
Table of Contents

介绍	1.1
KickOff Party	1.2
1 教学目的	1.2.1
2 软件和工具	1.2.2
3 教学方法	1.2.3
4 WEB基本知识和历史	1.2.4
5 课前自学	1.2.5
Intro to HTML	1.3
1 初识HTML	1.3.1
2 HTML常用术语	1.3.2
3 解剖网站	1.3.3
4 动手 - 修改更多HTML	1.3.4
Intro to CSS	1.4
1 初识CSS	1.4.1
2 CSS常用术语	1.4.2
3 CSS盒子模型	1.4.3
4 使用Chrome开发者工具调试样式	1.4.4
5 动手 - 修改更多CSS样式	1.4.5
Intro to Bootstrap	1.5

Introduction

此教程是专门为 Girls Coding Day 女性编程素养教育而设计。

工作坊结束后，学员将具备能力在Bootstrap模板的基础上，做出 [Girls Coding Day](#) 这样的静态网站。

学员将获得：

1. 修改模板的基本HTML和CSS的能力。
2. 修改bootstrap 的能力。
3. 部署在Github Page的能力。
4. 自学编程的基本能力。
5. 一群热爱编程的闺蜜以及友好的程序员社群。

Kickoff Party

本章节目录

1. 下载和安装工作坊所需要的软件
2. 了解工作坊提倡的学习方法
3. 了解基本的Web基础知识
4. 自学HTML和CSS基本概念

1.1 教学目的

课程结束后具备能力在*Bootstrap*模板的基础上，做出[Girls Coding Day](#)这样的个人网站。

1. 修改模板的基本*HTML*和*CSS*的能力。
2. 修改*bootstrap* 的能力
3. 部署在*Github Page*的能力

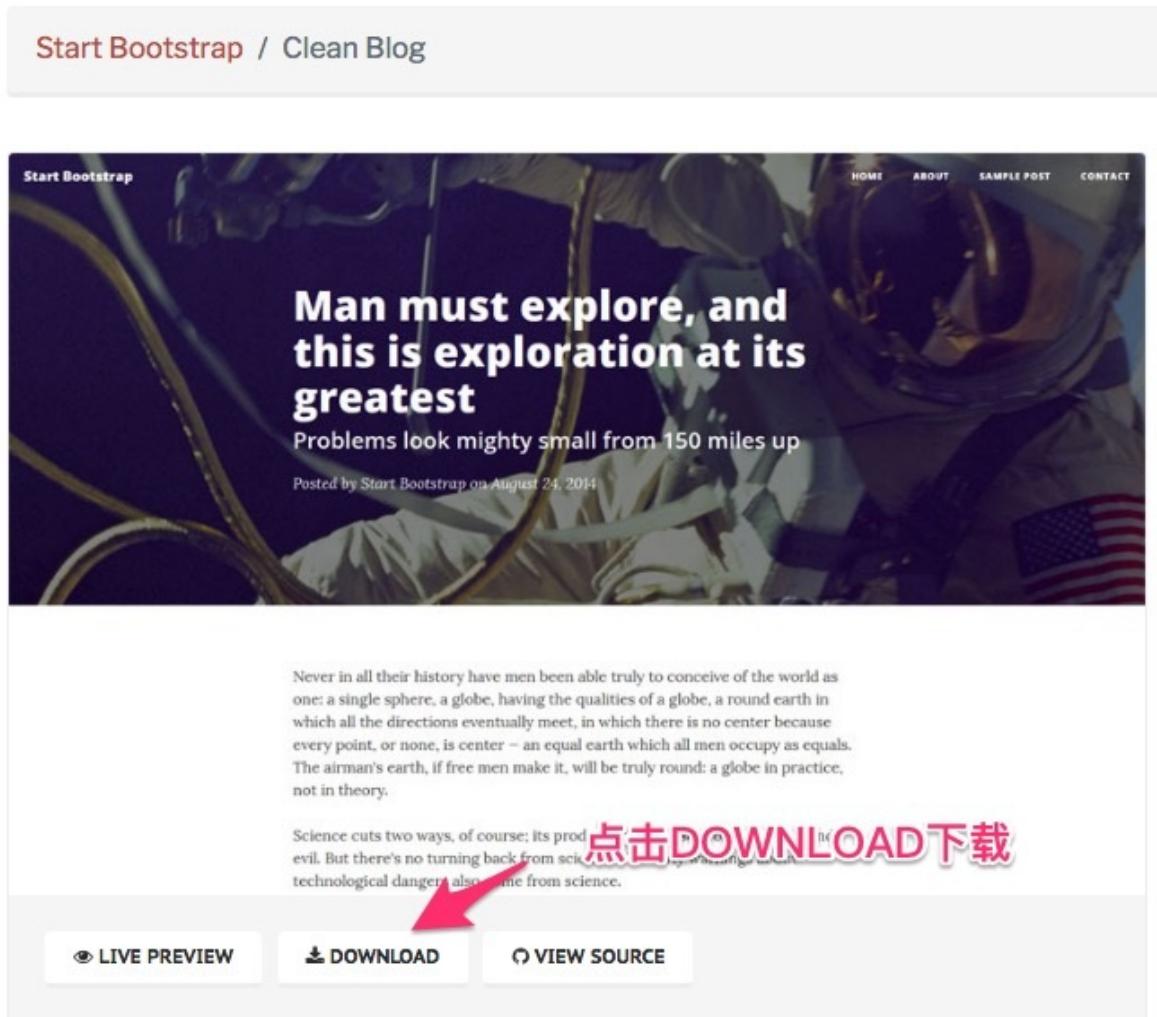
1.2 软件和工具

Cooking time: 30min active / 30-45min passive

- 下载并安装 [浏览器 Chrome](#)
- 下载并安装 [文本编辑器 Atom](#)
 - 你是否好奇「文本编辑器」是什么？那么赶紧用搜索引擎搜索一下关键字「文本编辑器」吧。
 - 略读 [文本编辑器 - 维基百科](#)
- 注册[GitHub](#)
 - Github能用来做什么？请略读 [怎样使用 GitHub？ - 知乎](#)
- 安装[GitHub Desktop](#)
 - 阅读并操作 [Authenticating to GitHub](#)
 - 按照阅读内容，配置好github.com 的帐号信息
- 注册 <http://codepen.io/>
- 下载概念图开源软件 [Cmap](#)
 - 阅读（可选）[概念图 - 维基百科](#), [自由的百科全书](#)

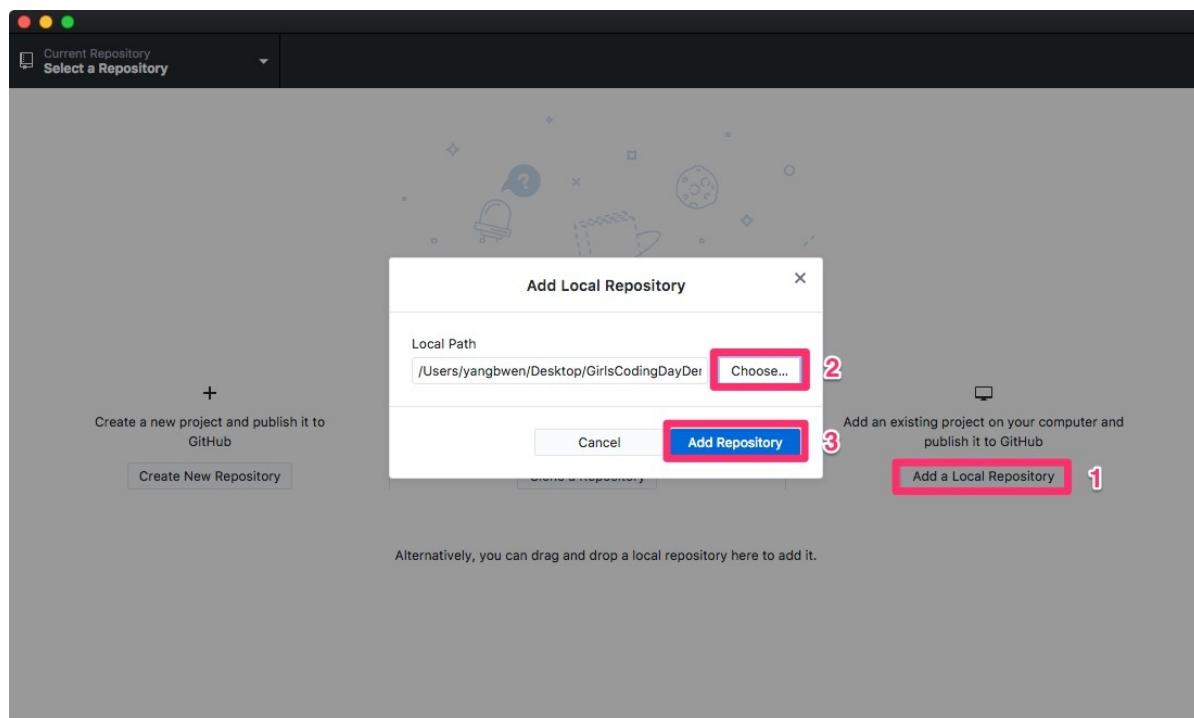
Github Desktop 配置

1. 去<https://startbootstrap.com/template-overviews/clean-blog/>, 下载免费Bootstrap模板，保存在桌面。

