

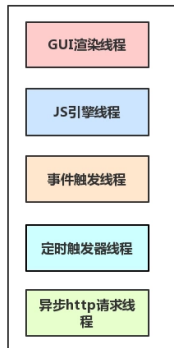
# 1. css 部分

## 1.1. 常见的浏览器有哪些？



## 1.2. 什么是浏览器的内核

内核就是浏览器所采用的渲染引擎，它是浏览器的重要组成部分。它决定了浏览器如何显示网页的内容以及页面的格式信息。负责对网页语法的解释（如标准通用标记语言下的一个应用 HTML、JavaScript）并渲染（显示）网页



## 1.3. 常用的浏览器内核有哪些

### Trident

IE 的内核，也就是国内双核浏览器的内核之一,此内核只能用于 Windows 平台，且不是开源的。Trident 内核一直延续到 IE11,IE11 的后继者 Edge 采用了新内核 EdgeHTML。

### Gecko

开源内核,后来被 FF (FireFox) 采用,最大优势是跨平台，在 Microsoft Windows、Linux、MacOs X 等主要操作系统中使用。

### Webkit

开源内核,Webkit 的鼻祖是 Safari,曾经的 Chrome 用的是 Webkit。注意：Webkit 其实包括是渲染引擎 Webcore(前身是 KHTML),以及 JS 引擎 JSCore

### Chromium

开源内核,chromium fork 自 webkit，代码可读性和编译速度得到提升。值得一提是谷歌专门研发了自己的 JS 引擎——V8，极大地提高了 JS 的运算速度。

## Presto

Opera Software 开发的浏览器排版引擎，它是世界公认最快的渲染速度的引擎，Opera7.0 开始使用。13 年 2 月后为了减少研发成本，Opera 放弃 Presto 宣布加入谷歌阵营，采用 chromium，之后也紧跟 Blink 的脚步。

### 1.4. CSS 优先级算法如何计算？

标签选择符：1

class 选择符：10

id 选择符：100

内联标签：1000      例如: `<p style=""></p>`

!important 声明的样式优先级最高，如果冲突再进行计算。

如果优先级相同，则选择最后出现的样式。

继承得到的样式的优先级最低。

### 1.5. CSS 选择器有哪些？哪些属性可以继承

id 选择器 (#myid)、类选择器 (.myclassname)、标签选择器 (div, h1, p)、相邻选择器 (h1 + p)、子选择器 (ul > li)、后代选择器 (li a)、通配符选择器 (\*)、属性选择器 (a[rel="external"])、伪类选择器 (a:hover, li:nth-child)

可继承的属性: font-size, font-family, color, line-height

不可继承的样式: border, padding, margin, width, height

### 1.6. link 引入样式和 import 引入样式的区别

1. @import 只能用于加载 css，link 除了可以加载 css，还可以加载其他资源

```
<link rel="stylesheet" type="text/css" href="/css/三二.css"/>
```

```
<link rel="icon" href="https://www.baidu.com/favicon.ico" >
```

2. link 写在 html 文件中，@import 写在 css 文件中

### 1.7. script, link, import, @import 的用途

script 引入 js 代码或者直接在 script 标签中插入 js 代码

link 既可以引入 css，也可以引入其他的

import 是 es6 里面的模块功能，可以导入其他模块的数据

@import 主要用来导入 css 代码

## 1.8. style 标签写在 body 后与 body 前有什么区别？

页面加载自上而下 当然是先加载样式。

写在 body 标签后由于浏览器以逐行方式对 HTML 文档进行解析,当解析到写在尾部的样式表(外联或写在 style 标签)会导致浏览器停止之前的渲染,等待加载且解析样式表完成之后重新渲染,在 windows 的 IE 下可能会出现 FOUC 现象(即样式失效导致的页面闪烁问题)

## 1.9. 如何异步加载 css 样式, js 代码

```
<link rel="preload" href="your-css-file.css" as="style">
```

那么如何设置 js 文件的异步加载呢？

有两种方式：

`<script defer>`

`<script async>`

这两种方式的区别就是：

defer 是在对页面解析的同时下载 js 文件，但是要等页面解析完，才开始执行 js 文件。

async 是在对页面解析的同时，下载 js 文件，一旦有下载完了，就先停止对页面的解析，执行下载完的那个 js 文件，当有多个 js 文件使用 async 的时候，不能保证执行顺序。

