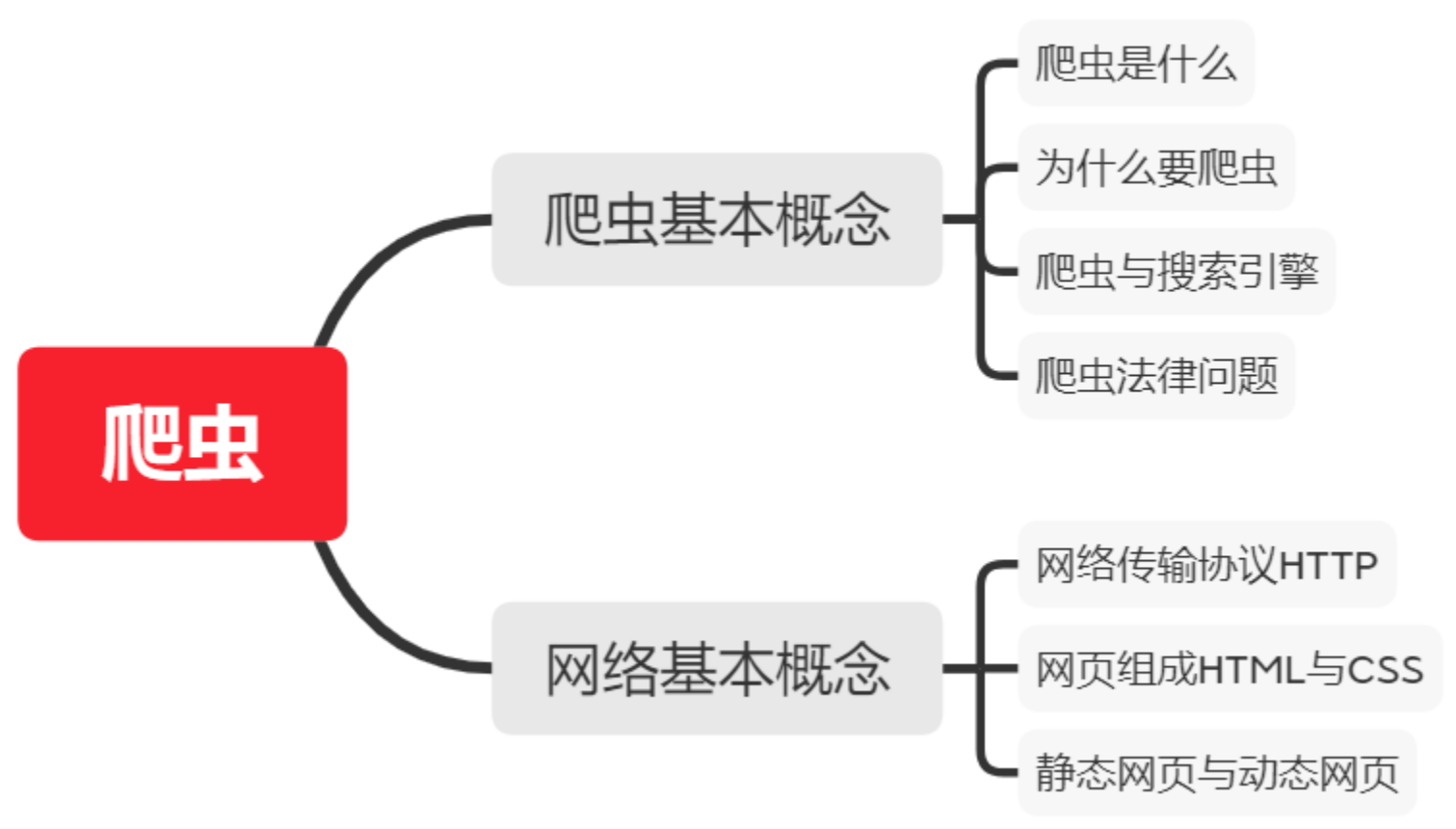
Python数据分析与机器学习前沿技术

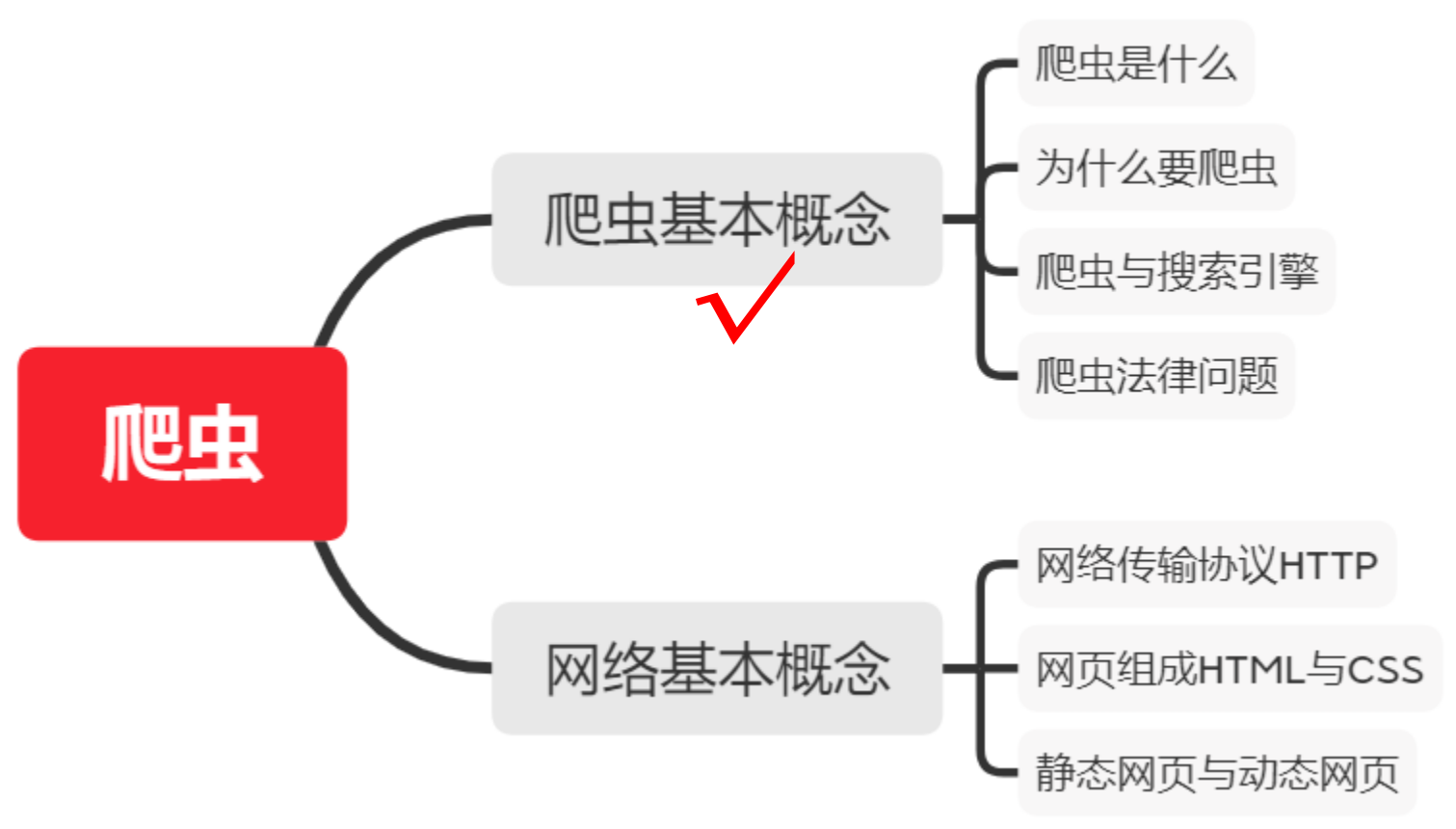
网络数据爬虫及信息抽取

2021.1

内容提纲

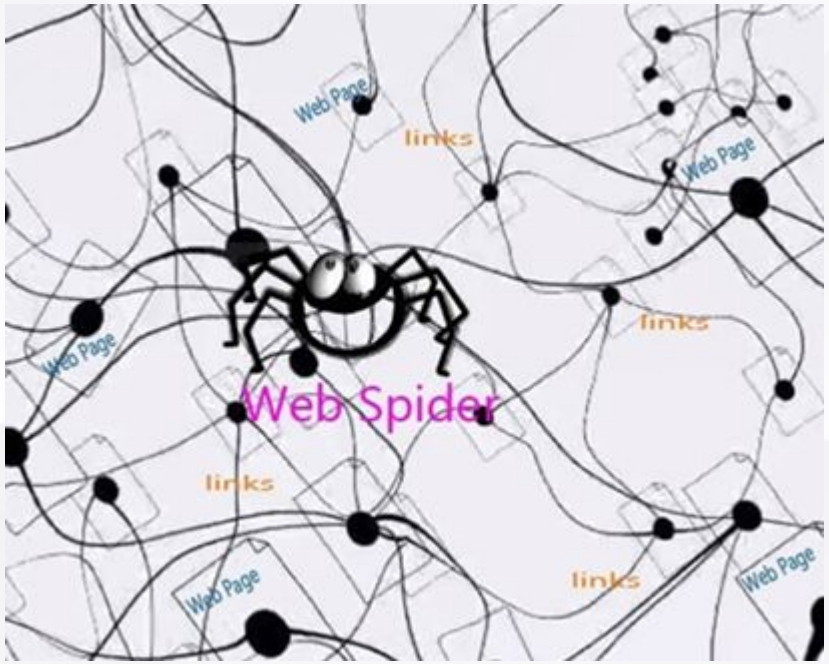


内容提纲



