Оглавление

[С чего начать? 1](#_Toc180831108)

[Шаблон структуры html-документа 2](#_Toc180831109)

[Родительский и дочерний элементы 3](#_Toc180831110)

## С чего начать?

1. Создаем папку, в которой будут храниться все ваши документы.

2. Запускаем среду разработки Visual Studio Code.

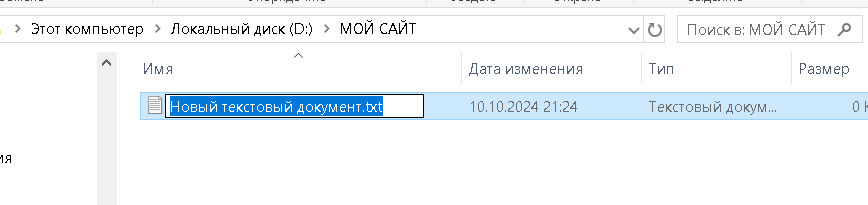
3. Жмем в левом верхнем углу File

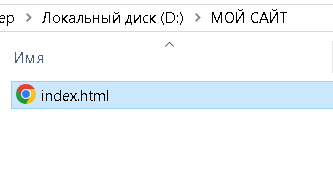
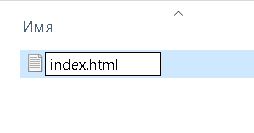
4. В меню выбираем Open Folder, выбираем свою созданную папку и жмем «Выбор папки»

3. Создаем файлы с расширением html и css (2 варианта)

Первый вариант

а) Создаем вручную текстовый файл в папке, даем имя и меняем расширение .txt на .html.

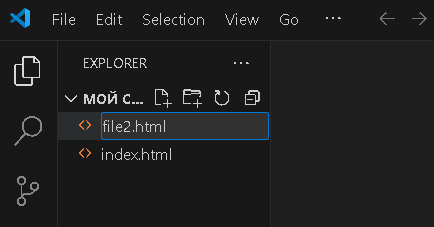
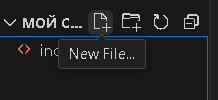




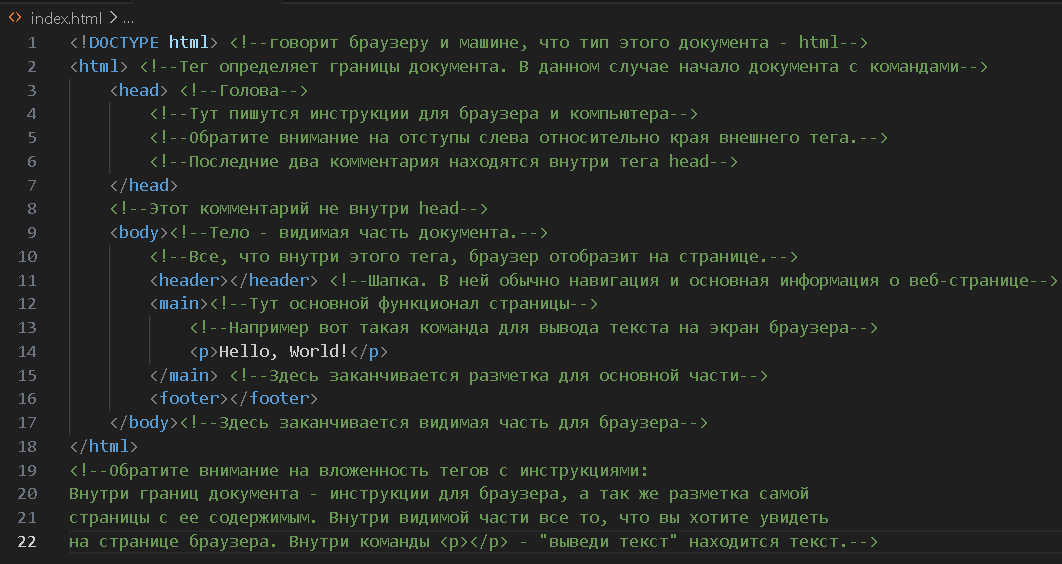
Второй

а) Жмем на иконку файла с плюсиком в меню рядом с названием папки

б) Даем ему название с расширением html



## Шаблон структуры html-документа



Начинайте писать html-документ с родительских элементов, а внутри уже прописывать дочерние. И так далее.

