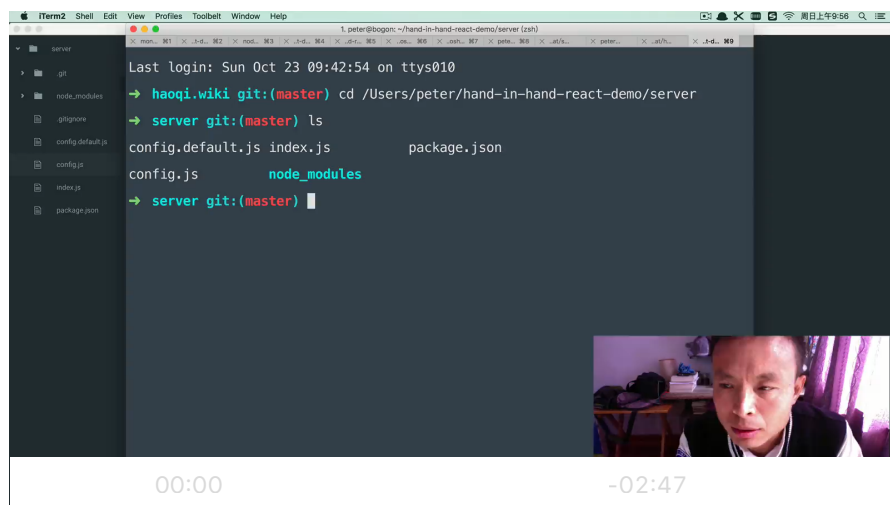


[首页](#) [目录](#)[退出](#)

第一章：搭建后端开发环境

新建一个简单的
Express 应用

▶ Express 应用开发必备
神器 Nodemon

Express 应用开发必备神器 Nodemon

上节课程中你也可能注意到了，当我们修改代码之后，需要重新启动 Express 应用，所做的修改才能生效。若之后的每次代码修改都要重复这样的操作，势必会影响开发效率，所以本节课程将介绍一位新的小伙伴 **Nodemon**，它会监测项目中的所有文件，一旦发现文件有改动，Nodemon 会自动重启 Express 应用，解决你的烦恼，成为你项目开发中的好帮手。

安装 nodemon 包

全局安装 **nodemon** 包，这样新创建的 Node.js 应

用都能使用 Nodemon 运行起来了。

```
npm install -g nodemon
```

安装完成之后，Nodemon 就可以启动你的 Express 应用了，先关闭当前正在执行的应用程序，然后再执行命令：

```
nodemon index.js
```

通过 Nodemon 启动应用之后，不管你是修改了代码，还是安装了新的 npm 包，Nodemon 都会重新启动应用。

Nodemon 配置文件

Nodemon 默认会监听当前目录下（也就是执行 nodemon 命令所在的目录）的所有文件，不过有些情况下，虽然项目文件发生了改动，但是不需要 Nodemon 重启应用，那如何让文件不被 Nodemon 监听呢？不需要监听的文件，可以通过设置 Nodemon 的配置文件排除掉，新建文件 `server/nodemon.json`，添加代码：

```
{
  "ignore": [
    "config.default.js"
  ]
}
```

Nodemon 配置文件是 JSON 文件，通过设置 `ignore` 属性值，一个由文件名组成的字符串数组，指定不需要监听的文件。

手动重启 Nodemon

有时候可能 Nodemon 还在运行的时候，你需要手动重启它，在这种情况下你不需要关闭正在运行的 Nodemon 进程然后再重启 Nodemon，只要在 Nodemon 命令运行的终端 窗口中输入 `rs` 两个字符，然后再按下回车键，就能重启 Nodemon 了。

顺便说一下，Nodemon 不是只专门服务于 Node.js 应用的，它还可以用于其它语言开发的应用，具体可以参考官方文档 [运行非 node 脚本](#) 一节的内容。

欢迎添加 Peter 的微信：happypeter1983

冀ICP备15007992号-3