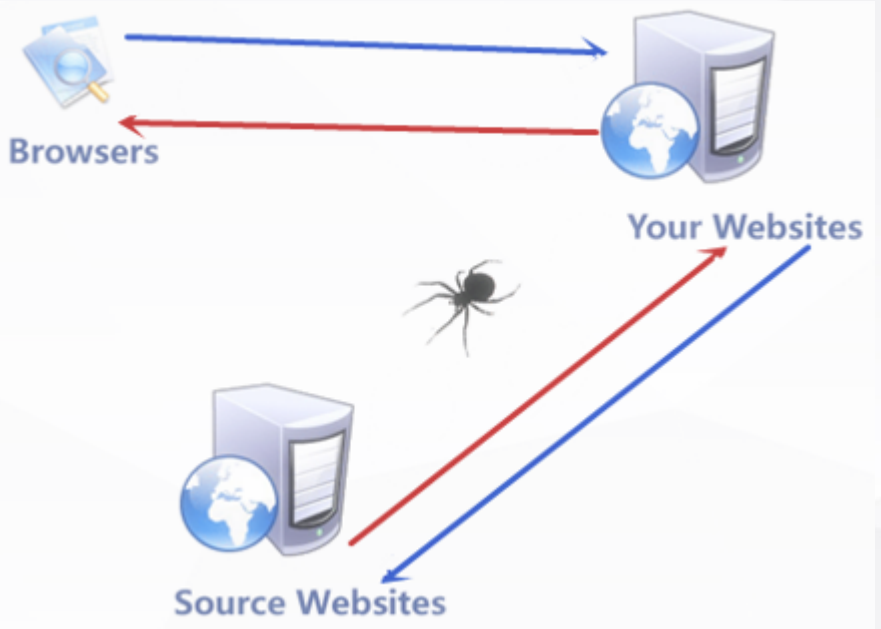
爬虫的定义

- **网络爬虫**（又被称为网页蜘蛛，网络机器人）就是**模拟浏览器发送网络请求**，**接收请求响应**，一种按照一定的规则，自动地抓取互联网信息的程序。
- 爬虫就是**模拟浏览器的行为**，越像越好，越像就越不容易被发现。
- 另外一些不常使用的名字还有蚂蚁、自动索引、模拟程序或者蠕虫。



