

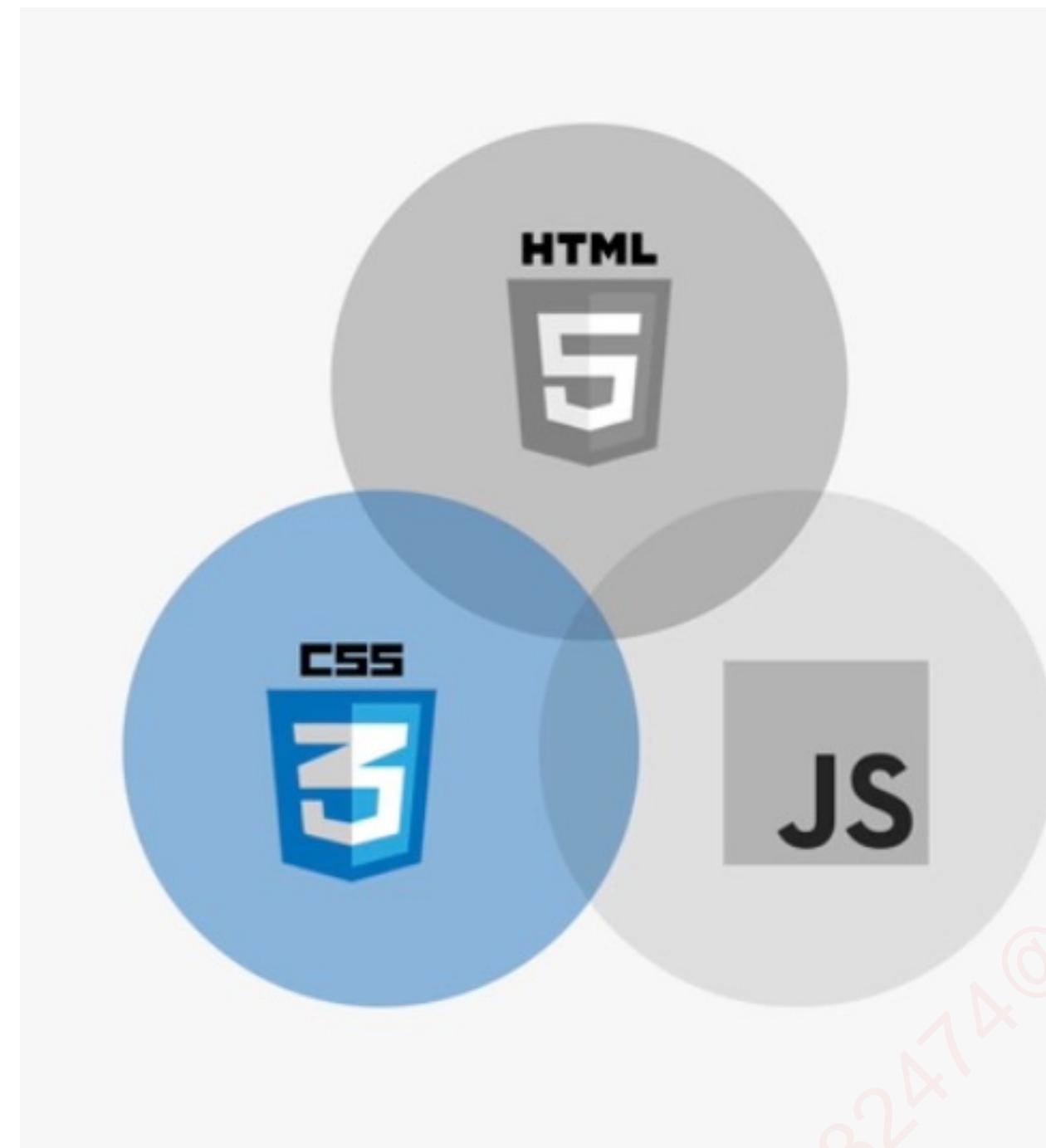


02 CSS





What is CSS



CSS - Cascading Style Sheets

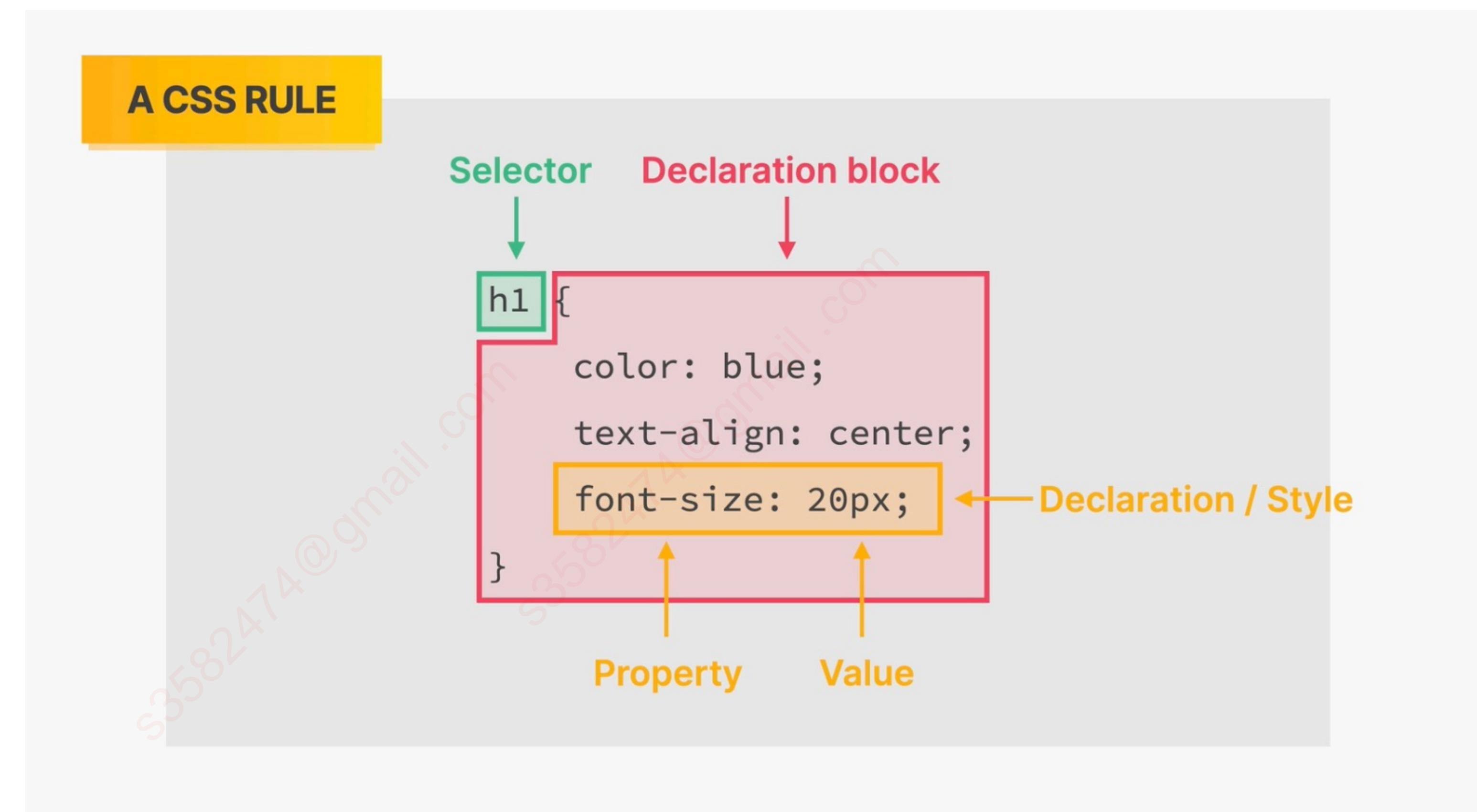
CSS describes the visual style and presentation of the content written in HTML

