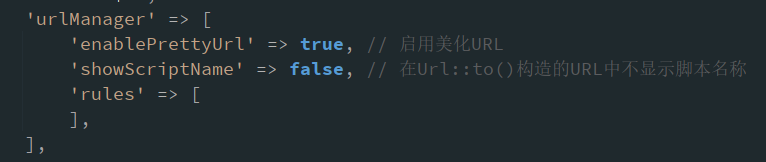
# 美化路由

## 启用Yii自带的美化路由

在配置文件中设置 urlManager 组件，如下：



原来的路由：

<http://study_yii.com/index.php?r=user/user/register>&id=100

美化后：

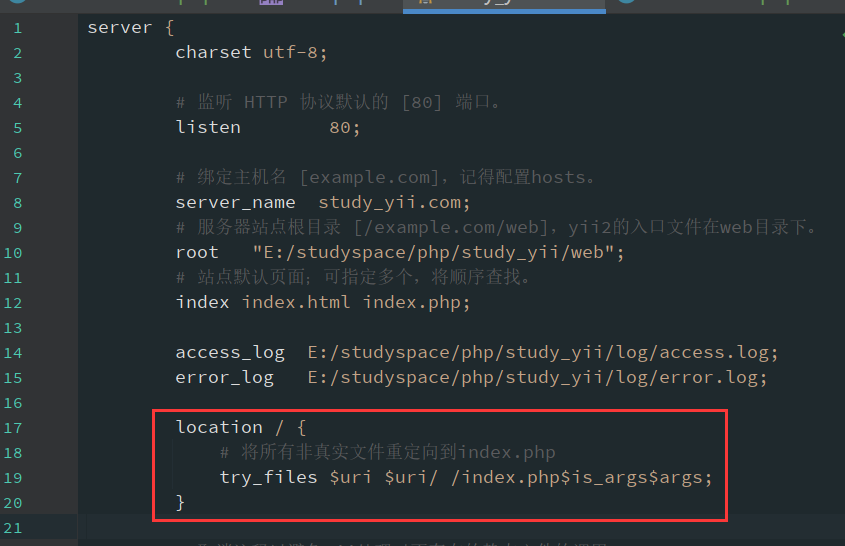
<http://study_yii.com/index.php/user/user/register>?id=100

注意：

* 此配置只会美化路由，不会美化参数。
* 一般不搞seo的话，没必要美化参数，因为每个请求地址都需要单独配置，非常麻烦。

## URL去掉index.php

需要在nginx的conf/vhosts/网站对应.conf文件中配置



去掉后URL：

<http://study_yii.com/user/user/register>?id=100

# CSRF验证

Yii默认开启CSRF验证，如果POST请求没有提交对应的CSRF token。会报400错误。

## 表单提交csrf token

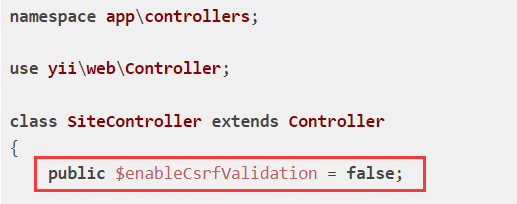
<input name="\_csrf" type="hidden" id="\_csrf" value="<?= Yii::$app->request->csrfToken ?>">

## 关闭CSRF验证

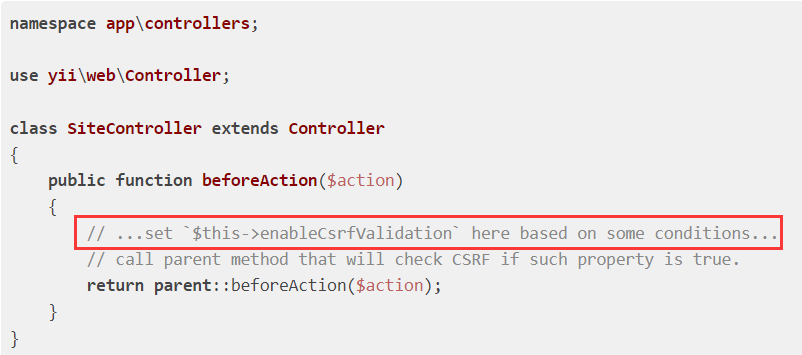
不推荐直接在主配置文件中的request组件关掉CSRF验证，这会导致整个应用都没有CSRF校验，不安全。