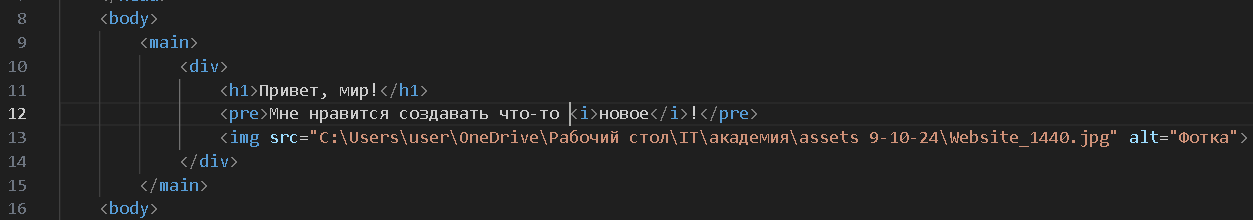
Для 1 html-документа существует только одна «голова» и одно «тело». 1 человек имеет только 1 голову и тело, html-документ тоже. Так же и с шапкой, подвалом, main.

Чтобы не было проблем с отступами, пишите открытый тег, Visual Studio Code сам автоматически пропишет вам закрытый тег, если закрытый тег у элемента вообще есть. Если у вас внутри тега будут какие-то еще элементы (теги), то поставьте курсор между открытым и закрытым тегом и нажмите Enter. В этом случае курсор автоматически окажется внутри тегов элемента с нужным отступом. И каждый раз, когда вы захотите добавить в родительский тег новый дочерний элемент, ставьте курсор в самый конец элемента (после последнего > в строке) и нажимайте Enter.

Любой компилятор читает ваш код, ваши команды по порядку, так и браузер.

Атрибуты всегда прописываются внутри открытого тега после его названия. Значение атрибута помещается в двойные кавычки.

## Родительский и дочерний элементы



Родительский элемент тот, который внешний, дочерний тот, который внутри.

Body родительский для main, main дочерний для body.

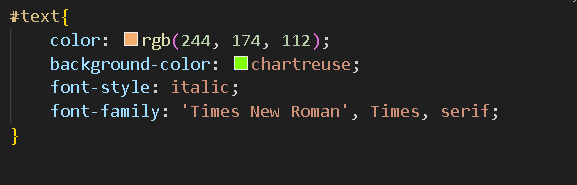
Main дочерний для div, div дочерний для main.

Div родительский для h1, pre, img, эта троица – дочерние для div.

i дочерний для pre, pre – родительский для i.

## CSS-синтаксис

Так выглядит синтаксис и структура в css-документе.



#name – обращение к id html-элементов

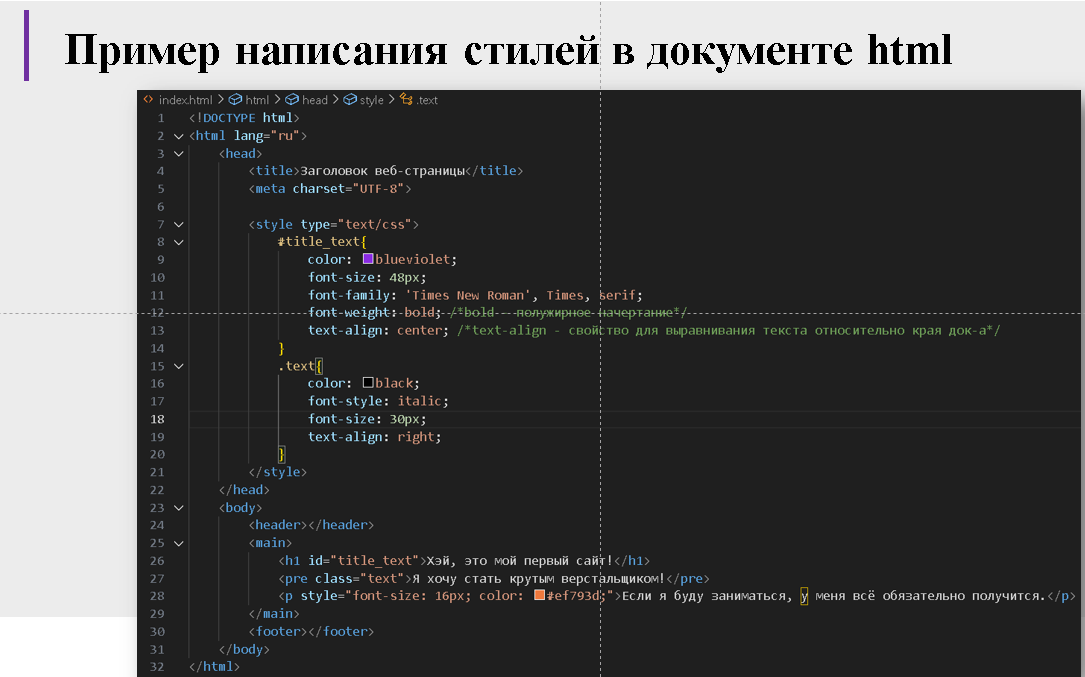
.name – обращение к class html-элементов

