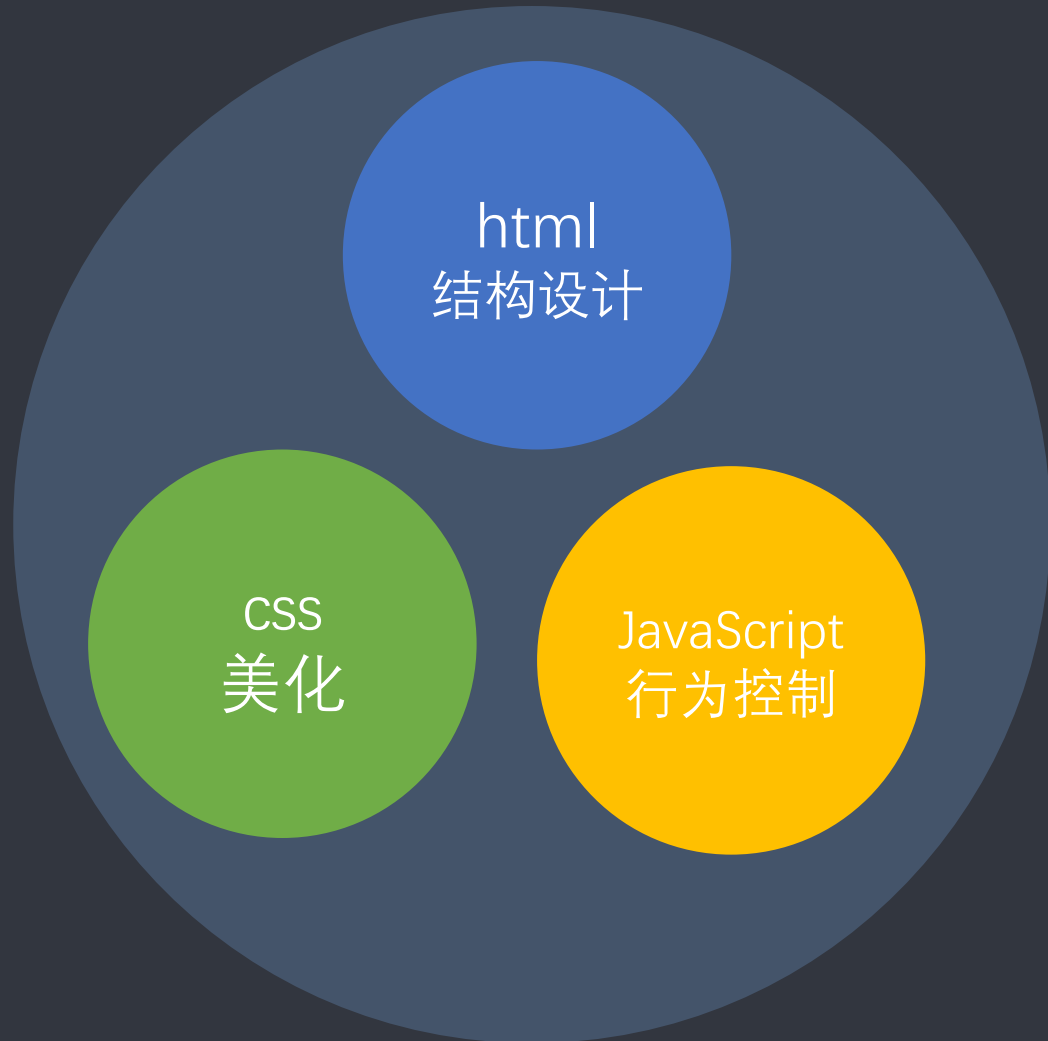


# CSS极速入门 —— 选择器

主讲人与课程设计：耕耕

wx: suency

# HTML, **CSS**, JAVASCRIPT 三者关系



*As you can see, **HTML** can make something.*

*Then **CSS** can make it beautiful*

*So, **JavaScript** make it to do something*

# 内容提纲

- 概念
- 引入方式
- 语法



基础

- 选择器
- 常用属性
- 盒子模型
- 定位方式
- 布局



重点核心

- BFC
- 单位
- 高级技巧
- 常见问题?



