# Nodejs

## Nodejs的理解

nodejs是一个基于Chrome V8 引擎的JS运行环境，也就是让javascript运行在服务器（server）端，

NodeJS使用了一个事件驱动，非阻塞式的I/O模型，使得其轻量又高效。

Nodejs包管理器npm是全球最大的开源生态系统。

nodejs是服务端的js平台

工作原理：nodejs的工作原理其实就是事件循环。每一条nodejs的逻辑都是写在回调函数里面的，而回调函数都是返回之后才异步执行的

## Mongodb



MongoDB 是一个基于分布式文件存储的数据库。由 C++ 语言编写。旨在为 WEB 应用提供可扩展的高性能数据存储解决方案。

MongoDB 是一个介于关系数据库和非关系数据库之间的产品，是非关系数据库当中功能最丰富，最像关系数据库的。

在高负载的情况下，添加更多的节点，可以保证服务器性能。

MongoDB 将数据存储为一个文档，数据结构由键值(key=>value)对组成。MongoDB 文档类似于 JSON 对象。字段值可以包含其他文档，数组及文档数组。

简单的使用方法：打开mongod.exe –dbpah 你要运行mongo的url

show dbs 查看你有多少个表 db 显示当前操作 help 帮助 show collections 看有哪些表 use 名字 切换和使用 db.stats() db.dagou.insert()写入 db.dagou.find() 查看

### 安装配置

1.下载 <http://www.mongodb.org/downloads>，双击安装完成（安装路径尽量简单，不要有中文）

启动数据库：

2.打开命令行，跳转到安装目录的 bin 目录执行下面命令

mongod.exe --dbpath E:\DB\MONGO\db

安装成功后可以在浏览器访问

<http://127.0.0.1:27017/> （成功之后会出现一串英文）

这里每次觉得麻烦的话，可以配置window服务

访问成功则表示安装成功且已启动数据库服务

## Express

### 概念的介绍

. Express 是一个简洁而灵活的 node.js Web应用框架, 提供了一系列强大特性帮助你创建各种 Web 应用，和丰富的 HTTP 工具。

使用 Express 可以快速地搭建一个完整功能的网站，它有一套健壮的特性，可用于开发单页、多页和混合Web应用。

网上查询：Connect拓展了Node ,Express拓展了Connect.(中间件)

### 安装

npm install express --save

## socket

### 概念

WebSocket提供了一个受欢迎的技术，以替代我们过去几年一直在用的Ajax技术。这个新的API提供了一个方法，从客户端使用简单的语法有效地推动消息到服务器。让我们看一看HTML5的WebSocket API：它可用于客户端、服务器端。而且有一个优秀的第三方API，名为Socket.IO。

WebSocket API是下一代客户端-服务器的异步通信方法。该通信取代了单个的TCP套接字，使用ws或wss协议，可用于任意的客户端和服务器程序。WebSocket目前由W3C进行标准化。WebSocket已经受到Firefox 4、Chrome 4、Opera 10.70以及Safari 5等浏览器的支持。

WebSocket API最伟大之处在于服务器和客户端可以在给定的时间范围内的任意时刻，相互推送信息。WebSocket并不限于以Ajax(或XHR)方式通信，因为Ajax技术需要客户端发起请求，而WebSocket服务器和客户端可以彼此相互推送信息；XHR受到域的限制，而WebSocket允许跨域通信。

Ajax技术很聪明的一点是没有设计要使用的方式。WebSocket为指定目标创建，用于双向推送消息。

## 模块

Data方法：

// 通过req的data事件监听函数，每当接受到请求体的数据，就累加到post变量中

req.on('data', function(chunk){

post += chunk;

})

url模块：url :协议：//域名||主机名||主机IP：端口/路径

QueryString模块

Http模块

事件event模块

文件fs模块

Steam模块

## 路由



### 路由的必备要素

1.开启服务 2.有效的请求地址 3.路由的映射表（规则）

### 理解

路由选择的核心是路由，顾名思义，路由指的就是我们要针对不同的URL有不同的处理方式

Get 请求小数据，而post请求是大数据，数据的交互。

### 语句的方法

each: 如果想对同一个集合中的所有元素都执行同一个异步操作。

map: 对集合中的每一个元素，执行某个异步操作，得到结果。所有的结果将汇总到最终的callback里。与each的区别是，each只关心操作不管最后的值，而map关心的最后产生的值。

filter: 使用异步操作对集合中的元素进行筛选, 需要注意的是，iterator的callback只有一个参数，只能接收true或false。

reject: reject跟filter正好相反，当测试为true时则抛弃

reduce: 可以让我们给定一个初始值，用它与集合中的每一个元素做运算，最后得到一个值。reduce从左向右来遍历元素，如果想从右向左，可使用reduceRight。