HyBrid App开发（三）

目录

[HyBrid App开发（三） 1](#_Toc467340734)

[一、综合案例之App项目 2](#_Toc467340735)

[1、实现动态的添加功能 2](#_Toc467340736)

[2、实现文件上传功能 4](#_Toc467340737)

[3、数据的显示之分页程序 5](#_Toc467340738)

[4、MiddleWare中间件 5](#_Toc467340739)

[二、整合App与Laravel框架（App接口） 7](#_Toc467340740)

[1、App接口原理图 7](#_Toc467340741)

[2、在服务器端定义api接口 7](#_Toc467340742)

[3、在HyBrid App手机端向服务器端发送请求 8](#_Toc467340743)

[三、Vuejs框架 9](#_Toc467340744)

[1、下载Vuejs 9](#_Toc467340745)

[2、放置vuejs到script文件夹中 9](#_Toc467340746)

[3、使用Vuejs遍历数据到视图 9](#_Toc467340747)

[4、为所有音乐添加音乐链接，打开音乐播放器 10](#_Toc467340748)

[5、使用api.openWin传递音乐地址 10](#_Toc467340749)

[6、在win\_music.html页面接收音乐地址 10](#_Toc467340750)

[四、使用APICloud中的模块让音乐响起来 11](#_Toc467340751)

[1、进入开发者控制台 11](#_Toc467340752)

[2、进入模块库，选择添加模块 11](#_Toc467340753)

[3、生成自定义loader 11](#_Toc467340754)

[4、查看netAudio手册，让音乐响起来 12](#_Toc467340755)

[5、让滑块随着音乐滑动 12](#_Toc467340756)

[6、让光盘旋转起来 13](#_Toc467340757)

[7、定义播放按钮与停止按钮 13](#_Toc467340758)

[五、实现App下拉刷新功能 14](#_Toc467340759)

[1、在frame\_news.html页面定义下拉刷新 14](#_Toc467340760)

[2、使用下拉刷新获取最新数据 14](#_Toc467340761)

[3、定义fnLoadList(isPull) 15](#_Toc467340762)

[4、skip累加 15](#_Toc467340763)

[5、把最新数据追加到列表头部 16](#_Toc467340764)

