



# HTML&CSS1

林梓涵



# 简要内容

## ■ HTML

- 简介
- 标签
- 注释

## ■ CSS

- 简介
- 基础语法
- 选择器
- 伪类

HTML

# 简介

- HTML(HyperText Markup Language)不是一种编程语言，而是一种标记语言，它以标签的形式来描述语言
- 一个HTML文档包括标签和纯文本

# 标签

`<p>hello world</p>`

- `<p>`为开始标签，表示一个元素的开始
- `</p>`为结束标签，表示元素的结束
- 开始标签和结束标签之间的东西被称为内容
- 开始标签、结束标签、内容合在一起被称为元素
- 元素可以含有空内容
- 空元素在开始标签中结束
  - `<br>`, `<link>`, `<img>`...

# 标签

- 标签可以嵌套
- 标签可以含有属性
  - 属性值必须使用引号包裹，单引号或双引号皆可
  - class, id
- 元素分为块级元素和内联元素
  - 块级元素会出现在新行，而内联元素不会

```
<div class="mydiv">  
  <p>hello world</p>  
</div>
```

# 一些基本的标签

- `<!DOCTYPE>`

- `<!DOCTYPE>`标签用于声明文档类型，一般情况下只需要写`<!DOCTYPE html>`即可，不用闭合，位于最开始位置

- `<html>`

- `<html>`元素包裹着整个页面，它是页面的根元素

- `<meta>`

- 位于文档头部，提供有关页面的元信息，如charset

- `<title>`

- 位于文档头部，用于定义文档标题，显示在浏览器标签上

- `<body>`

- 包含了所有显示在页面上的内容

# 一些基本的标签

- `<head>`
  - 用于定义文档的头部，包含了所有头部元素，处于`<html>`标签之后和`<body>`标签之前
- `<h1>-<h6>`
- `<p>`
- `<div>`
- `<form>`
- `<input>`
- `<button>`
- `<!--...-->`注释标签
- ...



CSS

# 简介

- CSS(Cascading Style Sheets)层叠样式表
- HTML仅仅决定网页的内容，CSS被设计出来决定网页的外观
- 样式可存在于四处：
  - 浏览器缺省设置
  - 外部样式表
  - 内部样式表（位于<head>标签内部）
  - 内联样式（在HTML元素内部）
- 优先级从上至下依次升高

# 基础语法

- CSS规则包括选择器以及若干条声明
  - 每条声明包括一个属性和一个值
  - 选择器可以分组
  - 一般情况下，子元素继承父元素的属性
  - <https://developer.mozilla.org/zh-CN/>
  - <http://www.w3school.com.cn>
  - CSS属性无法具体讲述，请在上述网页查阅
- 
- `Selector{property1: value1; property2: value2}`
  - `p, h1 {color: black; font-family:"Source Han Serif";}`

# 选择器

## ■ 后代选择器

- `<div><p>some word</p></div><p></p>`
- `div p {color: red;}`

## ■ id选择器

- `<p id = "p0">some word</p>`
- `#p0 {color: red;}`

## ■ 类选择器

- `<p class="p0">some word</p><h1 class = "p0">some word</h1>`
- `.p0 {color: red;}`
- `p.p0 {color: red;}`

# 选择器

- 属性选择器

- `<a href="http://info.Tsinghua.edu.cn">link</a>`
- `[href] {color: red;}`

- 通配符选择器

- `*{color: red;}`

- 子元素选择器

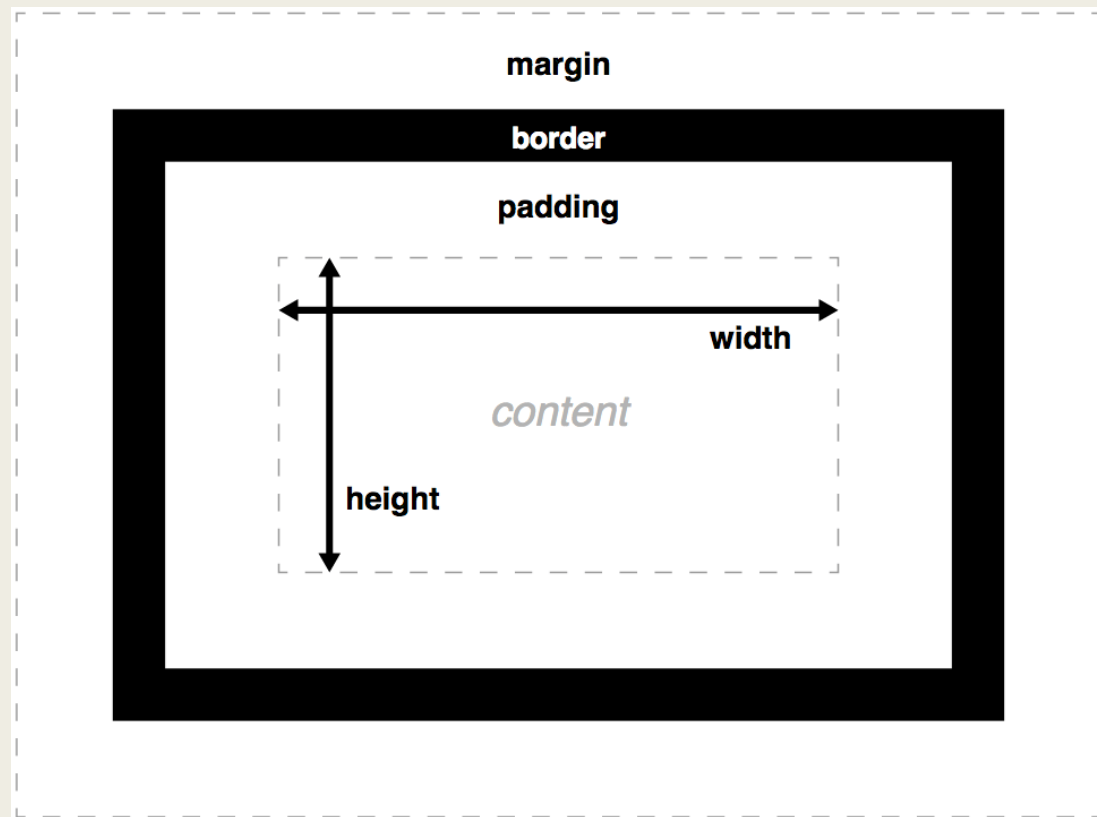
- `<div><p>some word</p></div><p>some word</p>`
- `div > p {color: red;}`