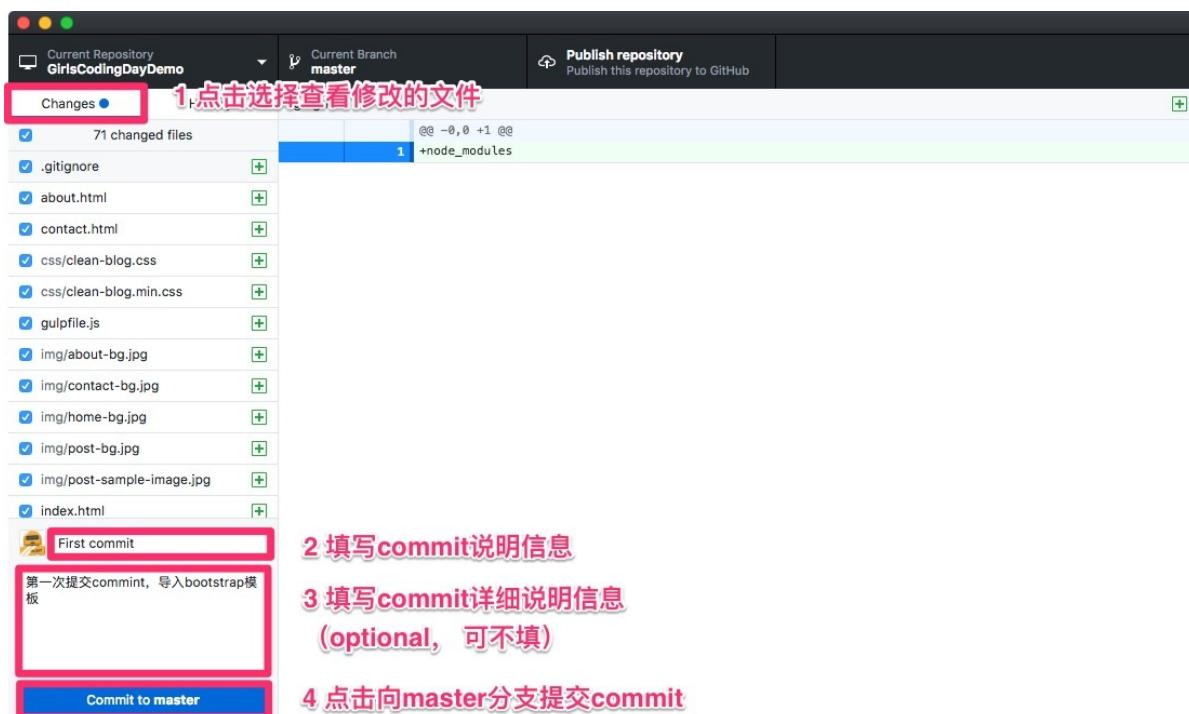


2. 解压缩，文件夹重命名为 GirlsCodingDayDemo
3. 打开Github Desktop，添加刚刚解压缩的文件夹为本地仓库。如下图

2 软件和工具

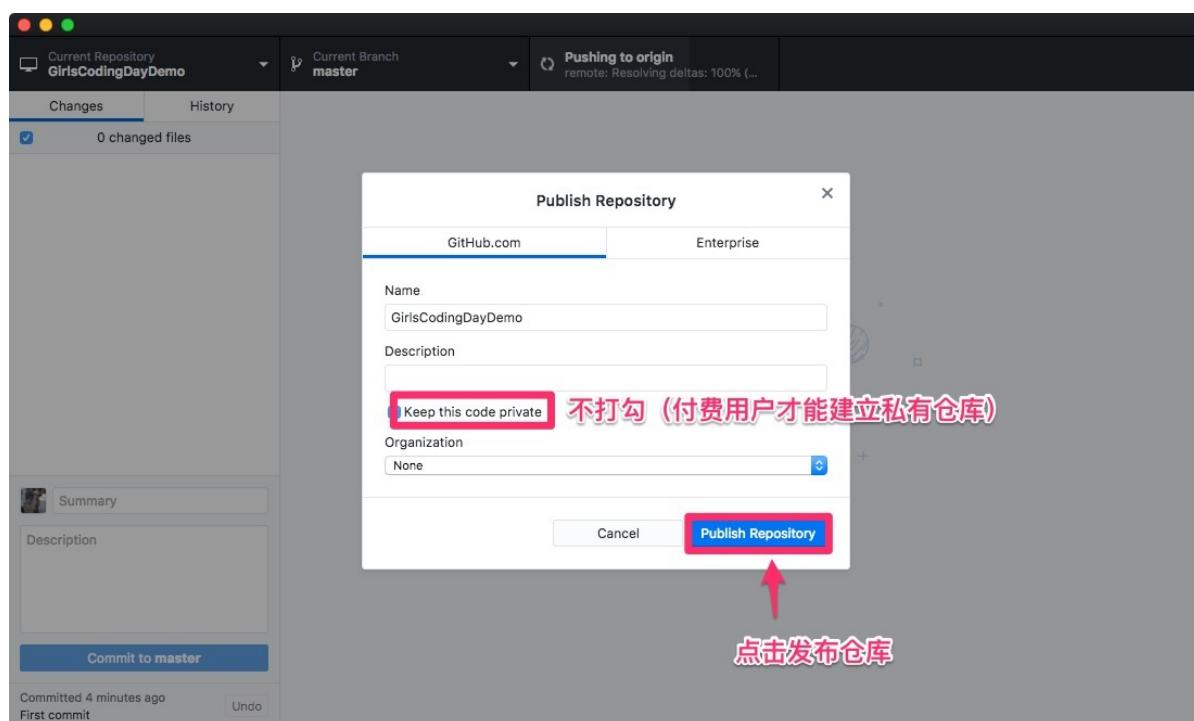
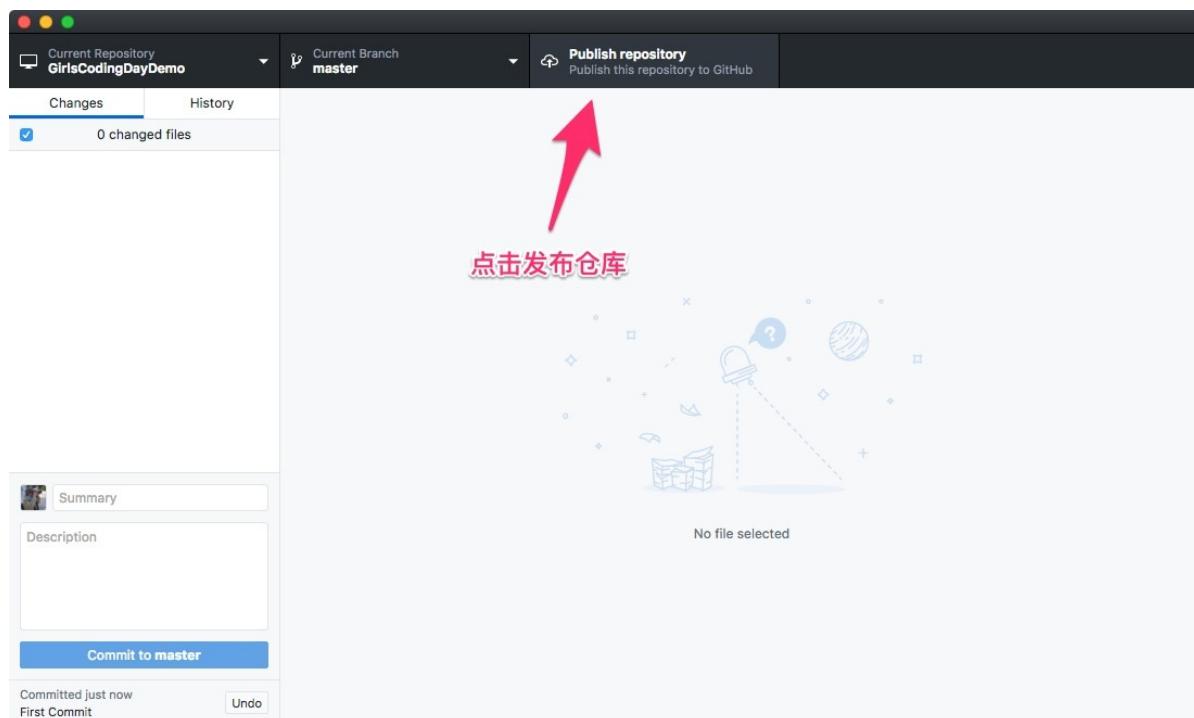