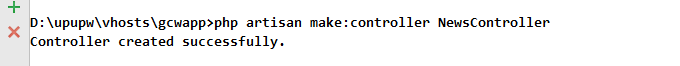
# 一、综合案例之App项目

## 1、实现动态的添加功能

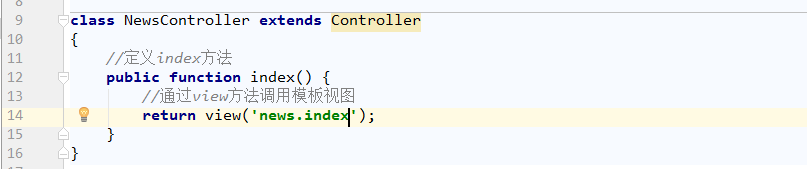
① 更改后台左侧链接导航



② 使用php artisan生成news控制器



复制NewsController.php到app\Http\Controllers\Admin文件夹，编写代码：



③ 在resources\views文件夹中创建一个news文件夹，然后创建index.blade.php视图文件

④ 定义路由



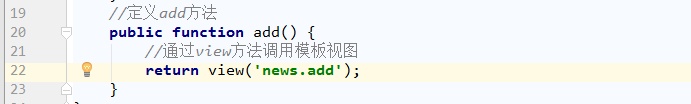
⑤ 更改动态页面中的新增动态导航连接



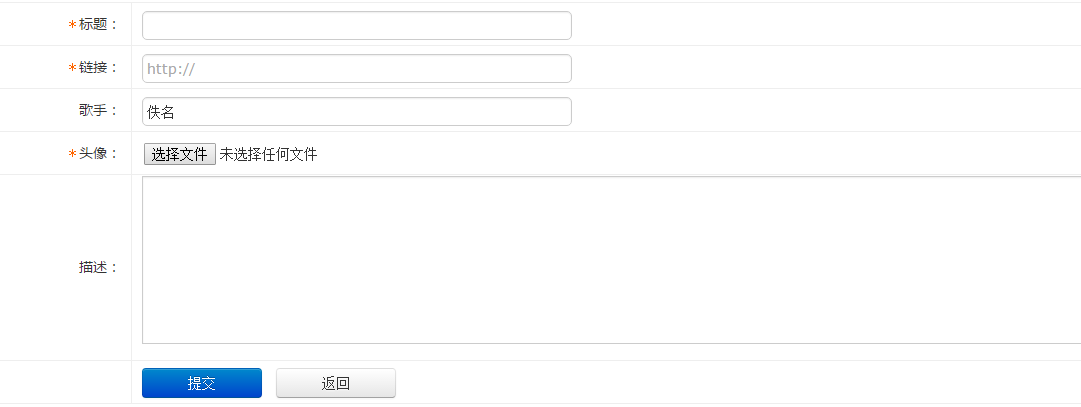
⑥ 在routes.php路由中，设置请求连接



⑦ 在NewsController控制器中定义add方法



⑧ 定义add.blade.php添加模板

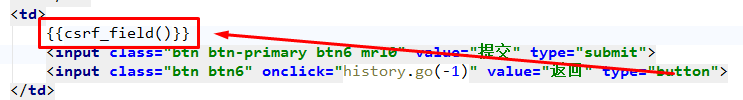


⑨ 定义action提交页面，并在News控制器端实现添加功能

定义action提交页面：