CSS consists of countless properties that developers use to format the content: properties about font, text, spacing, layout, etc.



推荐阅读 <https://www.w3schools.com/css/>

What is CSS



推荐阅读 <https://www.w3schools.com/css/>



Three ways to apply CSS to html

External

Internal

Inline

External style sheet

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

Internal style sheet

```
<head>
<style>
body {
    background-color: linen;
}

h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
```



Inline style

```
<h1 style="color:blue;margin-left:30px;">This is a heading</h1>
```

NOT recommend

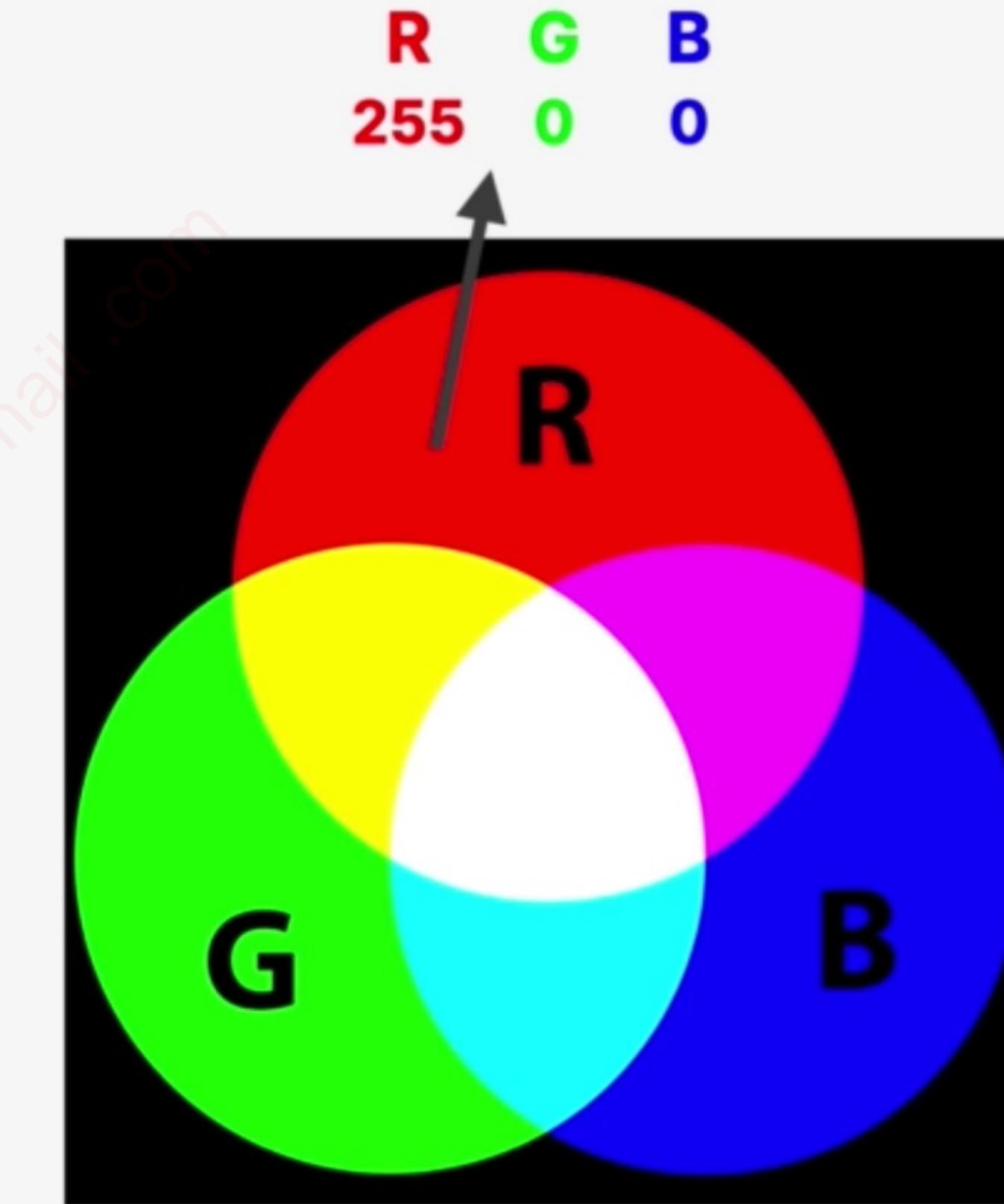
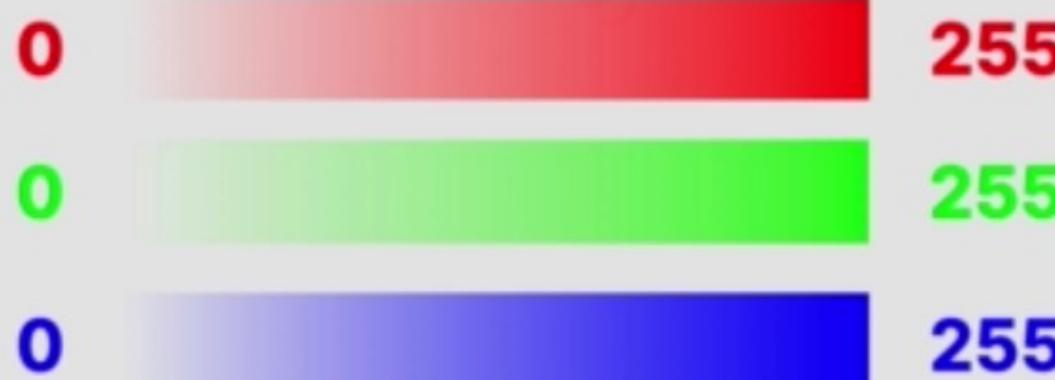
Only for components