4. 提交commit到master分支，（注意changes）



5. 发布到Github

2 软件和工具



6. 访问github.com 网站，你可以看到你刚刚通过Github Desktop软件在github.com 网站新建了一个仓库。点击Settings

2 软件和工具

1 点击Settings

设置这个仓库为Github Pages

GitHub Pages

✓ Your site is published at <http://sundevilyang.com/GirlsCodingDayDemo/>

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Source

Your GitHub Pages site is currently being built from the master branch. [Learn more](#).

master branch ▾ Save

2 选择master分支

Theme chooser

Select a theme to build your site with a Jekyll theme. [Learn more](#).

Choose a theme

Custom domain

Custom domains allow you to serve your site from a domain other than [sundevilyang.com](#). [Learn more](#).

Save

Enforce HTTPS — Unavailable for your site because you have a custom domain configured ([sundevilyang.com](#))
HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site.
When HTTPS is enforced, your site will only be served over HTTPS. [Learn more](#).

7. 使用Atom编辑器打开桌面的文件夹 GirlsCodingDayDemo 。找到 index.html 文件，修改其中两处HTML代码，然后保存（一定要保存！）。

2 软件和工具

The screenshot shows a code editor window with the following details:

- Project:** GirlsCodingDayDemo
- File:** index.html
- Content:** The code is a standard HTML document. It includes meta tags for charset, viewport, and author, followed by a title element containing "Girls Coding Day's Blog". The body section contains a navigation bar with a brand link labeled "Girls Coding Day".
- Annotations:** Two specific lines of code are highlighted with red boxes:
 - Line 12: <title>Girls Coding Day's Blog</title>
 - Line 45: Girls Coding Day
- Status Bar:** Shows file statistics (0 0 0), last save time (12:30), and a git status indicating master branch, 2 changes, and 1 file updated.

8. 使用chrome浏览器打开 index.html 预览修改效果



Man must explore, and this is exploration at its greatest

Problems look mighty small from 150 miles up

Posted by Start Bootstrap on September 24, 2014

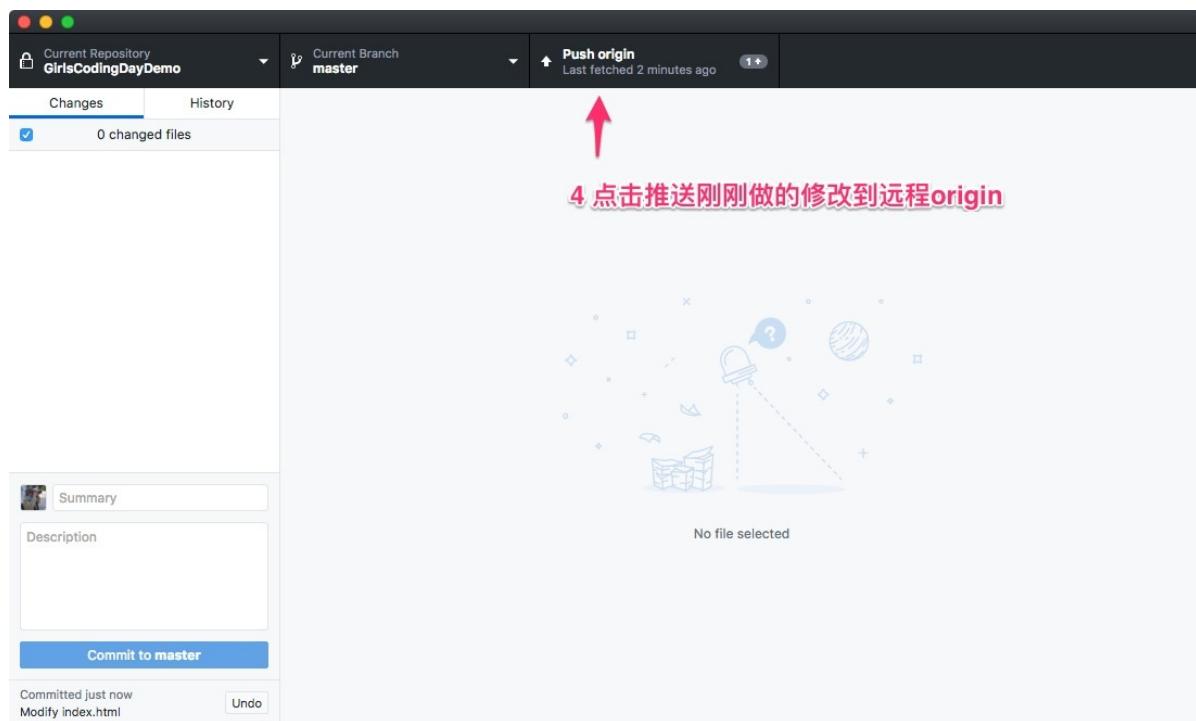
I believe every human has a finite number of heartbeats. I don't intend to

