настройках в разделе Subsettings выберите режим Custom Subsettings и выберите параметры как на рисунке 107.

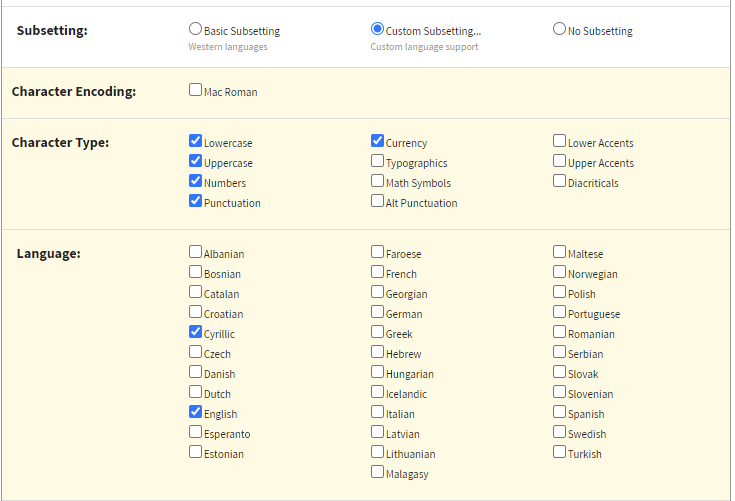


Рисунок 107 – Ручная настройка параметров оптимизации шрифта

* 1. Нажмите кнопку Download Your Kit и дождитесь скачивания шрифтов.
  2. В проекте создайте папку fonts и разместите в ней все нужные шрифты в формате woff и woff2.
  3. В архиве с шрифтами есть файл stylesheet.css скопируйте из него все содержимое и вставьте в самое начало файла style.css.
  4. Измените пути к файлам шрифтов в url (добавьте ../fonts/).

1. Для селектора body задайте свойства шрифтов по умолчанию. Информацию необходимо брать из графического макета:

font-family: "montserratregular", "Arial", sans-serif;

font-size: 14px;

line-height: 22px;

font-weight: 400;

color: #000;

1. Напишите пробный текст в файле index.html и проверьте работы стилей.
2. С помощью стилей уберите у всех ссылок подчеркивание.
3. У всех селекторов классов container задайте стили:

position: relative;

max-width: 1110px;

margin: 0 auto;

**Задание 4. Верстка главного меню сайта**

* + 1. Задайте стили для класса header:

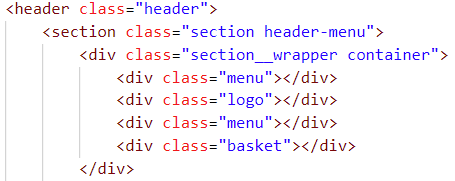
min-height: /\*взять из макета\*/;

background-color:/\*взять из макета\*/;

* + 1. Для селектора header-menu задайте стили:

min-height: 90px;

* + 1. Добавьте блоку с главным меню класс section\_\_wrapper и четыре блока div:

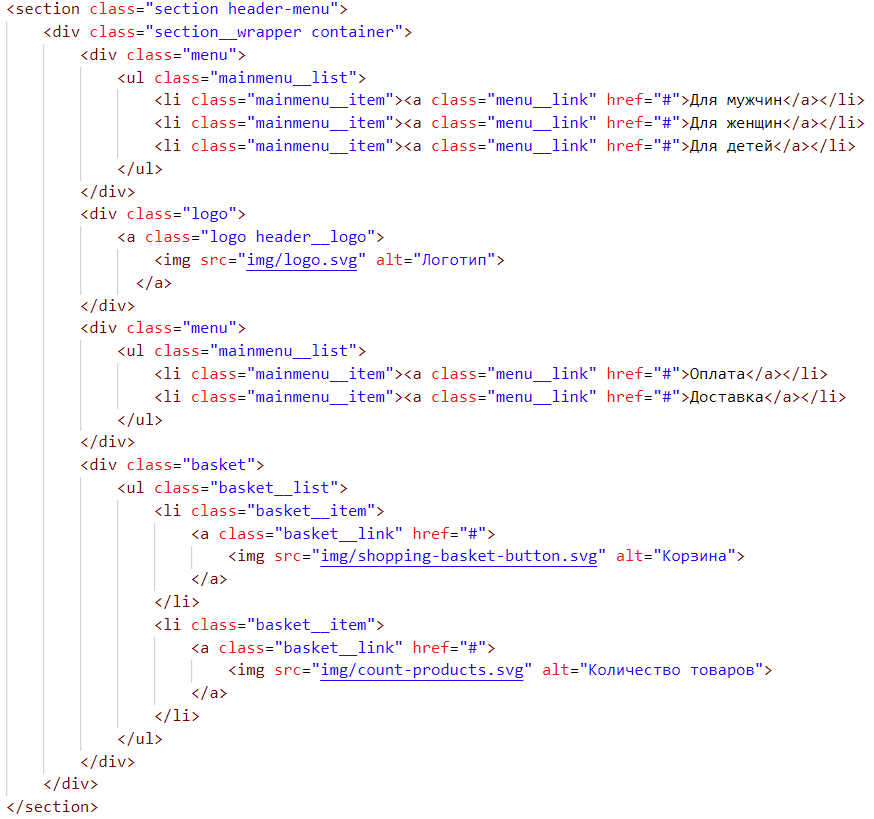


* + 1. В стилях задайте правила для section\_\_wrapper:

display: flex;

flex-direction: row;

* + 1. Добавьте HTML-код:



Используйте экспорт картинок из графического макета.

* + 1. Просмотрите результат (рис. 108).

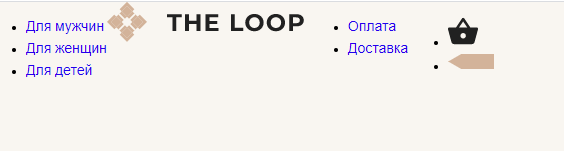


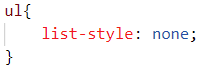
Рисунок 108 – Наполнение меню контентом

* + 1. С помощью стилей сделайте все списки flex-контейнерами (стили для классов у элементов ul). Результат работы должен быть как на рисунке 109.



Рисунок 109 – Списки меню стали flex-контейнерами

* + 1. У всех списков на странице уберите маркеры:



* + 1. С помощью выравнивания (используйте flex-свойства) и отступов приведите меню к виду как на рисунке 110.
    2. Добавьте иконки выпадающего списка у некоторых пунктов меню (согласно макету).
    3. Задайте стили для текста пунктов меню.



Рисунок 110 – Результат верстки главного меню

**Задание 5. Верстка баннера**

Баннер, находящийся сразу после главного меню и является слайдером, добавим ему управляющие кнопки (рис. 111). Выделим следующие зоны: зона слайда, зона кнопок навигации по слайдам и кнопки «вперед» и «назад».

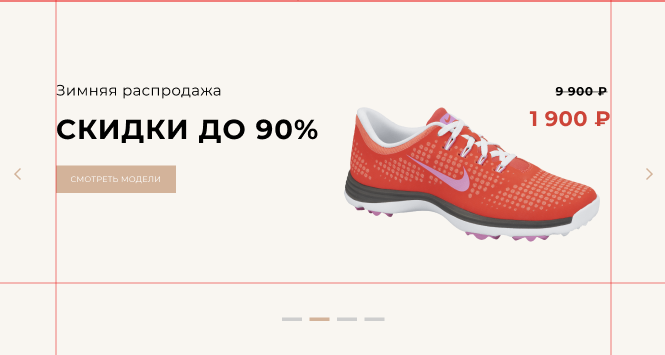
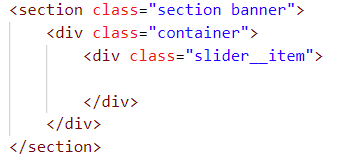
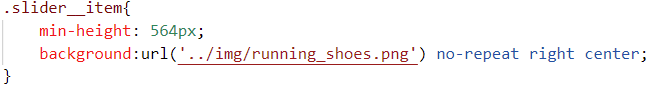


