

## 06 初始化后端项目

更新时间：2019-07-26 09:41:39



“ 更多一手资源请+V：Andyqc1  
知识是一种快乐，而好奇则是知识的萌芽。 qq：3118617541 ”

——培根

这节课我们来搭建一下后端项目，后端项目我们主要使用的是Express。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。

### 安装express-generator

正如前面介绍的，Express 是一个基于 Node.js 后端的 MVC 框架，Express Generator是一个 Express 项目脚手架生成工具，你可以把它理解成 vue cli 的作用，我们需要首先安装 Express Generator：

```
npm install express-generator -g
```

### 初始化后端项目

安装完成之后就可以执行命令，初始化后端项目：

```
express wecircleServer --no-view
```

`--no-view` 即不需要模版引擎这些工具。这里我们采用的 `--no-view` 配置，主要是我们的是一个前后端分离项目，我们的后端项目只负责提供对应的API接口，所有的页面渲染由前端的vue完成。

当然，在初始化项目时，有一些其它选项可以配置，可以参考下面这些文档：

```
$ express -h

Usage: express [options] [dir]

Options:

-h, --help      输出使用方法
--version      输出版本号
-e, --ejs       添加对 ejs 模板引擎的支持
--hbs          添加对 handlebars 模板引擎的支持
--pug          添加对 pug 模板引擎的支持
-H, --hogan     添加对 hogan.js 模板引擎的支持
--no-view      创建不带视图引擎的项目
-v, --view <engine> 添加对视图引擎（view）<engine> 的支持 (ejs|hbs|hjs|jade|pug|twig|vash)
                 （默认是 jade 模板引擎）
-c, --css <engine> 添加样式表引擎 <engine> 的支持 (less|stylus|compass|sass) （默认是普通的 css 文件）
--git          添加 .gitignore
-f, --force     强制在非空目录下创建
```

之后我们可以得到如下的目录结构：

```
|— bin
|   |— www          //项目启动脚本
|— public          //存放静态资源，后续我们的前端dist打包的项目会存放在此
|   |— images
|   |— javascripts
|   |— stylesheets
|   |   |— style.css
|— routes          //每个模块的路由文件
|   |— index.js
|   |— users.js
|— app.js          //项目的主入口文件，负责路由的配置，数据库初始化
|— package.json
```

更多一手资源请+V：Andyqc1  
aa：3118617541

这样看起来也很清爽有木有，其中 **routes** 这个文件夹主要用来存放一些路由文件，并且到时会根据模块来写，即用户模块，消息模块，私信聊天模块等等。

其中路由也是负责 **API** 接口的主要职责，包括 **post** 和 **get** 这些方法。

**public** 这个文件夹存放的都是静态资源文件，我们后面的前端项目打包完成后会放在这个文件里面。

### 启动后端项目

后端项目的脚手架完成之后，我们就可以先尝试去启动它，使用下面的命令：

```
npm install
npm run start
```

程序启动之后默认监听**3000**端口，我们可以试一下，浏览器输入**http://localhost:3000**会看到内容如下：

# Express

Welcome to Express

其中的静态资源文件已经可以通过 `localhost:3000/javascripts/xx.js` 访问到了，证明我们的服务已经启动成功。

当然这个只是初期的脚手架，和前端一样，我们后面会在此基础上添砖加瓦！

小结

本章节主要讲解了 `Express`，以及使用 `Express Generator` 生成我们的后端项目，并介绍了后端的目录结构，为后续的后端开发做准备。

}