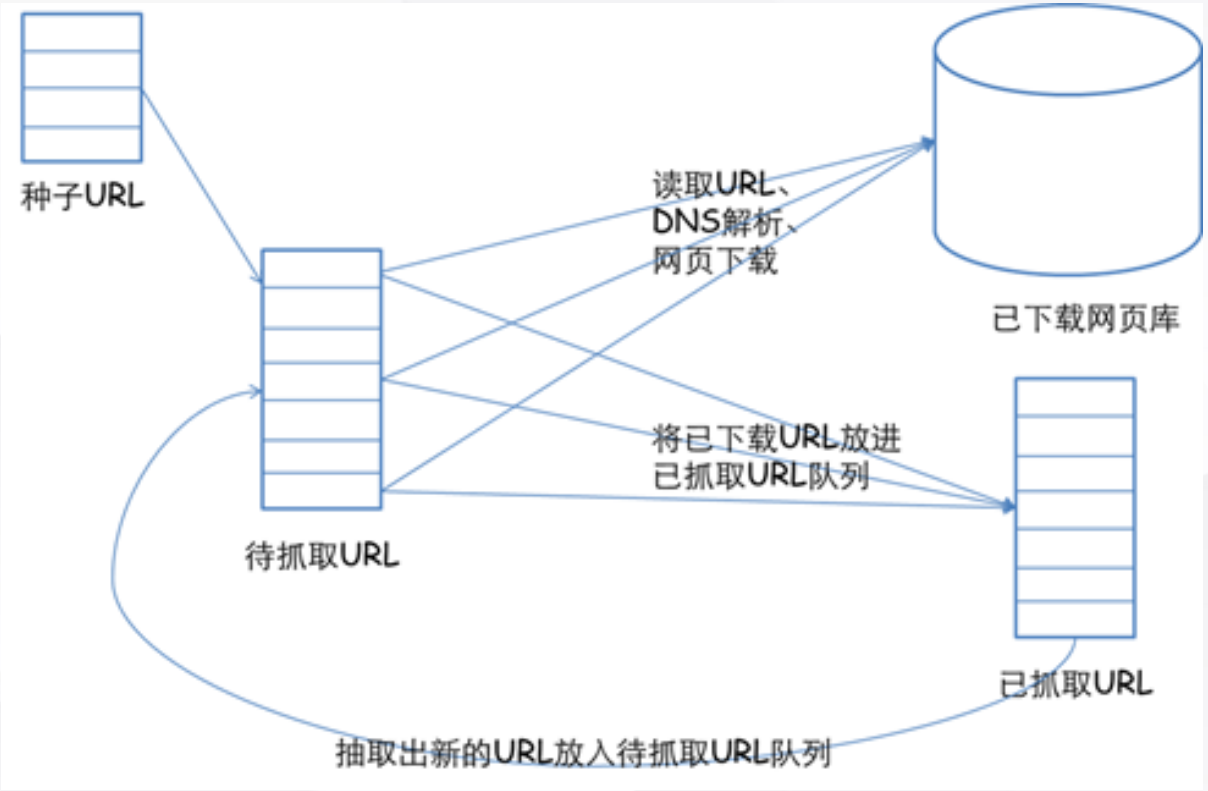
爬虫的基本结构

- 在网络爬虫的系统框架中，主过程由控制器，解析器，资源库三部分组成。
- **控制器** 负责给多线程中的各个爬虫线程分配工作任务。
- **解析器** 负责下载网页，进行页面的处理，主要是将一些JS脚本标签、CSS代码内容、空格字符、HTML标签等内容处理掉，爬虫的基本工作是由解析器完成。
- **资源库** 用来存放下载到的网页资源，一般都采用大型的数据库存储，如Oracle数据库，并对其建立索引。



