

# CSS Box Model

---

**WEB  
COURSE  
ORT DNIPRO**

---

**ORT**[\*\*DNIPRO.ORG/WEB\*\*](https://ortdnipro.org/web)

# Cascading Style Sheets

# Синтаксис CSS

**CSS селектор**, говорит к каким тегам (элементам) будет применяться описываемый стиль (**css selector**).

Значение которое устанавливается для свойства (**value**).

```
h2 { color: red; font-size: 16pt; }
```

Имя свойства, которое устанавливается (**property**).

Код на языке **CSS** состоит из стилевых правил, каждое правило содержит **селектор** (указание на то какие теги необходимо оформить этим правилом) и набор **стилевых свойств**, которые и задают оформление (на примере ***color***, ***font-size*** и др.).

# 1. CSS Box Model

или

# О размерах элемента

# CSS Box Model на практике

```
1  
2  
3    html:5  
4  
5    h1>lorem10  
6  
7    div*5>lorem200  
8  
9
```

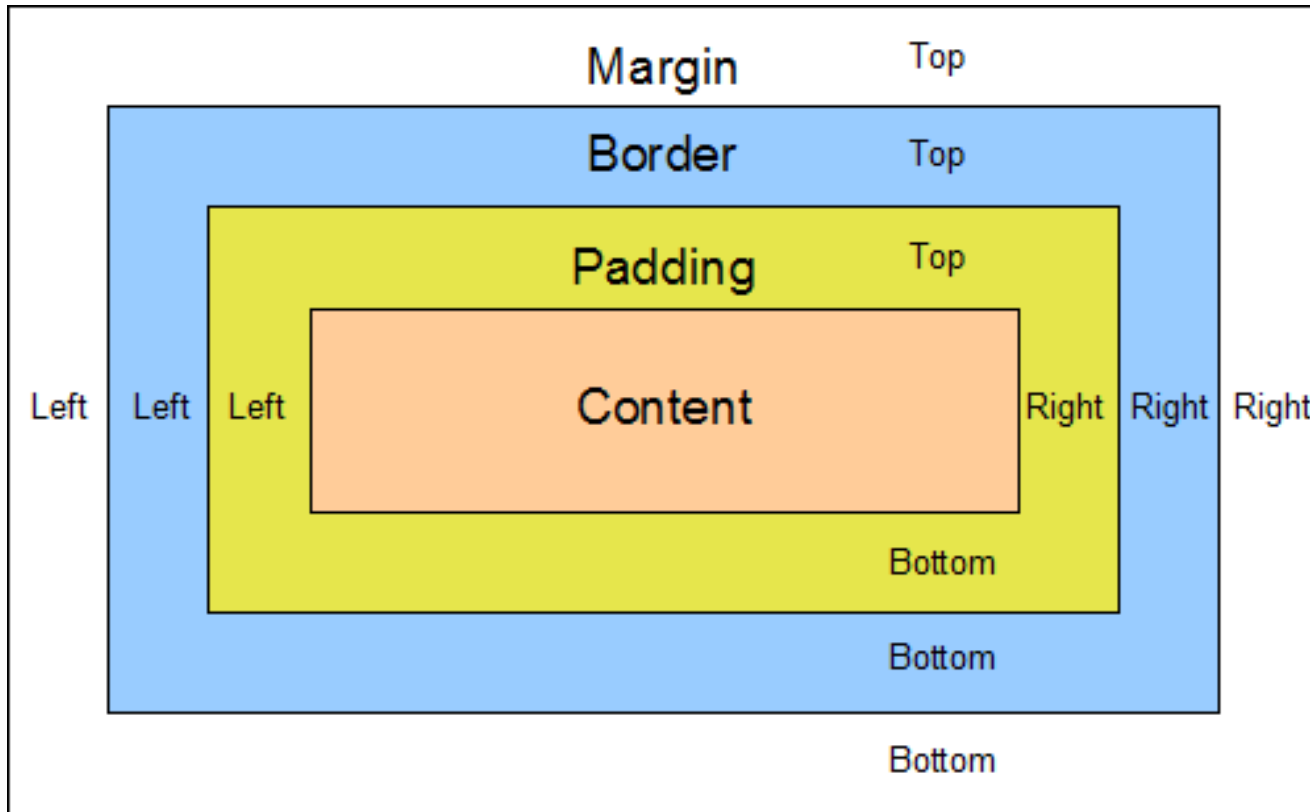
Подготовьте разметку на основе таких команд **Emmet**'а в **VSCode** (команды выполняются по одной)

1) Каркас HTML-разметки;

2) Один заголовок **<h1>** с 10-ю словами;

3) Пять блоков **<div>** с 200-ю словами каждый.