补充内容

# 什么是css

1. CSS 指层叠样式表 (Cascading Style Sheets)
2. 样式定义如何显示 HTML 元素
3. 给html打扮穿衣服

HTML



CSS



JavaScript



# 引入方式

1. 外部样式表 —— 在html中使用<link>标签, 引入外部的文件

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

2. 内嵌套样式表 —— 在html中使用<style>标签, 内容写在该标签里面

```
<style type="text/css">
    .main{
        color: pink;
    }
</style>
```

3. 内联样式表 —— 在html任意标签中使用style属性, 内容是属性值

```
<div class="main" style="color:purple;">aaa</div>
```

# 基础语法



# 第一个案例 —— 尝鲜

## html代码

```
<div></div>
```

## css代码

```
div {  
    width: 200px;  
    height: 200px;  
    background-color: pink;  
}
```

效果：div元素宽度200像素，高度200像素，并且背景颜色为粉色

# CSS总览

定位 & 布局	尺寸 & 补白	边框	背景 & 颜色	字体 & 文本	文本 & 书写	列表 & 表格	用户界面
<b>定位</b> position z-index top right bottom left clip	<b>尺寸</b> width min-width max-width height min-height max-height	<b>border</b> border-width border-style border-color border-top border-top-width border-top-style border-top-color border-right border-right-width border-right-style border-right-color border-bottom border-bottom-width border-bottom-style border-bottom-color border-left border-left-width border-left-style border-left-color border-radius border-top-left-radius border-top-right-radius border-bottom-right-radius border-bottom-left-radius box-shadow border-image border-image-source border-image-slice border-image-width border-image-outset border-image-repeat	<b>背景</b> background background-color background-image background-repeat background-attachment background-position background-origin background-clip background-size	<b>字体</b> font font-style font-variant font-weight font-size font-family font-stretch font-size-adjust	<b>文本装饰</b> text-decoration text-decoration-line text-decoration-color text-decoration-style text-decoration-skip text-underline-position text-shadow	<b>列表</b> list-style list-style-image list-style-position list-style-type	<b>appearance</b> text-overflow outline outline-width outline-color outline-style outline-offset nav-index nav-up nav-right nav-down nav-left cursor zoom box-sizing resize ime-mode user-select pointer-events
<b>布局</b> display float clear visibility overflow overflow-x overflow-y	<b>外补白</b> margin margin-top margin-right margin-bottom margin-left	<b>border-bottom</b> border-bottom-width border-bottom-style border-bottom-color border-left border-left-width border-left-style border-left-color border-radius border-top-left-radius border-top-right-radius border-bottom-right-radius border-bottom-left-radius box-shadow border-image border-image-source border-image-slice border-image-width border-image-outset border-image-repeat	<b>颜色</b> color opacity <color> Color Name HEX RGB RGBA HSL HSLA transparent currentColor	<b>文本</b> text-transform white-space tab-size word-break word-wrap overflow-wrap text-align text-align-last text-justify word-spacing letter-spacing text-indent vertical-align line-height text-size-adjust	<b>书写模式</b> direction unicode-bidi writing-mode	<b>表格</b> table-layout border-collapse border-spacing caption-side empty-cells	<b>内容</b> content counter-increment counter-reset quotes