Рисунок 111 – Структура слайдера

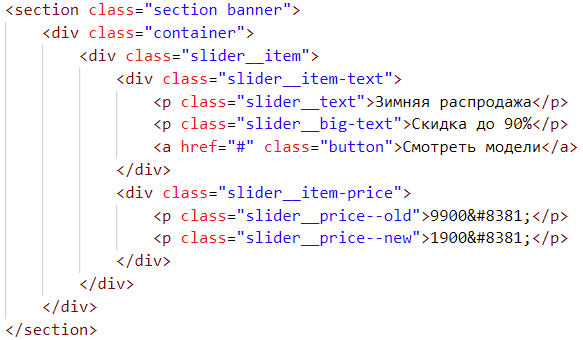
* + 1. Создайте дополнительный блок с классом slider\_\_item:



* + 1. У слайда будет фоновая картинка (в качестве примера, кроссовок). Задайте изображение кроссовка в качестве фоновой картинки и параметры ее отображения у созданного ранее класса:



* + 1. Внутри slider\_\_item разместите текстовую информацию и кнопку согласно макету. Для этого можно сделать блок с классом slider\_\_item flex-контейнером с двумя flex-элементами внутри:



* + 1. Оформите внешний вид этих блоков с помощью стилей:



Результат работы представлен на рисунке 112.

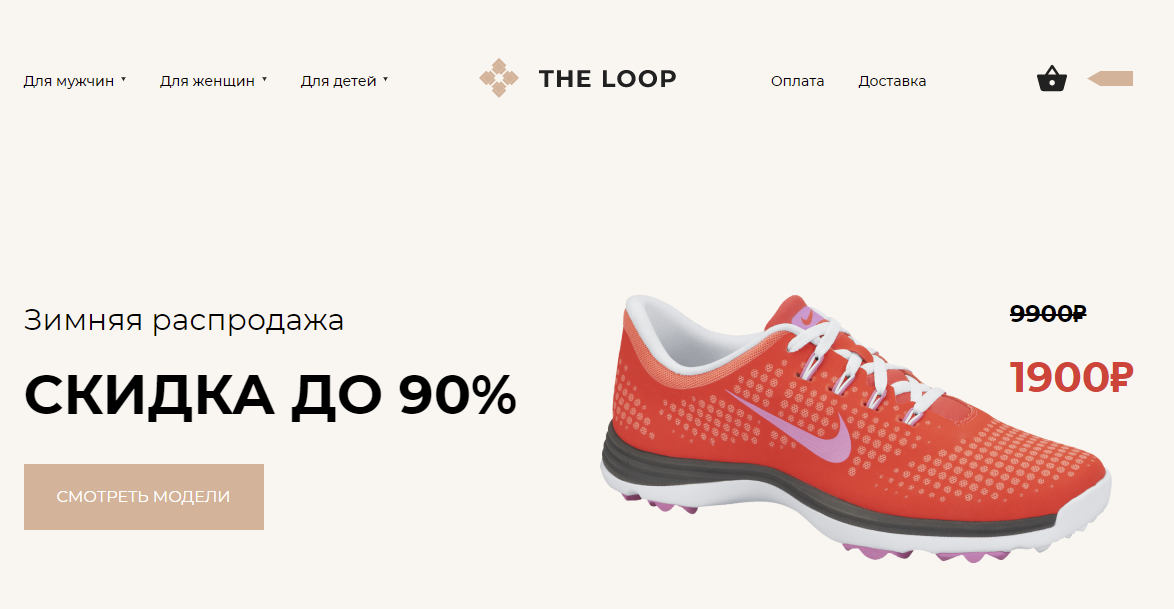


Рисунок 112 – Создание баннера

**Задание 6. Верстка управляющих кнопок слайдера с помощью абсолютного позиционирования**

Управляющие кнопки (вперед и назад) должны располагаться слева и справа от слайдера, но за его пределами (рис. 111). Для подобного позиционирования элементов используем position:absolute.

* + 1. Добавьте необходимые блоки в раздел слайдера:

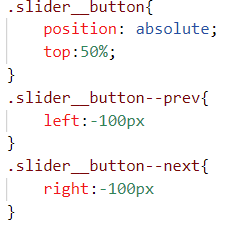


* + 1. Кнопки появятся на странице, но не в том месте (рис. 113).



