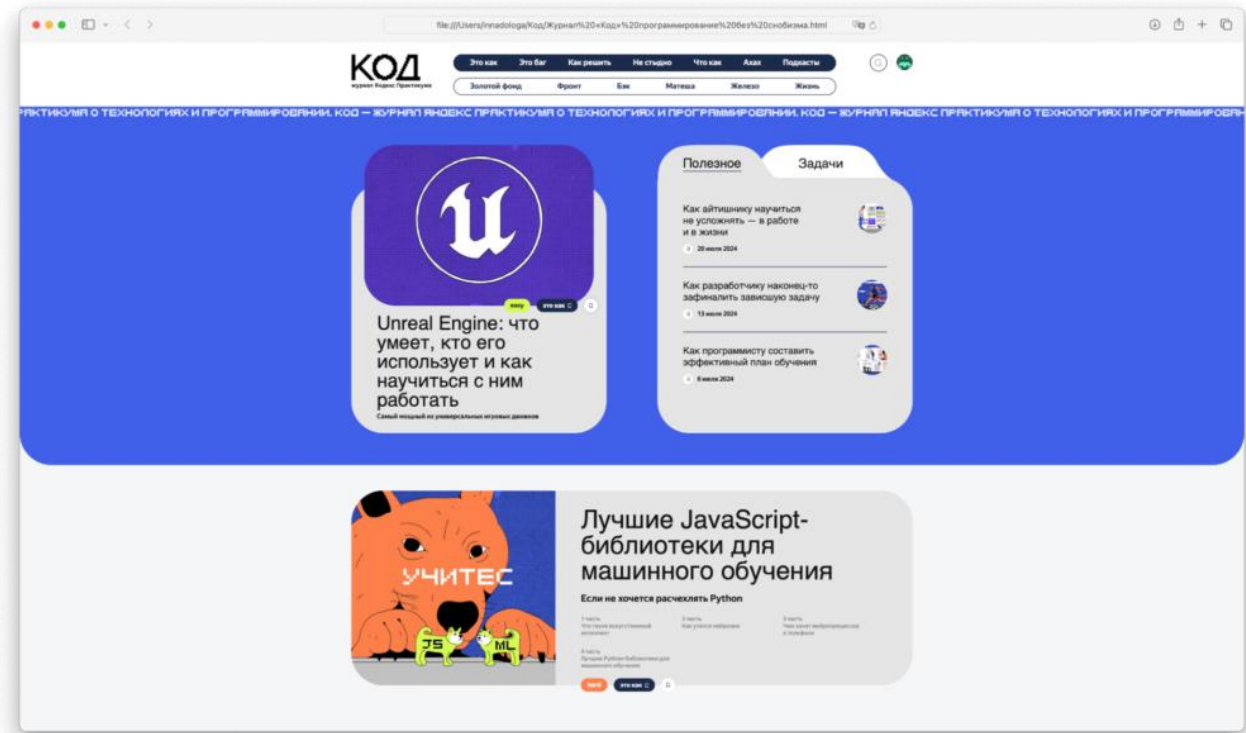
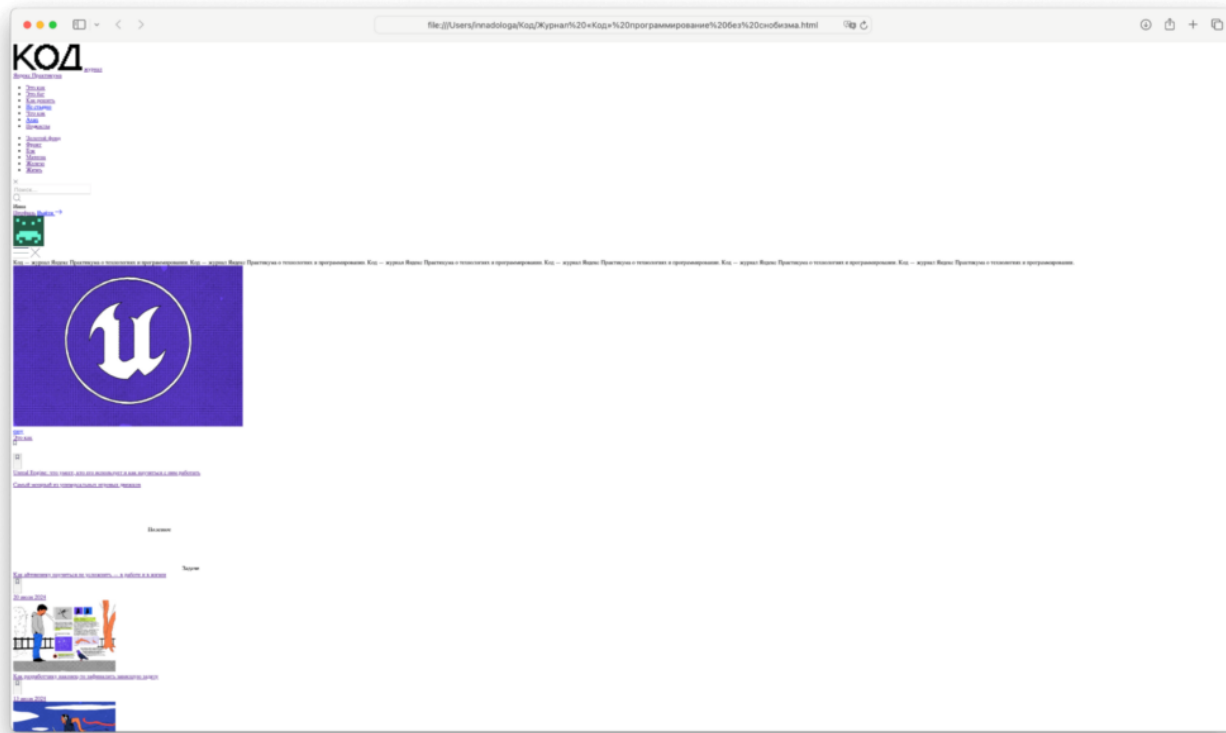