一般在控制器中或者某个方法中按需关闭即可。

控制器关闭：

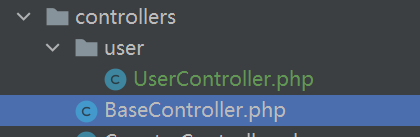


要对每个自定义方法禁用 CSRF 验证，您可以使用：



# 基础代码架构

## 控制器

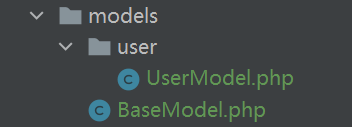


controllers目录下增加BaseController，其它控制器继承此类。

BaseController.php继承yii\web\Controller

按模块增加对应控制器目录，如user模块增加user目录。

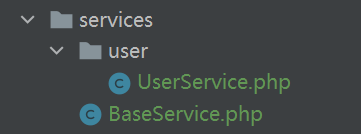
## 模型



结构同控制器

BaseModel.php继承 \yii\db\ActiveRecord（ActiveRecord继承自yii\base\Model）

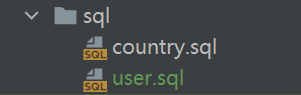
## 新增业务层



结构同控制器

BaseService.php无继承

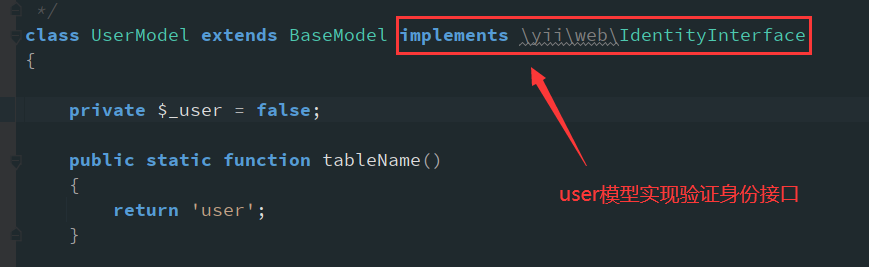
## SQL目录

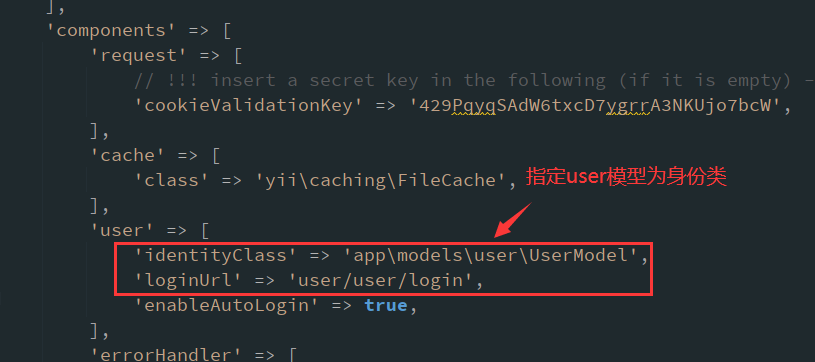


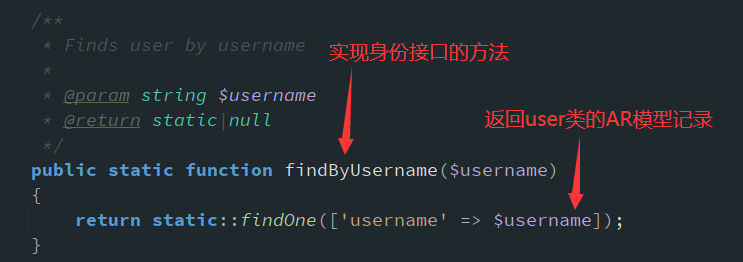
按数据表存放表的一切操作SQL。

# 配置user组件

## 配置







因为user组件配置的身份类是userModel，所以这里返回的用户信息就是userModel的AR模型。

## 使用