Рисунок 113 – Добавление кнопок управления слайдером

* + 1. У обертки слайдера (блок slider\_\_wrapper) установите свойство position:relative, чтобы кнопки позиционировались относительно него.
    2. А для кнопок задайте стили:



* + 1. Просмотрите результат (рис. 114).



Рисунок 114 –Изменение положения кнопок слайдера

* + 1. Измените свойство для кнопок, чтобы они соответствовали макету.

**Задание7. Закрепление панели меню вверху окна браузера**

С помощью position:fixed закрепим главное меню вверху окна браузера.

* + 1. Для класса header-menu дополнительно к заданным CSS-правилам задайте еще следующие:
* позиционирование – фиксированное,
* ширину – 100%,
* позицию сверху – 0 (top:0),
* z-index: 100 (это свойство задает глубину расположения элемента, положительное значение позволяет переместить меню на передний план страницы).
  + 1. Свойство position:fixed выведет элемент из потока объектов и освободит его место на страницы. Поэтому следующий за ним блок banner поднялся к верху окна. Для устранения этого эффекта задайте внешний отступ у класса banner равный высоте главного меню (90px).
    2. Просмотрите результат работы (рис. 115). Уменьшите размер браузера по вертикали и использую скролл-бар поперемещайтесь по странице.



Рисунок 115 – Фиксация панели главного меню вверху окна сайта

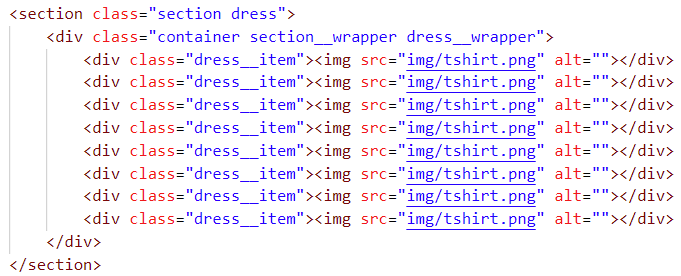
**Задание 8. Верстка пятой секции (dress)**

В этом задании с помощью flexbox-ов будет сверстан раздел слайда, представленный на рисунке 116. Сама секция станет flex-контейнером, в котором находятся восемь flex-элементов. Если взглянуть на макет внимательно, то можно заметить, что фон у блоков чередуется – #f4f4f4 и #f9f6f1.

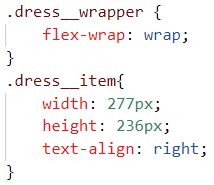


Рисунок 116 – Макет секции сайта (dress)

* + 1. Экспортируйте из макета изображение футболки.
    2. Добавьте код в index.html:



* + 1. Определите стили для новых классов:

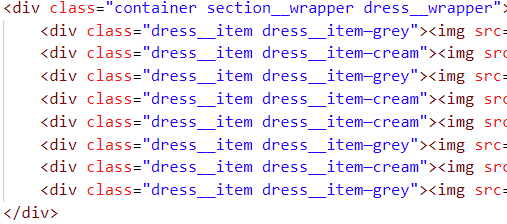


* + 1. Уменьшите размер картинки.
    2. Просмотрите результат (рис. 117).



Рисунок 117 – Секция dress

* + 1. Измените фоновые цвета, для этого задайте у соответствующих блоков классы dress\_\_item--grey и dress\_\_item--cream:



* + 1. Задайте стили для этих классов для определения фонового цвета блока. Для dress\_\_item--grey – #f4f4f4, а для dress\_\_item--cream – #f9f6f1.
    2. С помощью абсолютного позиционирования выровняйте картинку по нижнему краю, а также добавьте надпись. Не забудьте установить у класса dress\_\_item свойство position:relative.
    3. Просмотрите результат (рис. 118).



Рисунок 118 – Результат верстки секции dress

**Задания для самостоятельного выполнения**

* + 1. Сверстайте все остальные секции и элементы страницы в соответствии с графическим макетом.
    2. С помощью расширения PerfectPixel в Google Chrome проверьте соответствие сверстанной страницы графическому макету.
    3. С помощью псевдо класса :hover (а также :active и :visited) добавьте интерактивность элементам страницы – ссылкам и кнопкам. Также можно добавить интерактивные эффекты отlельным блокам на странице, на ваше усмотрение.