<b>特殊布局</b> <fraction> <grid> fr gr	<b>内补白</b> padding padding-top padding-right padding-bottom padding-left	<b>媒体查询</b> width height device-width device-height orientation aspect-ratio device-aspect-ratio color color-index monochrome resolution scan grid	<b>基本选择器</b> 元素选择符 通配选择符(*) 类型选择符(E) ID选择符(E#id) 类选择符(E.class)	<b>伪选择器</b> 伪类选择符 E.link E.visited E.hover E.active E.focus E.lang(fr) E.not(s) E.root E.first-child E.last-child E.only-child E.nth-child(n) E.nth-last-child(n) E.first-of-type E.last-of-type E.only-of-type E.nth-of-type(n) E.nth-last-of-type(n) E.empty E.checked E.enabled E.disabled E.target @page.first @page.left @page.right	<b>多列</b> columns column-width column-count column-gap column-rule column-rule-width column-rule-style column-rule-color column-span column-fill column-break-before column-break-after column-break-inside	<b>伸缩盒</b> 伸缩盒(旧) box-orient box-pack box-align box-flex box-flex-group box-ordinal-group box-direction box-lines	<b>变换 &amp; 动画</b> 变换 transform transform-origin transform-style perspective perspective-origin backface-visibility
<b>浏览器属性</b> Only IE scrollbar-3dlight-color scrollbar-darkshadow-color scrollbar-highlight-color scrollbar-shadow-color scrollbar-arrow-color scrollbar-face-color scrollbar-track-color scrollbar-base-color filter behavior	<b>函数</b> 函数 calc() toggle()	<b>度量</b> 角度 <angle> deg grad rad turn 时间 <time> s ms 频率 <frequency> Hz kHz	<b>关系选择符</b> 包含选择符(E F) 子选择符(E>F) 相邻选择符(E+F) 兄弟选择符(E~F)	<b>伪类选择符</b> E.link E.visited E.hover E.active E.focus E.lang(fr) E.not(s) E.root E.first-child E.last-child E.only-child E.nth-child(n) E.nth-last-child(n) E.first-of-type E.last-of-type E.only-of-type E.nth-of-type(n) E.nth-last-of-type(n) E.empty E.checked E.enabled E.disabled E.target @page.first @page.left @page.right	<b>长度</b> <length> em ex ch rem vw vh vmax vmin cm mm q in pt pc px	<b>伸缩盒(新)</b> flex flex-grow flex-shrink flex-basis flex-flow flex-direction flex-wrap align-content align-items align-self justify-content order	<b>过渡</b> transition transition-property transition-duration transition-timing-function transition-delay
<b>Only webkit</b> -webkit-text-fill-color -webkit-text-stroke -webkit-text-stroke-width -webkit-text-stroke-color -webkit-box-reflect -webkit-tap-highlight-color -webkit-user-drag -webkit-line-clamp -webkit-overflow-scrolling	<b>生成内容</b> counter() counters() attr()	<b>语法与规则</b> !important /*comment*/ @import @charset @media @font-face @page @keyframes @supports	<b>属性选择符</b> E[att] E[att="val"] E[att!="val"] E[att\$="val"] E[att*="val"] E[att =“val”]	<b>伪类选择符</b> E.link E.visited E.hover E.active E.focus E.lang(fr) E.not(s) E.root E.first-child E.last-child E.only-child E.nth-child(n) E.nth-last-child(n) E.first-of-type E.last-of-type E.only-of-type E.nth-of-type(n) E.nth-last-of-type(n) E.empty E.checked E.enabled E.disabled E.target @page.first @page.left @page.right	<b>数字</b> <number> <integer> <percentage>	<b>图像</b> <image> image() image-set() <gradient> linear-gradient() radial-gradient() repeating-linear-gradient() repeating-radial-gradient()	<b>动画</b> animation animation-name animation-duration animation-timing-function animation-delay animation-iteration-count animation-direction animation-play-state animation-fill-mode
			<b>数字</b> <number> <integer> <percentage>	<b>数字</b> <number> <integer> <percentage>			<b>杂项</b> CSS Hack 条件Hack 属性级Hack 选择符级Hack 打印 page page-break-before page-break-after page-break-inside