name – обращение к тегу html-документа

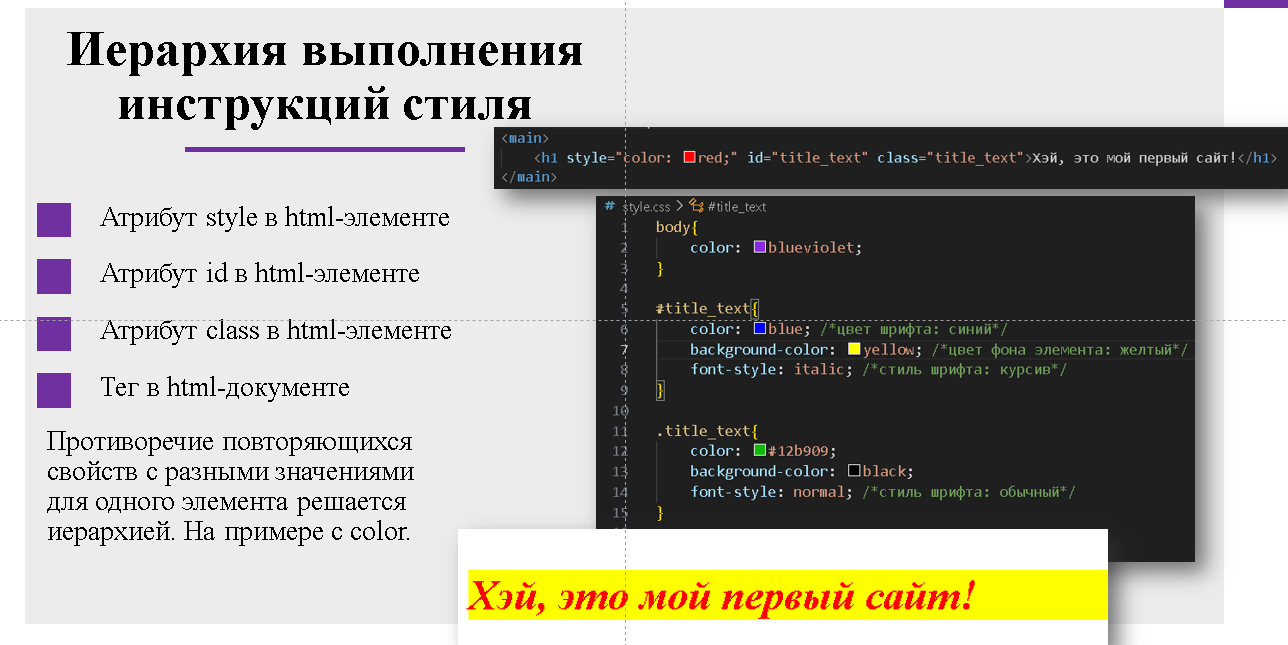
**Правила:**

1. Набор инструкций стиля в css-документе для элемента html-документа должен иметь название
2. Набор инструкций стиля в css-документе для элемента html-документа должны располагаться внутри фигурных скобок
3. Свойство от значения свойства должны разделяться двоеточием
4. Инструкция должна завершиться ; Это знак для машины, что инструкция завершена, а дальше будет описание новой инструкции
5. Если внутри скобок набора свойства повторяются, то будет выполняться та, которая идет раньше.
6. Наименование инструкций в css-документе должно совпадать с наименованием id или class или тега элемента в html-документе
7. У одного элемента id и class могут иметь одинаковые наименования
8. У нескольких элементов может быть одинаковое наименование id и class, тогда для разных элементов будут выполняться одинаковые инструкции стиля
9. В css-документе если у нескольких наборов инструкций id или class или тега повторяется свойство (с разными значениями), то выполнятся те «повторяющиеся» инструкции, которые указаны раньше или указаны в наборе, который выше по иерархии (иерархия в следующем слайде).
10. Если внутри скобок повторяется свойство, то выполняется то, которое указано раньше.
11. Для тэга и атрибута style в html-документе синтаксис тот же.