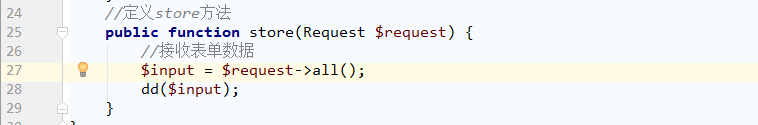
定义csrf，生成随机的token



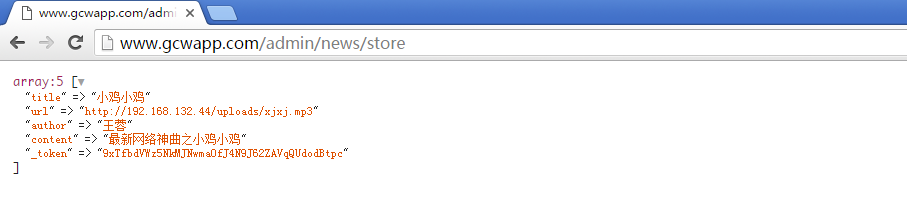
定义routes.php路由：



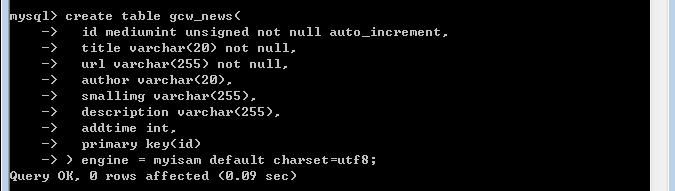
在News控制器中定义store方法：



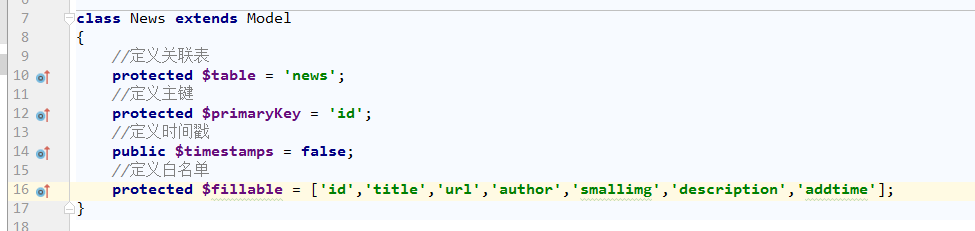
运行结果：



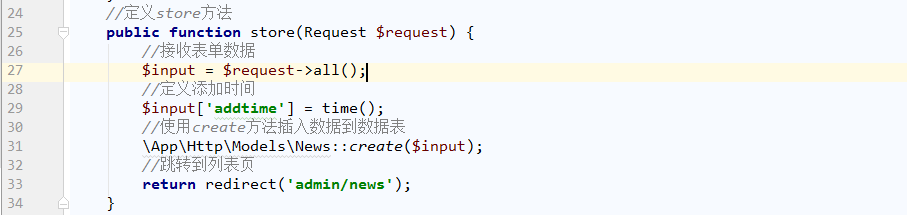
设计数据表：



定义News模型：

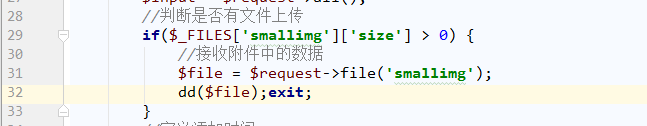


使用create方法实现添加操作：

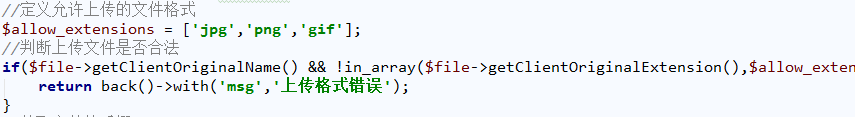


## 2、实现文件上传功能

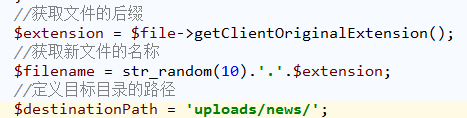
① 判断是否有文件上传



② 设置允许上传的文件格式



③ 设置相关参数（文件的后缀、新文件名称、目标路径）



④ 使用move方法移动文件到目标路径

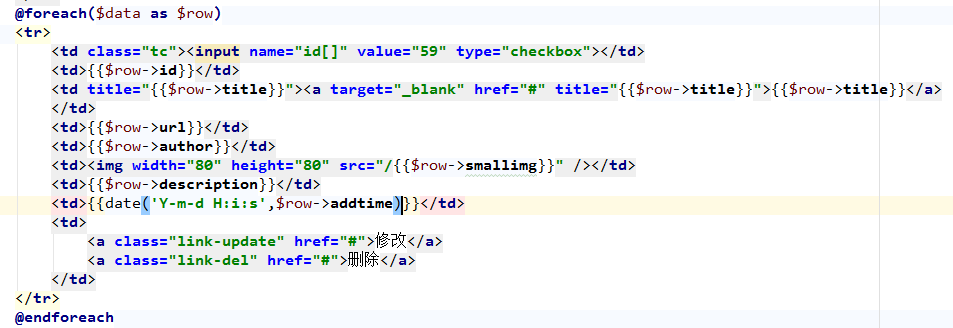