- 回到Github Desktop，可以看到修改的内容。提交commit，再点击Push origin同步到Github Pages。

1. 点击Changes

2 查看修改的文件无误

3 提交commit



10. 打开github pages的链接，可以看到自己刚刚修改的内容。

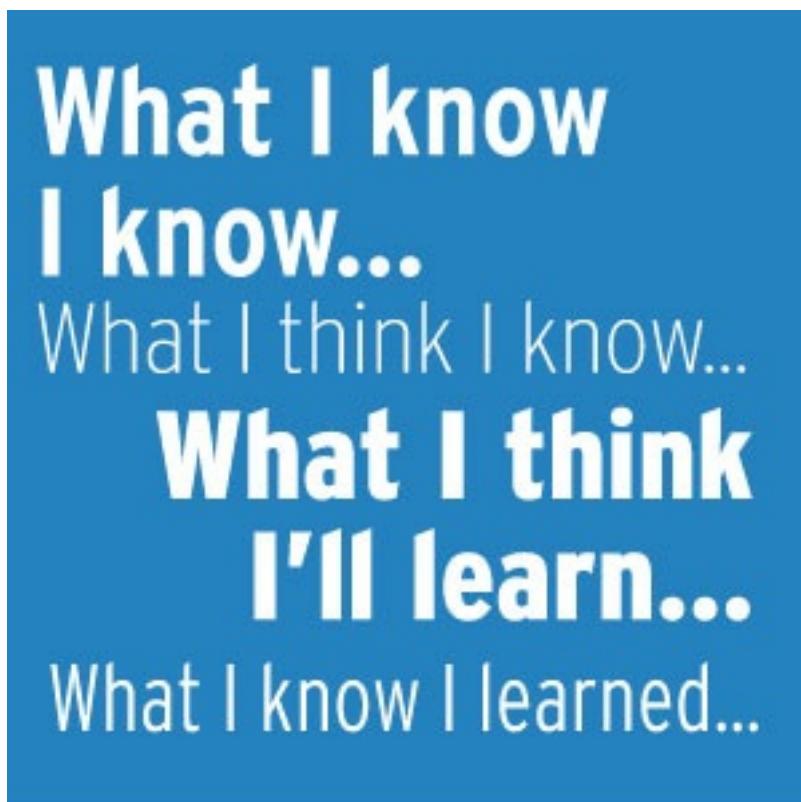
1.3 教学方法介绍

整个课程设计基于学习参与的三种方式：

1. 带着问题，探究式学习。
2. 自发动，自主学习。
3. 指南不如匠人，社会化学习。

探究式学习

什么是定向思维？



定向思维，与发散思维相对，以基础为“经验”，在动力定型驱使下的按照既定方向或程序进行思维的活动过程。

但淋雨或、洗澡或开车时候在我们脑海中闪现的随机想法称作不定向思维，它不同于有目的的定向思维。

虽然，许多失败都缘于定向思维，但技能学习不同。在新进入技能领域之前，带着一个有经验前辈写下的关键问题，新手不容易迷茫并有能在浩瀚的概念中有定向前行。

- 带着问题流量使用工具、软件的网站（例如上面安装软件，我们并不是简单地让大家下载个软件而已，而是注入了需要思考的问题）

- 带着问题看阅读材料
- 带着问题做练习
- 带着问题观察身边其他人

自主学习

如果你本身在互联网行业，恰巧身边有很多程序员。你会发现一个事实：『**程序员的编程技能都是自学成才！**』。不管是科班出身还是非科班出身，他们更多的人愿意将自己获得的编程技能途径定义为「自学」。

正如Alvin Toffler所说

21世纪的文盲不再是哪些不会读书写字的人，而是那些不会学习，忘记所学，以及不能温故知新的人。

如果，你有这样的口头禅「告诉我，你想要我学什么，我就学给你看」，你可能不适合编程，因为我们希望你能「自己教自己」。

当你证明自己可以成为自主学习者，可以独立思考，可以找到和发现有效运用所学知识解决显示世界中难题的方法，你的自信心会得到极大的提升。自我获得力量增强的感觉，驱使自己为了未来而继续学习。

但要是失败了呢？失败是痛苦的，但是却也可以转化成一种有意义的学习经历。当你认识到做错了事情并知道下次怎样可以做得更好，你也获得了一种强有力终身学习的技能。

自主学习的八项技能

- 找出并评估有用的信息来源：信息源头不正确会导致一系列错误，例如我们阅读材料尽量只是W3School、维基百科等第一手文档，而不是其他二手加工过的资料。
- 在有用的信息源中辨别出重要的信息：设定阅读目标、找出中心思想和重要细节
- 以有意义的方式组织信息：用Cmap画概念图
- 撰写日志和博文：写作输出是最好的知识管理
- 安排好时间：第一要义是**时间哲学**，可参考文章 [「感觉身体被掏空」到掌控世界 -- 上班族一周如何拿出15小时学习](#)
- 背诵最小核心知识：**把经过历史和经验反复验证的知识概念深入到自主心智**。如果一个成年人还跟小学一年级的学生一样，需要去理解为什么三加四等于七，而不是把这个概念作为本能反应出来，那么他是无法和别人竞争的，因为效能太低。而我们为了能够提高学习效率，必须在学习过程中记住一些关键知识。
- 运用问题解决体系
- 监督自己的学习（元认知）

社会化学习

教是最高形式的理解。

告诉我，我会忘记。给我看，我会记住。让我教，我会理解。

如果仅仅依靠自己的行动、通过试错来学习，那么学习绝对是一件非常辛苦、费力的事情，而且可能是危险或有害的。

回归学习的本质，人类大多数行为是通过模仿和观察习得的，通过交流、分享、应用、反馈、争论、甚至争辩，最终帮助个体对于某个事物形成较为全面的认知。

因此我们非常有必要，大家坐在一个屋子里，共同完成一些学习。哪怕没有言语交流，只是彼此坐在一起，也是非常棒的“交流”。

三个提高思维能力的方法

- 概念图
- 刻意练习
- 费曼技巧

1.4 WEB基本知识和历史

Cooking time: 30min active / 30-45min passive

Learning by Reading(10mins): [网络是如何工作的 - 学习 Web 开发 | MDN](#)

本小节会覆盖的主题

- 托管主机: 计算机上的文件如何成为公共网站
- 主机的组件和网站架构
- 各种可选主机提供商以及如何选择

本地开发环境

- 打开Atom软件, 在桌面创建文件 `index.html`, 并用文本编辑器写入你最喜欢的诗篇。
- 现在, `index.html` 只是一个保存在你自己电脑上面的文件.
- 你可以通过文件夹 (Windows 电脑) 或Finder (Mac电脑) 看到文件的地址类似为
`/Users/yangbwen/Desktop/index.html`

主机的作用是给这个新创立的文件一个地方, 好让互联网上的任何人都可以看见。

这个地方可以是你的电脑吗?

默认设置互联网上随随便便一个人是不能访问你电脑的, 但你可以给他们权限。

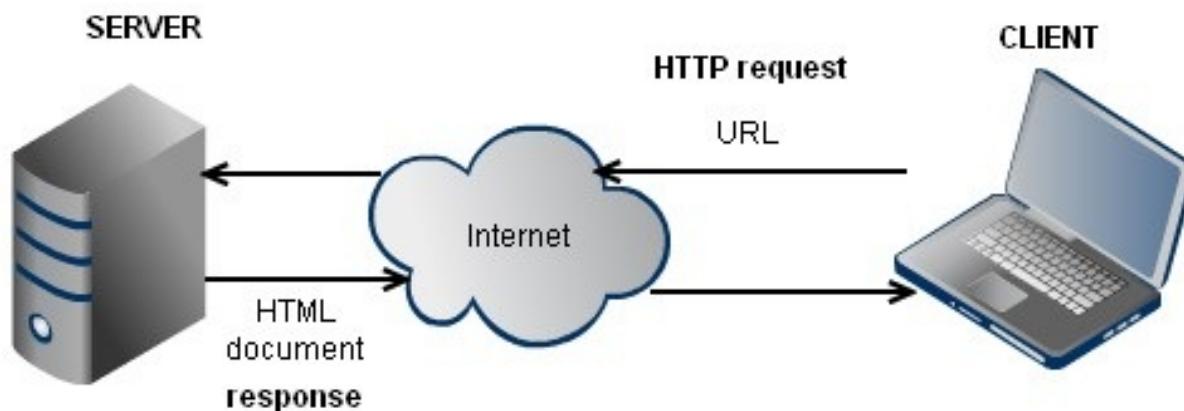
- 例如: 你可以在你的电脑上面设置一个互联网服务器 (webserver) 。
- 缺点: 如果你的电脑关机了别人就不能访问了。
- 通常做法: 把你的这些文件放在一个一直开机的地方。

服务器 (Server) 与主机 (Hosting)

托管主机: 把你文件的地方通过托管在一台由其他人管理电脑的服务器 (server) 上。

1. 阅读 [服务器 - 维基百科, 自由的百科全书](#)
2. 阅读后回答: 服务器是软件还是硬件?