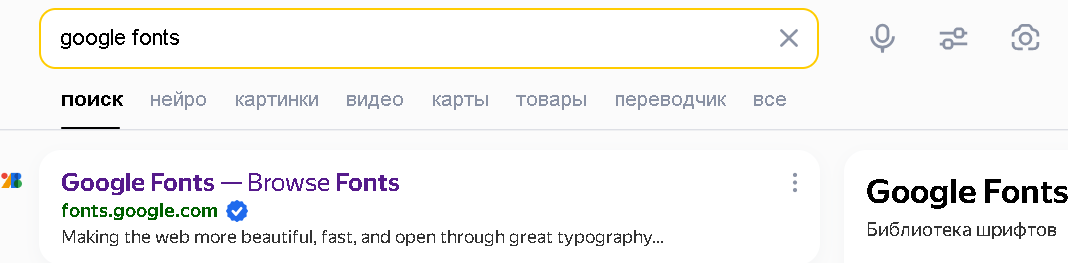
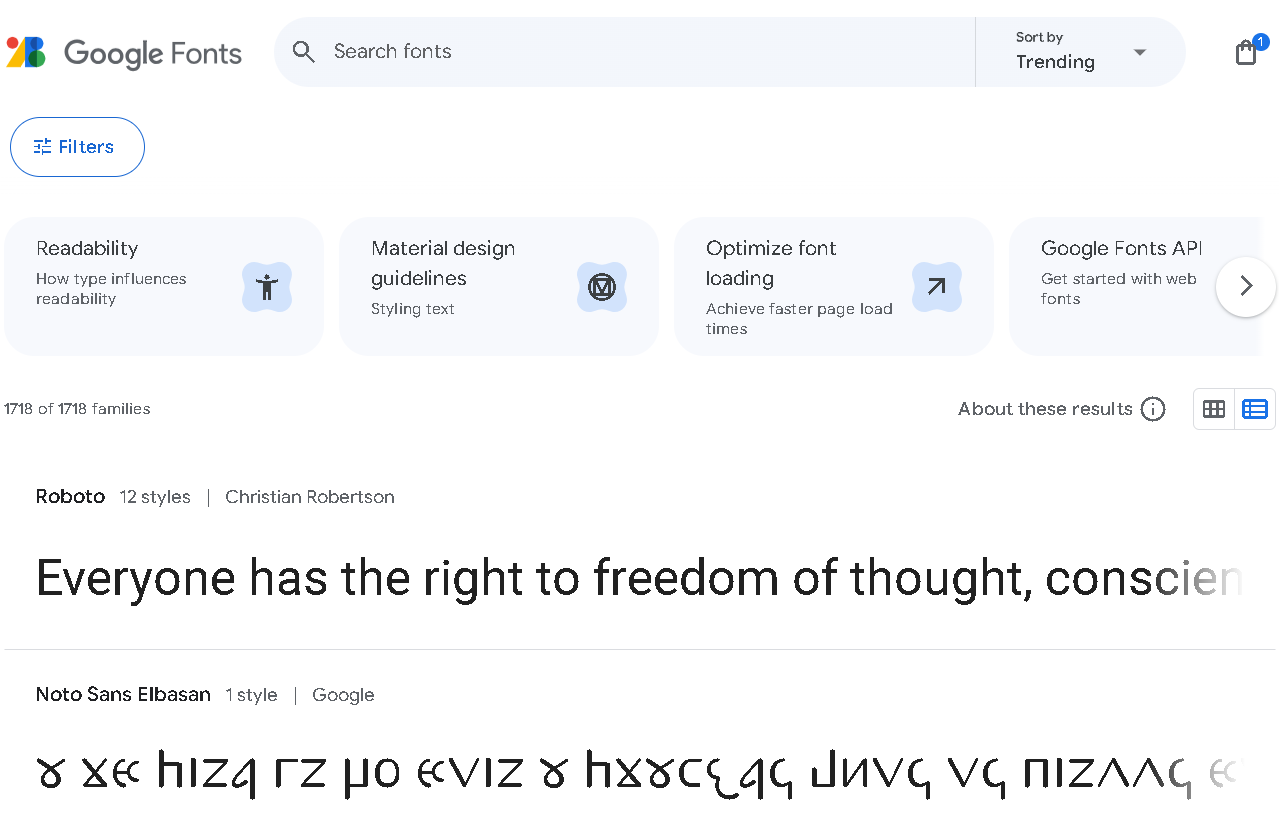
**Пример css в html-документе**

