**框架说明文档**

1、框架搭建...........................................................

1.1、控制器创建...................................................

1.2、模型创建.....................................................

2、框架使用...........................................................

2.1、控制器属性介绍...............................................

2.2、控制器集成方法...............................................

2.3、模型增删改查.................................................

2.4、模型标准写法.................................................

3、后台JS说明

框架说明：新框架相比之前的框架有几点比较创新的写法。

1.Common/Service下的ControllerService和ModelService作为所有控制器和模型的基类，每一个控制器和模型都要继承自他们，控制器基类内封装了一些常用方法、权限控制及用户登录信息存储取出操作，减少写程序过程中的代码逻辑。

模型基类内封装的是对数据的增删改查和统计，但大部分数据不只是查的一个表，

所以具体的方法还是要写在对应的Model内，但是查询的时候用封装的方法会方便一些。

2.Vendor下的Txunda目录

目录下有Param.Txunda.php等几个类文件，用的时候只要引入就可以使用，这几个类文件只有一个对外开放的方法供开发者调用，已经全部封装在ControllerService内，直接使用$this即可调用

这些类文件封装的是一些常用的操作，比如参数检查，文件上传，发送手机、邮箱验证码，消息推送等

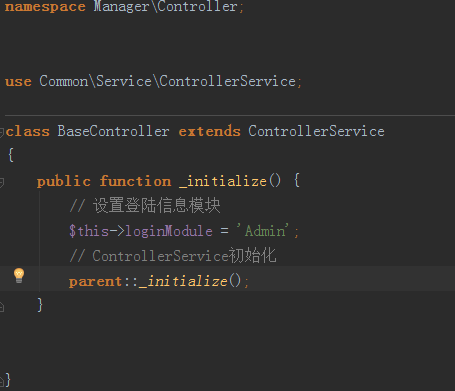
3.对后台的模板的修改，去掉了JS渲染模板步骤，开发起来更方便

4.常用JS的封装，比如模板赋值，删除数据弹窗，Ajax请求，文件上传等操作

1.1、控制器创建

每个控制器都要继承自ControllerService，为了减少操作，则：

每个模块下都应有一个BaseController，如图

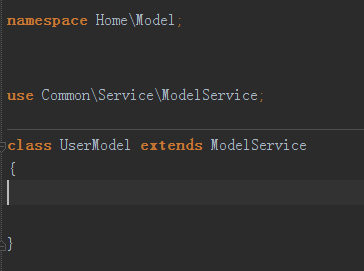


$this->loginModule这个属性的赋值用于多个模块使用同一个登录信息，如果只有单一模块则不用写，后面会说到这个

每个控制器都要继承自BaseController，这个就不说了

1.2、模型创建

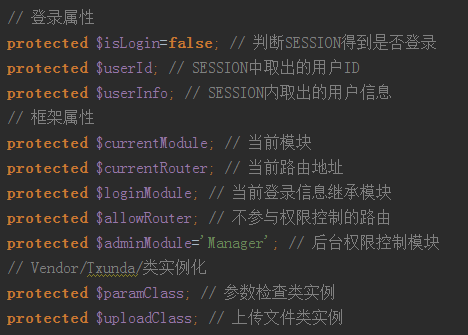
每个模型都要继承自ModelService，如图：



如果是多个模块公用的Model，则放在Common/Model下

2.1、控制器属性介绍

ControllerService有几个属性，不可被覆盖，开发中要注意

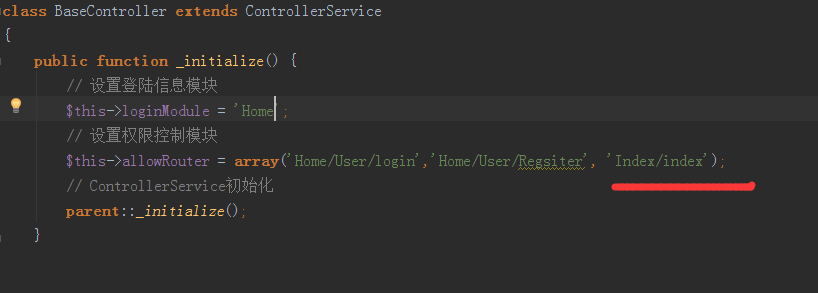


路由地址拼接格式如下：Manager/User/login

allowRouter属性是不参与权限控制的地址，比如登录页等，赋值方式如下

$this->allowRouter = array(‘Home/User/login’, ‘Home/User/Regsiter’);

要在初始化方法内赋值，并写在parent::\_initializt()上方，如下：



如果不写Home，也就是不写模块名称，则默认填充当前模块的名称

这是为了考虑多模块共用一个登陆信息，所以路由地址要精确到

模块名/控制器名/方法名

模块名：首字母大写 控制器名：首字母大写 方法名：和真实方法名等同

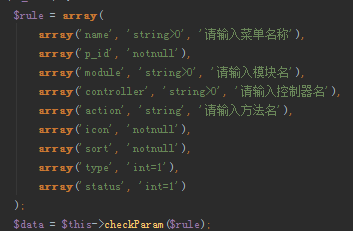
adminModule属性是后台模块名，因为只有后台模块会有菜单权限控制，默认是Manager，如果要改的话可以直接在ControllerService内改，也可以在BaseController的初始化方法内重新赋值

LoginModule属性是判断登录信息的模块，默认为当前模块的名称，如果多个模块公用一个，则需要自己在BaseController内复制

2.2 控制器集成方法

1. 参数判空checkParam($fields)