认识名词: 单机、Client/Server、Browser/Server



客户端提出请求，服务器端满足客户端（通常）。请求就像问题：客户端询问他们，服务器回答。对于我们的web讨论，client = browser。虽然任何类型的计算机可以用作服务器，但服务器通常比其他计算机更大和更强大。客户端应用程序在与您交互的客户端计算机上运行（如您的浏览器），服务器应用程序在其他地方的机器上运行，这样一来客户端或客户端应用程序就可以和服务器应用程序通话

1. 使用搜索引擎搜索上述关键词「Client/Server、Browser/Server」，阅读自己搜索到的文章。
2. 教练带大家讨论C/S 和 B/S。
3. 问题：怎么去判断是B/S 还是C/S？例如小程序是哪种模式？微信手机端是哪种模式？微信网页端是哪种模式？

托管客户端应用程序

您的 `index.html` 和 `style.css` 文件都是客户端代码（client code），浏览器运行这些文件就会显示页面。为了使任何人的浏览器都可以加载它们，我们将它们放在其他地方（主机）的计算机上，该计算机上的服务器（server）将是看门人，让任何人的浏览器加载您的文件（客户端代码）。在这个例子中，你的代码是客户端代码，服务器只是作为看门人。

托管主机提供商

类别	说明 (what)	什么情况下用 (when)	比较 (why)
文件存储	放置文件的地方;示例： S3, Github Pages, Dropbox, 百度盘, 七牛。	当你只需要静态文件即可。	适用于静态页面或客户端应用程序，不足以用于服务器应用

软件打包服务	可以获取域名，可选择预安装软件;示例：wordpress.org, bluehost.com。	当你需要一个廉价方法来做网站（例如，为一个小企业），并要一切“能工作而已”，并不需要定制，那么你可以使用此类型。	想要创建和管理内容的用户，对于想要更多控制的技术人员或者想要使用软件组件不在允许列表上
自建	可以设置和管理您自己的所有网站组件，包括数据库，服务器，域名等。例如：Amazon EC2, Heroku, Linode, Digital Ocean, 阿里云，青云等。	当你需要完全控制你的软件或业务要求时；例如，大多数科技公司都是建立自己的网站架构和托管解决方案。	更耗时，往往更便宜。需要技术人员来实现和维护，但完全可控，部署/托管任何类型的客户端或服务器应用程序的能力

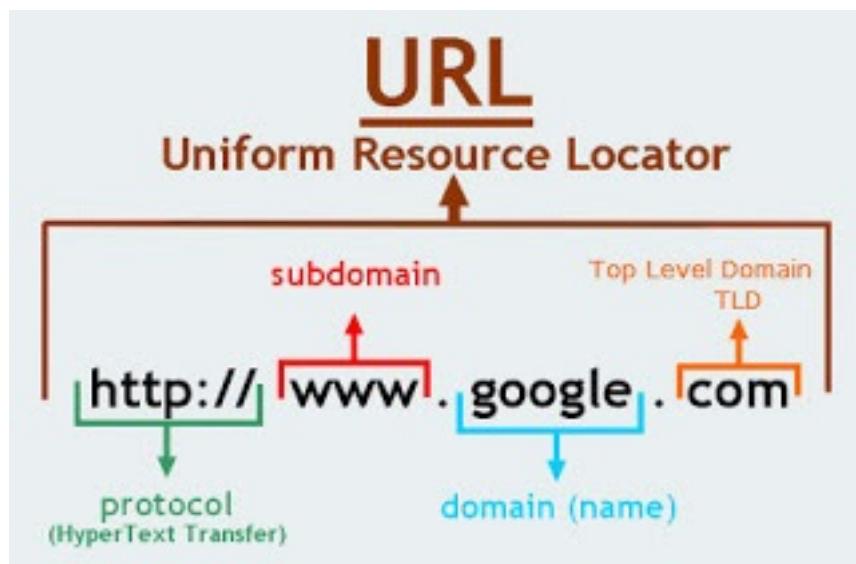
域名托管 Domain hosting

- IP addresses

IP被在线电脑用来互相识别，通常是四组数字。通过简单的方法查找您的IP地址 [What Is My IP Address?](#)

阅读 [IP地址 - 维基百科，自由的百科全书](#)

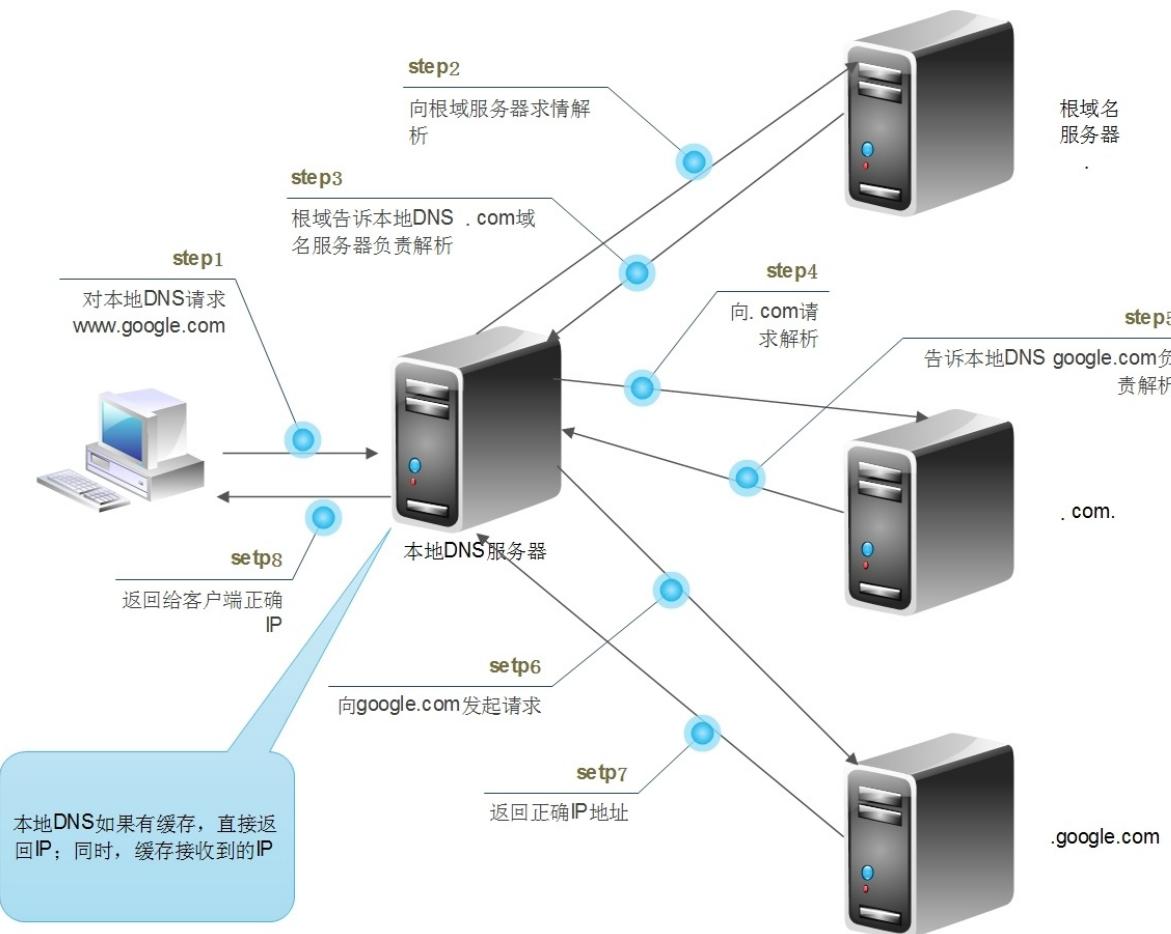
- Domains



- Domain Name System (DNS)