# CSS Box Model



**CSS Box Model** - описание размерностей элемента: высоты - **height**, ширины - **width**, отступов (внутренних – **padding**, внешних - **margin**), рамки – **border**.

<https://webref.ru/css/type/size>

<https://webref.ru/css/type/margin>

<https://webref.ru/css/type/padding>

<https://webref.ru/css/type/border>

# CSS Box Model

`margin: 10px;` {  
    `margin-top: 10px;`  
    `margin-right: 10px;`  
    `margin-bottom: 10px;`  
    `margin-left: 10px;`

`margin: 10px 20px 30px 40px;` {  
    `margin-top: 10px;`  
    `margin-right: 20px;`  
    `margin-bottom: 30px;`  
    `margin-left: 40px;`

Отступы (как **padding** так и **margin**) можно задать для каждой стороны в отдельности. Свойство **border** также возможно установить для каждой стороны отдельно. Есть возможность задать сразу четыре значения через пробел (**top, right, bottom, left**). А также для левого и правого отступа **margin** есть возможность задать автоматических расчёт отступов при помощи значения **auto**.

# Свойство **box-sizing**

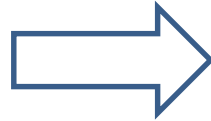
```
32 article{  
33     width: 500px;  
34     padding: 20px;  
35     margin: 50px auto;  
36     border: 5px solid green;  
37     border-radius: 25px;  
38     box-shadow: 5px 5px 5px gray;  
39  
40     box-sizing: border-box;  
41 }
```

Свойство **box-sizing: border-box** – задаёт альтернативный вариант расчёта размерности элемента, при котором в заданную ширину (и/или высоту) должны входить не только контент, но и внутренние отступы (**padding**) и рамка (**border**), т.е. **весь блок целиком** (**margin** не входит в размеры блока).



