

CSS 基础概念

从入门到工作：CSS全解

版权声明

本内容版权属杭州饥人谷教育科技有限公司（简称饥人谷）所有。

任何媒体、网站或个人未经本网协议授权不得转载、链接、转贴，或以其他方式复制、发布和发表。

已获得饥人谷授权的媒体、网站或个人在使用时须注明「资料来源：饥人谷」。

对于违反者，饥人谷将依法追究 responsibility。

联系方式

如果你想要购买本课程

请微信联系 [xiedaimala02](#) 或 [xiedaimala03](#)

如果你发现有人盗用本课程

请微信联系 [xiedaimala02](#) 或 [xiedaimala03](#)

目录

- CSS 的历史
- CSS 是艺术
- 体系化学习
- 文档流
 - ✓ Normal Flow
- 盒模型
 - ✓ content box & border box

CSS 是谁发明的

李爵士的挪威同事赖先生首先提出 CSS

Håkon Wium Lie

赖先生生平

• 生平

- ✓ 1991年获得MIT媒体实验室视觉研究理学硕士学位
- ✓ 1994年提出CSS概念而闻名
- ✓ 1999年任Opera的CTO
- ✓ 2005年他写公开信给比尔盖茨问为什么IE不支持Web标准，盖茨说IE 7马上发布，他写了 Acid2 用来验证
- ✓ 2006年通过了博士论文答辩
- ✓ 2006年他倡议 Web 网页应使用通用字体格式
- ✓ 2007年他倡议浏览器可以支持video标签

• 观点

- ✓ 微软的IE常常成为他的批评对象
- ✓ 他也是Web打印概念的倡导者，用HTML和CSS写书

CSS 的牛 X 之处在哪？

层叠样式表

层叠指什么？

- 样式层叠

- ✓ 可以多次对同一选择器进行样式声明

- 选择器层叠

- ✓ 可以用不同选择器对同一个元素进行样式声明

- 文件层叠

- ✓ 可以用多个文件进行层叠

- 这些特性使得 CSS 极度灵活

- ✓ 这也为 CSS 后来被吐槽留下了隐患

CSS 的版本

版本	时间	重点
CSS 1	1996年	不用管
CSS 2	1998年	添加定位、z-index、media，不用管
CSS 2.1	2004~2011年	使用最广泛的版本（IE支持）
CSS 3	1999年开始起草	现代版本，分模块（IE 8 部分支持）
CSS 4*	分模块升级	

我怎么知道 哪些浏览器支持哪些特性呢

方法一：几十种浏览器全部跑一遍

方法二：使用 caniuse.com

caniuse.com 使用方法

- 请收藏此网站

- ✓ 输入你关心的样式，比如 border-radius 或 filter
- ✓ 查看大部分浏览器的支持情况
- ✓ 如果想看更多，点击 Show All
- ✓ 下方会详细说明兼容 bug 有哪些（翻译成中文）

- 这个网站是怎么运作的

- ✓ 网站主一开始自己测试了一部分浏览器
- ✓ 社区的前端工程师帮助测试各种各样的浏览器
- ✓ UC 浏览器和 QQ 浏览器当然主要是中国开发者测试
- ✓ 大家把测试结果提交到 GitHub 上
- ✓ 这就是开发者社区的力量

CSS 是艺术

就像画画、折纸

你需要用感性思维来理解 CSS

不要用理性思维

即不要问「为什么会这样」

而是说「原来是这样」

浏览器说是怎么样，就是怎么样

当然有极少情况是浏览器出错了

为什么 `color: red` 能让字变红

不要问「为什么」

要说「原来是这样」

所见即所学

体系化学习

- 学一门语言必须学会什么

- ✓ 语法（怎么写代码）
- ✓ 如何调试（怎么知道自己代码写错了）
- ✓ 在哪查资料（其实就是为了抄代码）
- ✓ 标准制定者是谁

- 如何学

- ✓ Copy - 抄文档、抄老师
- ✓ Run - 放在自己的机器上运行成功
- ✓ Modify - 加入一点自己的想法，然后重新运行

语法超级简单

• 语法一：样式语法

```
选择器 {  
    属性名: 属性值;  
    /*注释*/  
}
```

• 注意事项

- ✓ 所有符号都是英文符号，如果写错了，浏览器会警告
- ✓ 区分大小写，a 和 A 是不同的东西
- ✓ 没有 // 注释
- ✓ 最后一个分号可以省略，但建议不要省略
- ✓ 任何地方写错了，都不会报错，浏览器会直接忽略
- ✓ 那我怎么知道自己写没写错呢？一会说