The RGB Model



👉 **RGB Model:** Every color can be represented by a combination of **RED**, **GREEN** and **BLUE**

👉 Each of the 3 base colors can take a value between **0** and **255**, which leads to 16.8 million different colors



Color Representation



1

RGB / RGBA NOTATION

- 👉 Regular RGB model

```
rgb(0, 255, 255)
```



- 👉 RGB with transparency ("alpha")

```
rgba(0, 255, 255, 0.3)
```

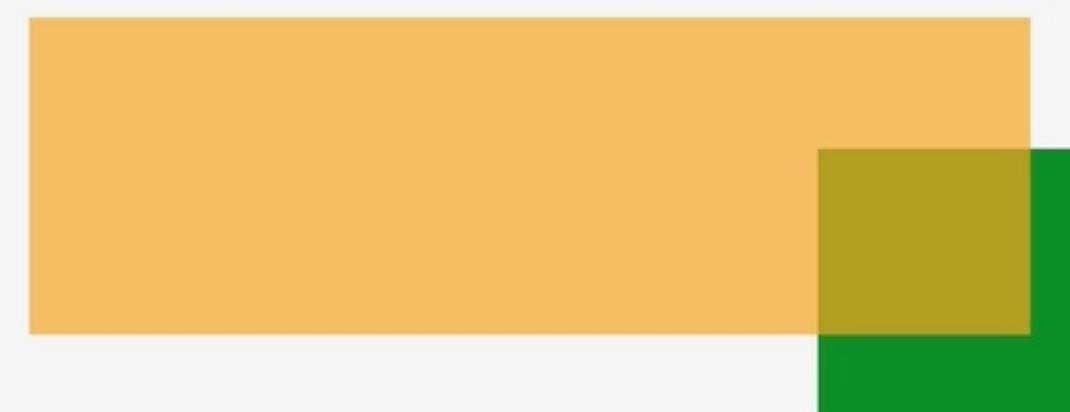


```
#f4b33f
```

```
rgb(244, 179, 63)
```



```
rgba(244, 179, 63, 0.7)
```



2

HEXADECIMAL NOTATION

- 👉 Instead of using a scale from 0 to 255, we go from **0** to **ff** (255 in hexadecimal numbers)

```
#00ffff
```

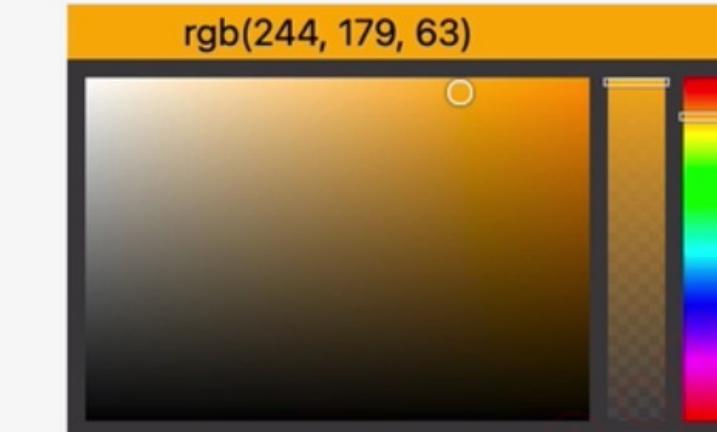


- 👉 Shorthand, when all colors are identical pairs

```
#off
```



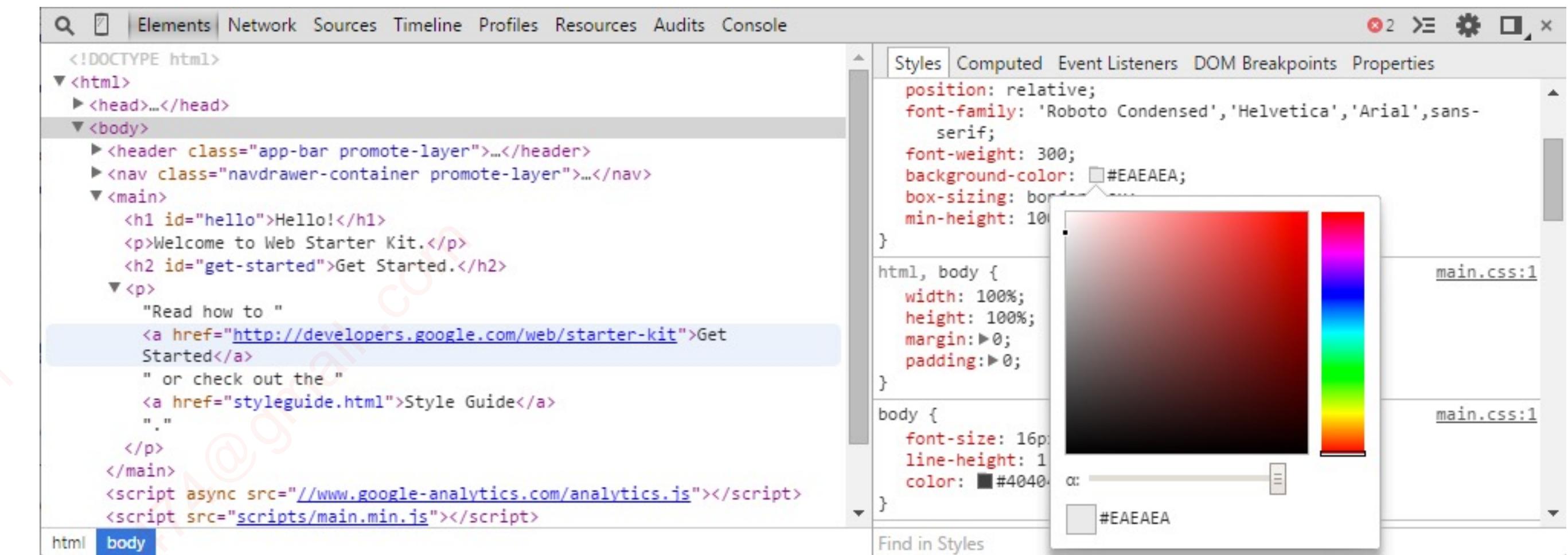
💡 In practice, we mostly use **hexadecimal** colors, and **rgba** when we need transparency