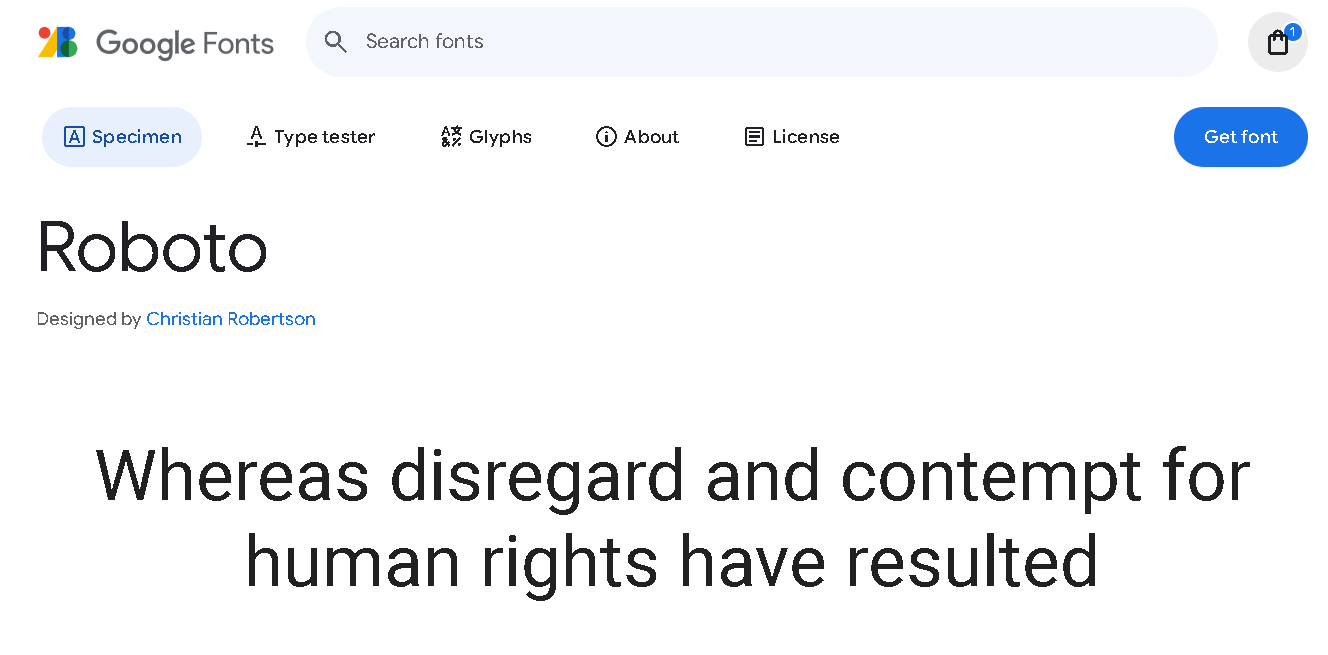


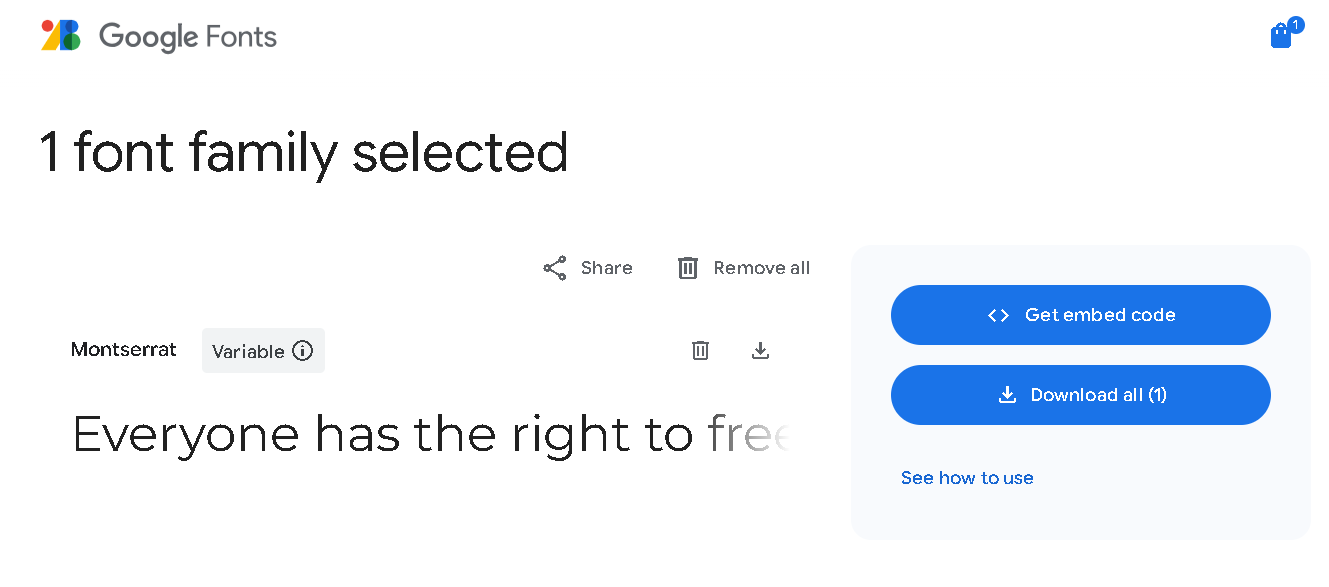
## Иерархия



## Подключение шрифтов

1. Ищем сайт Google Fonts. 
2. Через поисковик или ниже в списке выбираем нужный шрифт.
3. Нажимаем синюю кнопку Get Font. Чуть выше на иконке пакета должен появиться синий кружок с количеством добавленных шрифтов.



1. Когда выбрали все нужные шрифты, нажимаем на иконку пакета. Далее нажимаем синюю кнопку <> Get embed code. Кавычки намекают на то, что это нам надо для разметки. 
2. Слева мы можем выбрать для нас нужные настройки. Справа верхнее окошко с разметкой мы полностью копируем и вставляем в head. В нижнем окошке нам представлен css-документ. Я рекомендую это все не копировать, а лишь скопировать оттуда свойство font-famaly со значением. В любом случае мы будем прописывать для нашего сайта собственные инструкции и описание, которое явно будет отличаться от данного.

## Смещение и размещение контейнеров и элементов



Свойство display:

* none (ничего не отображает);
* block (единый блок-прямоугольник, располагающий содержимое вертикально, обтекающий по размерам свое содержимое по высоте, растягивающийся на всю ширину страницы; размеры можно явно указать);

Примеры: <h1>, <div>, <p>, <pre>