## 1.10. 盒模型是什么？有几种呢

盒模型规定了网页元素如何显示以及元素间相互关系，它本质上是一个盒子，封装周围的 HTML 元素。css 定义所有的元素都可以拥有像盒子一样的外形和平面空间。即都包含内容区、补白（填充）、边框、边界(外边距)，这就是**盒模型**。

盒模型分为两种：

第一种：W3C 盒模型（标准盒模型）

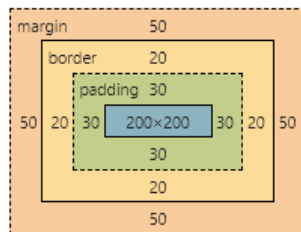
实际宽度 = 内容的宽度 (width) + border + padding + margin

```


.div1 {
width: 200px;
height: 200px;
padding: 30px;
border: 20px solid red;
margin: 50px;
}

div {
display: block;
}


```



## 第二种：怪异盒模型(又称 ie 盒模型)

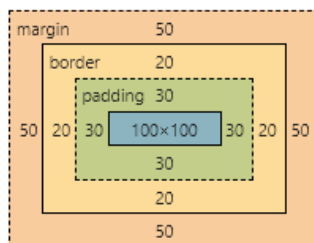
宽度 = 内容宽度 (content+border+padding) + margin

```


.div2 {
width: 200px;
height: 200px;
padding: 30px;
border: 20px solid red;
margin: 50px;
box-sizing: border-box;
}

div {
display: block;
}


```



## 1.11. box-sizing 属性的作用是什么

用来控制元素的盒子模型的解析模式，默认为 content-box

context-box: W3C 的标准盒子模型, 设置元素的 height/width 属性指的是 content 部分的高/宽

border-box: IE 传统盒子模型。设置元素的 height/width 属性指的是 border + padding + content 部分的高/宽

## 1.12. w3c 是什么

网际网路联盟(World Wide Web Consortium,简称 W3C), 又称 W3C 理事会专门制定一些网络标准的组织。

### 1.13. 元素竖向的百分比设定是相对于容器的高度吗

当按百分比设定一个元素的宽度时，它是相对于父容器的宽度计算的，但是，对于一些表示竖向距离的属性，例如 padding-top, padding-bottom, margin-top, margin-bottom 等，当按百分比设定它们时，依据的也是父容器的宽度，而不是高度。

### 1.14. 如何居中一个 div

已知 div 高度的情况

```
div{  
    width:300px;  
    border:2px solid red;  
    height:200px;  
    position: absolute;  
    left:50%;  
    top:50%;  
    margin:-100px -150px;  
}
```

未知 div 高度的情况

```
div{  
    width:300px;  
    border:2px solid red;  
    height:200px;  
    margin:auto;  
    position: absolute;  
    top:0;  
    right:0;  
    left:0;  
    bottom:0;  
}
```

### 1.15. display 有哪些值? 说明他们的作用

inline (默认) - 内联 典型代表: span, a

1. 不能设置宽高

## 2. 在一行显示

none - 隐藏

block - 块显示      典型代表: p,div,form,table,li

3. 可以设置宽高

4. 默认不在一行显示

inline-block - 内联块状显示      典型代表: button,input

5. 可以设置宽高

6. 在一行显示

table - 表格布局

grid - 网格布局

flex:弹性布局

## 1.16. position 有哪些值

static (默认): 按照正常文档流进行排列;

relative (相对定位): 不脱离文档流, 参考自身静态位置通过 top, bottom, left, right 定位;

absolute(绝对定位): 参考距其最近一个不为 static 的父级元素通过 top, bottom, left, right 定位;

fixed(固定定位,悬挂定位): 所固定的参照对象是可视窗口。

## 1.17. 用纯 CSS 创建一个三角形怎么做?

```
div{  
    width:0;  
    width:0;height:0;  
    border:60px solid transparent;  
    border-bottom-color:red;  
}
```

## 1.18. display:none 与 visibility: hidden 的区别?

display: none 不显示对应的元素, 在文档布局中不再分配空间 (回流+重绘)

visibility: hidden 不显示对应的元素, 在文档布局中仍保留原来的空间 (重绘)

opacity:0;      不显示对应的元素, 保留空间, 保留交互能力

## 1.19. position 跟 display、float 这些特性相互叠加后会怎么样?

display 属性规定元素应该生成的框的类型;position 属性规定元素的定位类型; float 属性是一种布局方式, 定义元素在哪个方向浮动。

类似于优先级机制:position:absolute/fixed 优先级最高,有他们在时,float 不起作用, display 值默认为调整为 block。

position 如果是 relative.float 优先级最高,其次是相对定位,最后是 display.