参数传参方式



或者



前者返回的是一个数组 后者返回的是一个值

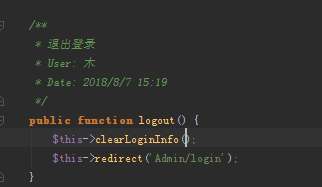
1. 设置登录信息setLoginInfo



主键为id并一定要在数组内，不然判断不出来登录状态

如果传null 则是退出登录或者使用

clearLoginInfo

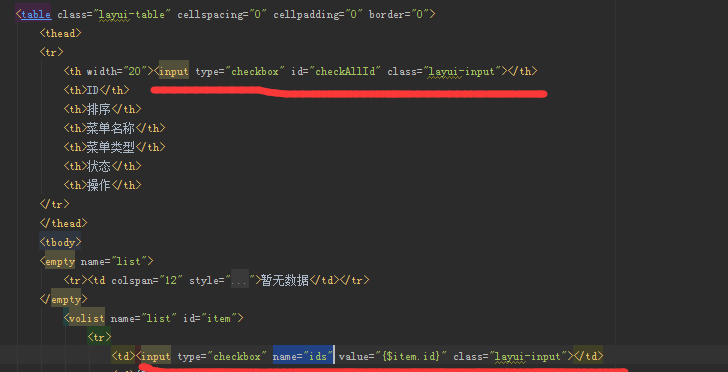


3、后台js说明

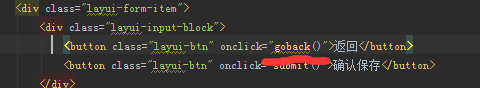
1. 列表页全选/反选

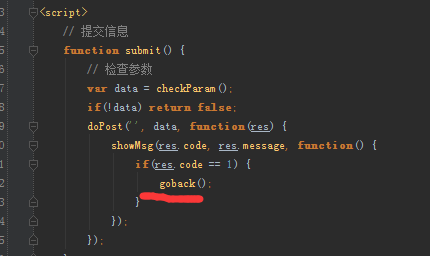
Jquery监听 id=”checkAllId”

name=”ids”



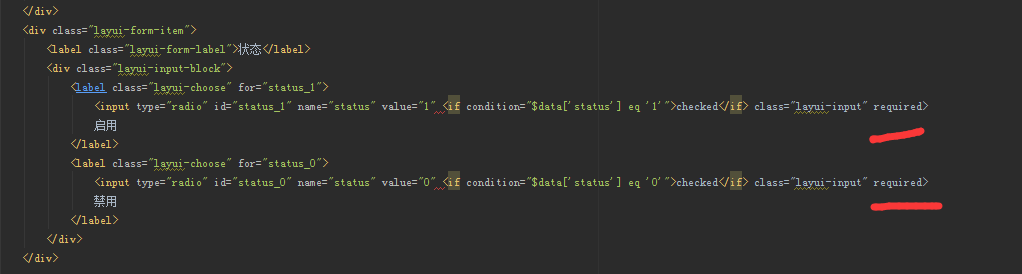
1. goback 方法 返回上级页面并刷新，用于编辑/添加之后 或者 返回按钮

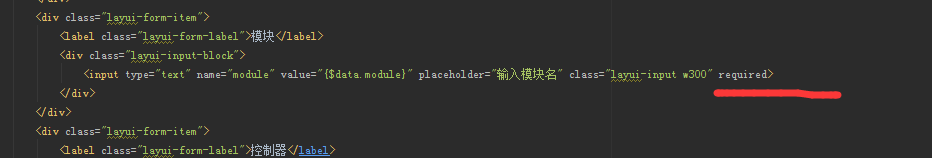


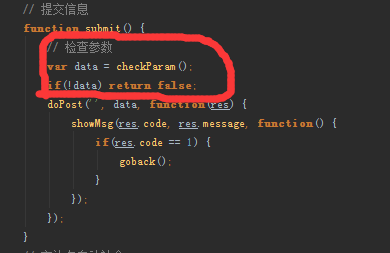


1. 参数判空

需要判空的输入框添加required参数，例：





使用：

所有的参数都要有 layui-input 这个class

如果参数错误会有提示

1. showMsg方法

参数1：icon 1-成功样式 0-失败样式

参数2：提示文字

参数3：提示完成后执行操作 注:1.5秒之后

上面的图片就是调用方法

1. doPost方法

参数1： url

参数2：data

参数3：回调方法

详情见上图

1. 字段修改



三个属性写对

第一个属性是固定的 第二个是请求地址 第三个是字段名

1. confirmAllAction 批量操作按钮绑定方法

参数1：请求地址

参数2：操作名称

会弹出个确认窗口，点击确认之后才会执行操作



选中的id会以ids参数名传一个数组过去

用法如下



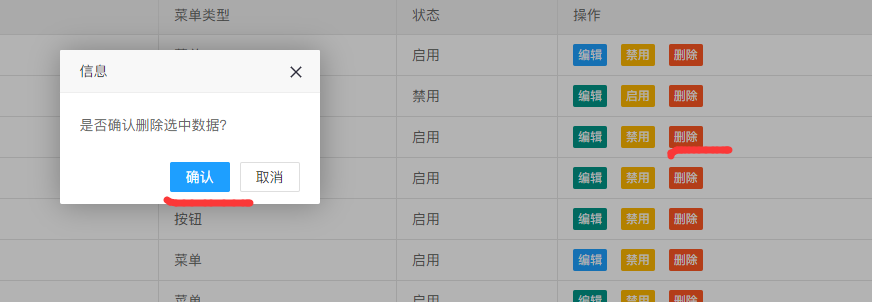
1. confirmOneAction

参数1：请求地址

参数2：操作名称

参数3：id





1. 发送短信接口
2. API – BaseController 方法

1.setAbsoluteUrl 用于后台编辑器上传图片补全路径使用