- 相邻兄弟选择器

- `<div><div><p>some word</p></div><p>some word</p></div>`
- `div + p {color: red;}`

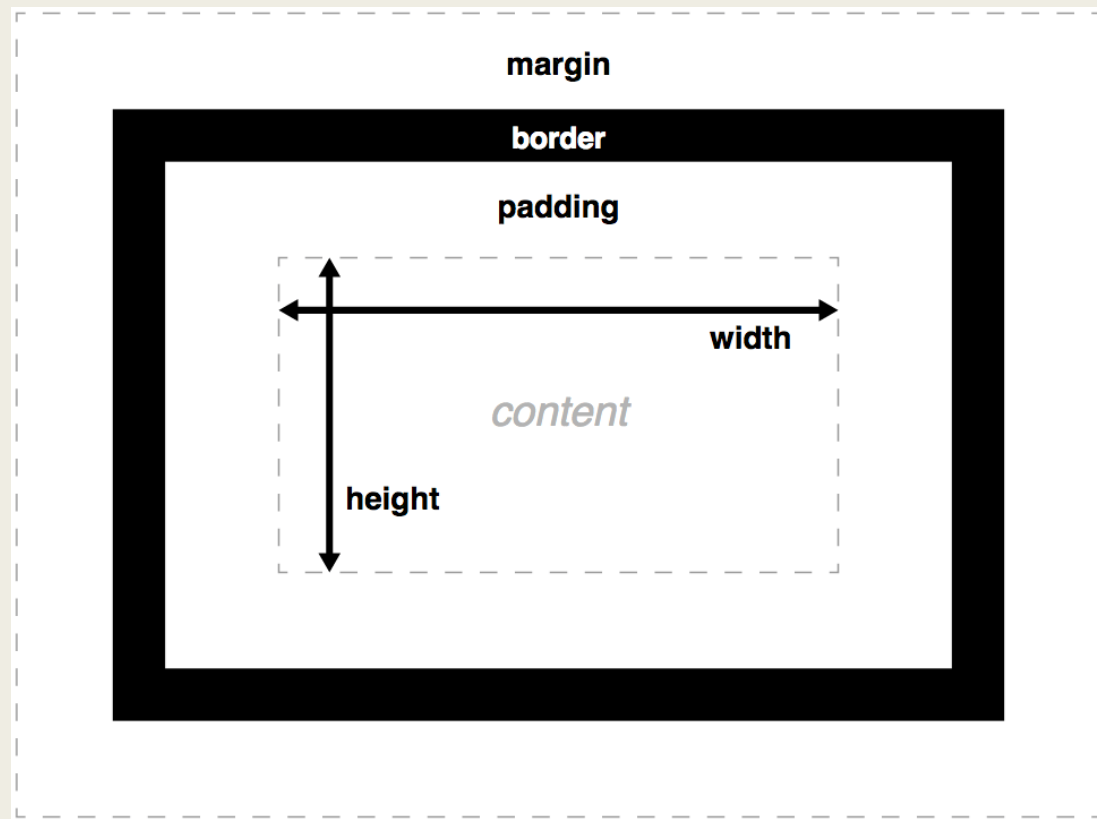
# 盒模型

- height和width默认设定了content box的高宽
- padding代表的是一个CSS box的内边距，介于content box和border之间。针对四边分别有padding-[top|right|bottom|left]选项
- 也可以只使用padding属性，将四边的大小按照上-右-下-左的顺序依次写出，当右和下未定义时，用上代替，左未定义时，用右代替，这被称为值复制
- box-sizing属性有三个值：content-box, border-box和inherit。content-box，他们决定了width和height所设定的对象