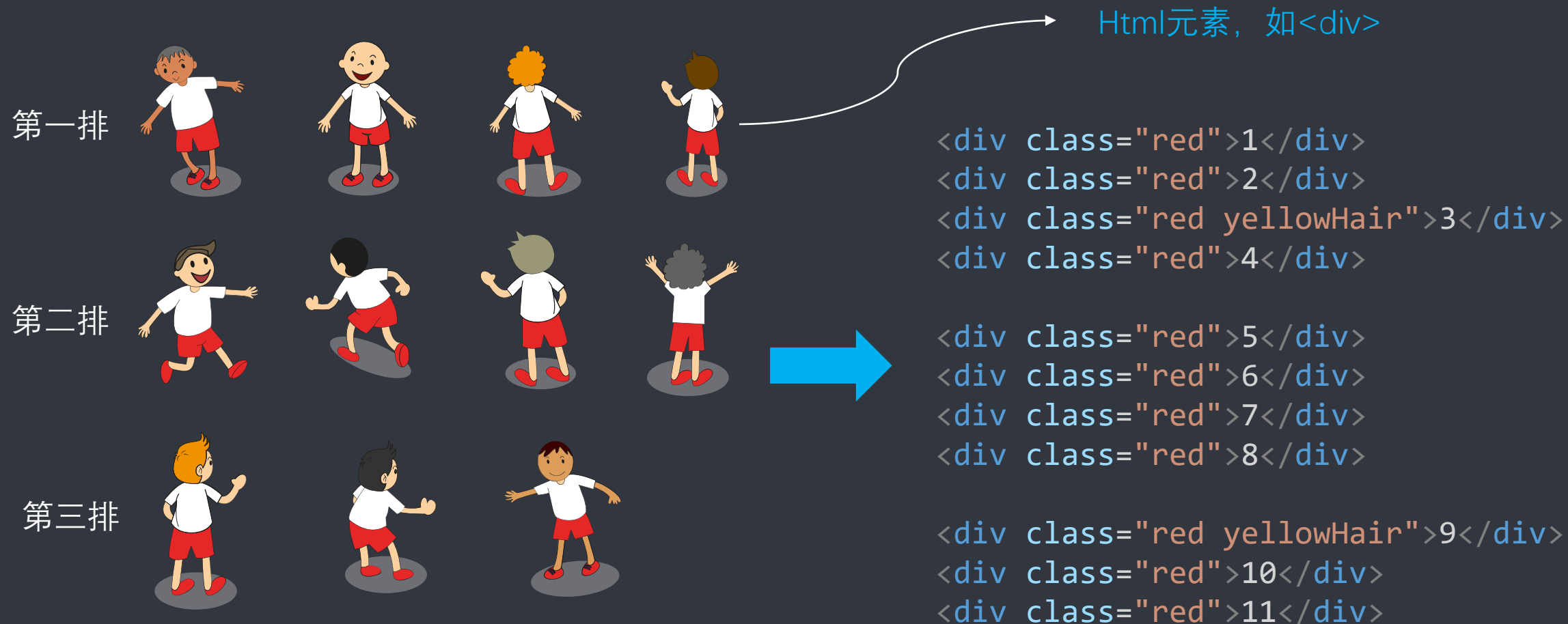
# 核心掌握属性总结

外边距 (margin) , 内边距 (padding) , 边框 (border) , 尺寸 (width, height) , 背景 (background) , 颜色 (color) , 字体 (font) , 文本 (text) , 布局属性, 选择器

选择器 (selector)

# 选择器是什么

在 CSS 中，选择器是一种模式，用于**选择需要添加样式的元素**。



# 选择器 (selector)

1. id选择器 ( # myid)

2. 类选择器 (.main)

3. 标签选择器 (div, h1, p)

4. 相邻选择器 (h1 + p)

5. 兄弟选择器 (h1 ~ p)

6. 子选择器 (div>span)

7. 后代选择器 (div span)

8. 通配符选择器 ( \* )

9. 属性选择器 (a[rel = "external"])

10. 伪类选择器 (p: hover)