## 3、数据的显示之分页程序

① 在index方法中，获取所有数据，如下图所示：



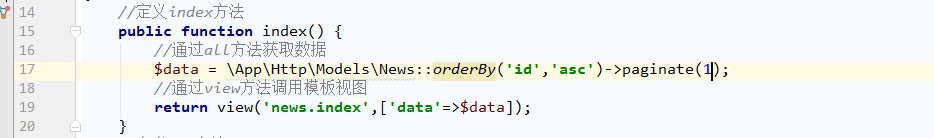
② 在news/index.blade.php模板文件中，遍历data数据



③ 实现数据分页程序

第一步：使用paginate或simplePaginate进行数据分页

第二步：在模板中通过$数据对象->render()显示分页连接



在模板中显示分页连接：



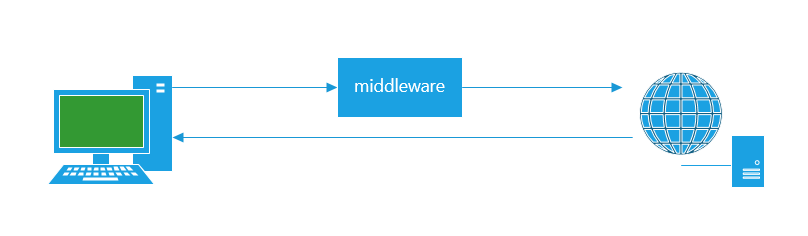
运行结果：



## 4、MiddleWare中间件

① 什么是中间件

HTTP 中间件为过滤访问你的应用的 HTTP 请求提供了一个方便的机制。例如，Laravel 默认包含了一个验证用户的中间件。如果没有经过身份验证，中间件将会将用户重定向至登录页面。然而，如果用户经过了验证，中间件将会允许请求继续在应用中执行下去。



② 创建中间件，基本语法：

php artisan make:middleware AdminLogin

示例代码：



运行结果：

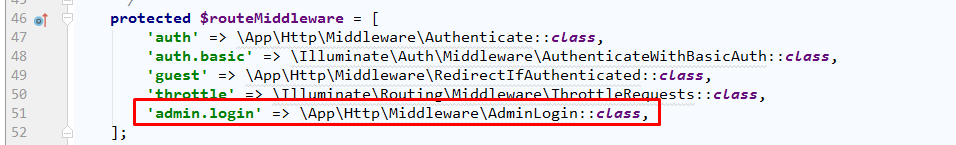
在app\Http\MiddleWare文件夹中会自动生成AdminLogin文件，如下图所示：