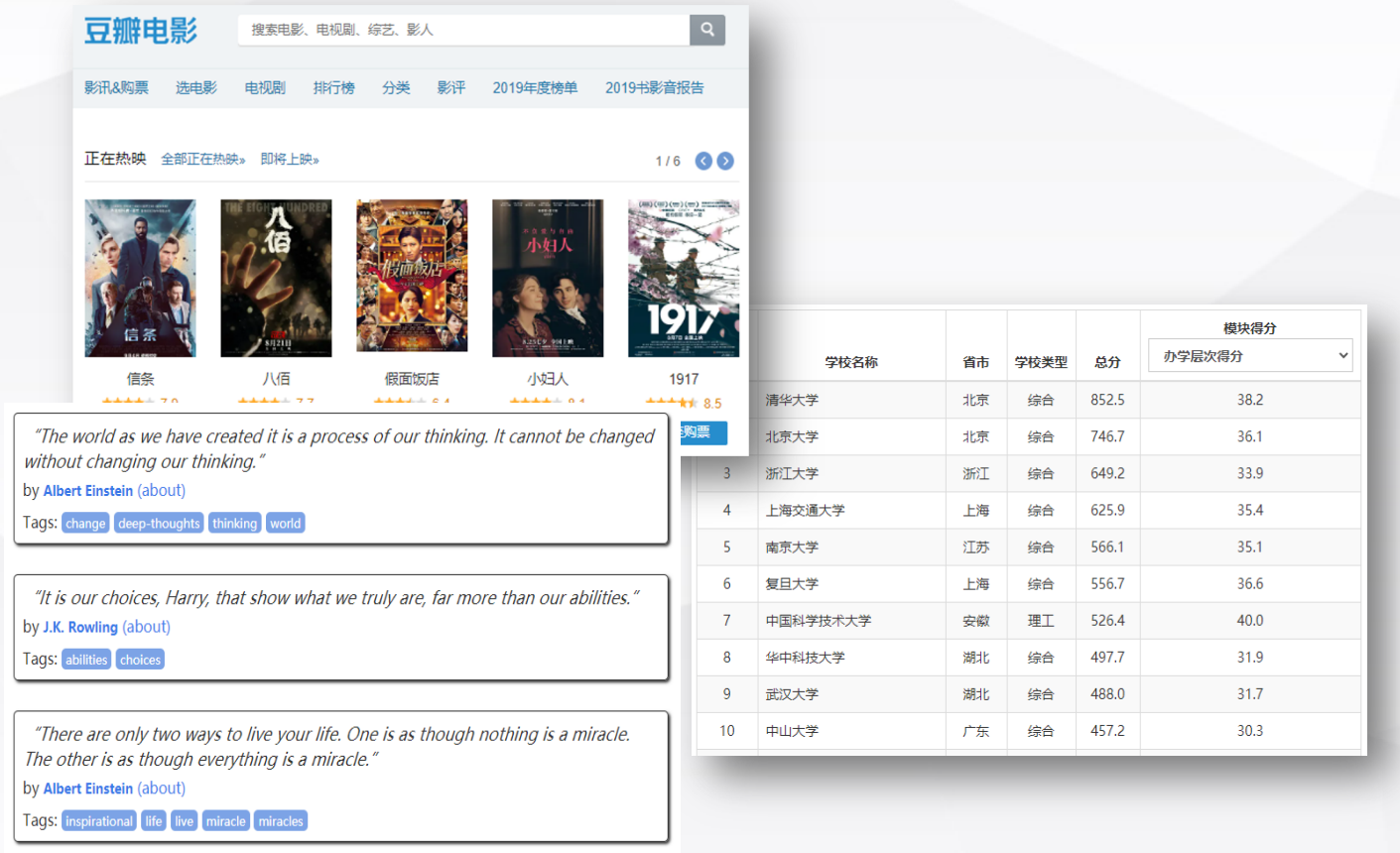
爬虫的工作流程

- 一个通用的网络爬虫的框架



爬虫的用途

- 搜索引擎
  - 百度、谷歌、必应搜索等
- 购物助手
  - 惠惠购物助手
- 数据分析
  - 社交网络分析
- 自动化操作
  - 抢票软件
  - 自动化关注，评论，回复
- ....



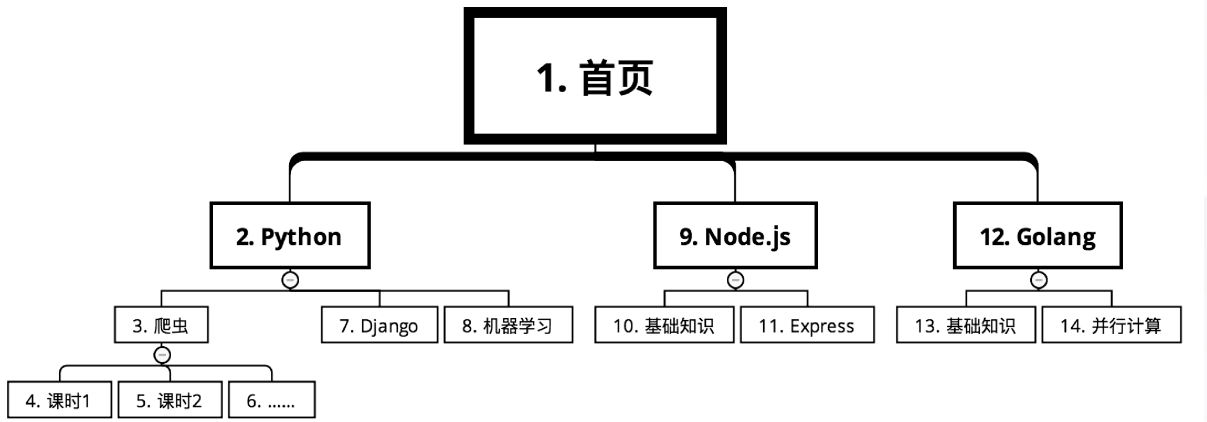
爬虫与搜索引擎

- **搜索引擎**（Search Engine）是指根据一定的**策略**、运用特定的**计算机程序**从互联网上搜集信息，在对信息进行组织和处理后，为用户提供检索服务，将用户检索相关的信息展示给用户的系统。
- **爬虫**（Crawler）是从搜索引擎机器人程序发展而来。虽然两者在功能上很相似，但是爬虫程序却可以通过分析遍历来的网页中含有的网页链接信息，自动获取下一步需要遍历的网页，这个过程可以自动的持续进行下去。爬虫是个非常形象的称呼，也有人称之为**蜘蛛**（Spider），它们都是一个意思，真像Internet上的一个蜘蛛爬虫，自由的跑来跑去，抓取所能获得的各种网页信息。

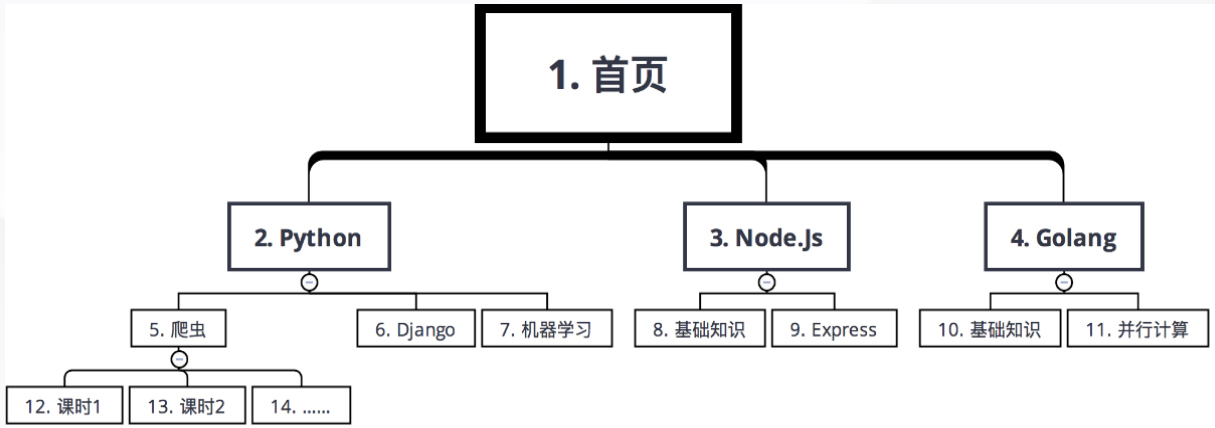
爬虫的抓取策略

- 在爬虫系统中，**待抓取URL队列**是很重要的一部分。待抓取URL队列中的URL以什么样的顺序排列也是一个很重要的问题，因为这涉及到先抓取那个页面，后抓取哪个页面。而决定这些URL排列顺序的方法，叫做**抓取策略**。

