



## 第二版：HTML 18 道

### 目录

第二版：HTML 18 道	1
1、前端需要注意哪些 SEO	2
2、<img>的 title 和 alt 有什么区别	2
3、HTTP 的几种请求方法用途	3
4、从浏览器地址栏输入 url 到显示页面的步骤	3
5、如何进行网站性能优化	4
1. 减少 HTTP 请求：合并文件、CSS 精灵、inline Image	4
2. 减少 DNS 查询：DNS 缓存、将资源分布到恰当数量的主机名	4
3. 减少 DOM 元素数量	4
1. 使用 CDN	4
2. 配置 ETag	4
3. 对组件使用 Gzip 压缩	4
1. 减小 cookie 大小	4
1. 将样式表放到页面顶部	5
2. 不使用 CSS 表达式	5
3. 使用<link>不使用@import	5
1. 将脚本放到页面底部	5
2. 将 javascript 和 css 从外部引入	5
3. 压缩 javascript 和 css	5
4. 删除不需要的脚本	5
5. 减少 DOM 访问	5
1. 优化图片：根据实际颜色需要选择色深、压缩	5
2. 优化 css 精灵	5
3. 不要在 HTML 中拉伸图片	5
6、HTTP 状态码及其含义	5
7、语义化的理解	6
8、介绍一下你对浏览器内核的理解？	7
9、html5 有哪些新特性、移除了那些元素？	7



10、HTML5 的离线储存怎么使用，工作原理能不能解释一下？	8
11、浏览器是怎么对 HTML5 的离线储存资源进行管理和加载的呢？	8
12、请描述一下 cookies, sessionStorage 和 localStorage 的区别？	9
13、iframe 有那些缺点？	9
14、WEB 标准以及 W3C 标准是什么？	10
15、xhtml 和 html 有什么区别？	10
16、Doctype 作用？严格模式与混杂模式如何区分？它们有何意义？	10
17、行内元素有哪些？块级元素有哪些？空(void)元素有那些？行内元素和块级元素有什么区别？	11
18、HTML 全局属性(global attribute)有哪些？	11