编写自定义代码，验证用户是否登录：

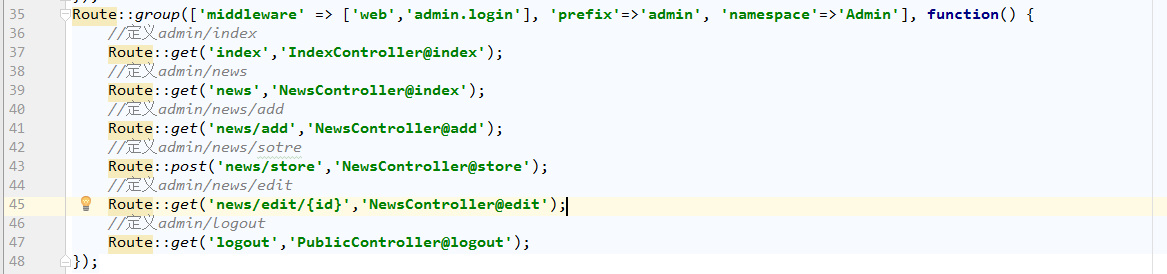


③ 启用AdminLogin中间件功能，在Kernel.php中启动AdminLogin中间件



④ 绑定需要验证session的请求

在routes.php中，定义需要验证的请求，并设置中间件：



⑤ 实现后台退出功能

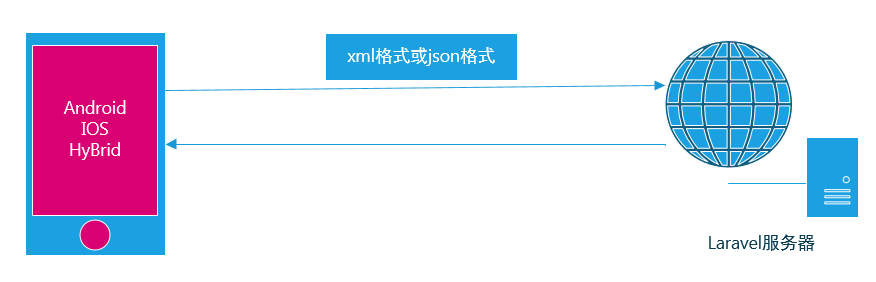


在Public控制器中定义logout，把session(‘admin’)置空，如下图所示：



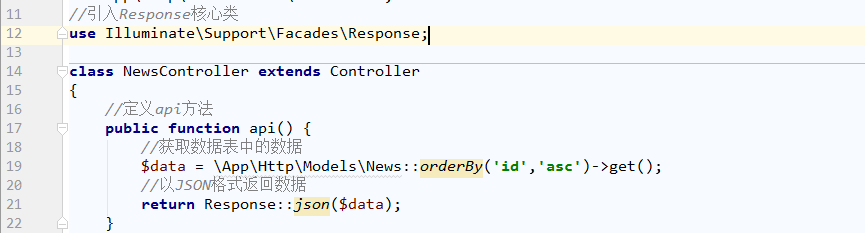
# 二、整合App与Laravel框架（App接口）

## 1、App接口原理图

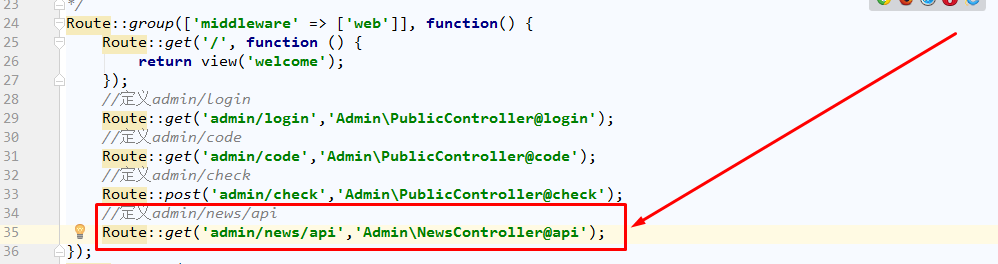


## 2、在服务器端定义api接口

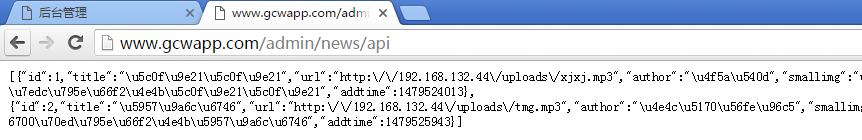
在NewsController控制器中，定义一个api方法：



在routes.php路由中，设置路由，如下图所示：



运行结果：

  
特别注意：如果我们没有真实域名，而采用虚拟域名，手机端是没有办法访问到的，必须使用当前计算机的IP地址作为域名地址。

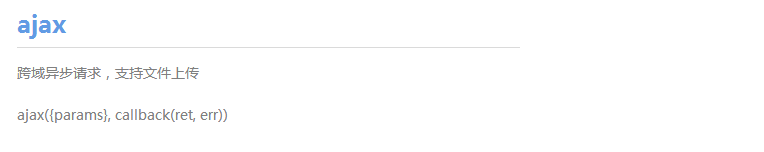


以上IP地址就是你自己计算机的IP地址。

## 3、在HyBrid App手机端向服务器端发送请求



基本语法：



示例代码：



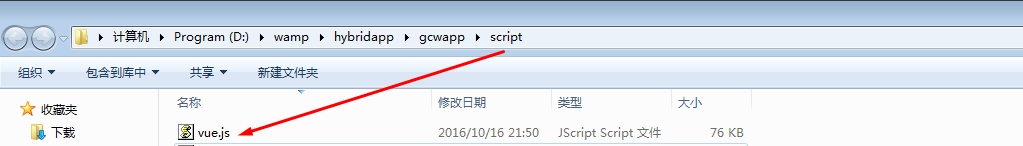
# 三、Vuejs框架

## 1、下载Vuejs

下载地址：<http://cn.vuejs.org/>

Vue.js（读音 /vjuː/, 类似于 view）是一个构建数据驱动的 web 界面的渐进式框架。Vue.js 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。