✓ 深度优先遍历策略



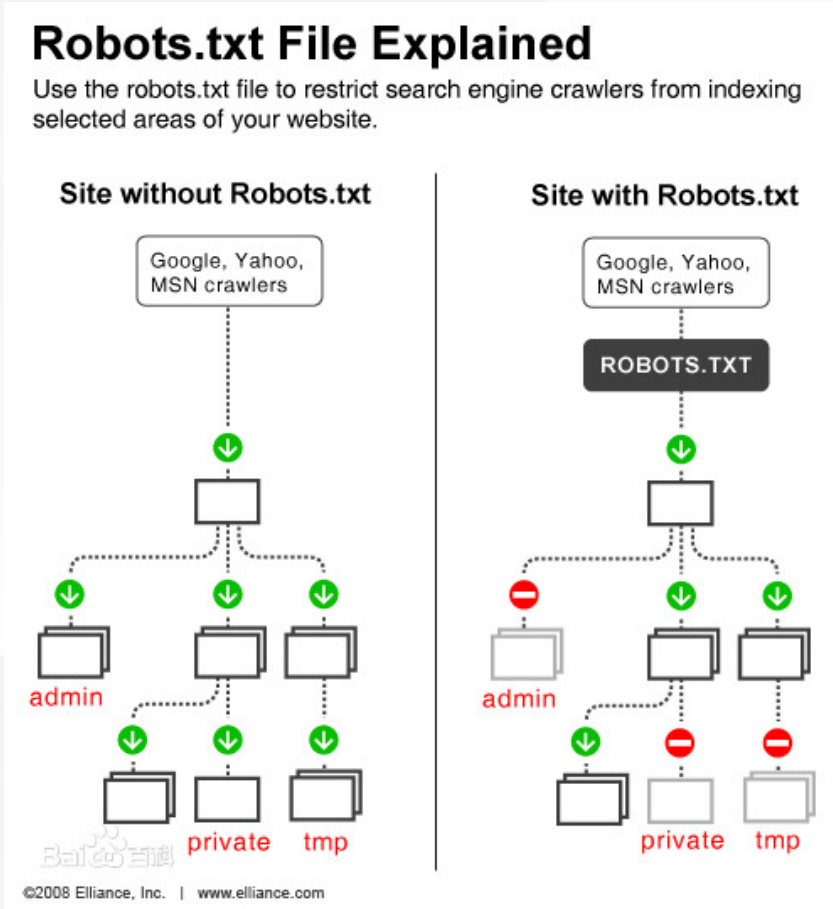
✓ 广度优先遍历策略



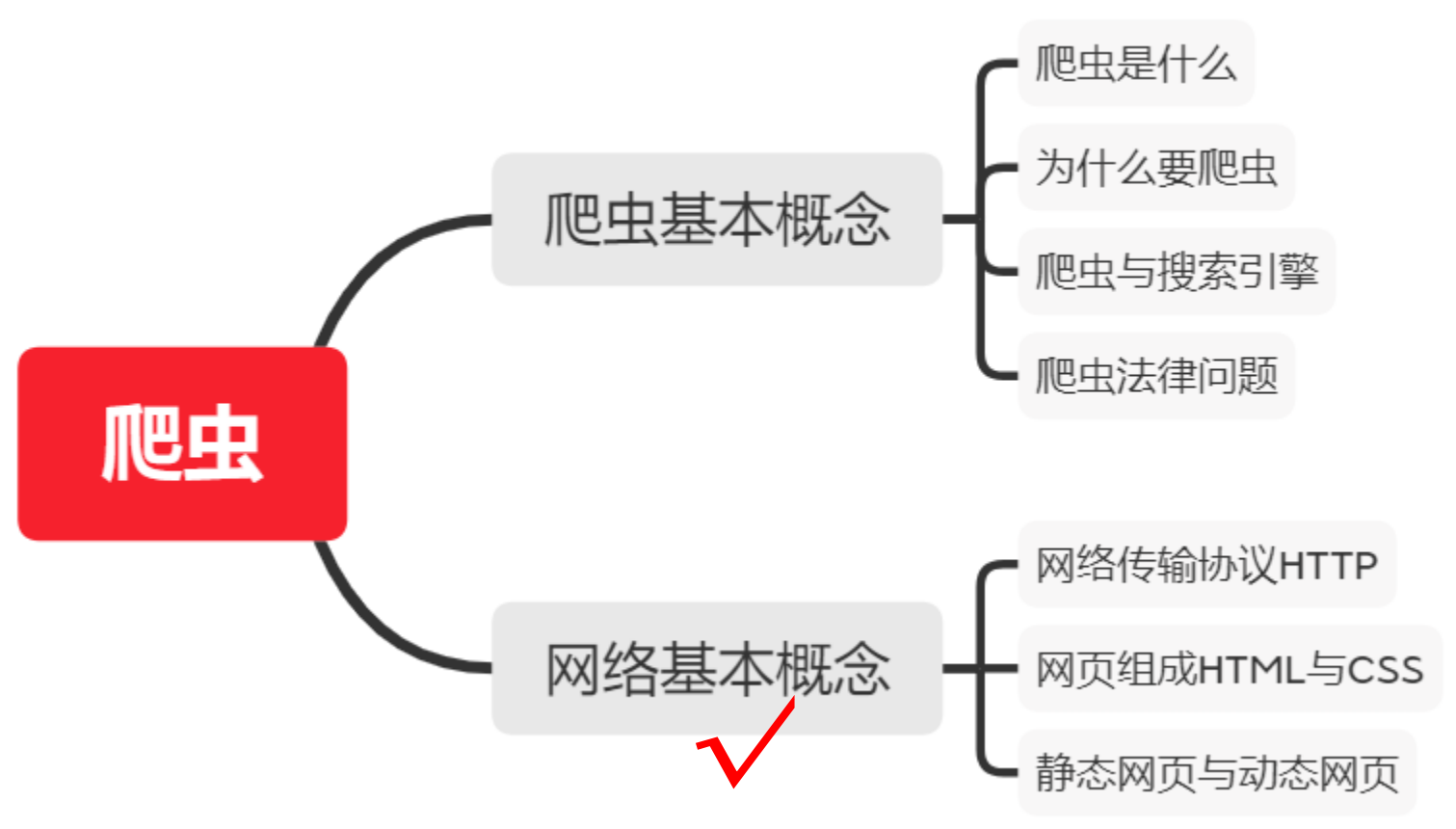
- 大站优先策略、 PageRank策略

爬虫与robots协议

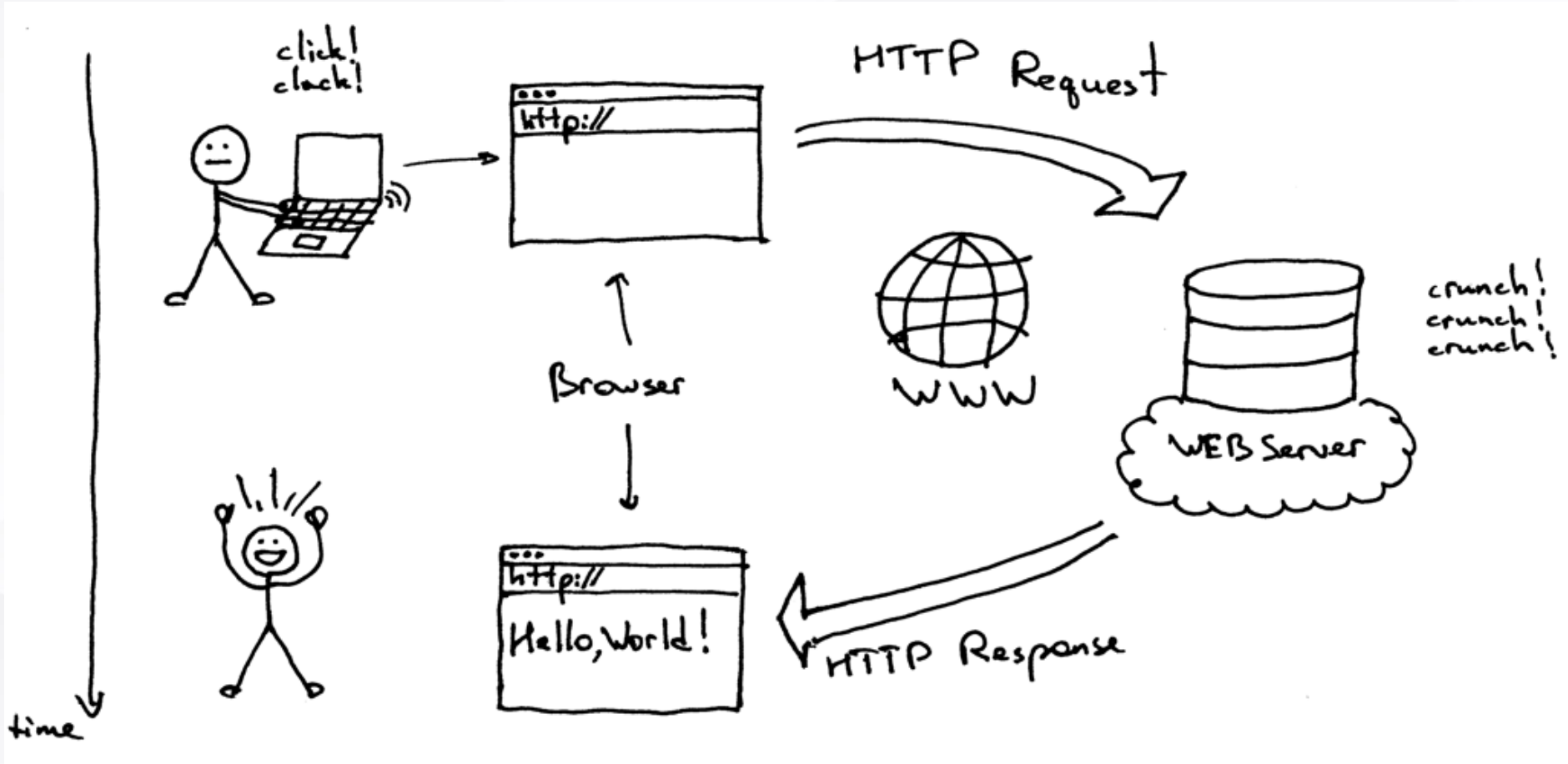
- 网络蜘蛛需要抓取网页，不同于一般的访问，如果控制不好，则会引起**网站服务器负担过重**。比如，淘宝就因为雅虎搜索引擎的网络蜘蛛抓取其数据引起淘宝网服务器的不稳定。
