HTML

语义化：

是指对文本内容的结构化选择合乎语义的标签

优点：对机器：有利于搜索引擎的爬虫爬取有效信息，有利于seo，对于读屏软件可以根据文章自动生成目录

对开发人员：增强了可读性，结构更加清晰，有利于团队开发与维护

标签：header footer main setion aside nav article

script 中async 和defer 的区别：

script:一般的流程，从上至下，遇到script就停止解析html，去下载并执行脚本，该过程会阻塞html解析

defer：遇到defer不会停止解析html，此时异步下载脚本但不会执行，当html解析完毕再统一执行下载好的脚本，执行过程也是按脚本出现顺序

async：遇到async，当脚本加载时是异步加载，不会阻塞解析html，当脚本加载完毕，此时会停止解析html，立马去执行加载完毕的脚本，此时会阻塞html解析

DOCTYPE的作用：

作用：告诉浏览器以什么样的模式来解析文档，必须声明在文档第一行

两种模式：

标准模式：CSS1Compat，使用w3c的标准解析渲染，浏览器以其支持的最高标准呈现页面

兼容模式：BackCompat, 浏览器使用自己的怪异模式解析渲染，在该模式中，页面以一种比较宽松的向后兼容的方式显示，能保证一些老网站正常访问

src和href的区别：

src表示对资源的引用，它会将指向的资源插入当前标签位置

href表示超文本的引用，它会指向一些网络资源，建立与当前文档的链接关系

SGML、HTML、XML、XHTML的区别：

SGML：标准通用标记语言，是一种定义电子文档结构和描述其内容的国际标准语言，是所有电子文档标记语言的起源

HTML：超文本标记语言，用于规定怎样显示网页

XML：可扩展标记语言，他可以自己创建标签，html标签都是固定的且数量有限

XHTML：跟html没太大区别，更加严格，标签必须小写，标签必须有闭合标签

HTML5新特性：

1. 媒体播放标签：video、audio
2. 本地离线存储：localstorage 和 sessionstorage
3. 语义化标签：header、footer。。。。
4. 绘图：canvas
5. 表单控件：search、number、email、time。。。
6. webwork、websocket
7. 拖拽

移除：

纯表现元素：s 删除线、u 下划线、big更大字体、center 居中对齐文本等 可以css来代替

对可用性产生负面影响的元素：frame、frameset、noframes，不利于文档组织

浏览器内核的理解：

决定了浏览器如何显示网页内容以及页面格式信息，分为渲染引擎和js引擎

渲染引擎：负责渲染页面，默认情况在可以显示html、xml文档以及图片，也可以借助一些浏览器扩展插件显示其他类型数据，如pdf格式

Js引擎：解析和执行js来实现网页动态效果

IE：Trident内核 只能在windows 不能跨平台

Chrome：以前Webkit ，现在Blink

Firefox：Gecko内核 代码公开

Safari：Webkit

Opera：Blink

文档的预解析：

当执行js脚本时，另一个线程解析剩下的文档，并加载后面需要通过网络加载的资源，这种方式可以使资源并行加载从而使整体速度更快。预解析并不改变DOM树，它将这个工作交给主解析过程，预解析只解析外部资源的引用

浏览器的渲染原理：

1. 解析文档，生成DOM树，DOM树由DOM元素及属性节点构成
2. 解析css，生成css规则树
3. 根据DOM树和css规则树生成渲染树，渲染树节点为渲染对象包含颜色等属性，渲染对象与DOM元素相对应，对应关系不是一对一的，不可见的DOM元素不会插入渲染树
4. 回流，渲染对象插入渲染树中，没有大小和位置，当浏览器生成渲染树后会根据渲染树来布局，此时会计算出每个节点在页面中的位置和大小
5. 重绘，遍历渲染树，调用渲染树对象的paint方法将内容绘制到屏幕上，该过程是逐步完成的，解析一部分就显示一部分，不会等到全部解析完成才去构建和布局渲染树

回流和重绘

回流（重排）：DOM的变化影响了元素的几何信息（位置、大小）此时浏览器需要重新计算元素的几何信息，将其显示在正确的位置，表现为重新生成布局，重新排列元素。

重绘：当一个元素的外观属性（如颜色，字体等）发生变化，重新把元素外观绘制出来的过程。

触发回流重绘：元素位置、尺寸大小

1. 添加、删除、更新DOM元素
2. 边距、宽高、边框、填充
3. 浏览器尺寸改变
4. 计算offsetWidth和offsetHeight
5. 设置style属性的值
6. 修改默认字体

