# 总结

原生语言比较重要

## 一、基础知识

1. Internet

- IP

- URI、URL、URN

- DNS

- Http协议

2. 网络编程相关--

- Web Server基本原理

- 网络机器人

3. web

- Web 1.0 2.0 3.0 比较

- LAMP

Web service：通过网络调用其他网站的资源

Web service架构：尽量把非核心功能交给其他人去做，自己全力开发核心功能 平台无关、编程语言无关

Web server:网站服务器，每一个网站服务器都需要从网络接受http请求，然后提供http回复给请求者。HTTP回复一般包含一个html文件，有时也可以包含纯文本文件、图像或其他类型的文件。

## 二、浏览器端

1. HTML/XHTML

- 结构、表现、行为

- 基本语法、常用标记--

- html语义化\*（去年考过）

- html5（无插件范式，新功能）\*

Html5的新功能：

1. 简化语法，doctype声明更简单
2. <canvas>取代flash(无插件范式)
3. 新增<header>和<footer>标签，<section><article>标签，是文档结构更清晰，更有利于SEO
4. 删除<b>和<font>标签
5. 删除<frame><center><big>标签
6. Placeholder
7. 必要属性required

2. CSS

- why

- 几种方式

- CSS盒模型\*

- 响应式网页设计，主要手段。\*

- 优先级顺序和继承关系（css顺序，冲突机制）

（较简单）

3. javascript\*

- 基本语法

- "first-class" functions

- 事件驱动编程

- 面向对象

- 匿名函数

- 作用域、闭包及其用途

4. DOM

- XHTML/XML与DOM树

- DOM0，DOM2事件流（捕获、目标和冒泡）

- 观察者模式 -

5. Ajax \*

- RIA

- 同步、异步通信

- Ajax请求

- Ajax优缺点

- XSS

- SOP

- Two-request limit

作业中考察

## 三、服务器端PHP（比例不大）

- Lifecycle

- cgi\fastcgi\mod\_php模式

- 静态网页 动态网页 DHTML

- 基本语法--

- HTML form

- 浏览器-server交互

- 验证：不同方式

- 正则表达式

- 面向对象

- 设计模式\*

## 四、框架\*

- 概念

- 模板引擎

- MVC

- 框架的优缺点

流行框架、如何自己制作框架（面型对象、设计模式的应用）

## 五、优化\*（涉及较多）

- 基准测试（如何优化）

- 浏览器（对前端页面的解析机制，从url开始输入整个完整过程的理解）

- 优化思路，技术

基本原理的理解

## 六、XML、Web Services、Mashups（涉及较少）

- XML优缺点

- XML应用

- DTD与XML Schemas

- Web Services、Mashups只涉及基本概念与用途

注：没特别注明意味着主要关注基本概念与用途

题型

1. 基本概念（所有课件中涉及的，不限于总结）（是什么？怎么用？名词解释，缩写要写出中文或英文全称） 20

2. 简答题 40

3. 问答题 40

Further Readings

1. 高性能网站建设指南：前端工程师技能精髓。[美] [Steve Souders](http://book.jd.com/writer/Steve%20Souders_1.html)（[史蒂夫.桑德斯](http://book.jd.com/writer/%E5%8F%B2%E8%92%82%E5%A4%AB.%E6%A1%91%E5%BE%B7%E6%96%AF_1.html)） 著。电子工业出版社。

2. 高性能网站建设进阶指南：Web开发者性能优化最佳实践 [Perfrntance Best Practices For Web Developers Even Faster Web Sttes]。[美] [Steve Souders](http://book.jd.com/writer/Steve%20Souders_1.html)（[史蒂夫.桑德斯](http://book.jd.com/writer/%E5%8F%B2%E8%92%82%E5%A4%AB.%E6%A1%91%E5%BE%B7%E6%96%AF_1.html)） 著；[口碑网前端团队](http://book.jd.com/writer/%E5%8F%A3%E7%A2%91%E7%BD%91%E5%89%8D%E7%AB%AF%E5%9B%A2%E9%98%9F_1.html) 译。[电子工业出版社](http://book.jd.com/publish/%E7%94%B5%E5%AD%90%E5%B7%A5%E4%B8%9A%E5%87%BA%E7%89%88%E7%A4%BE_1.html)。

3. 高性能JavaScript。[Nicholas C. Zakas](http://book.jd.com/writer/Nicholas%20C.%20Zakas_1.html)（[尼古拉斯·泽卡斯](http://book.jd.com/writer/%E5%B0%BC%E5%8F%A4%E6%8B%89%E6%96%AF%C2%B7%E6%B3%BD%E5%8D%A1%E6%96%AF_1.html)） 著；[丁琛](http://book.jd.com/writer/%E4%B8%81%E7%90%9B_1.html) 译。电子工业出版社。

4. 大型网站技术架构 核心原理与案例分析。[李智慧](http://book.jd.com/writer/%E6%9D%8E%E6%99%BA%E6%85%A7_1.html) 著。电子工业出版社

5. 高性能浏览器网络

react.js AngularJS node.js …… vue.js

https://coggle.it/diagram/Vz9LvW8byvN0I38x

多看笔试和面试题

可以增强SEO的方式：

1. <meta>标签（<meta name=”description” content=”…” />

<meta name=”keywords” content=”标签名” />）

1. <article>和<section>标签

块级元素和内联元素

内联元素中不能包含块级元素，块级元素中可以包含内联元素

<a>中可以放<div>，a里不能放button、a

为什么不建议使用table布局

1. Table比其他html占用更多字节（速度更慢）
2. 阻挡浏览器渲染引擎的渲染顺序
3. 显示图片时需要把单个、有逻辑性的图片切成多个图
4. 在某些浏览器中table里的文字拷贝会出现问题
5. Table会影响其内部的某些布局属性的生效
6. 对于页面布局来说，从语义上看是不正确的
7. 不灵活