* inline (блок из совокупности прямоугольников, располагает содержимое горизонтально в строку, обтекает по размерам свое содержимое по высоте и ширине, размеры указать явно нельзя; есть перенос);

Примеры: <a>

* inline-block (единый блок-прямоугольник, располагает содержимое горизонтально в строку, обтекает по размерам свое содержимое по высоте и ширине, размеры указать явно можно; есть перенос);
* run-in (inline-block, часть block, располагается перед следующим block, устаревший);
* list-item (блоки для элементов списка, значок указывается свойством list-style, устаревший);
* table (блоки для отображения таблицы значениями display: table, display: table-row, display: table-cell);
* flex (размеры блоков автоматически подстраиваются под содержимое, по умолчанию отображает содержимое в строку) ;
* grid (отличие Grid от Flexbox в том, что через grid можно легко позиционировать любые объекты в строке и столбце. Также есть возможность накладывать друг на друга множества элементов. Автоматически множества располагаются по порядку, написанному в разметке).

## Выравнивание по центру

1 способ: задать элементу свойство margin: auto;

2 способ: задать родителю свойство position: relative; задать дочернему (-им) элементам свойство position: absolute; top: 0; bottom: 0; left: 0; right: 0;

3 способ: задать родителю display: flex; justify-content: center; align-items: center;