我们的网站：<https://tech.souyunku.com>

关注我们的公众号：**搜云库技术团队**，回复以下关键字

回复：**【进群】**邀请您进「**技术架构分享群**」

回复：**【内推】**即可进：北京，上海，广州，深圳，杭州，成都，武汉，南京，  
郑州，西安，长沙「**程序员工作内推群**」

回复 **【1024】**送 4000G 最新架构师视频

回复 **【PPT】**即可无套路获取，以下最新整理调优 PPT！

## 46 页《JVM 深度调优，演讲 PPT》

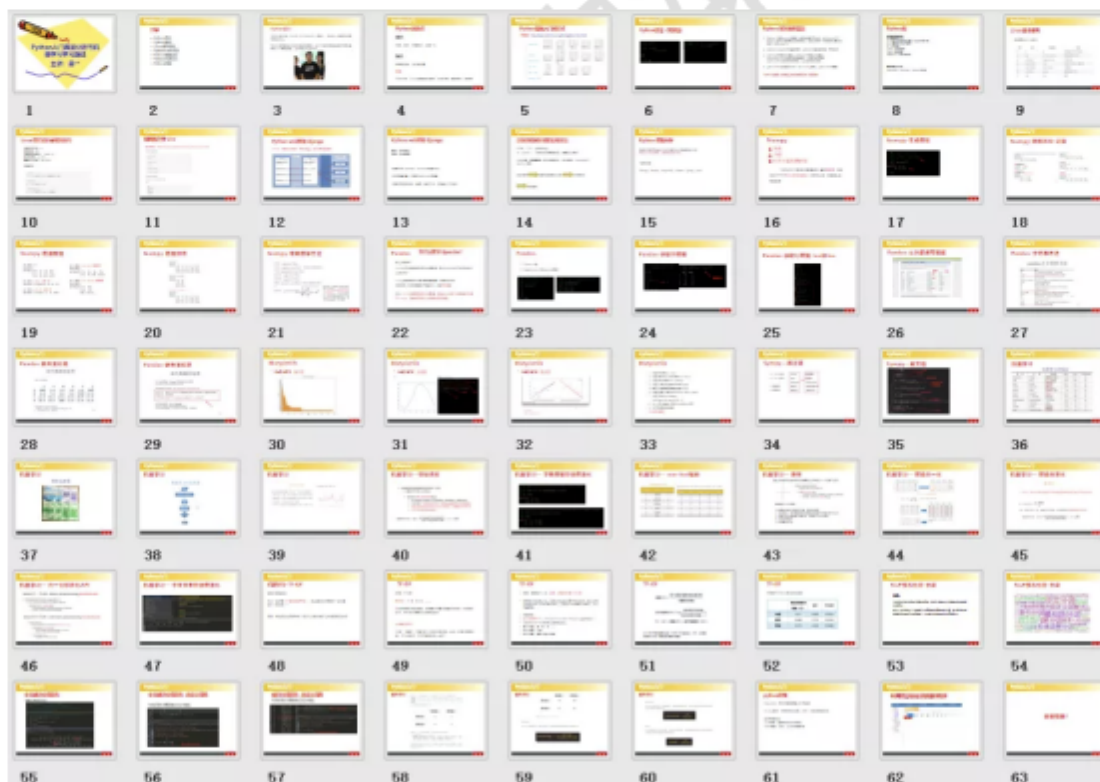




## 53 页《Elasticsearch 调优演讲 PPT》



## 63 页《Python 数据分析入门 PPT》





微信扫一掃

<https://tech.souyunku.com>

技术、架构、资料、工作、内推  
专注于分享最有价值的互联网技术干货文章

## 1、前端需要注意哪些 SEO

- 合理的 title、description、keywords：搜索对着三项的权重逐个减小，title 值强调重点即可，重要关键词出现不要超过 2 次，而且要靠前，不同页面 title 要有所不同；description 把页面内容高度概括，长度合适，不可过分堆砌关键词，不同页面 description 有所不同；keywords 列举出重要关键词即可
- 语义化的 HTML 代码，符合 W3C 规范：语义化代码让搜索引擎容易理解网页
- 重要内容 HTML 代码放在最前：搜索引擎抓取 HTML 顺序是从上到下，有的搜索引擎对抓取长度有限制，保证重要内容一定会被抓取
- 重要内容不要用 js 输出：爬虫不会执行 js 获取内容
- 少用 iframe：搜索引擎不会抓取 iframe 中的内容
- 非装饰性图片必须加 alt
- 提高网站速度：网站速度是搜索引擎排序的一个重要指标

## 2、<img>的 title 和 alt 有什么区别



- 通常当鼠标滑动到元素上的时候显示
- alt 是<img>的特有属性，是图片内容的等价描述，用于图片无法加载时显示、读屏器阅读图片。可提图片高可访问性，除了纯装饰图片外都必须设置有意义的值，搜索引擎会重点分析。

### 3、HTTP 的几种请求方法用途

- 1、GET 方法
- 发送一个请求来取得服务器上的某一资源
- 2、POST 方法
- 向 URL 指定的资源提交数据或附加新的数据
- 3、PUT 方法
- 跟 POST 方法很像，也是想服务器提交数据。但是，它们之间有不同。PUT 指定了资源在服务器上的位置，而 POST 没有
- 4、HEAD 方法
- 只请求页面的首部
- 5、DELETE 方法
- 删除服务器上的某资源
- 6、OPTIONS 方法
- 它用于获取当前 URL 所支持的方法。如果请求成功，会有一个 Allow 的头包含类似 "GET,POST" 这样的信息
- 7、TRACE 方法
- TRACE 方法被用于激发一个远程的，应用层的请求消息回路
- 8、CONNECT 方法
- 把请求连接转换到透明的 TCP/IP 通道

### 4、从浏览器地址栏输入 url 到显示页面的步骤





- 浏览器根据请求的 URL 交给 DNS 域名解析，找到真实 IP，向服务器发起请求；
- 服务器交给后台处理完成后返回数据，浏览器接收文件（HTML、JS、CSS、图像等）；
- 浏览器对加载到的资源（HTML、JS、CSS 等）进行语法解析，建立相应的内部数据结构（如 HTML 的 DOM）；
- 载入解析到的资源文件，渲染页面，完成。

## 5、如何进行网站性能优化

- content 方面

1. 减少 HTTP 请求：合并文件、CSS 精灵、inline Image
2. 减少 DNS 查询：DNS 缓存、将资源分布到恰当数量的主机名
3. 减少 DOM 元素数量

- Server 方面

1. 使用 CDN
2. 配置 ETag
3. 对组件使用 Gzip 压缩

- Cookie 方面



## 1. 减小 cookie 大小

- css 方面

1. 将样式表放到页面顶部
2. 不使用 CSS 表达式
3. 使用<link>不使用@import

- Javascript 方面

1. 将脚本放到页面底部
2. 将 javascript 和 css 从外部引入
3. 压缩 javascript 和 css
4. 删除不需要的脚本
5. 减少 DOM 访问