语法超级简单

- 语法二： at 语法

```
@charset "UTF-8";  
@import url(2.css);  
@media (min-width: 100px) and (max-width: 200px) {  
    语法一  
}
```

- 注意事项

- ✓ @charset 必须放在第一行
- ✓ 前两个 at 语法必须以分号 ; 结尾
- ✓ @media 语法会单独教学
- ✓ charset 是字符集的意思，但 UTF-8 是字符编码 encoding，这是历史遗留问题

如何调试？

怎么知道自己的代码对还是错？

如何调试 CSS

- 方法

- ✓ 使用 W3C 验证器 ([在线](#) / 命令行工具) 不用试了
- ✓ 使用 VSCode 看颜色
- ✓ 使用 WebStorm 看颜色
- ✓ 使用开发者工具看警告

- 如何使用开发者工具

- ✓ 找到你脑中的标签
- ✓ 看它是否有选择器
- ✓ 看它的样式是否被划掉
- ✓ 看它的样式是否有警告

Border

调试法

Border 调试法

- 步骤

- ✓ 怀疑某个元素有问题
- ✓ 就给这个元素加 border
- ✓ border 没出现？说明选择器错了或者语法错了
- ✓ border 出现了？看看边界是否符合预期
- ✓ bug 解决了才可以把 border 删掉

- 记住

- ✓ CSS 的 border 调试法
- ✓ 就相当于 JS 的 log 调试法
- ✓ 我会再每节课重复这个调试法

新人常见错误

- 低级错误

- ✓ 选择器拼写错误
- ✓ 属性名拼写错误
- ✓ 属性值拼写错误
- ✓ 大小写错误
- ✓ 没写分号
- ✓ 中文冒号
- ✓ 没写反花括号
- ✓ 没加单位

- 非低级错误

- ✓ 没有非低级错误

在哪查资料

- 网站推荐

- ✓ Google 搜索关键词时加 MDN
- ✓ CSS tricks (英文)
- ✓ 张鑫旭的博客

- 书籍推荐

- ✓ 不推荐买任何书
- ✓ CSS 和 HTML 一样，以练为主

在哪搜练习素材

- PSD

- ✓ Freepik 搜索 [PSD web](#)
- ✓ 如果下载慢，就把域名加入翻墙插件
- ✓ 中文免费 PSD 网站较少，需要多搜一下
- ✓ 365PSD 里的 [UI 套件](#) 还行

- 效果图（不提供下载）

- ✓ dribbble.com 顶级设计师社区
- ✓ 可以用肉眼模仿它

- 商业网站

- ✓ 直接模仿你常去的网站

不要沉迷临摹

每个类型的临摹一两个即可

PC 网站、手机网站、UI 套件

再多无益

标准制定者是谁

W3C

搜索 CSS spec 可以找到 CSS 最新标准

没人能看完它

你可以看看 CSS 2.1 标准的[中文版](#)