CSS. База

CSS (Cascading Style Sheets, «каскадные таблицы стилей») — это язык стилей, с помощью которого оформляют веб-страницы.



# Из чего состоит

CSS — декларативный язык. Это значит, что он не содержит явный код для стилизации, а описывает, как следует стилизовать элементы веб-страницы.

Состоит из:

- Комментарии
- Селекторы
- Свойства
- Значения

# Структура

Селектор, свойство и значение образуют правило. Правило, содержит блок деклараций — одно или несколько свойств и их значений в фигурных скобках. Каждая декларация состоит из пары «свойство: значение», разделённой точкой с запятой:

```
/* комментарий */  
селектор {  
    свойство: значение; -- декларация  
}
```

# Подключение

1. Прописать стили в HTML
2. Создать .css файл. Тогда внутри тега HEAD надо указать:

```
<link rel="stylesheet" href="styles.css">
```

# CSS-свойства

- font-family, font-size, font-weight, font-style — свойства шрифта;
- color, background-color, border-color — свойства цвета;
- text-align, text-decoration, text-transform — свойства текста;
- display, float, position, margin, padding — свойства макета;
- background-image, background-repeat, background-position — свойства фона.

# CSS значения

Это могут быть отдельные значения, числа, цветовые коды, ключевые слова или более сложные выражения.

- `font-family: Arial, sans-serif,`
- `color: #ff0000,`
- `background-image: url(image.png),`
- `margin: 10px 20px,`
- `text-align: center.`

# Типографика

- `font-family` определяет список шрифтов, которые будут использоваться для текста.
- Размер шрифта `font-size` определяет, какого размера должен быть текст.
- Стиль шрифта `font-style` определяет стиль шрифта, например курсив, как в этом примере:

`font-style: italic;`

- Толщина шрифта `font-weight` определяет толщину текста.  
Например, можно задать жирный шрифт:  
`font-weight: bold;`