将域名转换为允许计算机相互通信的IP地址。通过键入终端（Mac）或命令提示符（PC）查找GCD的IP地址：nslookup girlscodingday.org

Advice for Tutors: 教练解释下图



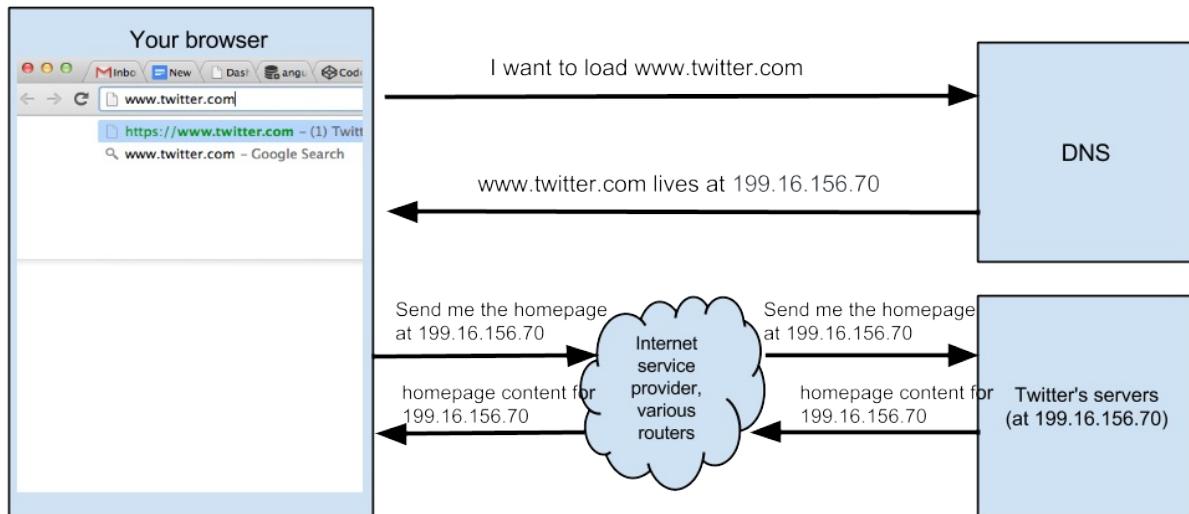
- 注册域名 Registrars

通常用来购买域名。良好的客服、易于使用的管理界面、价格实惠（无隐藏费）、允许将名称转让给其他注册商、ICANN认证

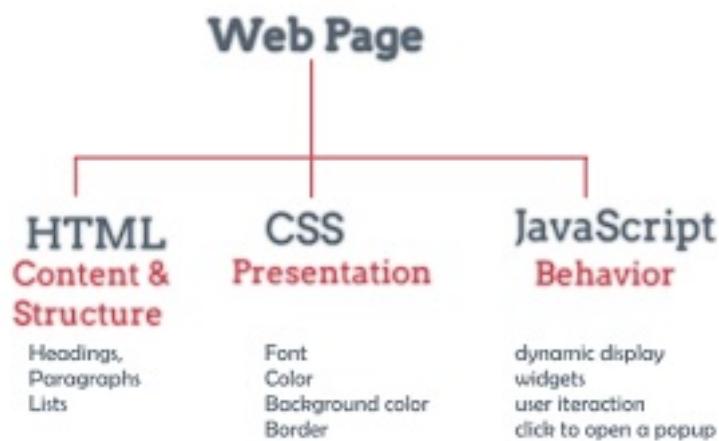
Advice for Tutors: 教练说说国内、国外都有哪些买域名的代理机构？国外和国内买域名有哪些优缺点？什么是备案（可选）？

一个网站的日常

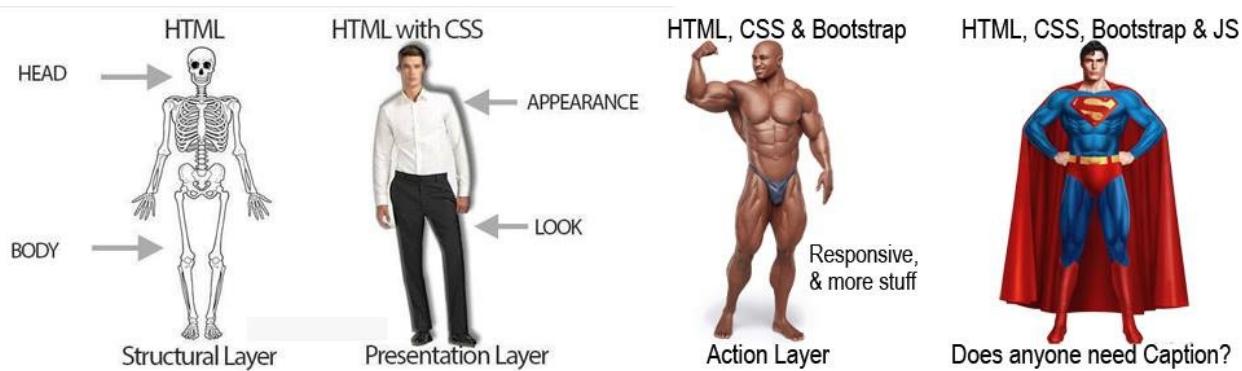
1. 在浏览器地址栏中输入网址
2. DNS 将您连接到托管服务器
3. 然后将文件发送回您的计算机进行显示
4. 有时，代码在发给你之前必须进行编译。



HTML/CSS/Bootstrap/Javascript 对于WEB分别意味着什么？



Advice for Tutors: 教练简单讲讲 HTML/CSS/JavaScript 分别是什么以及三者之间的关系。



Advice for Tutors: 教练讲讲

- 什么是Bootstrap?
- Web 1.0 vs Web 2.0 ?
- 浏览器大战

Tasks For Tutors :

1. 给学员演示如何使用atom
 - 打开一个项目
 - 找到index.html
 - 修改html常用标签: title/h1/h3/li/ol
2. 给学员演示Chome Developer Tools的简单用法
 - 快捷键调用开发者工具
 - 怎么看定位页面上某个HTML元素
 - 怎么查看某个HTML元素的CSS样式, 以及在开发者工具中调试样式属性

1.5 课前作业

- 从你的同组学员中，找一个你喜欢的或许也喜欢你的小伙伴，结对！
- 阅读[HTML 教程](#) 到「HTML 内联框架」章节
- 阅读[CSS 教程](#) 到 「CSS 伪元素」章节
- 把你自学到的HTML和CSS概念用概念图工具Cmap画出来。（「概念图」是一项受益终身的技能）

Intro to HTML

本章节目录

1. 了解一个网站的结构
2. 能阅读和理解所给源码中的HTML
3. 学会使用文本编辑器Atom阅读、修改HTML源码

初识HTML

Inquiry Learning

尝试向你的同学解释下列概念：

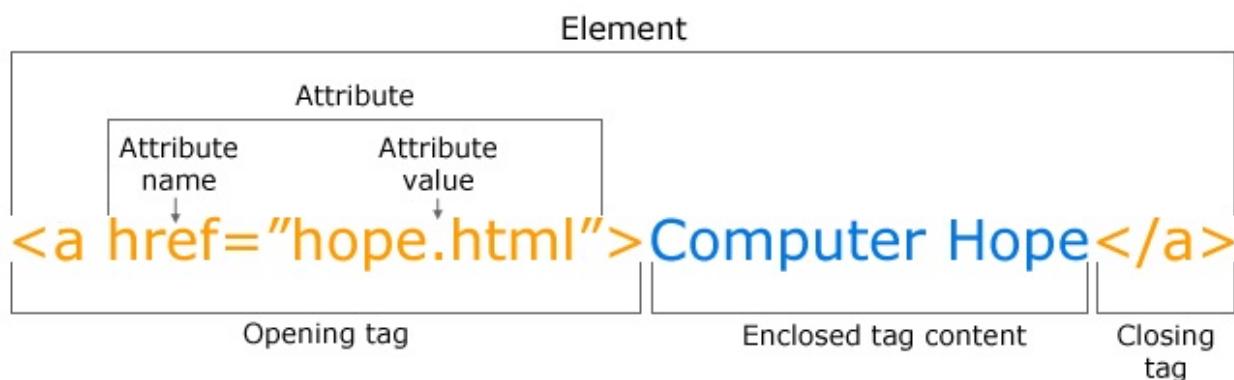
1. 什么是文本 (Text) ?
2. 什么是超文本 (HyperText) ?
3. 什么是标记语言 (Markup Language) ?
4. 什么是网页 (也叫HTML 文档) ?

Learning by Reading

阅读 [W3C School HTML 教程](#) 到 「HTML 内联框架」章节

HTML常用术语

Breakdown of an HTML Tag



元素 - Element

Inquiry Learning

1. 什么是HTML元素(HTML Element)?
2. HTML文档和HTML元素什么关系?

标签 - Tag

Inquiry Learning

什么是HTML标签 (HTML Tag) ?

属性 - Attribute

Inquiry Learning

1. 什么是属性?
2. 标签和属性是什么关系?
3. 能否识别属性的name/value形式?