- 图片方面

1. 优化图片：根据实际颜色需要选择色深、压缩
2. 优化 css 精灵
3. 不要在 HTML 中拉伸图片



## 6、HTTP 状态码及其含义

- 1XX：信息状态码
- 100 Continue 继续，一般在发送 post 请求时，已发送了 http header 之后服务端将返回此信息，表示确认，之后发送具体参数信息
- 2XX：成功状态码
- 200 OK 正常返回信息
- 201 Created 请求成功并且服务器创建了新的资源
- 202 Accepted 服务器已接受请求，但尚未处理
- 3XX：重定向
- 301 Moved Permanently 请求的网页已永久移动到新位置。
- 302 Found 临时性重定向。
- 303 See Other 临时性重定向，且总是使用 GET 请求新的 URI。
- 304 Not Modified 自从上次请求后，请求的网页未修改过。
- 4XX：客户端错误
- 400 Bad Request 服务器无法理解请求的格式，客户端不应当尝试再次使用相同的内容发起请求。
- 401 Unauthorized 请求未授权。
- 403 Forbidden 禁止访问。
- 404 Not Found 找不到如何与 URI 相匹配的资源。
- 5XX：服务器错误
- 500 Internal Server Error 最常见的服务器端错误。
- 503 Service Unavailable 服务器端暂时无法处理请求（可能是过载或维护）。

## 7、语义化的理解





- 用正确的标签做正确的事情!
- html 语义化就是让页面的内容结构化, 便于对浏览器、搜索引擎解析;
- 在没有样式 CSS 情况下也以一种文档格式显示, 并且是容易阅读的。
- 搜索引擎的爬虫依赖于标记来确定上下文和各个关键字的权重, 利于 SEO。
- 使阅读源代码的人对网站更容易将网站分块, 便于阅读维护理解

## 8、介绍一下你对浏览器内核的理解?

- 主要分成两部分: 渲染引擎(layout engineer 或 Rendering Engine)和 JS 引擎
- 渲染引擎: 负责取得网页的内容 (HTML、XML、图像等等)、整理讯息 (例如加入 CSS 等), 以及计算网页的显示方式, 然后会输出至显示器或打印机。浏览器的内核的不同对于网页的语法解释会有不同, 所以渲染的效果也不相同。所有网页浏览器、电子邮件客户端以及其它需要编辑、显示网络内容的应用程序都需要内核
- JS 引擎则: 解析和执行 javascript 来实现网页的动态效果
- 最开始渲染引擎和 JS 引擎并没有区分的很明确, 后来 JS 引擎越来越独立, 内核就倾向于只指渲染引擎

## 9、html5 有哪些新特性、移除了那些元素?

- HTML5 现在已经不是 SGML 的子集, 主要是关于图像, 位置, 存储, 多任务等功能的增加
- 绘画 canvas
- 用于媒介回放的 video 和 audio 元素
- 本地离线存储 localStorage 长期存储数据, 浏览器关闭后数据不丢失
- sessionStorage 的数据在浏览器关闭后自动删除



- 语义化更好的内容元素, 比如 article、footer、header、nav、section
- 表单控件, calendar、date、time、email、url、search
- 新的技术 webworker, websocket, Geolocation
- 移除的元素:
- 纯表现的元素: basefont, big, center, font, s, strike, tt, u`
- 对可用性产生负面影响的元素: frame, frameset, noframes
- 支持 HTML5 新标签:
- IE8/IE7/IE6 支持通过 document.createElement 方法产生的标签
- 可以利用这一特性让这些浏览器支持 HTML5 新标签
- 浏览器支持新标签后, 还需要添加标签默认的风格
- 当然也可以直接使用成熟的框架、比如 html5shim

## 10、HTML5 的离线储存怎么使用, 工作原理能不能解释一下?

- 在用户没有与因特网连接时, 可以正常访问站点或应用, 在用户与因特网连接时, 更新用户机器上的缓存文件
- 原理: HTML5 的离线存储是基于一个新建的.appcache 文件的缓存机制(不是存储技术), 通过这个文件上的解析清单离线存储资源, 这些资源就会像 cookie 一样被存储了下来。之后当网络在处于离线状态下时, 浏览器会通过被离线存储的数据进行页面展示
- 如何使用:
- 页面头部像下面一样加入一个 manifest 的属性;
- 在 cache.manifest 文件的编写离线存储的资源
- 在离线状态时, 操作 window.applicationCache 进行需求实现