## 1.20. 浮动有哪些影响, 清除浮动有哪些方式

父元素的高度无法被撑开, 影响与父元素同级的元素  
与浮动元素同级的非浮动元素 (内联元素) 会跟随其后  
参与布局的元素, 需要同时浮动, 不能部分浮动  
多个元素参与浮动布局, 高度要保持一致

清除浮动的方式

父级 div 定义 height(不太好用, 因为要设定高度)

最后一个浮动元素后加空 div 标签 并添加样式 clear:both。

包含浮动元素的父标签添加样式 overflow 为 hidden 或 auto。

## 1.21. margin 和 padding 分别适合什么场景使用?

何时使用 margin:

需要在 border 外侧添加空白

空白处不需要背景色

上下相连的两个盒子之间的空白, 需要相互抵消时。

何时使用 padding:

需要在 border 内侧添加空白

空白处需要背景颜色

上下相连的两个盒子的空白, 希望为两者之和

## 1.22. 什么是 margin 塌陷? 怎么解决?

外间距塌陷就是当我们设置内层盒子的外间距时, 实际作用到了外层盒子上面去。

解决方案:

(1) 为父盒子设置 border, 为外层添加 border 后父子盒子就不是真正意义上的贴合 (可以设置成透明: border: 1px solid transparent)

(2) 为父盒子添加 overflow: hidden;

(3) 为父盒子设定 padding 值;

(4) 为父盒子添加 position: fixed;

## 1.23. 什么是 margin 重叠? 怎么解决?

margin 重叠也叫做外间距重叠，外间距穿透，外间距合并。通常出现在父子元素，兄弟元素之间。当设定了垂直方向的外间距时，最终的取值是较大的数值。

**解决方案：**

1. 添加边框（只能解决父子间距穿透）
2. 外层设置 padding（只能解决父子间距穿透）
3. 外层设置 overflow:hidden（只能解决父子间距穿透）
4. 绝对定位: position:absolute;（只能解决父子间距穿透）

## 1.24. CSS 属性 overflow 属性定义溢出元素内容区的内容会如何处理？

参数是 scroll 时候，必会出现滚动条。

参数是 auto 时候，子元素内容大于父元素时出现滚动条。

参数是 visible 时候，溢出的内容出现在父元素之外。

参数是 hidden 时候，溢出隐藏。

## 1.25. css 中双冒号和单冒号有什么区别？

单冒号可以表示伪类，也可以表示伪元素。

在 cs3 中为了区分，用单冒号表示伪类，双冒号表示伪元素

常见的伪类包括 :hover :active :focus

常见的伪元素包括： ::before ::after ::first-letter ::first-line

## 1.26. 怎么让 Chrome 支持小于 12px 的文字？

```
p{font-size:12px; -webkit-transform:scale(0.8)}
```

## 1.27. 什么是响应式设计？响应式设计的基本原理是什么？

响应式网站设计(Responsive Web design)是一个网站能够兼容多个终端或者说不同宽度的屏幕，而不是为每一个终端做一个特定的版本。

基本原理是通过媒体查询检测不同的设备屏幕尺寸做处理。

## 1.28. viewport 是什么

一般移动设备的浏览器都默认设置了一个布局视口，用于解决早期的 PC 端页面在手机上显示的问题。

iOS, Android 基本都将这个视口分辨率设置为 980px，所以 PC 上的网页大多都能在手机上呈现，只不过元素看上去很小，一般默认可以通过手动缩放网页。

设置的视口是 980，实际上只有 375。 缩放比 980/375。

我们通过设置视口为设备宽度，如此设置 1px，实际上也是 1px

移动端设计稿的尺寸是多大，常用标准是 750px



移动端技巧：

百分比布局

flex 布局

rem      --      参照 html 的 font-size 的大小计算    font-size=设备宽度/750  
vw

## 1.29. 如何兼容低版本的 ie

```
if(!window.ActiveXObject || "ActiveXObject" in window)location.href="https://emuchong.com/x/update.html";
```

