# 第四阶段 Python数据分析

# Python 爬虫1

## 网校相关视频

Python爬虫开发

第一章网络爬虫基础

## 明确本次课知识点，明确重点难点

### 【知识点目标】

* 网络爬虫相关概念
* 网页基础
* 用urllib.request爬取整个网页
* 构造请求头
* 使用代理服务器

### 【重点】

* 网页基础
* 用urllib.request爬取整个网页
* 构造请求头
* 使用代理服务器

### 【难点】

* 构造请求头
* 使用代理服务器

## 复习巩固作业讲解

## 本次课程任务讲解

### 【知识点1】网络爬虫相关概念

1. 什么是网络爬虫

爬虫，即网络爬虫，大家可以理解为在网络上爬行的一只蜘蛛，互联网就比作一张大网，而爬虫便是在这张网上爬来爬去的蜘蛛咯，如果它遇到资源，那么它就会抓取下来。

1. 爬虫的分类

Crawler ，即Spider（网络爬虫），其定义有广义和狭义之分。

* 1. 狭义上指遵循标准的 http 协议，利用超链接和 Web 文档检索方法遍历万维网的软件程序；
  2. 而广义的定义则是能遵循 http 协议，检索 Web 文档的软件都称之为网络爬虫。

网络爬虫是一个功能很强的自动提取网页的程序，它为搜索引擎从万维网上下载网页，是搜索引擎的重要组成部分。

1. 爬虫可以做什么

爬取数据：文本、图像、视频......

1. 爬虫爬取步骤--三步走
   1. 爬哪里（URL）:获取整个页面数据
   2. 爬什么：找到你需要的内容
   3. 怎么爬：利用正则表达式、BeautifulSoup等对页面进行解析
2. 其他相关概念
   1. 浏览网页的过程

在用户浏览网页的过程中，我们可能会看到许多好看的图片，比如 http://image.baidu.com/ ，我们会看到几张的图片以及百度搜索框，这个过程其实就是用户输入网址之后，经过DNS服务器，找到服务器主机，向服务器发出一个请求，服务器经过解析之后，发送给用户的浏览器 HTML、JS、CSS 等文件，浏览器解析出来，用户便可以看到形形色色的图片了。

因此，用户看到的网页实质是由 HTML 代码构成的，爬虫爬来的便是这些内容，通过分析和过滤这些 HTML 代码，实现对图片、文字等资源的获取。

* 1. URL的概念及使用
     1. URL

URL，即统一资源定位符，也就是网址，统一资源定位符是对可以从互联网上得到的资源的位置和访问方法的一种简洁的表示，是互联网上标准资源的地址。互联网上的每个文件都有一个唯一的URL，它包含的信息指出文件的位置以及浏览器应该怎么处理它。

* + 1. URL的格式

URL由三部分组成:

第一部分是协议(或称为服务方式)。例如http://  
第二部分是存有该资源的主机IP地址(有时也包括端口号)。如127.0.0.1或域名www.baidu.com  
第三部分是主机资源的具体地址，如域名后的目录及文件名爬虫爬取数据时必须要有一个目标的URL才可以获取数据，因此，它是爬虫获取数据的基本依据

例如：<https://movie.douban.com/subject/1292052/>



### 【知识点2】网页基础

1. HTML结构

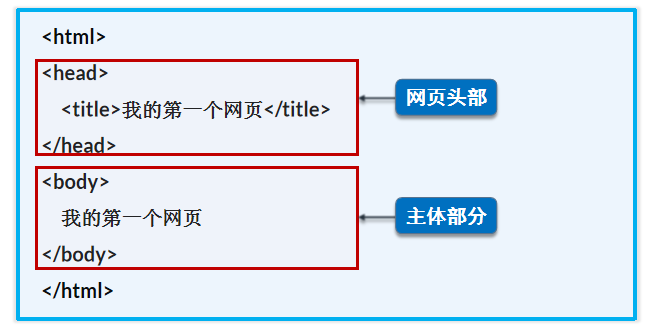
HTML是超文本标记语言，被web浏览器进行解析查看。它的文档制作简单，功能强大，支持很多其他格式文件的嵌入。