开始 CRM 学习法

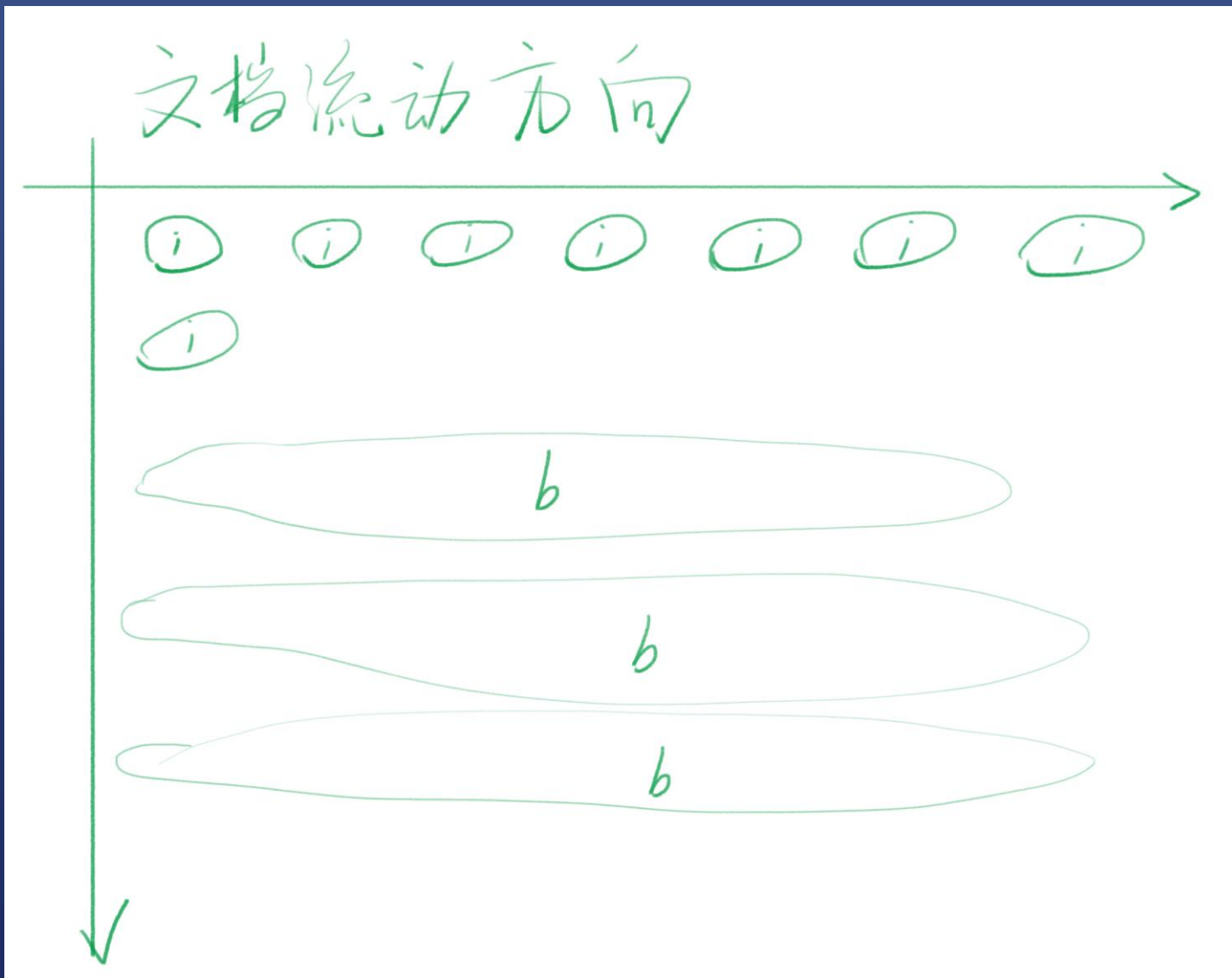
抄 - 运行 - 改

基本概念

- 要理解几个重要的概念
 - ✓ 文档流 Normal Flow
 - ✓ 块、内联、内联块
 - ✓ margin 合并
 - ✓ 两种盒模型（border-box 更符合人类思维）

新的html5规定，元素不分内联元素（span）和块级元素（div）。看样式。display:inline就是内联元素，display:block就是块级元素

文档流动方向



文档流

• 流动方向

- ✓ inline 元素从左到右，到达最右边才会换行
- ✓ block 元素从上到下，每一个都另起一行
- ✓ inline-block 也是从左到右 到达一行最后时不会把自己分成两块

• 宽度

inline里不能有block元素

- ✓ inline 宽度为内部 inline 元素的和，不能用 width 指定
- ✓ block 默认自动计算宽度，可用 width 指定 不是默认100%
- ✓ inline-block 结合前两者特点，可用 width