# 选择器权重

important的权重为**权益提升，最高级**

内联样式的权重是**1000**

ID选择器的权重为**100**

**类选择器**的权重为**10** —— **最常用**

标签选择器的权重为**1**

伪类选择器的权重为**10**

属性选择器的权重为**10**

伪对象选择器的权重为**1**

通配符选择器的权重为**0**

注：数字越大，权重越高，优先级也越高

## ID选择器：规定用#来定义（名字自定义）

针对某一个特定的标签来使用，**只能使用一次**。css中的ID选择器以“#”来定义。

**Html代码**      `<h2 id="mytitle">你好</h2>`

**css代码**            `#mytitle {  
                 border: 3px dashed green;  
                 }`

注意：id选择器的选择符是“#”，任何的HTML标签都可以有id属性

## 类选择器：规定用圆点.来定义

和id非常相似，任何的标签都可以携带id属性和class属性。**类选择器可以使用多次。**

### Html代码

```
<h2 class="oneclass">你好</h2>  
<div class="oneclass">你好2</div>
```

### css代码

```
.oneclass {  
    color: red;  
}
```

注意：类选择器可以被多种标签使用，同一个标签可以使用多个类选择器。用空格隔开。

## 标签选择器：选择器的名字代表html页面上的标签

标签选择器，选择的是页面上所有这种类型的标签，所以经常描述“共性”，无法描述某一个元素的“个性”。

Html代码 `<div>你好2</div>`

Css代码 

```
div {  
    color: red;  
}
```

注意：所有的标签，都可以是选择器。选择的所有，而不是一个。



## 子选择器：E>F

选择E下面的所有儿子F，**只有儿子**，没有孙子，重孙子等等

Html代码

```
<div class="parent">
  <div class="son">
    son
    <div>sonson</div>
  </div>
</div>
```

css代码

```
.parent>div {
  width: 100px;
  height: 100px;
}
```

## 后代选择器： E F

选择E下面的所有后代， **包括儿子，孙子，重孙子**等等

Html代码

```
<div class="parent">
  <div class="son">
    son
    <div>sonson</div>
  </div>
</div>
```

css代码

```
.parent div {
  width: 100px;
  height: 100px;
}
```

## 相邻选择器： E + F

选择E下一个邻居， **只是下一个且只有一个**

Html代码

```
<div class="brother">brother1</div>  
<p>brother2</p>  
<p>brother3</p>
```

css代码

```
.brother+p {  
    font-size: 100px;  
}
```

# 兄弟选择器： E ~ F

选择E下面**所有邻居**

Html代码

```
<div>brother0</div>  
<div class="brother">brother1</div>  
<p>brother2</p>  
<p>brother3</p>
```

css代码

```
.brother~p {  
    font-size: 100px;  
}
```

## 通配符选择器: \*

选择所有

Html代码

```
<div class="all">  
  <span>name</span>  
  <p>abc</p>  
</div>
```

css代码

```
.all * {  
  color: purple;  
  font-size: 50px;  
}
```

## 属性选择器: [attribute=value]

[attribute], [attribute=value], [attribute~=value], [attribute|=value],  
[attribute^=value], [attribute\*=value], [attribute\$=value]

**Html代码** `<div name="haha">abc</div>`

**css代码** `[name="haha"] {  
 font-size: 100px;  
}`

[attr|"bar"]'attr'属性值开头必须是bar的单词, 或者开头是bar-

[attr^="val"] 值开头三个字符必须是val

[attr\$="val"]属性值最后三个字符必须是val

[attr\*="val"]属性值任意位置包含val这三个字符即可

[attr~="val"]属性值任意位置包含val这单词

## 伪类选择器：（冒号：）

CSS 伪类用于向某些选择器添加特殊的效果。

Html代码

```
<div class="wei">abc</div>
```

css代码

```
.wei:hover {  
    color: skyblue;  
}
```

# 交集选择器

## Html代码

```
<div class="jiao">abc</div>  
<div>  
  <p class="jiao">cdf</p>  
</div>
```

## css代码

```
div.jiao {  
  font-size: 100px;  
}
```



# 问题1

这么多选择器如何选择？