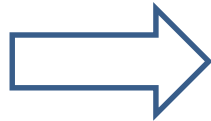
# Свойство **box-sizing**

**box-sizing:content-box**  
(по умолчанию)



**width** = **content-width**

**box-sizing:border-box**



**width** = **content-width** + **padding** + **border**

<https://webref.ru/css/box-sizing>

ElementsConsoleSourcesNetworkPerformanceMemory

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body>
    <article>...</article> == $0
      <!-- One Article -->
      <article>...</article>
      <!-- One Article -->
      <article>...</article>
      <!-- One Article -->
      <article>...</article>
      <!-- One Article -->
      <article>...</article>
      <!-- One Article -->
      <article>...</article>
      <!-- One Article -->
    </body>
  </html>
```

htmlbodyarticle

StylesEvent ListenersDOM BreakpointsProperties

Filter: :hov .cls +

element.style {
}

article {
 width: 500px;
 padding: 20px;
 margin: 50px auto;
 border: 5px solid green;
 border-radius: 25px;
 box-shadow: 5px 5px 5px grey;
}

The diagram illustrates the CSS Box Model for the `article` element. It consists of three nested rectangles:

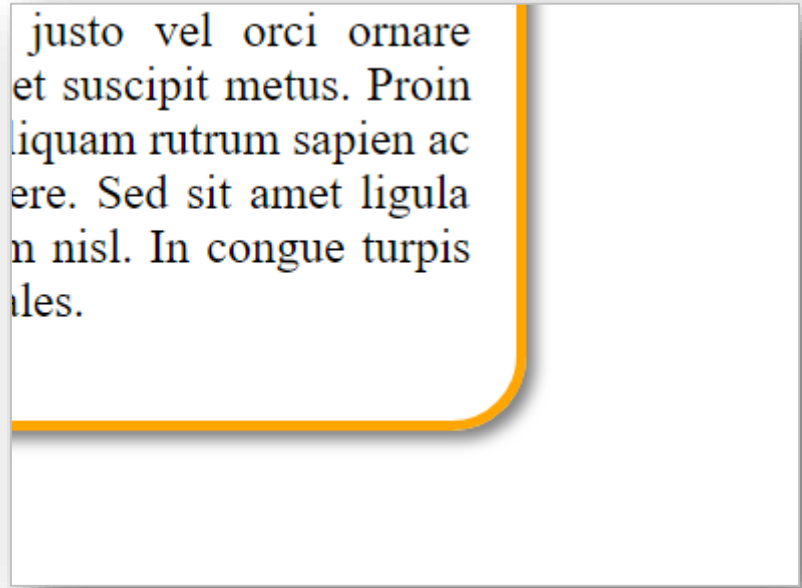
- Content Box (padding):** A light blue rectangle with dimensions `500 × 612.094`.
- Border Box (border):** A green rectangle surrounding the content box with a `border: 5px solid green`.
- Margin Box (margin):** An orange rectangle surrounding the border box with a `margin: 50px auto`.

The diagram also shows the `box-shadow: 5px 5px 5px grey` applied to the border box. The overall dimensions of the margin box are `100 × 100`.

# CSS Box Model

Консоль разработка  
(закладка **Elements** → **Styles**)  
может нам показать все  
размерности тега

# Скругление углов и тень от блока



justo vel orci ornare  
et suscipit metus. Proin  
liquam rutrum sapien ac  
ere. Sed sit amet ligula  
n nisl. In congue turpis  
les.

Свойство **border-radius** позволяет задать радиус округления уголков рамки. Свойство **box-shadow** позволяет задать отбрасывание тени блоком, набор параметров аналогичен свойству **text-shadow**.

<https://webref.ru/css/box-shadow>

<https://webref.ru/css/border-radius>

## 2. Единицы измерения

# Абсолютные vs. Относительные единицы измерения

**3px**

**5in**

**12cm**

**24pt**

...

**50%**

**30vw**

**100vh**

**3rem**

...

CSS поддерживает множество единиц измерения, но все они делятся на две группы: абсолютные и относительные.

<https://webref.ru/css/value/size>  
<https://webref.ru/course/css-basics/size>