## 1.30. 常用的网页布局方式

弹性布局：使用 百分比、rem 、flex

浮动布局：使用 float   clear

table 布局：一般用于后台显示，很少用

网格布局：

自适应布局和响应式布局：一般需要用 media 媒体查询完成，需要根据屏幕宽度的不同，写多个 css 样式文件

## 1.31. 几个常提到的名词

双飞翼，圣杯，流体布局

## 1.32. BFC 是什么

全称是 Block Formatting Context，是脱离文档流的独立的渲染区块，区块内的元素不会影响外面的元素

什么情况下会触发 BFC 渲染：绝对定位，悬挂定位，overflow:hidden，display:flex

## 1.33. 浏览器是怎样解析 css 的

浏览器解析 css 的时候，是从上往下解析的。如果权重一样的情况下，下面的 css 会覆盖上面的 css。如果权重不一样，会以权重高的作为渲染结果。比如!important>style>id>class>tag

## 1.34. 为什么要初始化 CSS 样式

因为浏览器的兼容性问题，不同浏览器对有些标签的默认值是不同的，  
如果没对 CSS 初始化往往会出现浏览器之间的页面显示差异。

思路一：全部去掉默认值

使用 reset.css

思路二：统一规定默认值  
使用 normalize.css

## 1.35. png,jpg,gif ,webp 是什么？

他们都表示一种图片的格式

png 是便携式网络图片 (Portable Network Graphics) 是一种无损数据压缩位图文件格式.优点是：压缩比高，色彩好。大多数地方都可以用。

jpg 是一种针对相片使用的一种失真压缩方法，是一种破坏性的压缩，在色调及颜色平滑变化做的不错。在 www 上，被用来储存和传输照片的格式。

gif 是一种位图文件格式，以 8 位色重现真色彩的图像。可以实现动画效果。

webp 格式是谷歌在 2010 年推出的图片格式，压缩率只有 jpg 的 2/3，大小比 png 小了 45%。缺点是压缩的时间更久了，兼容性不好，目前谷歌和 opera 支持。

## 1.36. 回流与重绘的区别是什么

回流会影响到页面的布局，重绘只影响元素的外观

## 1.37. psd 切图的使用方法

ps 创建的设计文件就是 psd 结尾。

设计师在设计一个网页的时候，但是这一个网页里面会使用到很多小的图片。

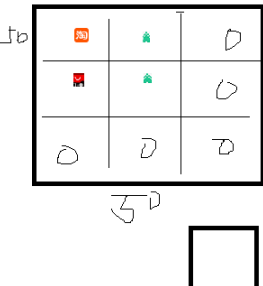
我们就要从 psd 文件中拿到这些图片。

就叫切片（切图）

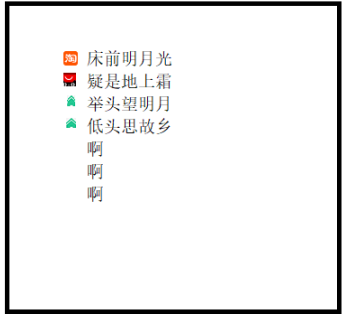
## 1.38. 雪碧图/精灵图是什么

是一个容纳了很多小图片的一中“大”图片。在实际项目中，使用精灵图，可以减少请求的次数，可以提高渲染的效率。在项目中是一个很好的选择

精灵图 sprite



50 50



需要发起n次请求去加载图片

```
<img src="" />
<img src="" />
<img src="" />
<img src="" />
```

使用精灵图的方式

```
background:url(精灵图.png) 0 0;
background:url(精灵图.png) 0 -50px;
background:url(精灵图.png) -50px 0;
```

## 1.39. 字体图标的使用

iconfont 注册账号，搜索想要的字体图标，添加到购物车，下载代码，引入到项目中，在项目中，通过 class="iconfont icon-xx" 的方式引入。

## 1.40. svg 图片的使用

svg 是矢量图，体积大。在放大的时候，也不会失真。效果特别好。

## 1.41. 蓝湖怎么使用

前端和 ui 的交互方式。

<https://lanhuapp.com/>

## 1.42. 设计稿的尺寸是多少

pc 端可以说 1460 ， 移动端，通常是 750 宽度

## 1.43. 版心是什么

主要内容区域