👉 Color picker in VS Code



How to Access DevTools



- Select the **Chrome menu**  at the top-right of your browser window, then select **Tools > Developer Tools**.
- Right-click on any page element and select **Inspect Element**.

npm install live-server -g
live-server



Dev Tool

- 在 Elements 面板中检查和实时编辑 DOM 树中的任何元素。
- 在 Styles 窗格中查看和更改应用到任何选定元素的 CSS 规则。
- 在 Computed 窗格中查看和修改选定元素的框模型。



Conflicting Selectors and Declarators

```
<p id="author-text" class="author">  
  Posted by Laura Jones on Monday, June 21st 2027  
</p>
```

```
.author {  
  font-style: italic;  
  font-size: 18px;  
}  
  
#author-text {  
  font-size: 20px;  
}  
  
p,  
li {  
  font-family: sans-serif;  
  color: #444444;  
  font-size: 22px;  
}
```

- 🤔 There are **multiple selectors** selecting the same element. Which one of them applies?
- 🤓 **All of them. All rules and properties are applied!**



Conflicting Selectors and Declarators

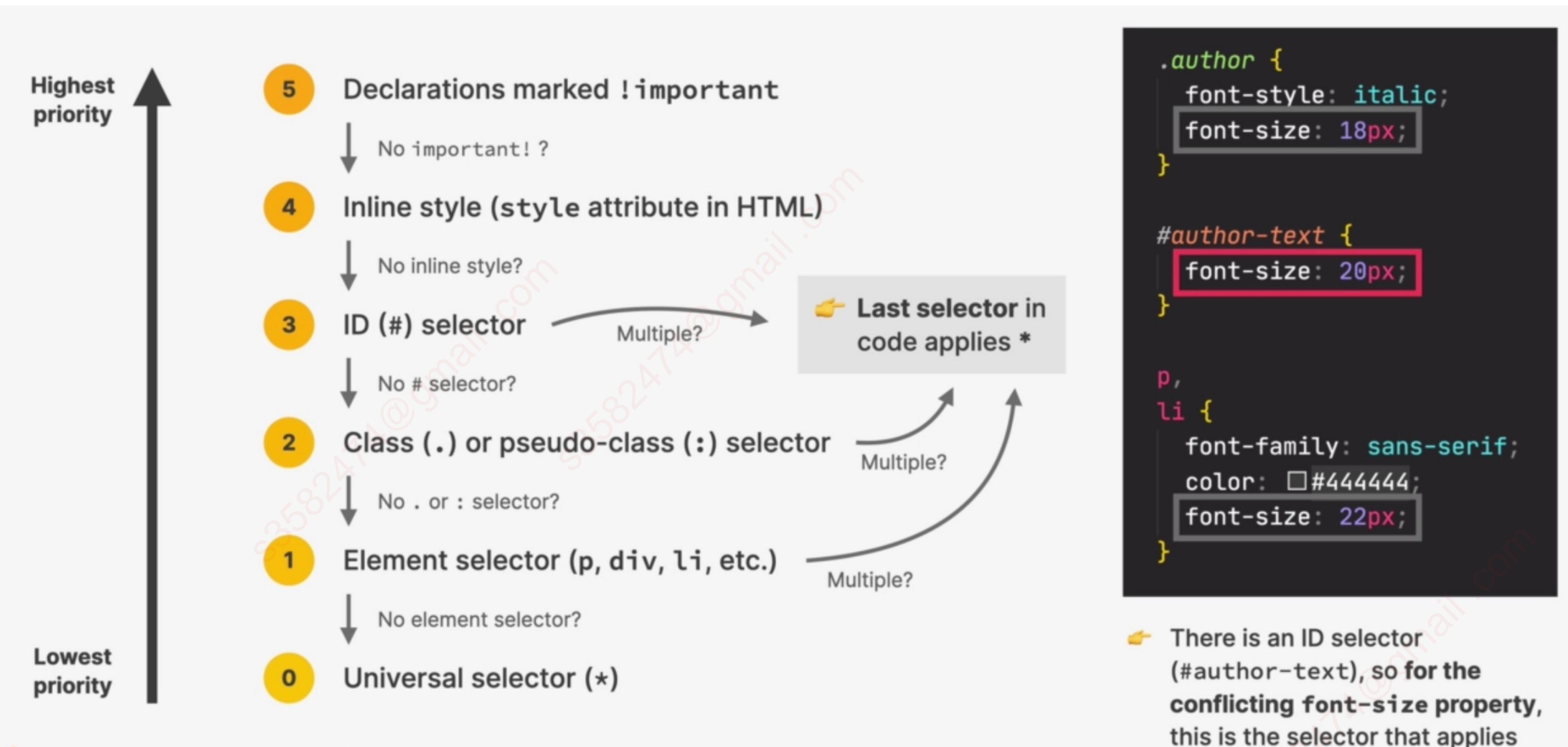
```
<p id="author-text" class="author">  
  Posted by Laura Jones on Monday, June 21st 2027  
</p>
```

```
.author {  
  font-style: italic;  
  font-size: 18px;  
}  
  
#author-text {  
  font-size: 20px;  
}  
  
p,  
li {  
  font-family: sans-serif;  
  color: #444444;  
  font-size: 22px;  
}
```

- 🤔 There are **multiple selectors** selecting the same element. Which **one of them applies?**
 - 🤓 **All of them. All rules and properties are applied!**
- ↓
- 🤔 But there are **conflicting font-size declarations!** Is it 18px, or 20px, or 22px?



Conflicting Selectors and Declarators





Inheritance

Parent element

```
<body>
  <nav>
    This is the navigation
  </nav>

  <h1>My website</h1>

  <p>
    The text is this paragraph
    is completely irrelevant 😊
  </p>

</body>
```

color: #444444;
font-size: 16px;
font-family: sans-serif;

color: #444444;
font-size: 16px;
font-family: sans-serif;

color: #444444;
font-size: 16px;
font-family: sans-serif;