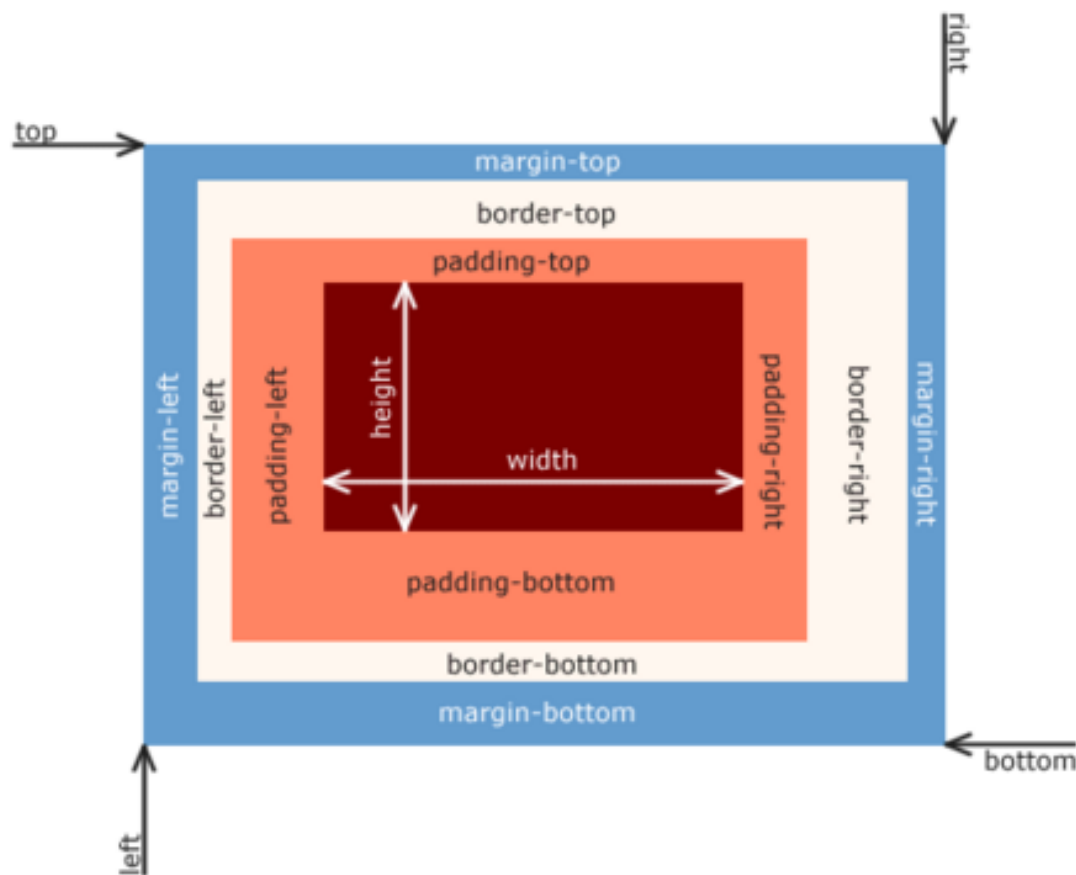
# Цвета

Цвет текста указывается с помощью свойства `color`, а цвет фона элемента — с помощью свойства `background-color`

- стандартным именем, например `color: blue;`,
- шестнадцатеричным кодом HEX — `color: #ff0000;;`,
- значениями RGB (красный, зелёный и синий) — `color: rgb(255, 0, 0);,`
- значениями RGBA, где к RGB добавляется прозрачность — `color: rgba(255, 0, 0, 0.5);,`
- значениями HSL (оттенок, насыщенность и светлость) — `color: hsl(0, 100%, 50%);,`
- значениями HSLA, где к HSL добавляется прозрачность — `color: hsla(0, 100%, 50%, 0.5);.`

# Блочная модель

Блочная модель, или Box Model, рассматривает каждый элемент как прямоугольную коробку



# Layout. display: block

- Элемент ведет себя как блочный.
- Занимает всю доступную ширину родительского элемента.
- Начинается с новой строки.
- Примеры элементов по умолчанию: <div>, <p>, <h1>—<h6>, <ul>, <li>.

```
div {  
  display: block;  
  width: 200px;  
  height: 100px;  
  background-color: lightblue;  
}
```