## 2、放置vuejs到script文件夹中



## 3、使用Vuejs遍历数据到视图

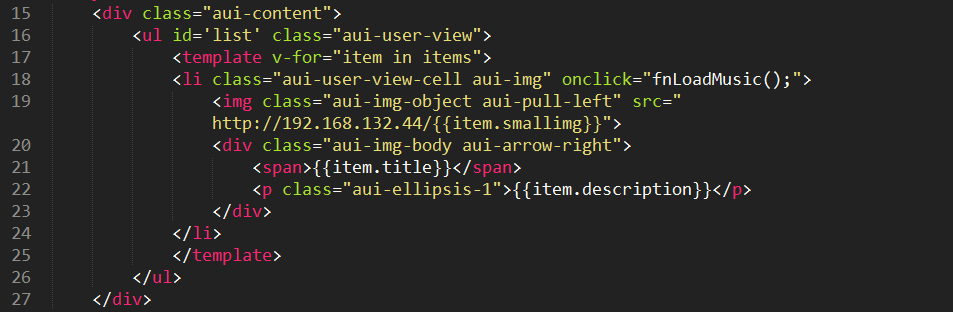
① 使用script标签引入vuejs



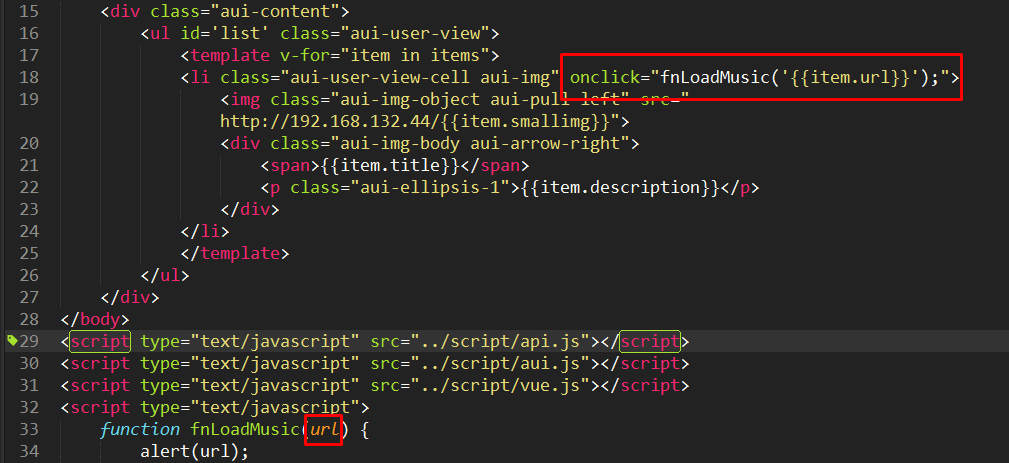
② 使用new Vue实现创建Vue对象



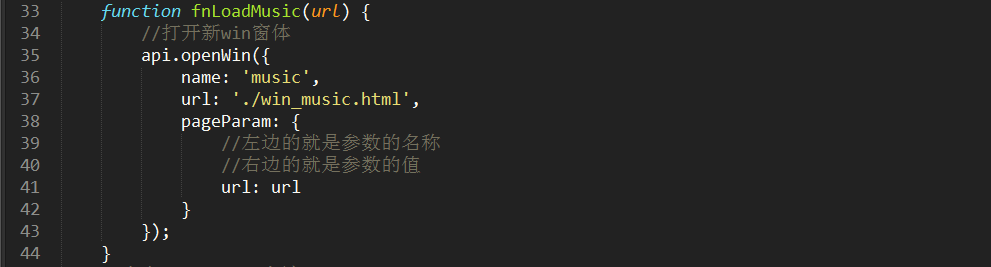
③ 使用模板视图遍历显示数据



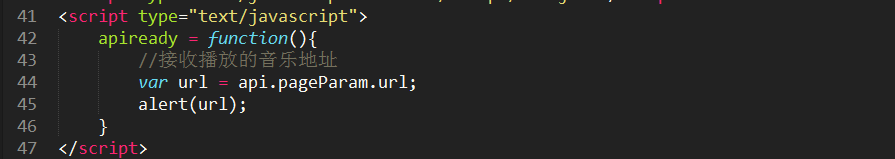
## 4、为所有音乐添加音乐链接，打开音乐播放器



## 5、使用api.openWin传递音乐地址

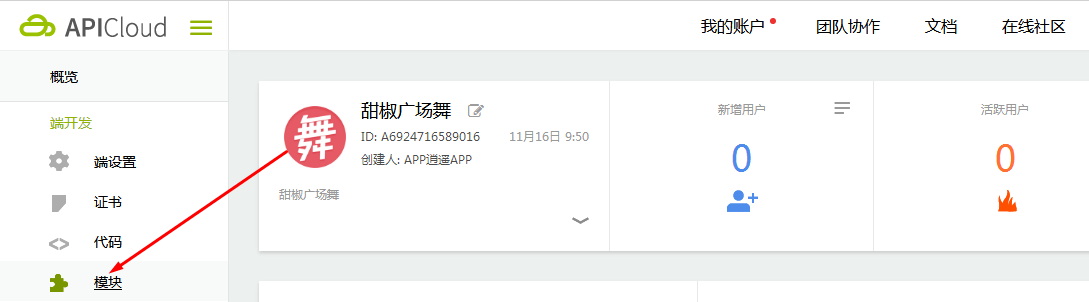


## 6、在win\_music.html页面接收音乐地址



# 四、使用APICloud中的模块让音乐响起来

## 1、进入开发者控制台



## 2、进入模块库，选择添加模块

① 添加netAudio音乐播放器模块