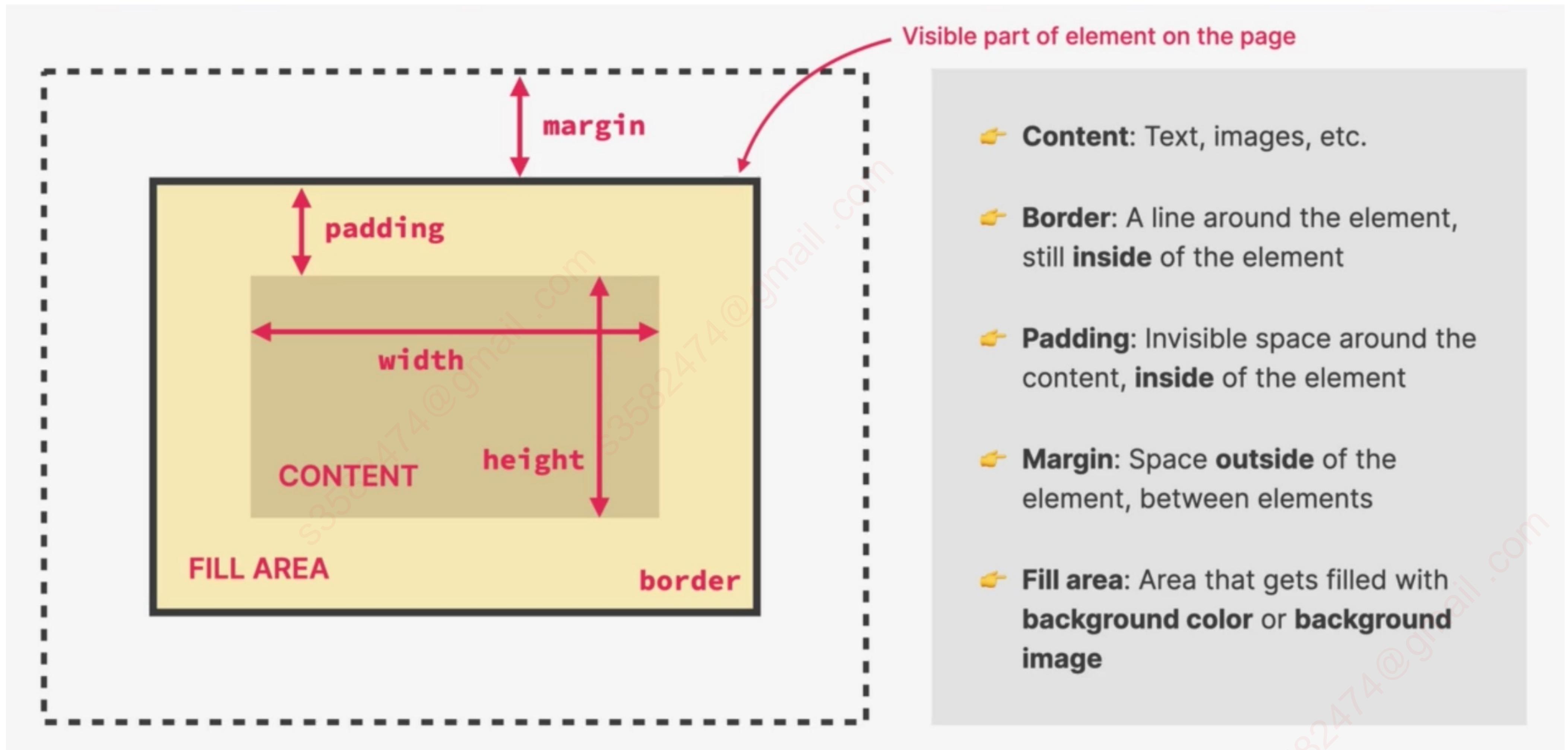
```
body {
  color: #444444;
  font-size: 16px;
  font-family: sans-serif;

  border-top: 10px solid #1098ad;
}
```

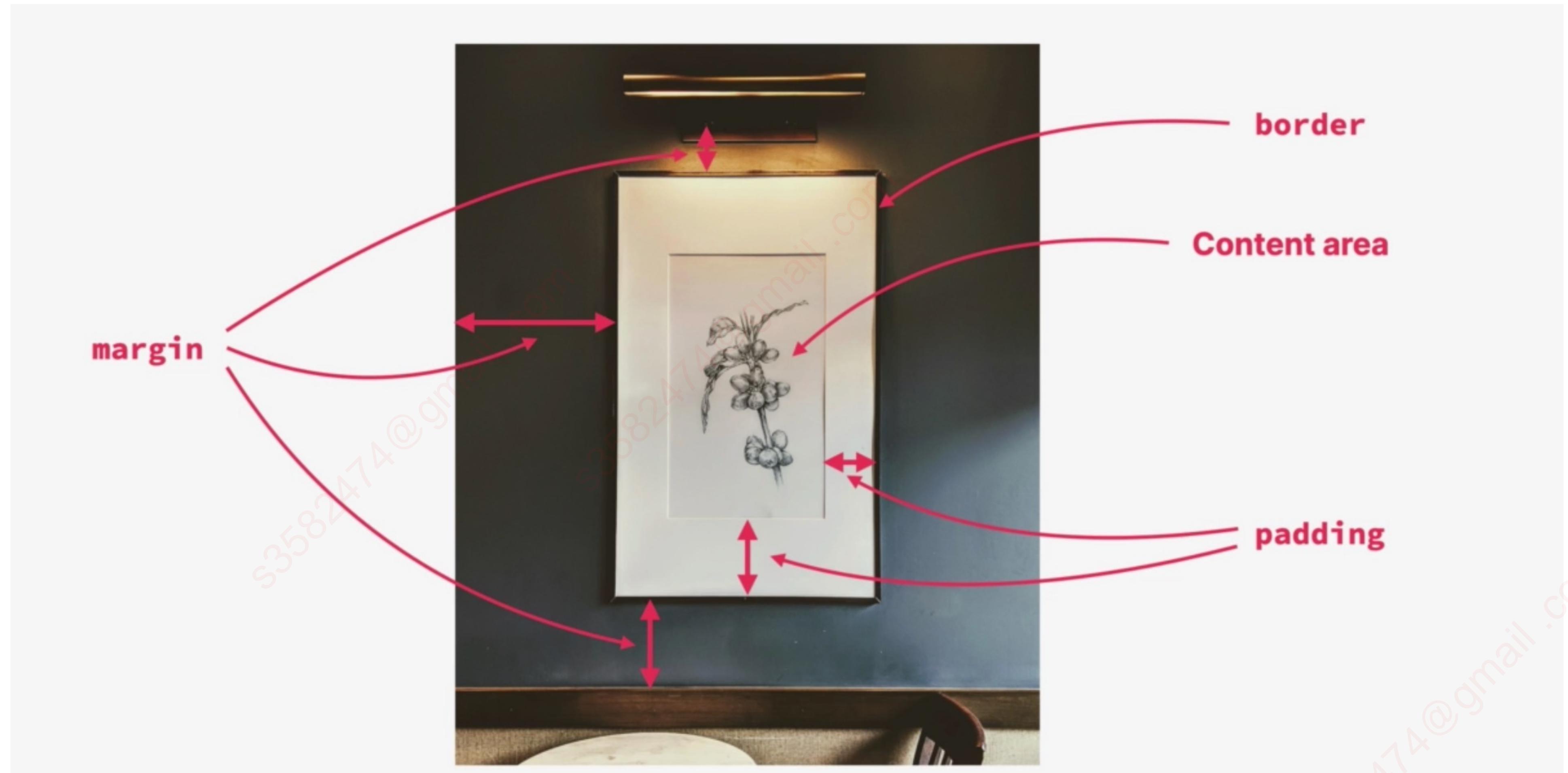
The border property does **NOT** get inherited

💡 Not all properties get inherited. It's mostly ones **related to text**: font-family, font-size, font-weight, font-style, color, line-height, letter-spacing, text-align, text-transform, text-shadow, list-style, etc.