• 高度

大部分时候是一样的，但是如果字体不一致，line-height只能间接决定。（如100px左右）

- ✓ inline 高度由 line-height 间接确定，跟 height 无关 如果span里什么都没有，高度为line-height
- ✓ block 高度由内部文档流元素决定，可以设 height 如果div里什么都没有，高度为0
- ✓ inline-block 跟 block 类似，可以设置 height

overflow 溢出

- 当内容大于容器

- ✓ 等内容的宽度或高度大于容器的，会溢出
- ✓ 可用 overflow 来设置是否显示滚动条
- ✓ auto 是灵活设置 溢出时显示滚动条
- ✓ scroll 是永远显示 永远显示滚动条
- ✓ hidden 是直接隐藏溢出部分
- ✓ visible 是直接显示溢出部分
- ✓ overflow 可以分为 overflow-x 和 overflow-y

脱离文档流

- 回忆一下

- ✓ block 高度由内部文档流元素决定，可以设 height
- ✓ 这句话的意思是不是说，有些元素可以不在文档流中

- 哪些元素脱离文档流

- ✓ float
- ✓ position: absolute / fixed

- 怎么让元素不脱离文档流

- ✓ 不要用上面属性不就不脱离了

两种盒模型

- 分别是

- ✓ content-box 内容盒 - 内容就是盒子的边界
- ✓ border-box 边框盒 - 边框才是盒子的边界

- 公式

- ✓ content-box width = 内容宽度
- ✓ border-box width = 内容宽度 + padding + border

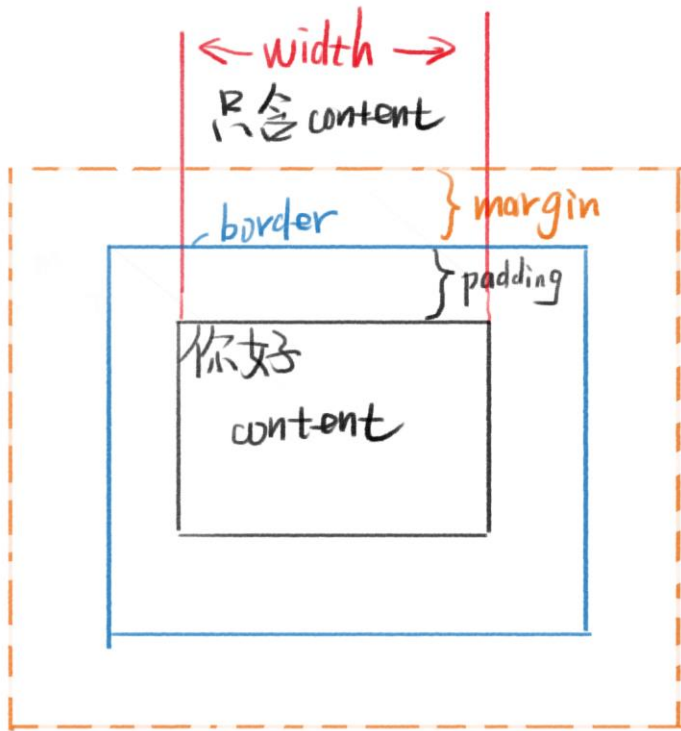
- 哪个好用

- ✓ border-box 好用
- ✓ 同时指定 padding、width、border 就知道为什么了

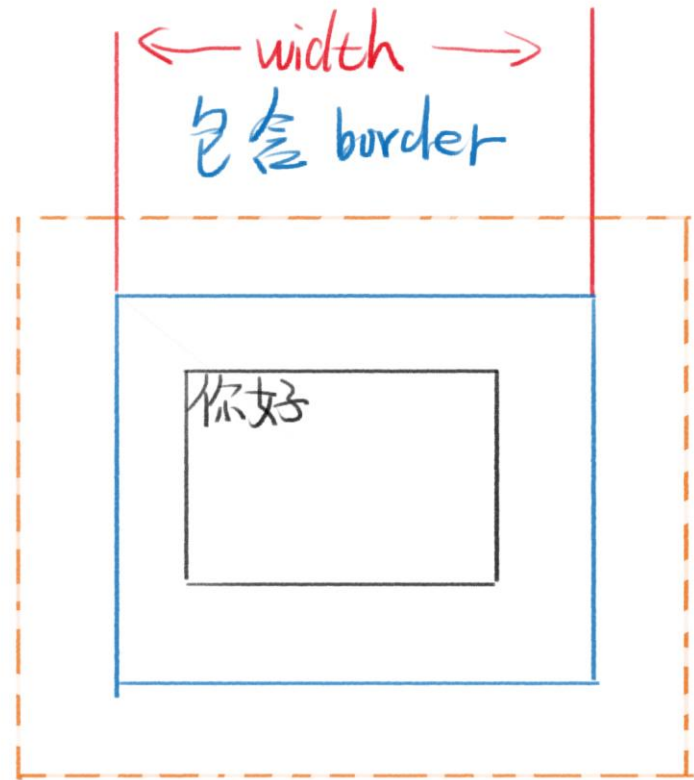
```
HTML +
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <title>JS Bin</title>
7 </head>
8
9 <body>
10  <div class="content-box">content box</div>
11  <div class="border-box">border box</div>
12 </body>
13
14 </html>
CSS +
1 .content-box{
2   margin: 25px;
3   border: 5px solid red;
4   padding: 15px;
5   box-sizing: content-box;
6   width: 100px;
7   height: 100px;
8 }
9
10 .border-box{
11   margin: 25px;
12   border: 5px solid red;
13   padding: 15px;
14   box-sizing: border-box;
15   width: 100px;
16   height: 100px;
17 }
```