# 盒模型

- margin是CSS box的外边距，位于border的外侧，其定义方式与padding相同
- 当两个box的margin相互接触时，它们之间的距离不是两个margin的值之和，而是取两者中较大的那个。这叫外边距的合并
- border是指一个box的边框，每个边框有三个要素：宽度、样式以及颜色
- 可以通过border-[style|color|width]的形式，按照上-右-下-左或值复制的形式来决定他们的值，也可以使用border-[top|right|bottom|left]-[style|color|width]的形式为各边单独决定



# 定位

## ■ display属性

- none 不显示该元素
- block 被显示为块级元素，前后有换行符
- inline 被显示为内联元素，前后无换行符
- inline-block 行内块元素
- inherit
- height和width属性对内联元素无效
- vertical-align属性仅对内联有效

## ■ <C:\Users\林小寒\Desktop\新建文件夹\vertical-align.html>



# 定位

## ■ position属性

- static 默认值，box正常形成
- relative 元素框相对于初始位置偏离（保留在文档流中）
- absolute 相对于其包含块定位（从文档流中被删除）
  - 相对于设置了position: [absolute|relative]的父元素进行偏移，如果没有，则相对于body
- fixed 相对于视窗定位

■ <C:\Users\林小寒\Desktop\新建文件夹\absolute.html>

■ <C:\Users\林小寒\Desktop\新建文件夹\relative.html>

■ <C:\Users\林小寒\Desktop\新建文件夹\fixed.html>

# 浮动

- float
  - left
  - right
- <C:\Users\林小寒\Desktop\新建文件夹\float.html>

# 作业

- 写一个注册表单
- <C:\Users\林小寒\Desktop\新建文件夹\register.html>

CSS

# transform

- `translate(x,y)`定义2D转换
- `scale(x,y)`定义2D缩放转换
- `rotate(angle)`定义2D旋转
- `skew(x-angle, y-angle)`定义沿着X和Y轴的2D倾斜转换
- <C:\Users\林小寒\Desktop\新建文件夹\transform.html>

# 水平居中

- `margin: 0 auto;`
  - 块的`position`不为`fixed`或`absolute`
- 如果是`absolute`?
  - 利用`transform`
  - 利用`absolute`进行定位
  - `flex`布局

# 垂直居中

- 绝对定位
- transform
- flex
- table-cell
- ...

# box-shadow

- `box-shadow: h-shadow v-shadow blur spread color inset`
- `h-shadow`: 水平阴影的位置
- `v-shadow`: 垂直阴影的位置
- `blur`: 模糊距离
- `spread`: 阴影尺寸
- `color`: 阴影的颜色
- `inset`: 将外部阴影改为内部阴影



# 伪类

## ■ 语法

- Selector: pseudo-class {property: value}
- 也可以和CSS类搭配使用

## ■ 链接

- a:link 未访问的链接
- a:visited 已访问的链接
- a:hover 鼠标移动到链接上
- a:active 选定的链接
- active需要放在最后，hover需要放在link和visited后
- 其他伪类请自行查阅

# transition

- 当元素从一种样式变换为另一种样式时添加效果
- `transition:property duration timing-function delay;`
- CSS属性
- 时间
- 速度曲线
- 延时

# @keyframes

- @keyframes规则通过定义动画序列中的关键帧的样式来控制CSS动画序列中的中间步骤
- @keyframes规则中的!important会被忽略
- 应该始终定义0%和100%选择器
- `@keyframes animationname {keyframes-selector {css-styles;}}`

# 伪元素

- `selector::pseudo-element {property: value;}`
- `::before`用于在元素前添加内容
- `::after`用于在元素后添加内容
- <C:\Users\林小寒\Desktop\新建文件夹\pseudo-element.html>

# 浏览器兼容性

- 为了使代码能够在尽可能多的浏览器上正常运行，需要为一些较新的CSS特性添加私有前缀
  - *-ms-* IE的私有前缀
  - *-webkit-* chrome, safari和Edge的私有前缀
  - *-moz-* firefox的私有前缀
  - *-o-* opera的私有前缀
- 可以在MDN上查询CSS属性的浏览器兼容性
- 回退方案

# 响应式设计

- 为了使用户在不同的浏览器，不同的设备上浏览网页时都能觉得很爽
- 反例：网络学堂
- 为了进行响应式设计，需要根据分辨率/dpr来对代码进行调整
- 一个比较好的方式是使用非px单位作为长度单位
- 另外也可以使用@media进行媒体查询的设置
- @media使你可以在同一个样式表里，针对不同的媒介使用不同的样式规则
- 移动端适配非常emm...
- bootstrap

# 长度单位

- `vh`: 视窗高度的1%
- `vw`: 视窗宽度的1%
- `vmin`: `min(vh, vw)`
- `vmax`: `max(vh, vw)`
- `em`: 以父元素的`font-size`为基准
- `rem`: 以根元素的`font-size`为基准