

Node.js 与 MongoDB 课程项目安排

同学们好，首先恭喜大家完成了《Node.js与Mongodb》的课程，简单回顾一下我们的课程体系：

在课程中我们首先学习了Node.js的安装以及简单使用，接着我们完成了一些系统自带的模块的学习，详细了解了HTTP协议以及Node.js的工程问题。在后半部分的课程中我们学习了Node.js和数据库相关的知识，这里我们主攻MongoDB，辅助的了解了Mysql。学习完成这些课程之后同学们就应该具备尝试编写复杂项目的能力了。在课程的最后我们安排了Node.js的性能和单元测试的内容，以及一个业内非常流行的webserver的框架Express的使用，通过这些，大家可以了解到当今工作中的一些趋势。

当然，课程并不能带领你从0开始到立刻精通，课程中我主要想培养大家的“攻城精神”，学会如何面对全新的事物、查阅文档、解决项目中的问题，这些比学会100个API的使用更为重要。除了学习课程外，还需要大家付出一定的心血，保持乐观好奇的精神，去不断地汲取更加深入高级的知识，这样才能在未来步入大师的殿堂。

课程之后我们安排了一个课程项目，在这个项目中大家一定会遇到各种问题，希望大家不是直接要答案，而是主动地去寻找答案，在决定寻求帮助之前，首先想想自己做了哪些努力和尝试，如果发现自己并没有做出任何努力的话，我建议还是先自己尝试探索一下，毕竟这样才能获得成长。

项目介绍

在这节课程之前，我们已经学习了Node.js以及MongoDB的使用，这里我需要大家用自己所学的知识，完成一个简单的站点的服务端开发，你可以根据自己的兴趣搭建一个自己的博客或者一个新闻系统或者一个简易论坛，无论你的选择是什么，我希望能满足下面的需求：

1. 没有捷径，虽然 `Express`，`Ghost`，`Koa` 之类的第三方产品可以让你秒速的完成这个项目，但是我更希望你是自己尝试写出了整个过程，上面列出的开源软件可以作为借鉴，当你卡壳的时候尝试看看他们的源码了解他们如何实现的，但是不要去尝试直接使用。
2. HTTP的实现，在项目中，我们要自己处理HTTP协议，你需要处理Node.js页面的200请求，所有静态资源（至少包括css,js,image）的200请求，如果对自己比较有信心希望你能实现304等其它HTTP状态。
3. 有分页的列表页，你需要完成一个列表页面，(比如博客或者新闻的列表页)，列表页面中必须包含分页功能，可以通过url的参数进行翻页。如果你对自己有信心，可以在列表的页面再加上分类。
4. 有发布页面，你需要实现至少一个发布页面，可以是发布新的博客/新闻，也可以是对某一篇博客/新闻进行评论，这里的提交请使用AJAX的POST的方式，返回的格式为HTTP 200 的 JSON。
5. 提供访问统计，你的博客/新闻页面需要有页面统计，每次访问统计加一，并且在页面中展示出来。
6. 可以修改自己发布的内容。
7. 如果对自己的能力很认可，我希望能实现用户的功能，每个用户都可以发布新的内容且每个用户只能修改自己发布的内容。

项目安排建议

下面给出一个具体的时间安排作为参考，你可以按照这个时间来进行开发，或者自己根据实际情况进行安排，但是如果决定自己安排时间，请在项目开始前就设计好每天的工作，并严格执行。

Day 1 出发前的准备 - 项目准备

这一天你需要准备各种项目所需，安装Node.js，安装MongoDB，初始化项目(npm init)，确保你的Node.js和MongoDB可以正常运行，建立对应的MongoDB用户以及项目数据库。

Day 2 千里之行，始于足下 - 简单Webserver的实现

写代码的第一天，今天我们要完成Webserver的建立，你要实现

- 在某个端口中可以访问你的Node.js项目（比如127.0.0.1:27017/）
- 能识别不同的访问地址，比如说访问127.0.0.1:27017/bloglist/page1，你能获取到['bloglist','page1']这样的数组，方便你后续的处理。

TIPS: 可以尝试使用Url,Path模块来处理用户访问的url

Day 3 规划你的未来 - 路由的设计

设计我们的路由，路由是指每个地址是用来做什么用的，下面给出一个示例：

/	[首页]
/static/**	[静态资源]
/blog	[博客列表]
/blog/page/2	[博客列表第二页]
/blog/article/5808bc603344715766a16631	[具体的某一篇博客]
/message	[留言板]
/message/page/2	[留言板第二页]

Day 4 Let's Do It - 静态资源

这次我们来处理静态资源的显示，比如访问 `/static/blog/index.js` 返回相应的js文件，访问 `/static/blog/abc.jpg` 返回相应的jpg文件。

TIPS: 访问具体的静态资源我们只需要去拿到对应目录的文件并返回给用户就可以了。你可能用到的模块 `File`，你可能需要了解的知识，HTTP头中的MIME（如 `Content-Type:image/jpeg`）

Day 5 Let's Do It - 动态资源

把相应的请求交给相应的模块来处理，比如：

- 访问 `/blog` 或者 `/blog/page/2` 时候，交给项目目录中的 `/bloglist.js` 来处理
- 访问 `/blog/article/5808bc603344715766a16631` 的时候，交给项目目录中的 `/article.js` 来处理

Day 6 我们的数据朋友 - 数据的发布I

今天我们来处理数据发布的内容，你需要建立一个发布的页面比如， `/blog/publish`，这个页面中你需要提供标题、内容以及发布按钮，点击发布按钮拿到标题和内容的内容通过AJAX发布提交到一个新的地址，比如 `/blog/publishdo`。

TIPS: 数据的发布按钮，需要你通过之前学的JS知识，拿到标题和内容，然后通过AJAX提交，如果这一块内容没有很好地掌握的话，可以咨询其他同学，或者前端课程的老师

Day 7 我们的数据朋友 - 数据发布II

今天我们来处理数据的保持，在 `/blog/publishdo` 页面，你需要接受前段发过来数据，并保存到数据库。

TIPS: 接受POST请求的数据我们需要用 `on('data')` 以及 `on('end')`，数据保持我们可以使用Mongoose链接mongoDB数据库进行保存。

Day 8 我们的数据朋友 - 数据的展示

今天我们需要从数据库拿到我们的数据并显示出来，比如：

- 访问 `blog/page/1` 的时候，我们要拿到所有博客的列表并展示第一页的数据
- 访问 `blog/article/5808bc603344715766a16631` 的时候，我们需要拿到博客ID为 `5808bc603344715766a16631` 的文章并显示出来

Day 9 & Day 10 完美之路 - 整理美化项目

前几天我们已经把全部的功能都开发完毕了，今天我们来美化我们的页面，让她看起来跟动人。当然了如果你能力不错，也可以尝试实现用户的功能哦。

好了，如果这么几天大家按照要求完成了一个项目的开发的话，那么恭喜你，你的Node.js又上了一层楼，不妨给自己一个奖励吧！最后祝大家学习愉快。