# Абсолютные единицы измерения

| Единица   | Описание                         |
|-----------|----------------------------------|
| <i>px</i> | CSS Пиксель                      |
| <i>in</i> | Дюйм (1 дюйм равен 2,54 см)      |
| <i>cm</i> | Сантиметр                        |
| <i>pt</i> | Пункт (1 пункт равен 1/72 дюйма) |
| ...       |                                  |

Не зависят от размера устройства и плотности точек на нём. Величина заданная при помощи абсолютных единиц измерения будет одинакова на всех устройствах.

<https://webref.ru/css/value/size>  
<https://webref.ru/course/css-basics/size>

# Относительные единицы измерения

| Единица | Описание   |
|---------|--|
| %       | Рассчитывается, как правило, от значения родительского элемента, но есть множество исключений. |

| Единица           | Описание  |
|-------------------|---|
| <b><i>vw</i></b>  | 1% от ширины области просмотра  |
| <b><i>vh</i></b>  | 1% от высоты области просмотра  |
| <b><i>rem</i></b> | Расчёт выполняется относительно размера <b>font-size</b> заданного для тега <code>&lt;html&gt;</code> . |
| ...               |   |

<https://webref.ru/css/value/size>  
<https://webref.ru/course/css-basics/size>

## Указание единиц измерений - **обязательно!**

padding: 5px; +

padding: 0px; +

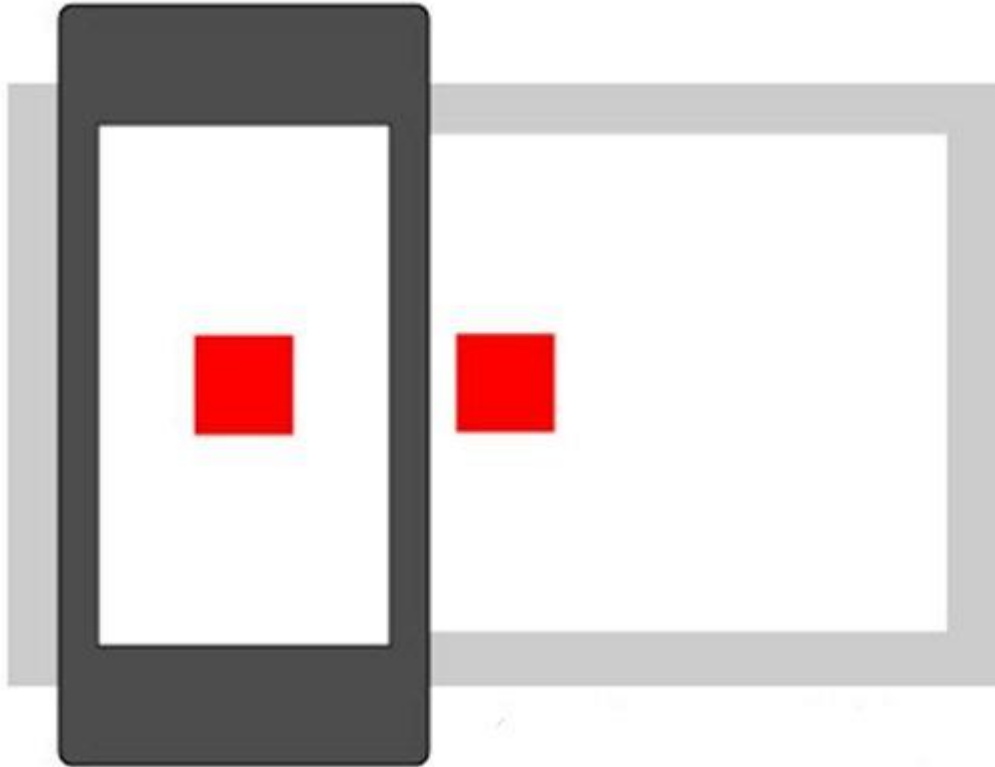
padding: 5 px; -

padding: 5; -

padding: 0; +



# CSS пиксель $\neq$ Физический пиксель



**CSS пиксель** – размер точки условно соответствующий  $\sim 1/96$  дюйма. Сколько физических пикселей используется для отрисовки одного **CSS пикселя** зависит от физического разрешения конкретного экрана.

# CSS пиксель $\neq$ Физический пиксель



**CSS пиксель** – размер точки условно соответствующий ~ **1/96** дюйма. Сколько физических пикселей используется для отрисовки одного **CSS пикселя** зависит от физического разрешения конкретного экрана.

### 3. **min-/max-** ограничения ширины и высоты

# Ограничения ширины/высоты

$\text{min-width} \leq \text{width} \leq \text{max-width}$

$\text{min-height} \leq \text{height} \leq \text{max-height}$

Подробнее: <https://habr.com/ru/post/483634/>

**Вариант применения:**  
границы (*свойства: min-width, max-width и т.д.*) заданы абсолютными величинами, а высота/ширина – относительными. Что даёт возможность ширине/высоте меняться (адаптироваться, на различных устройствах) но не выходить за допустимые рамки.

## 4. CSS-функции

`min()`, `max()`, `clamp()`

# CSS-функции `min()`, `max()`, `clamp()`