# Layout. display: inline

- Элемент ведет себя как строчный.
- Занимает только столько места, сколько нужно для содержимого.
- Не начинается с новой строки.
- Нельзя задавать ширину, высоту, вертикальные отступы (margin-top, margin-bottom).
- Примеры элементов по умолчанию: `<span>`, `<a>`, `<strong>`, `<em>`.

```
.span {  
    display: inline;  
    background-color: lightcoral;  
    padding: 5px;  
}
```

# Layout. display: inline-block

- Элемент ведет себя как строчный, но может иметь ширину, высоту и отступы.
- Не начинается с новой строки.
- Полезно для создания элементов, которые должны быть в одной строке, но иметь размеры

```
.button {  
  display: inline-block;  
  width: 100px;  
  height: 40px;  
  background-color: lightgreen;  
}
```

# Layout. display: grid

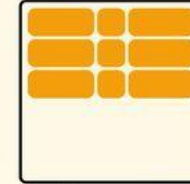
- Элемент становится grid-контейнером.
- Его дочерние элементы располагаются в сетке (Grid Layout).
- Полезно для сложных макетов

```
.grid-container {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  gap: 10px;  
}
```

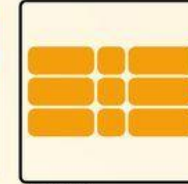
# Grid Layout

## CSS Grid Layout

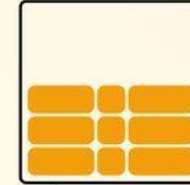
### align-content



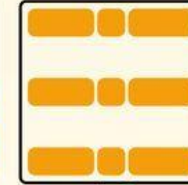
start



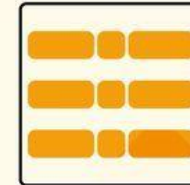
center



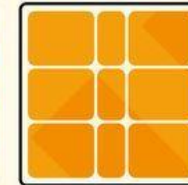
end



space-between

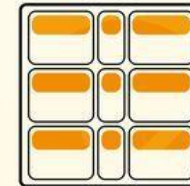


space-around

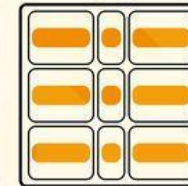


stretch

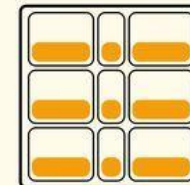
### align-items



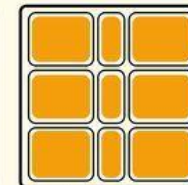
start



center

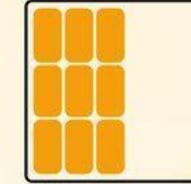


end

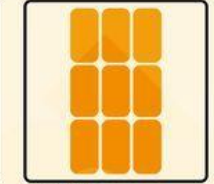


stretch

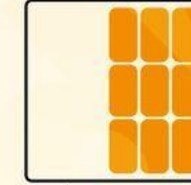
### justify-content



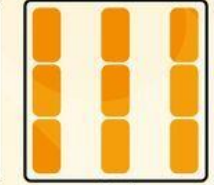
start



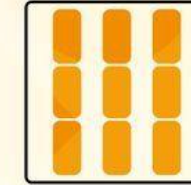
center



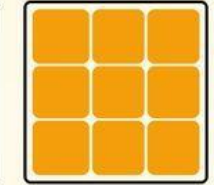
end



space-between

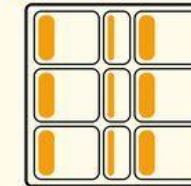


space-around

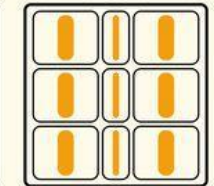


stretch

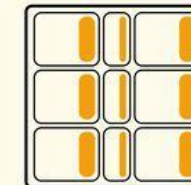
### justify-items



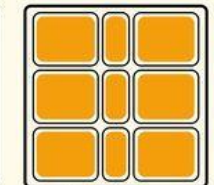
start



center



end



stretch

# Генераторы css

- <https://html-css-js.com/css/generator/>
- <https://webcode.tools/css-generator>

## Справочник CSS:

- <https://html5css.ru/cssref/default.php>