- 每个网络蜘蛛都有自己的名字，在抓取网页的时候，都会向网站标明自己的身份，用于标识此**网络蜘蛛的身份**。
- 让网站和网络蜘蛛进行交流。一方面让网站管理员了解网络蜘蛛都来自哪儿，做了些什么，另一方面也告诉网络蜘蛛哪些网页不应该抓取，哪些网页应该更新。



内容提纲



HTTP请求流程



HTTP概念

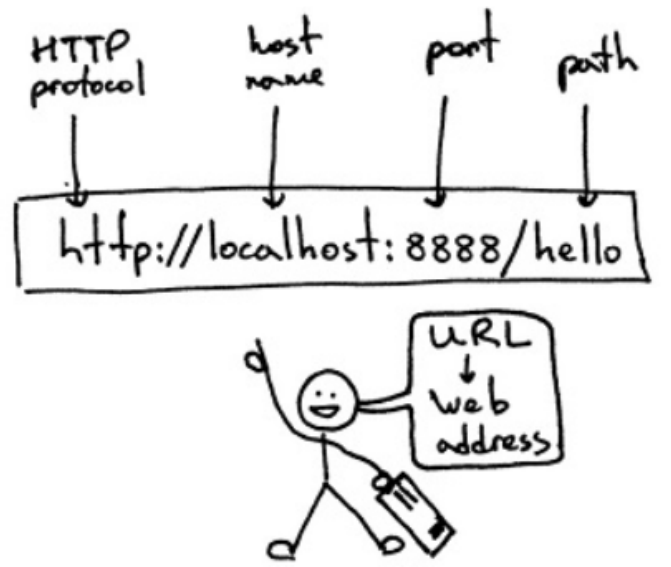
- 概念：HTTP（超文本传输协议）是应用层上的一种客户端/服务端模型的通信协议，它由请求和响应构成，且是无状态的。
- 协议：协议规定了通信双方必须遵守的数据传输格式，这样通信双方按照约定的格式才能准确的通信。