```
10
11  ✓  main{
12
13      width: min(960px, 70vw);
14
15      width: max(300px, 70vw);
16
17      width: clamp(300px, 70vw, 960px);
18
19  }
20
```

Позволяют комбинировать единицы измерения для получения эффекта адаптивности: **min()** выбираем минимальное значение из переданных, **max()** – максимальное, а **clamp()** придерживается второго значения, но не выходит за пределы первого и третьего (как минимума и максимума соответственно).

Подробнее: <https://habr.com/ru/company/ruvds/blog/501634/>

# 5. Изображения и размеры

# Изображения и размеры



По умолчанию размеры изображения задаются разрешением подключаемого файла и не ограничивается размерами родительского тега. Но размеры тега можно задать принудительно. В виде конкретной ширины или ограничения через **max-width** (**max-height**), или же задействовав функции (**min()**, **max()**, **clamp()**).

**Свойства которые будут полезны:**

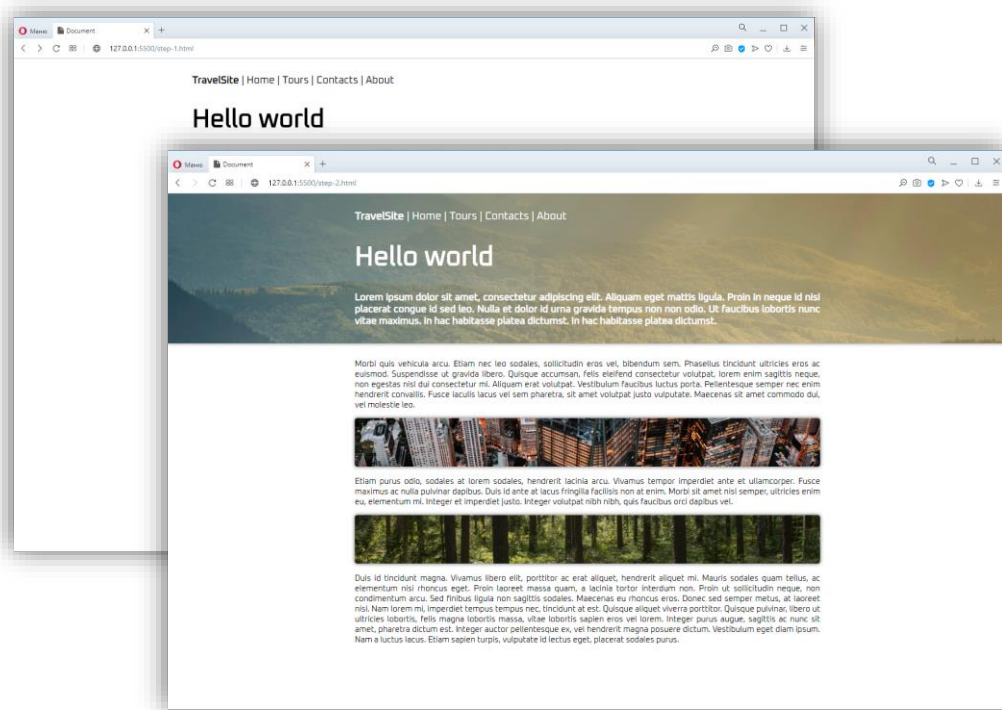
<https://webref.ru/css/object-fit>

<https://developer.mozilla.org/ru/docs/Web/CSS/object-position>



# Домашнее задание

# Домашнее задание | в 2 этапа



Создайте адаптивный «контейнер» в **70vw** от ширины видимой области браузера, но в рамках от **500** до **1100** пикселей. Внутри него разместите контент как на макете.