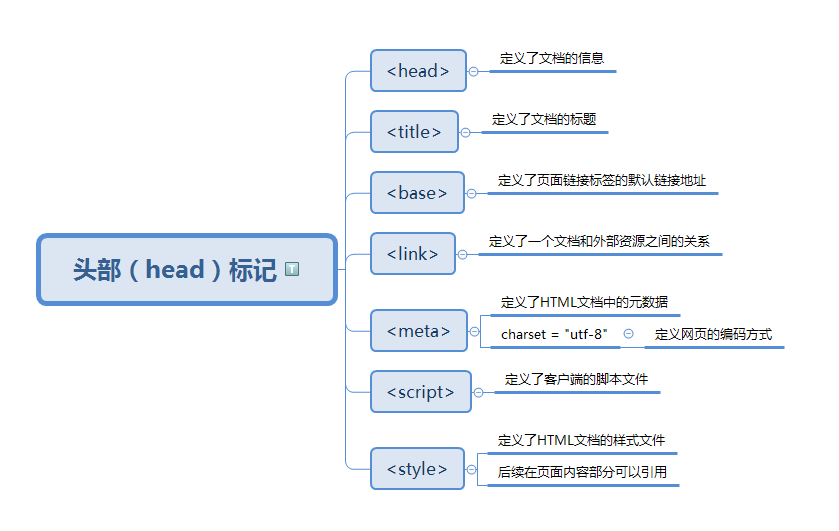
HTML主要由标签构成

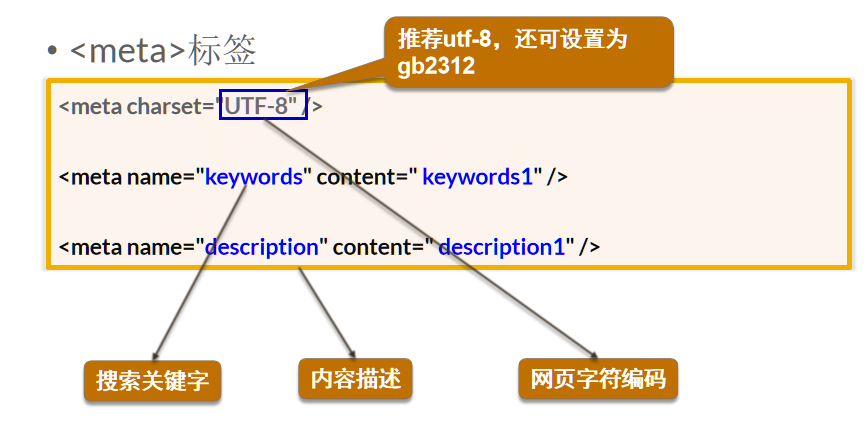
<html></html> 双标签：范围

<img src=”images/1.jpg”> 单标签：对象

”超文本“文档主要由头部（head）和主体（body）两部分组成。







***编码的变迁：从ascii到utf-8***

***ascii***

***gb2312***

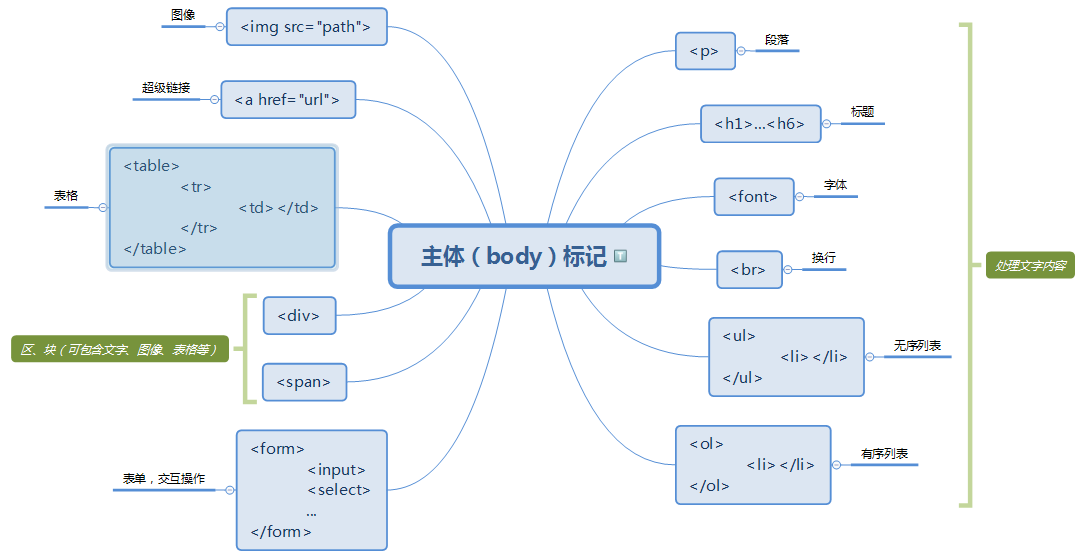
***gbk***

***big5***

***unicode***

***utf-8***

参考网页：<https://tgideas.qq.com/webplat/info/news_version3/804/808/811/m579/201307/218730.shtml>



1. 在HTML结构中找到需要的标签（定位）

文档主体部分可包含的标签元素有很多，各个标签元素的功能也不同，为了对标签进行区分，通常会为标签赋予一定的属性，通过属性值往往可以实现对标签的定位。

***<div id=‘id’ class=‘class’ attr=‘’>content</div>***

div是标签名

**id是为该标签设置的唯一标识**

**class是该标签的样式类**

attr是设置的其他属性

content是标签的文本内容

爬虫在爬取到网页后，就是利用这样的属性来找到相应的标签，再提取标签中的对应内容（文字、图像等）

参考网页：http://www.w3school.com.cn/html/index.asp

### 【知识点3】用urllib.request爬取整个网页

1. 获取整个网页内容  
    *import urllib.request  
    response = urllib.request.urlopen("http://www.baidu.com")  
    print(response.read().decode('utf-8'))*
   1. urllib：Python内置的HTTP请求库，不需安装即可使用