URL格式　　HTTP请求　　HTTP响应

- 无状态：无状态是指两次谅解通信之间是没有任何联系的，每次都是一个新的连接，服务端不会记录前后的请求信息。

URL地址格式

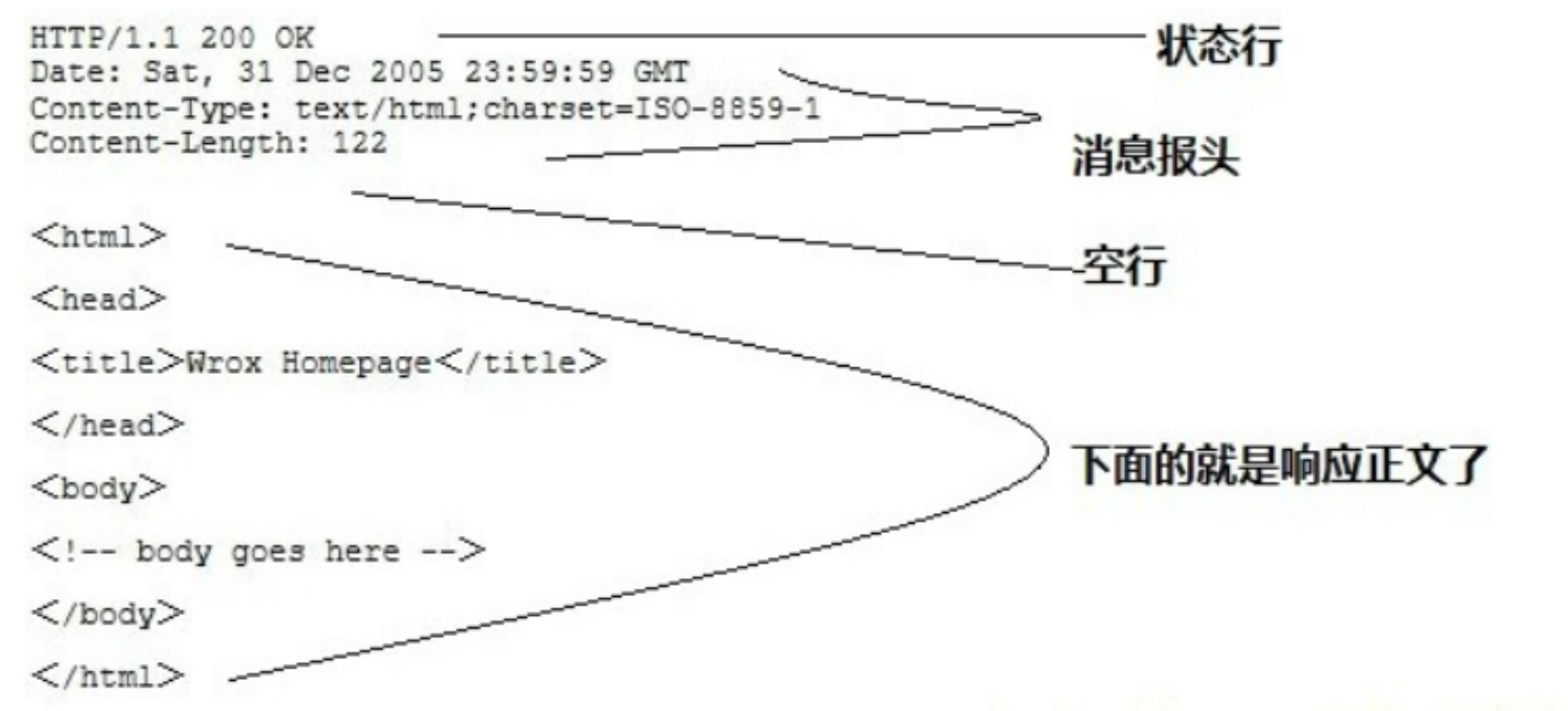
- 格式说明： scheme://host[:port]/path/…/[?query-string][#anchor]
  - scheme： 协议（例如： http, https, ftp）
  - host： 服务器的 IP 地址或者域名
  - port： 服务器的端口（如果是走协议默认端口， 缺省端口80）
  - path： 访问资源的路径
  - query-string： 参数， 发送给 http 服务器的数据
  - anchor： 锚（跳转到网页的指定锚点位置）



HTTP请求

HTTP响应

- HTTP响应由四个部分组成，分别是：状态行、消息报头、空行（回车符 + 换行符）和响应正文。



HTML与CSS

- 访问知乎Python社区
  - https://www.zhihu.com/column/Python
- 返回HTML文件如下

```
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Python中文社区</title>
</head>
<body>
<div>
    <h1 class="Column-Title">Python中文社区</h1>
    <p class="Column-Desc">同名微信公众号：Python中文社区，欢迎投稿</p>
</div>
<script src="https://static.zhihu.com/heifetz/vendor.js"></script>
<script src="https://static.zhihu.com/heifetz/column.app.js"></script>
</body>
</html>
```