content-box 和 border-box

content box



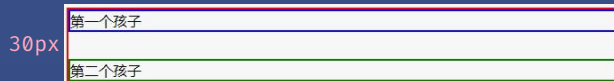
border box



margin 合并

(只有上下重叠, 左右不重叠)

```
<div class="parent">
  <div class="child1">第一个孩子</div>
  <div class="child2">第二个孩子</div>
</div>
```

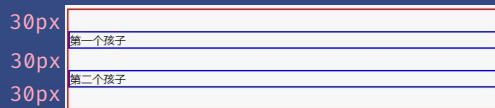


```
.parent{
  border:2px solid red;
}
.child1{
  border:2px solid blue;
  margin-bottom:30px;
}
.child2{
  border:2px solid green;
  margin-top:30px;
}
```

• 哪些情况会合并

- ✓ 父子 margin 合并
- ✓ 兄弟 margin 合并

```
<div class="parent">
  <div class="child">第一个孩子</div>
  <div class="child">第二个孩子</div>
</div>
```



```
.parent{
  border:2px solid red;
}
.child{
  border:2px solid blue;
  margin:30px 0; /*上下30px左右0*/
}
```

• 如何阻止合并

- ✓ 父子合并用 padding / border 挡住
- ✓ 父子合并用 overflow: hidden 挡住
- ✓ 父子合并用 display: flex, 不知道为什么
- ✓ 兄弟合并是符合预期的
- ✓ 兄弟合并可以用 inline-block 消除
- ✓ 总之要一条一条死记
- ✓ 而且 CSS 的属性逐年增多, 每年都可能新的

```
<div class="parent">
  <div class="child">第一个孩子</div>
  <div class="child">第二个孩子</div>
</div>
```

```
.parent{
  border:1px solid red; /*有border或者padding
  不会合并, 没有就会父子合并。*/
  margin:15px;
  background: green;
}
.child{
  border:2px solid blue;
  margin:30px 0; /*上下30px左右0*/
}
```

因为要把看不见的隐藏, 所以看不见的要包起来

为什么 CSS 不单独出个属性

用来禁用 margin 合并

You ask me, I ask who

你问我，我问谁

这就是 CSS 的恶心之处

不正交

基本单位

• 长度单位

- ✓ px 像素
- ✓ em 相对于自身 font-size 的倍数
- ✓ 百分数
- ✓ 整数
- ✓ rem: 等你把 em 滚瓜烂熟了再问 rem
- ✓ vw 和 vh
- ✓ 其他长度单位都用得很少, 不用了解

```
.div1{  
  font-size:20px;  
  border:1px solid red;  
  width:3em;  
}
```

• 颜色

- ✓ 十六进制 #FF6600 或者 #F60
- ✓ RGBA 颜色 rgb(0,0,0) 或者 rgba(0,0,0,1)
- ✓ hsl 颜色 hsl(360,100%,100%)

```
background: transparent;/*全透明*/
```

```
background: #FF6600;
```

```
background: rgb(123,0,255);
```

试一下做一个彩虹吧

定一个目标，然后搞定它

<http://js.jirengu.com/xacet/1/edit?html,css,output>

再见

下节课讲 CSS 布局