包含**request**、error、parse、robotparser等模块

* 1. urllib.request.urlopen方法
     1. 功能：获取网页信息
     2. 语法形式：

urllib.request.urlopen(url, data=None, [timeout])

- url:  需要打开的网址

- data：Post提交的数据

- timeout：设置网站的访问超时时间

* + 1. 结果：

response = urllib.request.urlopen("http://www.baidu.com")

print(response.read().decode('utf-8’))

# 使用read()方法读取响应对象中的文本，

# 注意：得到的文本数据格式为bytes类型，需要decode（）解码，转换成str类型。

* + 1. 其他方法：

read()：读取全部网页内容（最常用），返回bytes，需要用decode转换成str

readline()：读取行（首行），返回bytes

readlines()：读取全部内容并返回行的list，需要遍历

fileno()：读取文件描述符，是一个int值（linux/unix系统下的socket）

close()：关闭连接

info()：返回HTTPMessage对象，表示远程服务器返回的头信息

getcode()：返回HTTP状态码  
geturl()：返回URL对象

### 【知识点4】Python爬虫突破封禁--构造请求头

1. 相关概念
   1. request对象：是从客户端向服务器发出请求，包括用户提交的信息以及客户端的一些信息。客户端可通过HTML表单或在网页地址后面提供参数的方法提交数据，然后通过request对象的相关方法来获取这些数据。request的各种方法主要用来处理客户端浏览器提交的请求中的各项参数和选项。  
       python爬虫中的request其实就是通过python向服务器发出request请求，得到其返回的信息。
   2. POST和GET数据传送

常见的http请求方式有get、post、put、delete等

* + 1. get方式

get是比较简单的http请求，直接会将发送给web服务器的数据放在请求地址的后面，即在请求地址后面使用？key1=value1&key2=value2形式传递数据，只适合于数据量少，且没有安全性要求的请求。

[https://www.baidu.com/s?ie=utf-8&f=8&rsv\_bp=1&rsv\_idx=1&tn=baidu&wd=mathtype&oq=mathtype&rsv\_pq=9bcb70e7001b699d&rsv\_t=1948p47ET%2FZwqiCm2c8cr1%2BrrkX%2BjEXiqziVdKQkGJgaF8FKtcroiqQm4MI&rqlang=cn&rsv\_enter=0&rsv\_dl=tb](https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=1&rsv_idx=1&tn=baidu&wd=mathtype&oq=mathtype&rsv_pq=9bcb70e7001b699d&rsv_t=1948p47ET/ZwqiCm2c8cr1+rrkX+jEXiqziVdKQkGJgaF8FKtcroiqQm4MI&rqlang=cn&rsv_enter=0&rsv_dl=tb)

* + 1. post方式

post是将需要发送给web服务器的数据经过编码放到请求体中，可以传递大量数据，并且有一定的安全性，常用于表单提交。

http://www.lagou.com

1. 构造合理的HTTP请求头

有些网站不会同意程序直接用程序的方式进行访问，如果识别有问题，那么站点根本不会响应，所以为了完全模拟浏览器的工作，我们需要设置一些Headers HTTP的请求头的信息。

HTTP的请求头是在你每次向网络服务器发送请求时，传递的一组属性和配置信息。HTTP定义了十几种古怪的请求头类型，不过大多数都不常用。只有下面的七个字段被大多数浏览器用来初始化所有网络请求。



* 1. headers的Connection\Accept\Accept-Language\User-Agent属性

Connection：默认进行持久连接keep-alive，close表明当前正在使用的tcp链接在当天请求处理完毕后会被断掉；

Accept：代表浏览器可以接受服务器回发的内容的类型；

Accept-Language：浏览器可接受的语言；

User-Agent：向访问网站提供你所使用的浏览器类型、操作系统及版本、CPU 类型、浏览器渲染引擎、浏览器语言、浏览器插件等信息的标识。

* 1. 构造Request对象，使用请求头

req = urllib.request.Request(url,headers = headers string) #字典形式

page = urllib.request.urlopen(req).read()

text = page.decode(‘utf-8’)

参考：https://blog.csdn.net/qq\_34246164/article/details/81604891（常用请求头）

### 【知识点5】Python爬虫突破封禁--使用代理服务器

由于一直用同一个IP爬取目标网站的数据，如果访问次数过多，目标网站服务器会禁止你的访问，所以需要经常更换自己IP。这时候就需要代理服务器了。

* 1. 代理服务器原理

匿名

防火墙

* 1. 爬虫为何要使用代理服务器

防BAN

稳定高速

* 1. 代理服务器的分类

透明代理

匿名代理

高匿代理

* 1. 使用代理进行爬取的步骤

## 自主学习作业讲解

无

## 课程总结

#### ndarray的常用操作

## 下次自主学习任务布置

编写代码实现下列需求

需求1：判断字符串是否符合111-1111-1111的形式（电话号码）

如：str1="111-1111-1111"

需求2：找出字符串中符合电话号码格式的子字符串

如：str2=”alskdjlaksdjf111-1111-1111dskljfsda”