两个重要属性

- class

Learning by Reading

HTML class 属性 & HTML 类

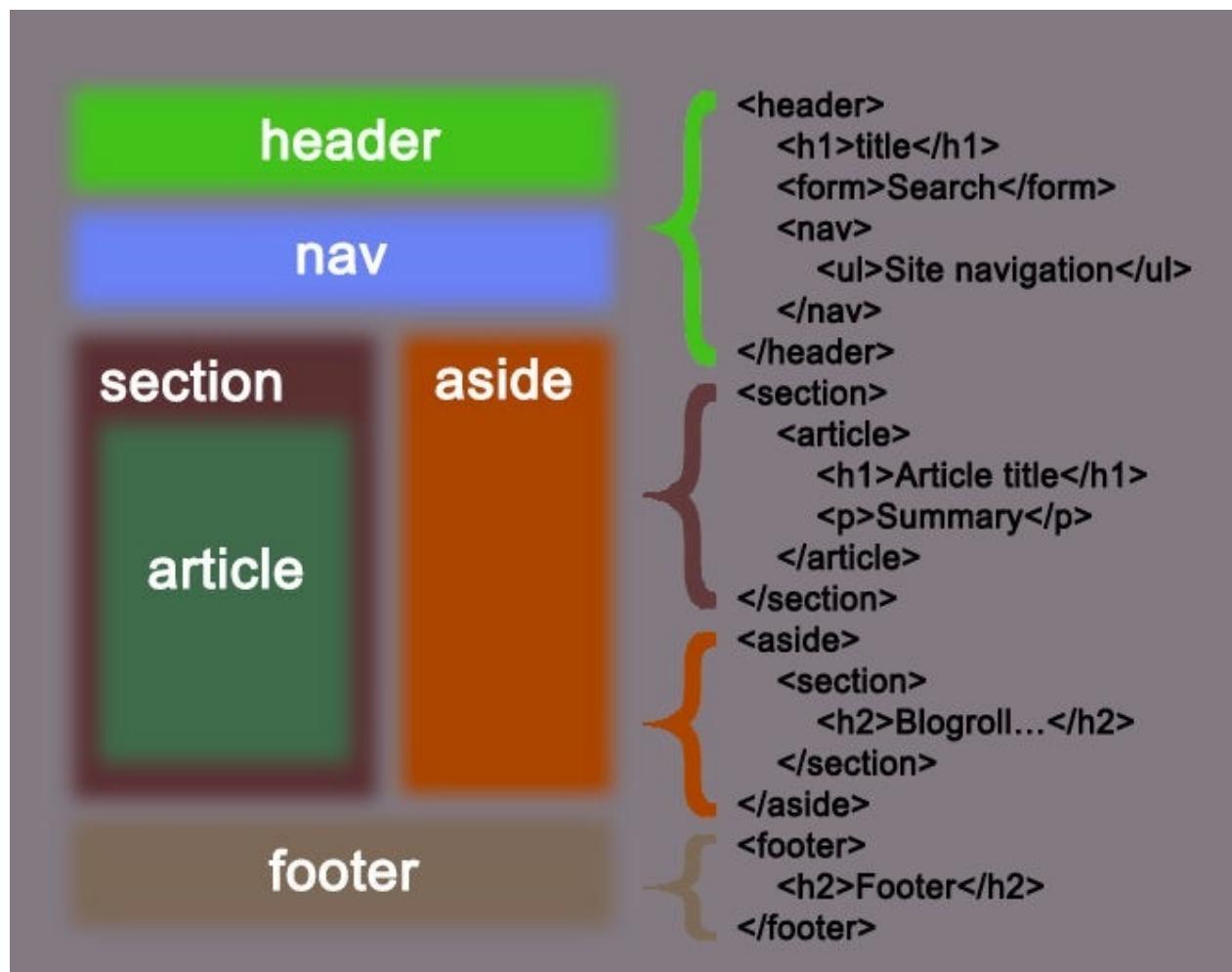
- id

Learning by Reading

HTML id 属性

解剖网站

HTML5 Structure



Inquiry Learning

尝试向你的同学解释下列概念：

`<header>` vs. `<head>` vs. `<h1>` `t` 三者的区别？

Advice for Tutor: 向同学们解释下图，使学员脑海中网页结构的画面与HTML元素一一对应起来

The screenshot shows a web browser displaying the 'Clean Blog' theme by Start Bootstrap. Several annotations with blue question marks and arrows point to specific HTML elements:

- A red arrow points to the title bar with the text "Clean Blog", labeled "Page Title".
- A red arrow points to the URL bar with the URL "https://blackrockdigital.github.io/startbootstrap-clean-blog/index.html", labeled "Website URL".
- A red box highlights the top navigation bar with links for "HOME", "ABOUT", "SAMPLE POST", and "CONTACT".
- A blue circle with a question mark points to the "Header" text, labeled "Header".
- A blue circle with a question mark points to the "Background Image" placeholder, labeled "Background Image".
- A blue circle with a question mark points to the "Clean Blog" h1 heading, labeled "h1 tag".
- A blue circle with a question mark points to the "A Clean Blog Theme by Start Bootstrap" text, labeled "span tag".
- A blue circle with a question mark points to the "Link" icon in the top right corner, labeled "li tag".

Content

Man must explore, and this is exploration at its greatest

Problems look mighty small from 150 miles up

Posted by Start Bootstrap on September 24, 2014

I believe every human has a finite number of heartbeats. I don't intend to waste any of mine.

Posted by Start Bootstrap on September 18, 2014

Science has not yet mastered prophecy

Learning by Doing

1. 找一个你喜欢的网站，调出Chrome开发者工具，向跟你结对的同学互相解剖此网站结构。
2. 打开网站 <https://codepen.io/pen/>， 在HTML编辑框建立自己的第一个HTML文档，并请教练审阅。
 - 所有HTML文档都必须符合特定的结构，要求至少包含以下声明和元素：`<DOCTYPE html>`，`<html>`，`<head>` 和 `<body>`。
 - 此练习必须包含以下HTML标签及其属性。
 - `h1`
 - `p`
 - `li`
 - `body`
 - `meta` (`charset`属性为`utf-8`)
 - `title`
 - `head`
 - `DOCTYPE`

■ html

Advice for Tutors:

1. 聊聊HTML的元素嵌套规则。讲解如何使用Chrome开发者工具定位HTML元素，查看HTML元素的嵌套实例。
2. 聊聊 自闭合元素（self-closing elements）。例如上面第二个练习题就只有开标签，没有闭标签。其他常见自闭合标签：`
` `<embed>` `<hr>` `` `<input>`
`<link>` `<meta>` `<param>` `<source>` `<wbr>`
3. 聊聊 如何使用 <https://validator.w3.org/> 代码验证工具检查HTML代码质量，有三种方式：网址；上传代码文件；直接输入代码。还有 [CSS 验证器](#)

动手 - 修改更多HTML元素



1. 把导航栏的英文换成中文，并添加一列「报名 Girls Coding Day」，链接到 <https://girlscodingday.org/> (考察知识点：考察学员 Lists, 参考 [HTML 列表](#))
2. 修改 `<header>` 元素中的内容 (考察知识点：Text-Based Element, 参考 [HTML 文本格式化](#))
 - 修改 `<h1>` 元素中内容为「Girls Coding Day」
 - 增加 `<h3>` 标签 (考察知识点：Headings系列)
 - `` 标签嵌套一个 `<a>` 标签 (链接为 Bootstrap模板下载地址 <https://startbootstrap.com/template-overviews/clean-blog/>) (考察知识点：Paragraphs, Opening Links in a New Window, 参考资料 [HTML 链接](#))
 - 将博客二字加粗 (考察知识点：Bold Text with Strong)
 - 将制作二字变成斜体 (考察知识点：Italicize Text with Emphasis)

Inquiry Learning :

- i. 什么是块元素?
- ii. 什么是内联元素?
- iii. Div元素和span元素哪个是块元素? 哪个是内联元素?