CSS Box modal



CSS Box model



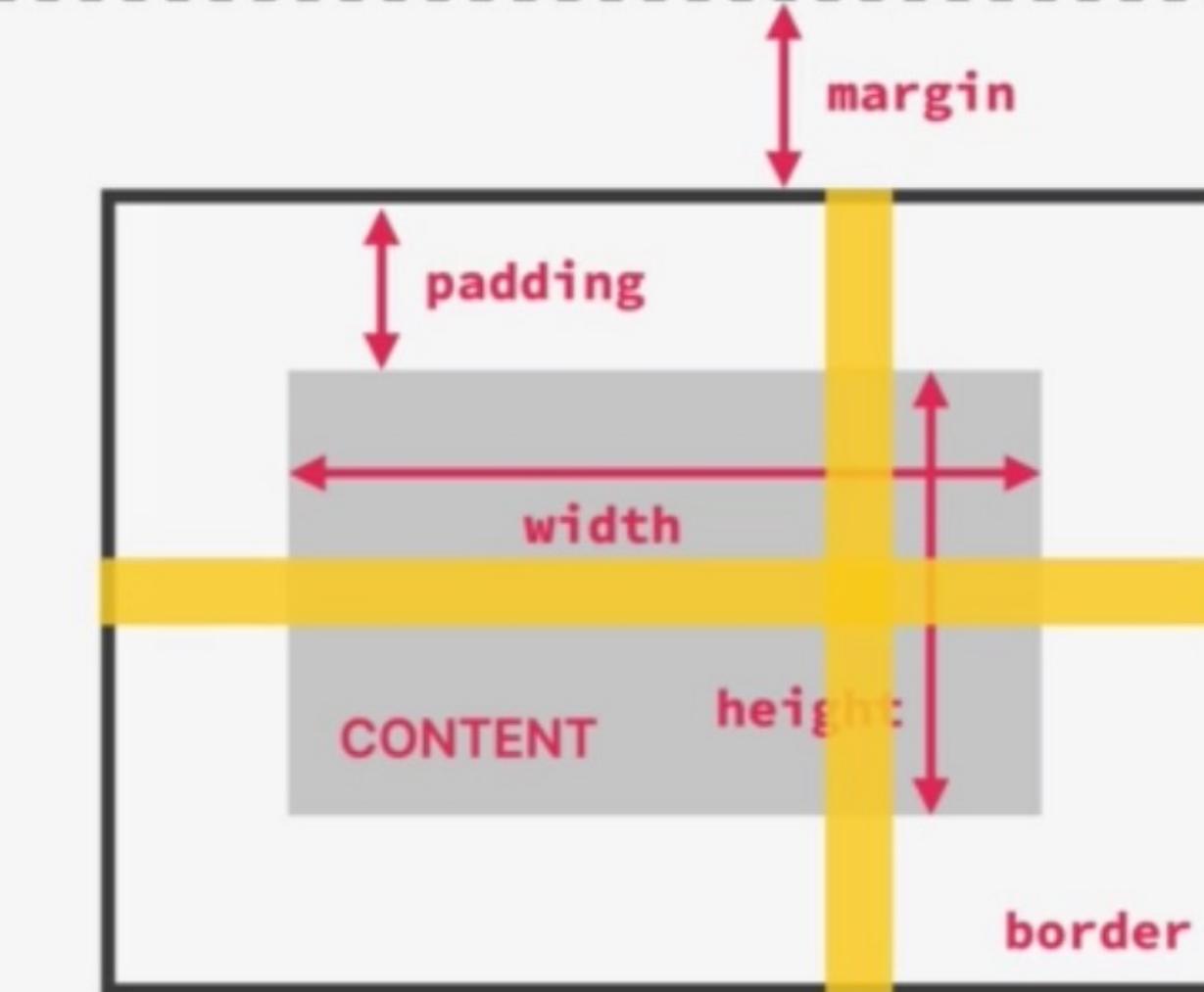
CSS Box modal

Final element width = left border + left padding + width + right padding + right border

Final element height = top border + top padding + height + bottom padding + bottom border

👉 We can specify all these values using CSS properties

⌚ This is the default behavior, but we can change it



BLOCK-LEVEL ELEMENTS

- 👉 Elements are formatted visually as **blocks**
- 👉 Elements occupy **100% of parent element's width**, no matter the content
- 👉 Elements are **stacked vertically** by default, one after another
- 👉 The box-model **applies as showed** earlier

Default elements: body, main, header, footer, section, nav, aside, div, h1-h6, p, ul, ol, li, etc.

With CSS: display: block

The Basic Language of the Web: HTML



Posted by **Laura Jones** on Monday, June 21st 2027

```
> <!> https://www.google.com/chrome</a>
<!> <div class="header-black-bg"></div>
<!> <!--NEED FOR TRANSPARENT HEADER IN MOBILE-->
> <header id="nav" class="header header-3"><a href="https://www.google.com/chrome">Google Chrome</a>
<!> <!--FEATURES 7 HALF IMG-->
> <div class="page-section bg-gray-light clearfix">
  ::before
    <div class="fes7-img-cont col-md-5">
      <div class="fes7-img" style="background-image: url('https://www.google.com/chrome/img/icon_1x.png');"></div>
    </div>
  <div class="container"></div>
  ::after
    </div>
```

All modern websites and web applications are built using three *fundamental* technologies: HTML, CSS and JavaScript. These are the languages of the web.

In this post, let's focus on HTML. We will learn what HTML is all about, and why should learn it.

What is HTML?

Ipsum dolor sit amet consectetur adipisicing elit. Quam recusandae reprehenderit vitae ratione veritatis corrupti sit ut vero, dolores nulla exercitationem eos quod iusto incident, preferendis alias tenetur. Est, vel!