HTML教程：https://www.w3school.com.cn/html/index.asp

HTML

- 网页内容的载体

CSS

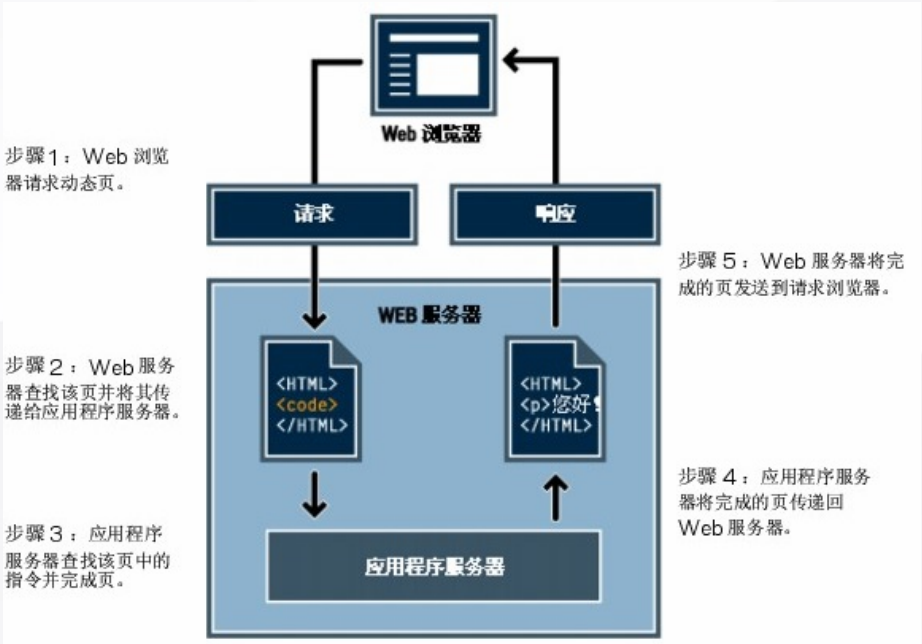
- 文字色彩、字体、动画效果

JS

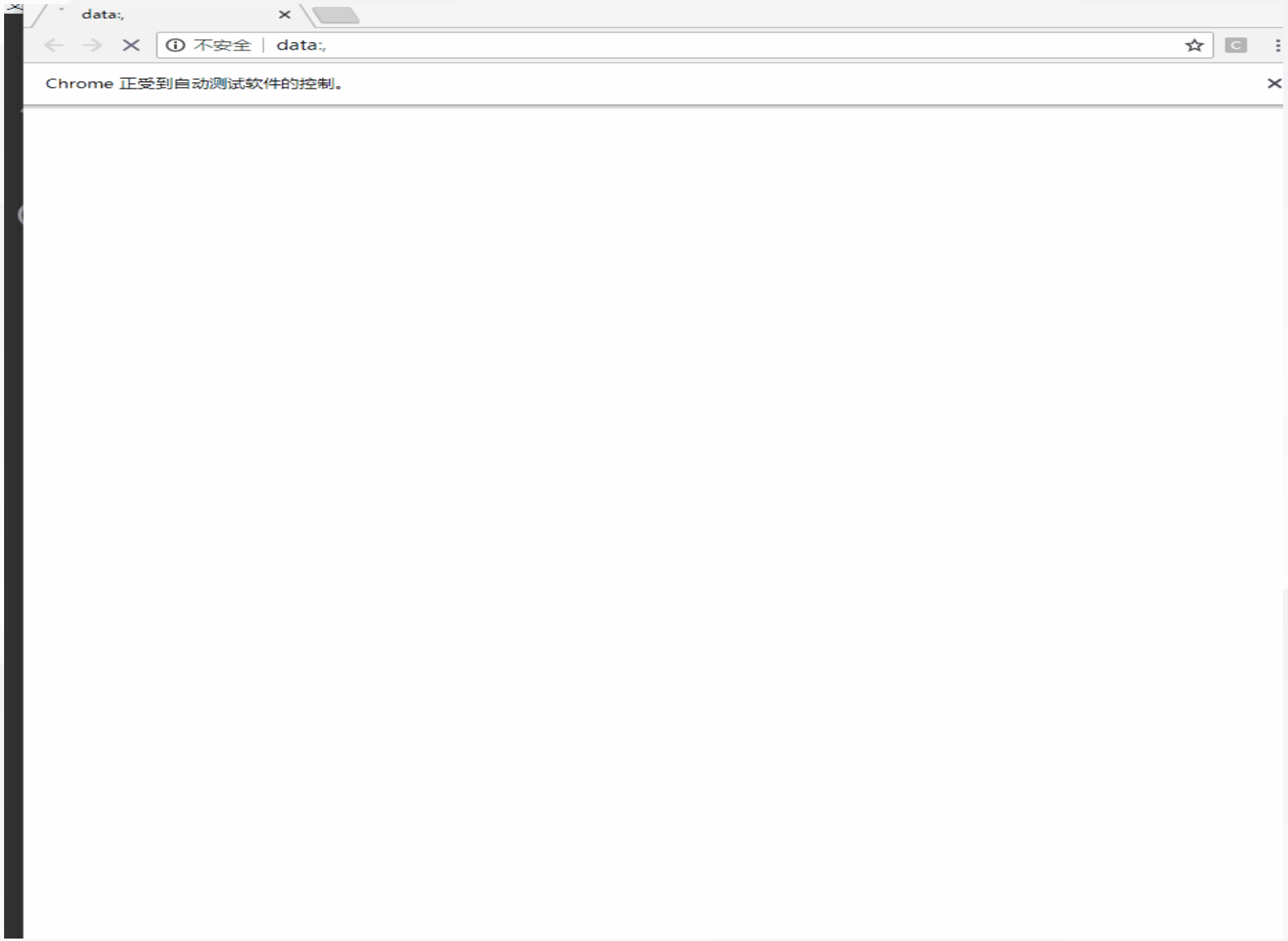
- 用户与网页的交互

动态网页

- 所谓的**动态网页**，是指跟**静态网页**相对的一种网页编程技术。静态网页，随着HTML代码的生成，页面的内容和显示效果就基本上不会发生变化了——除非你修改页面代码。而动态网页则不然，页面代码虽然没有变，但是显示的内容却是可以随着**时间**、**环境**或者**数据库**操作的结果而发生改变的。
- 爬虫取到的页面仅仅是一个静态的页面，即网页的源代码，就像在浏览器上的“**查看网页源代码**”一样。一些动态的东西如JavaScript脚本执行后所产生的信息，是抓取不到的。



Selenium让爬虫更像用户



- 框架底层使用**JavaScript模拟**真实用户对浏览器进行操作，Selenium测试直接运行在浏览器中，代码执行时，可以自动打开浏览器/表单输入/按钮点击，就像真实用户在操作的一样。

谢谢