Learning by Reading : 阅读 [HTML](#) `<div>` 和 `` , 理解块元素和内联元素

- 去 <https://unsplash.com/> 下载图片到 `img` 文件夹，修改图片名字，替换掉背景图 (考察知识点：Image, 参考资料 [HTML 图像](#))

3. 修改 `<footer>` 元素中的内容

- 修改twitter图标为微信图标
- facebook图标为微博图标， 如果有微博请链接到自己的微博

Advice for tutor

- 什么是 Font Awesome (参考: [Font Awesome 中文网](#)) ?
- 怎么使用 Font Awesome ?
- 添加邮箱图标，点击图标可以发送邮件。邮件需要包含主题 (subject)和正文 (body) (参考: [Mailto Links | CSS-Tricks](#)) (备注: Linking to an Email Address)
- 修改copyright段落内容，改成表格样式。 (考察知识点: Organizing Data with Tables, 参考 [HTML 表格](#))

Intro to CSS

本章节目录

1. 引用样式CSS
2. 了解CSS基本知识练习其基本用法
3. 使用编辑器Atom重置CSS样式

初识CSS

Inquiry Learning

尝试向你的同学解释下列概念：

1. 什么是层叠样式表（Cascading Style Sheets）？
2. CSS 规则的两个主要构成部分是什么？
3. HTML和CSS什么关系？

Learning by Reading

阅读[CSS 教程](#)到「CSS 伪元素」章节

引用 CSS

1. HTML 外部样式表

为了让CSS效果能在HTML中显示出来，我们需要在HTML文档中引用CSS文件。引用CSS最好的办法是把所有的样式代码放在额外的一个文件中，在HTML文档中的元素中引用此文件。

Inquiry Learning：想一想这样引用CSS的好处？

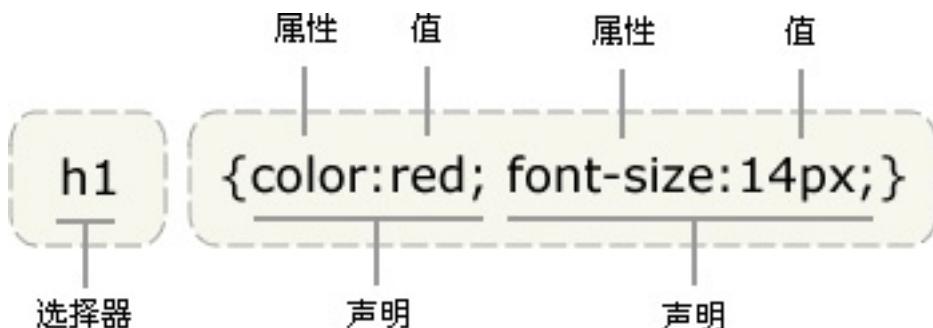
2. 另外两种引用CSS的方法：

- 内部样式表
- 内联样式

Inquiry Learning：当同一个HTML元素被不止被以上三种方法定义时，会使用哪个样式呢？

Learning by Reading：阅读[HTML CSS & CSS 简介 & 如何创建 CSS](#)

CSS常用术语



选择器 - Selector

Inquiry Learning

1. 选择器跟HTML元素什么关系？
2. 如何把相同的声明赋给多个HTML元素？ 提示：选择器分组
3. 子元素会继承父元素的样式，子元素不想继承，该怎么办呢？
4. 什么是「派生选择器、*id* 选择器、类选择器、属性选择器、伪类」？
5. 类选择器 可以在单个HTML文档中出现多次，*id*选择器 也可是在单个HTML文档中出现多次吗？
6. *id*选择器 和 派生选择器 可以混合使用吗？

声明 - Declaration

Inquiry Learning

1. 属性和值用哪个英文标点符合隔开呢？
2. 一个属性可以包含多个值吗？
3. 如果要定义不止一个声明，需要用哪个英文标点符号将每个声明分开？
4. 属性和值一定要英文小写吗？

属性 - Property

大概了解各个属性的效果，不要求熟练掌握。

Inquiry Learning

- 你知道下列CSS背景属性都有什么样式效果吗？

属性	描述
background	简写属性，作用是将背景属性设置在一个声明中。

<code>background-attachment</code>	背景图像是否固定或者随着页面的其余部分滚动。
<code>background-color</code>	设置元素的背景颜色。
<code>background-image</code>	把图像设置为背景。
<code>background-position</code>	设置背景图像的起始位置。
<code>background-repeat</code>	设置背景图像是否及如何重复。

- 你知道下列CSS文本属性的效果吗？

属性	描述
<code>color</code>	设置文本颜色
<code>direction</code>	设置文本方向。
<code>line-height</code>	设置行高。
<code>letter-spacing</code>	设置字符间距。
<code>text-align</code>	对齐元素中的文本。
<code>text-decoration</code>	向文本添加修饰。
<code>text-indent</code>	缩进元素中文本的首行。

- 你知道下列CSS字体属性的效果吗？

属性	描述
<code>font</code>	简写属性。作用是把所有针对字体的属性设置在一个声明中。
<code>font-family</code>	设置字体系列。
<code>font-size</code>	设置字体的尺寸。
<code>font-size-adjust</code>	当首选字体不可用时，对替换字体进行智能缩放。（CSS2.1 已删除该属性。）
<code>font-stretch</code>	对字体进行水平拉伸。（CSS2.1 已删除该属性。）
<code>font-style</code>	设置字体风格。
<code>font-variant</code>	以小型大写字体或者正常字体显示文本。
<code>font-weight</code>	设置字体的粗细。

- 你知道如何设置链接样式吗？

- 你知道下列CSS列表属性的效果吗？

属性	描述
list-style	简写属性。用于把所有用于列表的属性设置于一个声明中。
list-style-image	将图象设置为列表项标志。
list-style-position	设置列表中列表项标志的位置。
list-style-type	设置列表项标志的类型。

- 你知道下列CSS表格属性的效果吗？

属性	描述
border-collapse	设置是否把表格边框合并为单一的边框。
border-spacing	设置分隔单元格边框的距离。
caption-side	设置表格标题的位置。
empty-cells	设置是否显示表格中的空单元格。
table-layout	设置显示单元、行和列的算法。

值 - Value

Inquiry Learning

值为若干单词，则要给值加引号。用双引号还是单引号呢？

Learning by Reading

其他常用术语 [CSS Terms and Definitions](#)

盒子模型 - Box Model

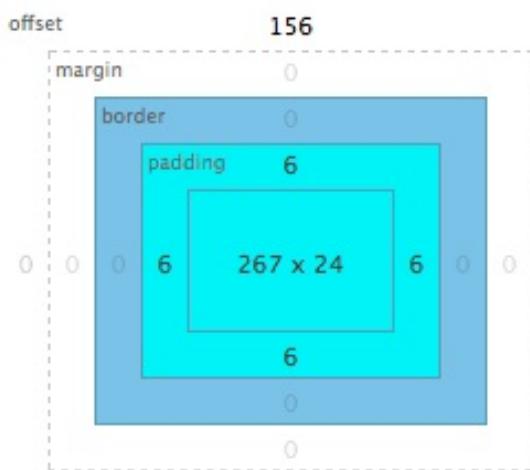
我们已经具备一些基本的HTML和CSS知识，知道一个网页大概长什么样子。现在我们深入讨论**盒子模型**，看看HTML元素到底如何在浏览器中展示的。

根据[CSS 框模型概述](#)，网页文档中所有的元素都是矩形的盒子（rectangular box），有宽度（width）、高度（height）、内边距（padding）、边框（border）、外边距（margin）。

重要的事情说三遍：网页中的每个元素都是矩形的盒子。网页中的每个元素都是矩形的盒子。网页中的每个元素都是矩形的盒子。

Advice for tutor

1. 请教练以下图和代码为例，讲讲盒子的大小



*Firebug's Box Model Display
(not to scale, but very useful)*

```
div {
    width: 267px;
    height: 24px;
    padding: 6px;
    border: 0px;
    margin: 0px;
}
```

Margin 比较特别，它不会影响盒子本身的大小，但是它会影响和盒子有关的其他内容，因此 margin 是盒模型的一个重要的组成部分。

盒子本身的大小是这样计算的：| 盒子大小 | 计算方法 || ----- | -----
----- | | **Width** | width + padding-left + padding-right + border-left +
border-right | | **Height** | height + padding-top + padding-bottom + border-top +

border-bottom |

使用Chrome开发者工具调试样式

Advice for tutor

请教练使用Chrome开发者工具，简单演示如何调试盒子模型属性以及其他样式。