In HTML, each element is made up of 3 parts:

1. **The opening tag**
2. **The closing tag**
3. **The actual element**

You can learn more at the [MDN Web Docs](#).

INLINE ELEMENTS

- 👉 Occupies only the space **necessary for its content**
- 👉 Causes **no line-breaks** after or before the element
- 👉 Box model applies in a different way: **heights and widths do not apply**
- 👉 **Paddings and margins** are applied **only horizontally** (left and right)

Default elements: a, img, strong, em, button, etc.

With CSS: display: inline

The Basic Language of the Web: HTML



Posted by **Laura Jones** on Monday, June 21st 2027

```
<!-->
<!--HEADER BOXED FONT WHITE TRANSPARENT-->
<div class="header-black-bg"></div>
<!--NEED FOR TRANSPARENT HEADER IN MOBILE-->
> <header id="nav" class="header header-1 header-dotted header-transparent header-white">
  <!--FEATURES 7 HALF IMG-->
  <div class="page-section bg-gray-light clearfix">
    ::before
    <div class="fes7-img-cont col-md-5">
      <div class="fes7-img" style="background-image: url('https://www.google.com/chrome%20target=_blank');"></div>
    </div>
    <div class="container"></div>
    ::after
  </div>
</header>
```

All modern websites and web applications are built using three **fundamental** technologies: HTML, CSS and JavaScript. These are the languages of the web.

In this post, let's focus on HTML. We will learn what HTML is all about, and why should learn it.

What is HTML?

Quam ipsum dolor sit amet consectetur adipisicing elit. Quam recusandae reprehenderit vitae ratione veritatis corrupti sit ut vero, dolores nulla exercitationem eos quod iusto incident, preferendis alias tenetur. Est, vel!

In HTML, each element is made up of 3 parts:

1. The opening tag
2. The closing tag
3. The actual element

BLOCK-LEVEL BOXES

- 👉 Elements formatted visually as blocks
- 👉 100% of parent's width
- 👉 Vertically, one after another
- 👉 Box-model applies as showed

INLINE-BLOCK BOXES

- 👉 Looks like inline from the **outside**, behaves like block-level on the **inside**
- 👉 Occupies only content's space
- 👉 Causes no line-breaks
- 👉 Box-model applies as showed

display: inline-block

INLINE BOXES

- 👉 Occupies only content's space
- 👉 Causes no line-breaks
- 👉 Box model is different: heights and widths do not apply
- 👉 Paddings and margins only horizontal (left and right)



Positioning

W3school examples:

https://www.w3schools.com/css/css_positioning.asp



Positioning

NORMAL FLOW

- 👉 Default positioning
- 👉 Element is “**in** flow”
- 👉 Elements are simply laid out according to their order in the HTML code

Default positioning
`position: relative`

ABSOLUTE POSITIONING

- 👉 Element is removed from the normal flow: “**out of** flow”
- 👉 No impact on surrounding elements, might overlap them
- 👉 We use top, bottom, left, or right to offset the element from its **relatively positioned container**

`position: absolute`



Positioning



Units

- 像素 (px) 是一种绝对单位 (absolute units) , 因为无论其他相关的设置怎么变化, 像素指定的值是不会变化的。其他的绝对单位如下 :
- mm, cm, in
- pt, pc: 点 (Points (1/72 of an inch)) , 十二点活字 (picas (12 points.))



- Screen measurements
 - pixels(px)
- Relative measurements
 - %,em, pt
 - kyleschaeffer.com/development/css-font-size-em-vs-px-vs-pt-vs/
- Color
 - #rrggb , rgb(r,g,b), rgba(r,g,b,a)
- Fonts
 - sans-serif, google fonts(<https://fonts.googleapis.com/>)
- Functional notation
 - **background:**
`url(http://www.example.org/image);`



Units

- vw, vh
- %
- 0
- auto



Units

em

- 1em与当前元素的字体大小相同（更具体地说，一个大写字母M的宽度）。CSS样式被应用之前，浏览器给网页设置的默认基础字体大小是16像素，这意味着对一个元素来说1em的计算值默认为16像素。但是要小心—em单位是会继承父元素的字体大小，所以如果在父元素上设置了不同的字体大小，em的像素值就会变得复杂。

Units

rem

- REM (root em) 和em以同样的方式工作，但它总是等于默认基础字体大小的尺寸；继承的字体大小将不起作用，所以这听起来像一个比em更好的选择



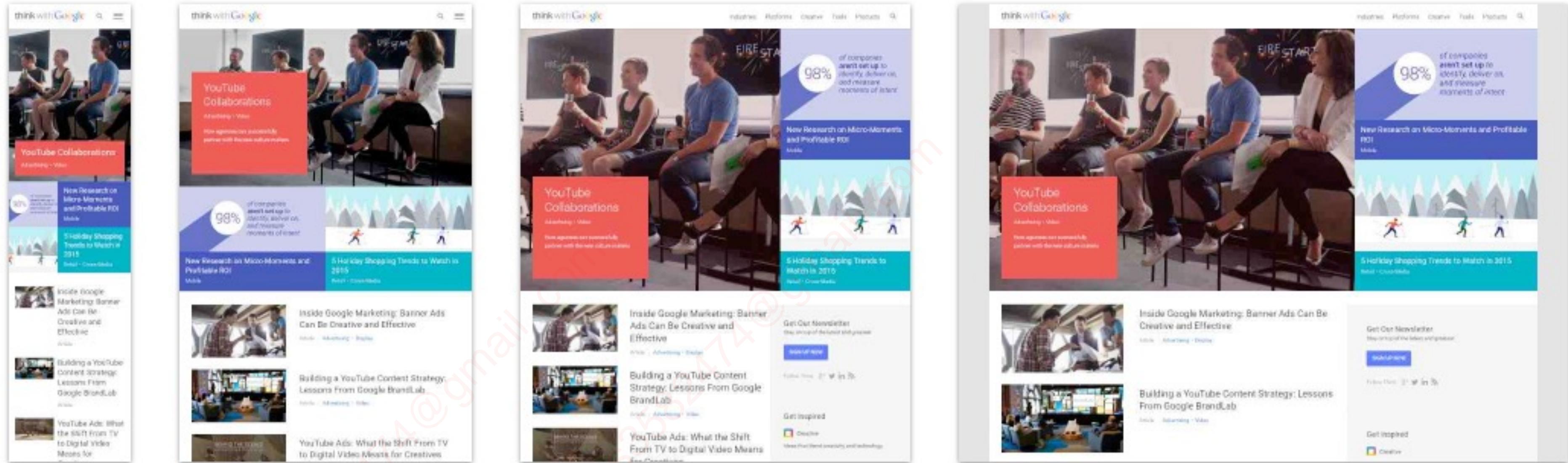
Normalize CSS

Normalize.css

makes browsers render all elements more consistently and in line with modern standards. It precisely targets only the styles that need normalizing.