CACHE MANIFEST

#v0.11

CACHE:



```
js/app.js
css/style.css
NETWORK:
resource/logo.png
FALLBACK:
/ /offline.html
```

## 11、浏览器是怎么对 HTML5 的离线储存资源进行管理和加载的呢

- 在线的情况下，浏览器发现 html 头部有 manifest 属性，它会请求 manifest 文件，如果是第一次访问 app，那么浏览器就会根据 manifest 文件的内容下载相应的资源并且进行离线存储。如果已经访问过 app 并且资源已经离线存储了，那么浏览器就会使用离线的资源加载页面，然后浏览器会对比新的 manifest 文件与旧的 manifest 文件，如果文件没有发生改变，就不做任何操作，如果文件改变了，那么就会重新下载文件中的资源并进行离线存储。
- 离线的情况下，浏览器就直接使用离线存储的资源。

## 12、请描述一下 cookies, sessionStorage 和 localStorage 的区别？

- cookie 是网站为了标示用户身份而储存在用户本地终端（Client Side）上的数据（通常经过加密）
- cookie 数据始终在同源的 http 请求中携带（即使不需要），记会在浏览器和服务端间来回传递



- sessionStorage 和 localStorage 不会自动把数据发给服务器，仅在本地保存
- 存储大小：
- cookie 数据大小不能超过 4k
- sessionStorage 和 localStorage 虽然也有存储大小的限制，但比 cookie 大得多，可以达到 5M 或更大
- 有效期时间：
- localStorage 存储持久数据，浏览器关闭后数据不丢失除非主动删除数据
- sessionStorage 数据在当前浏览器窗口关闭后自动删除
- cookie 设置的 cookie 过期时间之前一直有效，即使窗口或浏览器关闭

### 13、iframe 有那些缺点？

- iframe 会阻塞主页面的 Onload 事件
- 搜索引擎的检索程序无法解读这种页面，不利于 SEO
- iframe 和主页面共享连接池，而浏览器对相同域的连接有限制，所以会影响页面的并行加载
- 使用 iframe 之前需要考虑这两个缺点。如果需要使用 iframe，最好是通过 javascript 动态给 iframe 添加 src 属性值，这样可以避开以上两个问题

### 14、WEB 标准以及 W3C 标准是什么？

- 标签闭合、标签小写、不乱嵌套、使用外链 css 和 js、结构行为表现的分离

### 15、xhtml 和 html 有什么区别？



- 一个是功能上的差别
- 主要是 XHTML 可兼容各大浏览器、手机以及 PDA，并且浏览器也能快速正确地编译网页
- 另外是书写习惯的差别
- XHTML 元素必须被正确地嵌套，闭合，区分大小写，文档必须拥有根元素

## 16、Doctype 作用？严格模式与混杂模式如何区分？它们有何意义？

- 页面被加载的时，link 会同时被加载，而@import 页面被加载的时，link 会同时被加载，而@import 引用的 CSS 会等到页面被加载完再加载  
import 只在 IE5 以上才能识别，而 link 是 XHTML 标签，无兼容问题  
link 方式的样式的权重 高于@import 的权重
- <!DOCTYPE> 声明位于文档中的最前面，处于 <html> 标签之前。告知浏览器的解析器， 用什么文档类型 规范来解析这个文档
- 严格模式的排版和 JS 运作模式是 以该浏览器支持的最高标准运行
- 在混杂模式中，页面以宽松的向后兼容的方式显示。模拟老式浏览器的行为以防止站点无法工作。 DOCTYPE 不存在或格式不正确会导致文档以混杂模式呈现

## 17、行内元素有哪些？块级元素有哪些？ 空(void)元素有哪些？行内元素和块级元素有什么区别？

- 行内元素有：a b span img input select strong
- 块级元素有：div ul ol li dl dt dd h1 h2 h3 h4...p





- 空元素: `<br>` `<hr>` `<img>` `<input>` `<link>` `<meta>`
- 行内元素不可以设置宽高, 不独占一行
- 块级元素可以设置宽高, 独占一行

## 18、HTML 全局属性(global attribute)有哪些

- class: 为元素设置类标识
- data-\*: 为元素增加自定义属性
- draggable: 设置元素是否可拖拽
- id: 元素 id, 文档内唯一
- lang: 元素内容的语言
- style: 行内 css 样式
- title: 元素相关的建议信息