回流必会发生重绘，重绘不一定发生回流

如何减少回流重绘：

1. 批量处理DOM ，尽量少适用offsetTop等DOM属性
2. 最小化重绘重排，如样式集中改变
3. 使用绝对定位或固定定位使元素脱离文档流（在制作复杂动画时适用）
4. Css3硬件加速（GPU加速）如适用translate、opacity不会触发回流重绘

SessionStorage、localStorage、cookie

共同点：都保存在浏览器端

不同点：

SessionStorage：当前会话有效，存储大小5M，本地存储

LocalStorage：永久有效，存储大小5M，本地存储

Cookie：过期时间之前有效，存储大小4K，它会在浏览器和服务器之间来回传递（在http请求中携带）

Iframe优缺点：

优点：

原封不动的把嵌入的网页展现出来

多个网页引用iframe只需修改iframe的内容，就可以实现调用每一个页面的更改

缺点：

阻塞主页面的onload事件

影响页面并行加载，与主页面共享连接池，而浏览器对相同域的链接有限制

（解决以上两个问题：动态给iframe添加src属性值）

不利于SEO

增加服务器的http请求

兼容性差，很多移动设备无法完全显示框架

CSS

Css新特性：

1. 文字阴影
2. 动画
3. 图像变换（transform）
4. 滤镜
5. 选择器
6. 媒体查询
7. 渐变
8. Border-shadow、border-radius边框与圆角
9. Css3选择器如nth、not等
10. 过渡

Css选择器和优先级

Id选择器 #、类选择器 . 、标签选择器 div 、属性选择器 input[type=“text”] 、通配符选择器 \*、伪类选择器 a:hover、伪元素选择器 div::after、相邻兄弟选择器 +、子代选择器 >、后代选择器

优先级：！Important > 内联样式（style=”“） > id选择器 > 类选择器、属性、伪类 > 标签、伪元素 > 通配符选择器

Rgba和opacity 设置透明度的区别：

Rgba只作用于当前元素的颜色或背景，子元素不会继承其效果，opacity作用于当前元素以及元素内的所有内容

浏览器如何解析css选择器：

从右向左，从左向右如果发现不符合规则需要回溯，损失性能。

Display:none和visibility:hidden两者的区别：

Display：none，隐藏后不占用文档流，会引起回流重绘

Visibility：hidden，隐藏后占用文档流，会引起重绘，它具有继承性，

Transition支持visibility但不支持display

Transform、transition、animation的作用

Transform：描述了元素的静态样式，本身不会呈现动画效果，可以对元素进行图形变换

Transition：过渡，从一个效果逐渐转变为另一个效果，它需要一个触发事件

Animation：动画，自发的，可以立即且一直播放，不要触发事件

Line-height如何继承？

父元素的line-height为具体数值，子元素则继承该数值

父元素的line-height为比例值，子元素继承该比例值

父元素的line-height为百分比，子元素则为 父元素font-size\*百分比

如何让Chrome支持10px的文字

1、font-size:12px;-webkit-tranform:scale(0.84);

2、Font-size:20px;-webkit-transform:scale(0.5);

position属性值：

1. Static:默认定位，正常文档流
2. Relative：相对定位，相对于元素原来的位置进行定位，元素人占据原来的空间
3. Absolute：绝对定位，相对于最近已定位的父元素进行定位，父元素无定位则相对于根元素（html文档）进行定位，脱离文档流，不会占据原来空间
4. Fixed：固定定位，相对于浏览器窗口进行定位，为固定位置，窗口滚动也不会移动，脱离文档流，不占据原来空间
5. Sticky：粘性定位，可以认为fixed和relative的混合，元素在一定阈值内为相对定位，超过阈值为固定定位，必须指定top、left、right、bottom其中一个阈值才可以使粘性定位生效，否则为相对定位
6. Inherit：从父元素继承position属性值

Css盒模型：

标准盒模型：width包括content

怪异（IE）盒模型：width=content+border+padding

Box-sizing属性：

Content-box：标准盒模型

Border-box：IE盒模型

Inherit：继承父元素的box-sizing值

BFC格式化上下文

规定了box内部如何布局，且这个区域内的子元素不会影响到外面的元素

布局规则：

1. 内部box在垂直方向上排放
2. 同一个BFC相邻的box的margin会发生重叠
3. Box的margin左边与包含块的左border相接触
4. Bfc区域不会与float 浮动元素重叠
5. Bfc是一个独立容器，容器中的子元素不会影响外面的元素
6. 计算bfc高度时，浮动元素也参与计算

创建bfc：

1. 根元素，即html
2. Float不为none
3. Position为absolute与fixed
4. Display为inline-block，table-cell，table-caption
5. Overflow不为visible

使用场景：

去除边距重叠问题、清除浮动、阻止元素被浮动元素覆盖

元素水平居中

JS

Vue