# Hello world

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam eget mattis ligula. Proin in neque id nisi placerat congue id sed leo. Nulla et dolor id urna gravida tempus non non odio. Ut faucibus lobortis nunc vitae maximus. In hac habitasse platea dictumst. In hac habitasse platea dictumst.

Morbi quis vehicula arcu. Etiam nec leo sodales, sollicitudin eros vel, bibendum sem. Phasellus tincidunt ultricies eros ac euismod. Suspendisse ut gravida libero. Quisque accumsan, felis eleifend consectetur volutpat, lorem enim sagittis neque, non egestas nisl dui consectetur mi. Aliquam erat volutpat. Vestibulum faucibus luctus porta. Pellentesque semper nec enim hendrerit convallis. Fusce iaculis lacus vel sem pharetra, sit amet volutpat justo vulputate. Maecenas sit amet commodo dui, vel molestie leo.



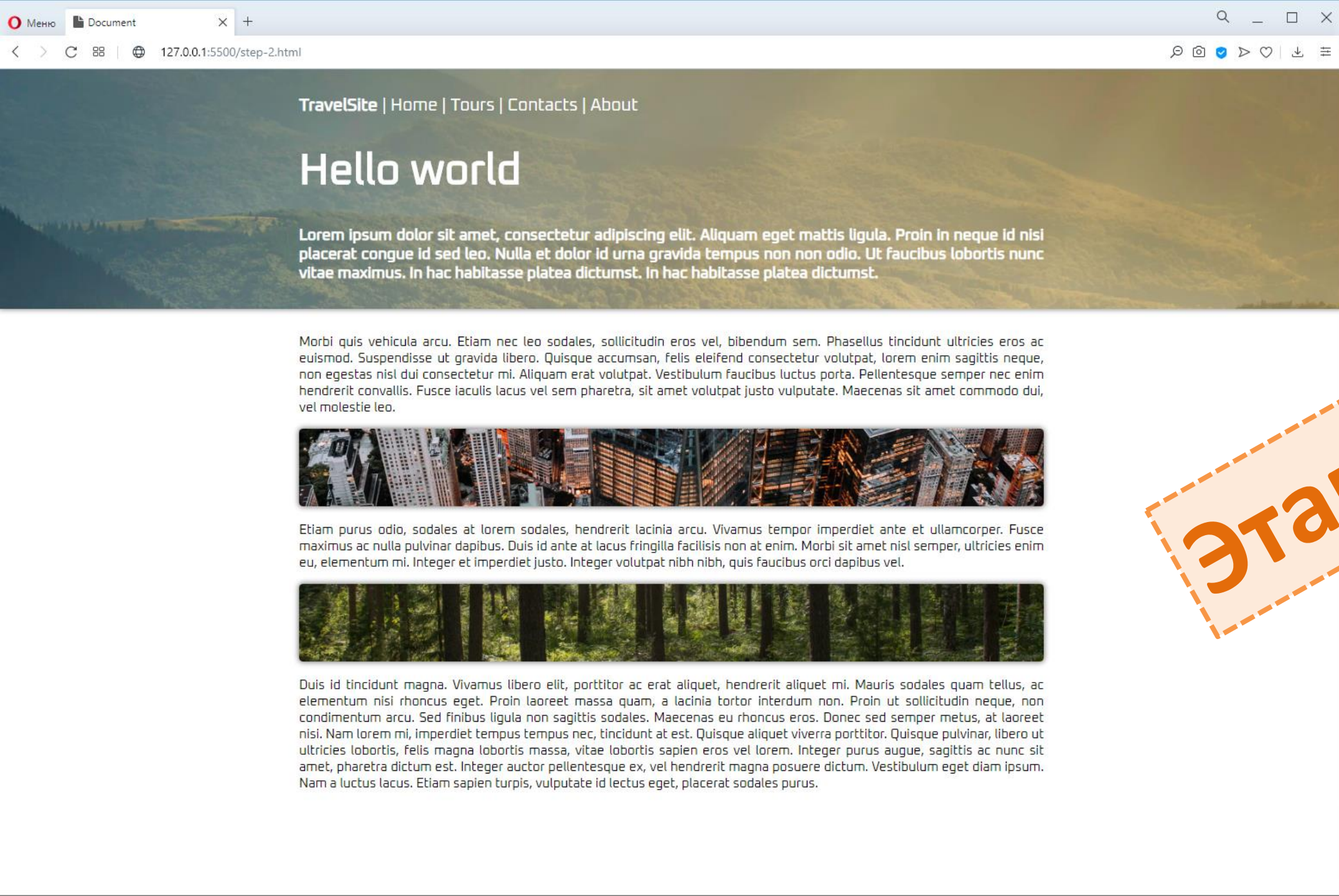
Etiam purus odio, sodales at lorem sodales, hendrerit lacinia arcu. Vivamus tempor imperdiet ante et ullamcorper. Fusce maximus ac nulla pulvinar dapibus. Duis id ante at lacus fringilla facilisis non at enim. Morbi sit amet nisl semper, ultricies enim eu, elementum mi. Integer et imperdiet justo. Integer volutpat nibh nibh, quis faucibus orci dapibus vel.