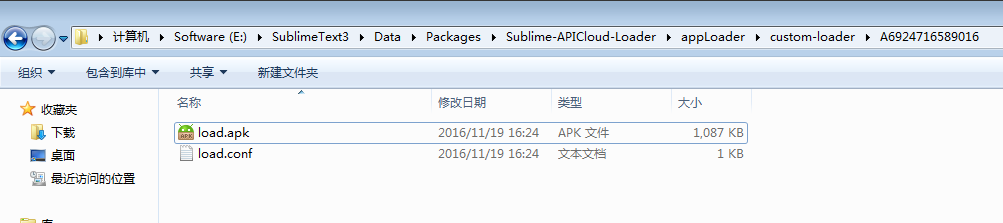
② 添加在线分享模块



## 3、生成自定义loader



下载后覆盖本地的load.apk，然后更改load.conf文件，如下图所示：



## 4、查看netAudio手册，让音乐响起来

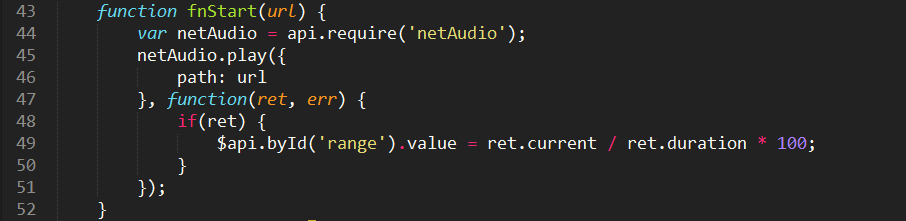


## 5、让滑块随着音乐滑动

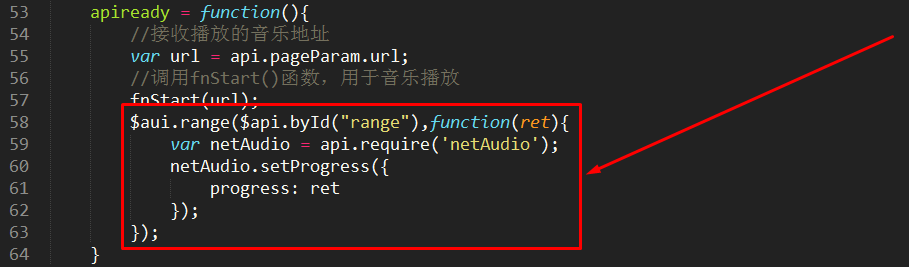
① 引入滑块JS



② 设置滑块的值

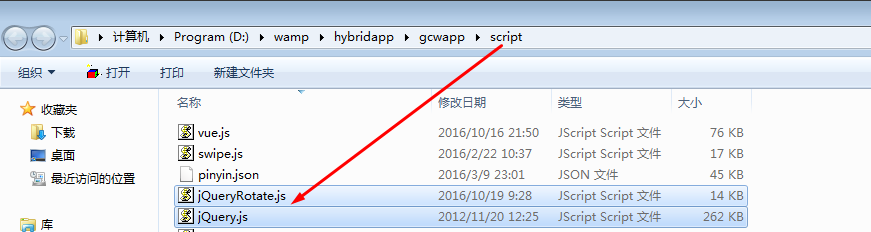


③ 拖动滑动，播放音乐的相应位置



## 6、让光盘旋转起来

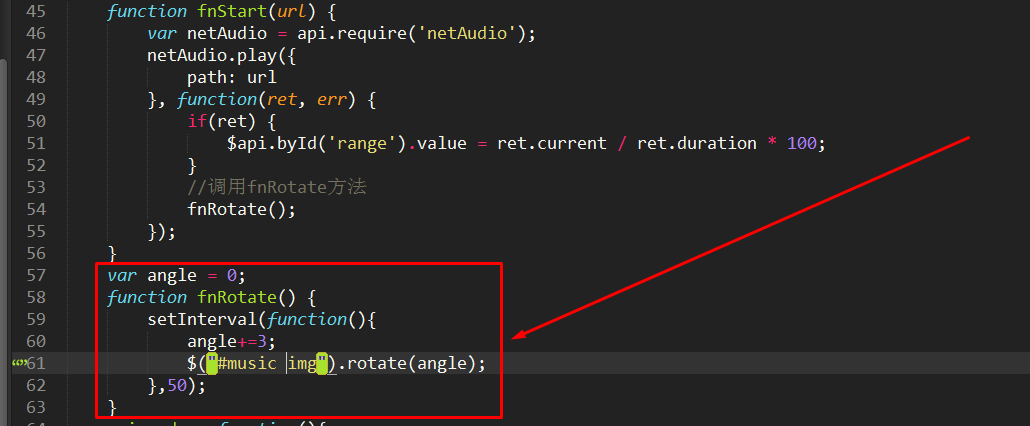
① 引入jquery.js与jqueryrotate.js



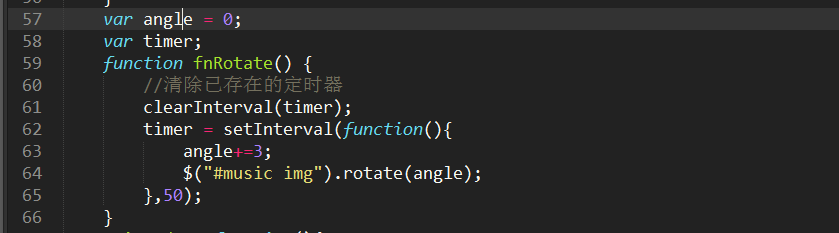
② 使用script标签导入到win\_music.html页面