Responsive Web Design



[https://developers.google.com/web/fundamentals
als/design-and-
ui/responsive/videos/resize.webm](https://developers.google.com/web/fundamentals/als/design-and-ui/responsive/videos/resize.webm)



Popular CSS Library



[Bootstrap](#)



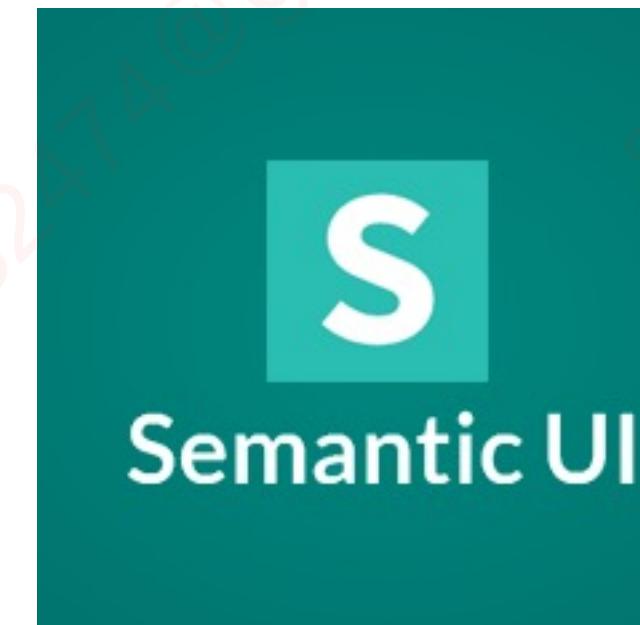
[Foundation](#)



[Pure CSS](#)



[Mobi CSS](#)



[Semantic UI](#)

推荐阅读 [The 5 Most Popular Frontend Frameworks of 2017 Compared](#)

Assignment 1

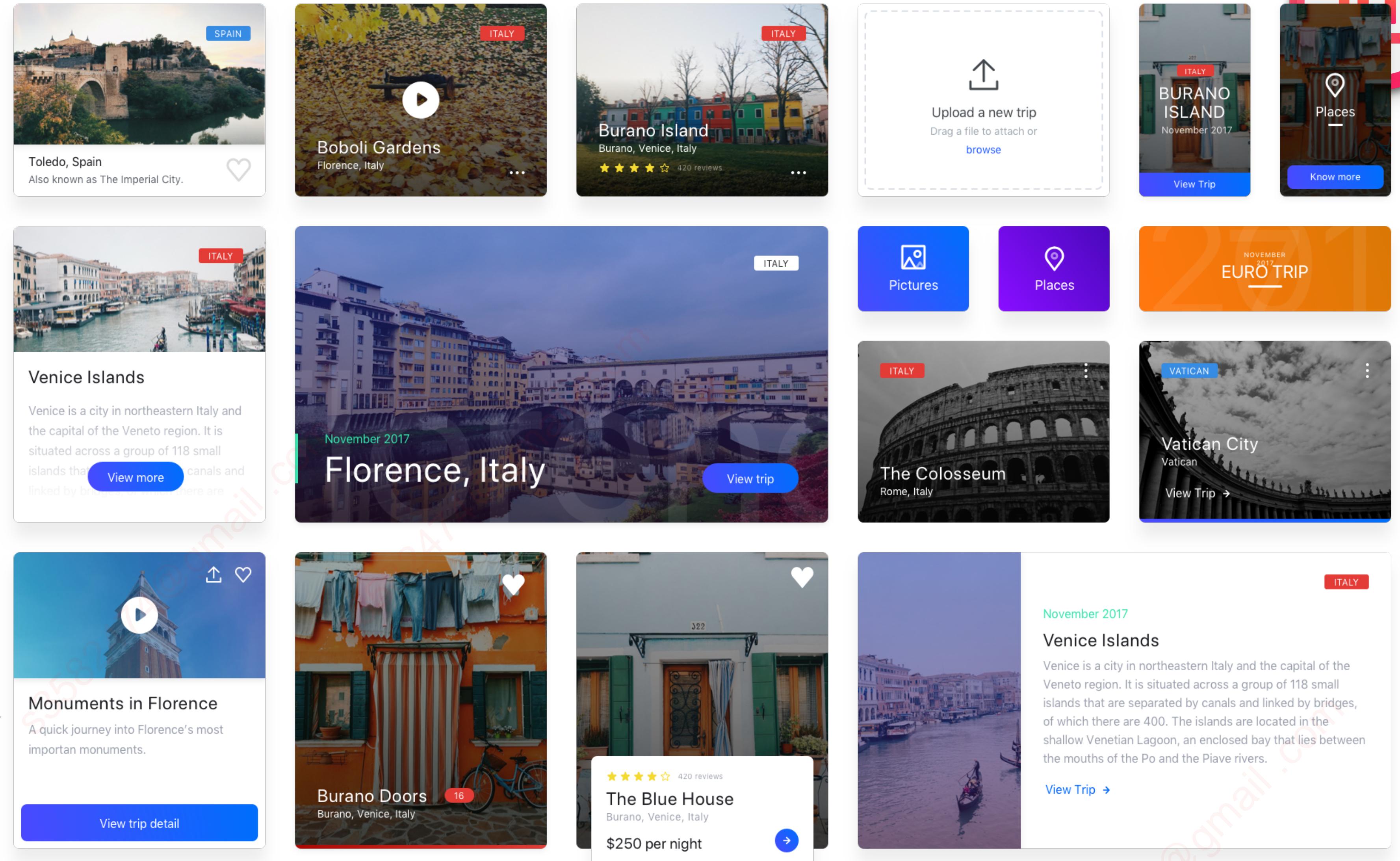
Card Design

要求

挑选一个card
按照设计完成
使用html, CSS, BEM

Deliver:

上传到Google Drive下
Assignment目录





CSS Resources

<https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>

<https://caniuse.com/>



联系方式

1. 微信联系求职顾问



匠人小花



悉尼匠人小嘿



墨尔本匠人小萌



阿德匠人小德

2. 匠人官网报名

Jiangren.com.au



澳洲匠人学院

电话 : 0421-672-555

微信 : jracademy

邮箱 : hello@jiangren.com.au

网址 : jiangren.com.au



匠人找工

免费发雇主招聘信息，免费发招聘者求职信息

微信 : jiangren_jobs