Duis id tincidunt magna. Vivamus libero elit, porttitor ac erat aliquet, hendrerit aliquet mi. Mauris sodales quam tellus, ac elementum nisi rhoncus eget. Proin laoreet massa quam, a lacinia tortor interdum non. Proin ut sollicitudin neque, non condimentum arcu. Sed finibus ligula non sagittis sodales. Maecenas eu rhoncus eros. Donec sed semper metus, at laoreet nisi. Nam lorem mi, imperdiet tempus tempus nec, tincidunt at est. Quisque aliquet viverra porttitor. Quisque pulvinar, libero ut ultricies lobortis, felis magna lobortis massa, vitae lobortis sapien eros vel lorem. Integer purus augue, sagittis ac nunc sit amet, pharetra dictum est. Integer auctor pellentesque ex, vel hendrerit magna posuere dictum. Vestibulum eget diam ipsum. Nam a luctus lacus. Etiam sapien turpis, vulputate id lectus eget, placerat sodales purus.

Этап #1

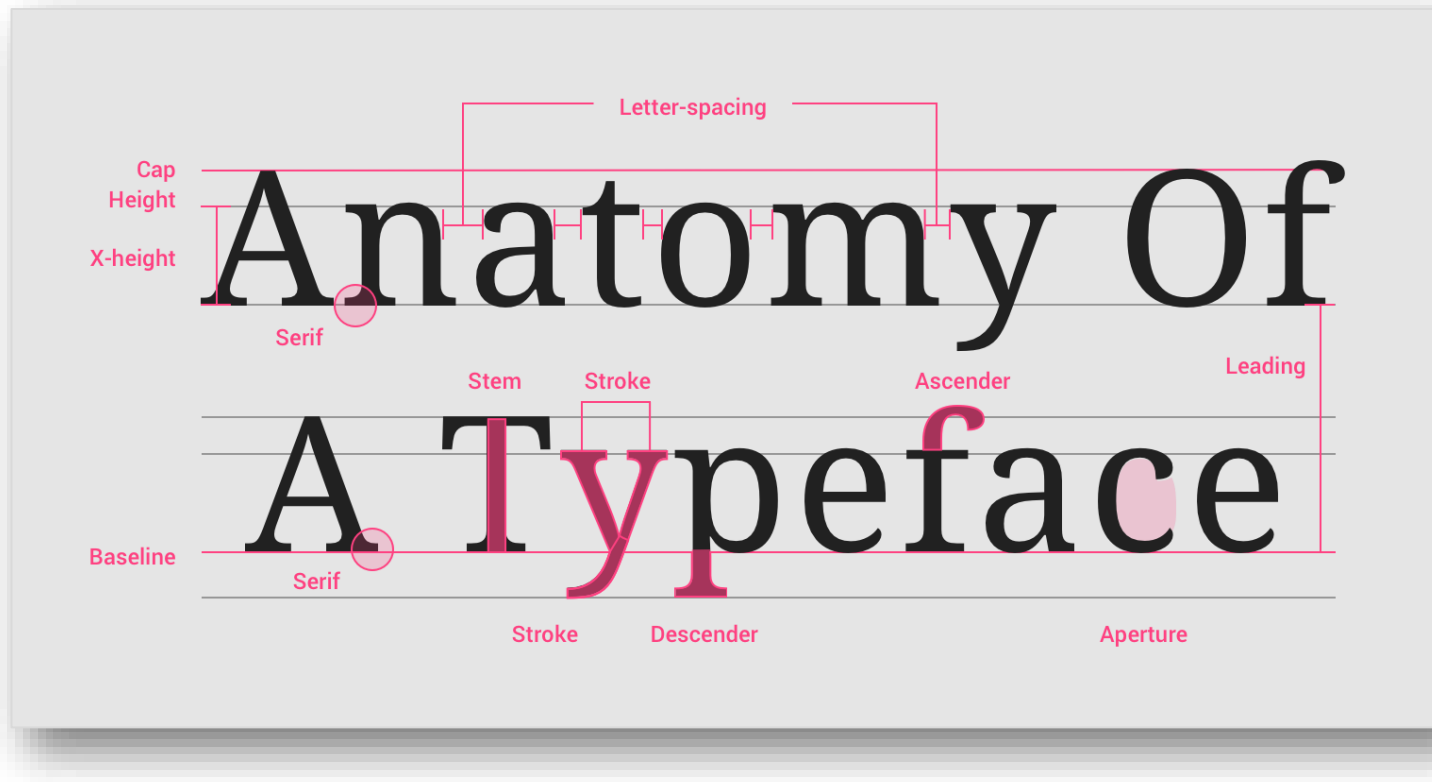




Этап #2

**К следующему  
занятию...**

# Типографика – наука о шрифтах



Предварительные знания – лучший помощник в обучении, поэтому к следующему занятию жду, что **посмотрите небольшой ролик о типографике.**

[https://youtu.be/Bq76TtKO\\_s](https://youtu.be/Bq76TtKO_s)