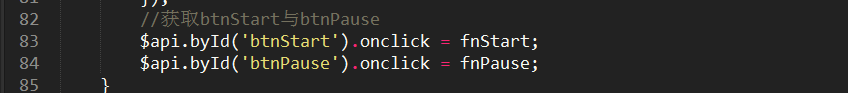
③ 让光盘旋转起来



④ 清除定时器，避免定时器叠加



## 7、定义播放按钮与停止按钮

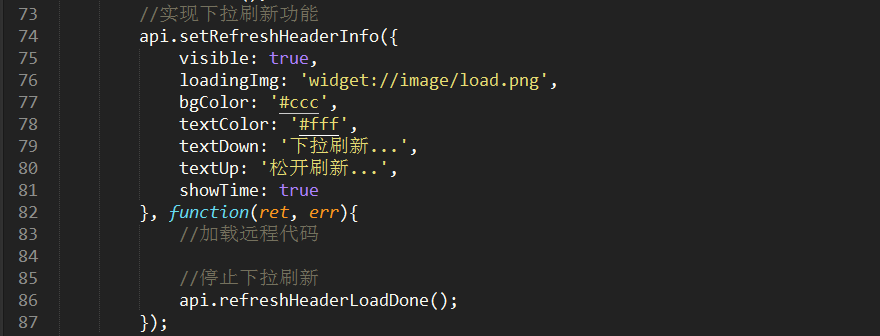


# 五、实现App下拉刷新功能

## 1、在frame\_news.html页面定义下拉刷新



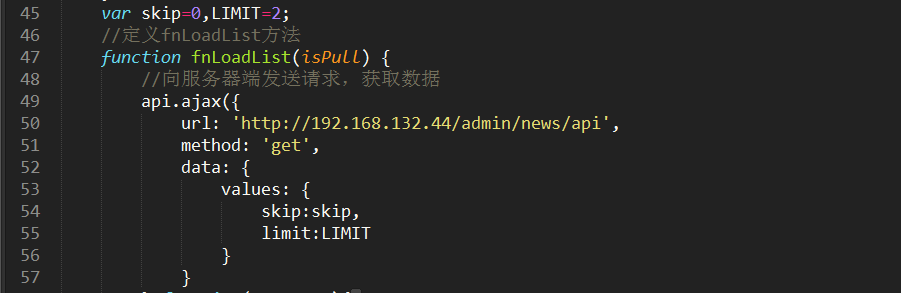
特别注意：使用完下拉刷新后，一定要记得停止下拉刷新，否则会一致处于下拉刷新状态！



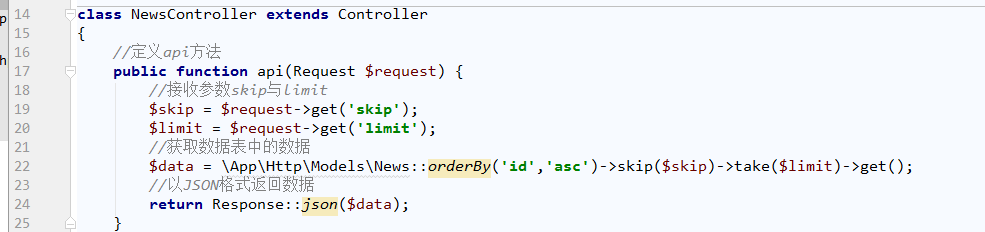
## 2、使用下拉刷新获取最新数据



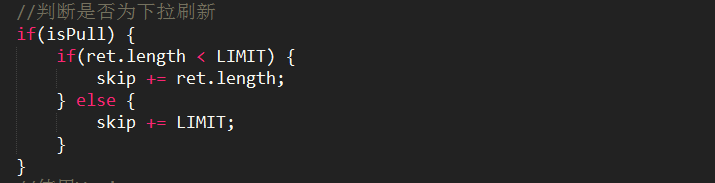
## 3、定义fnLoadList(isPull)



在Laravel框架的api方法中，接收skip与limit



## 4、skip累加



## 5、把最新数据追加到列表头部

