1、基础安装：composer create-project --prefer-dist yiisoft/yii2-app-basic basic

2、安装assets：conposer.json中写入

"replace": {

"bower-asset/jquery": ">=1.11.0",

"bower-asset/inputmask": ">=3.2.0",

"bower-asset/punycode": ">=1.3.0",

"bower-asset/yii2-pjax": ">=2.0.0"

},

3、域名配置：wampserver --- > apache

<VirtualHost \*:80>

DocumentRoot "E:\wamp64\www\yii\_demo\web "

ServerName yii.msshuol.cn

<Directory "E:\wamp64\www\yii\_demo\web ">

Options FollowSymLinks ExecCGI

AllowOverride All

Order allow,deny

Allow from all

Require all granted

</Directory>

</VirtualHost>

127.0.0.1 yii.msshuol.cn

4、默认控制器：'defaultRoute' => 'index',//默认控制器

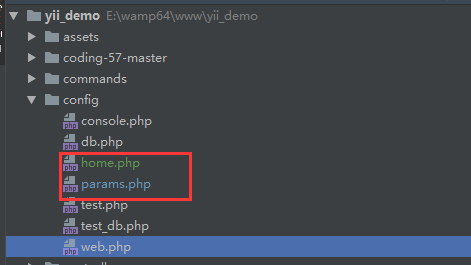
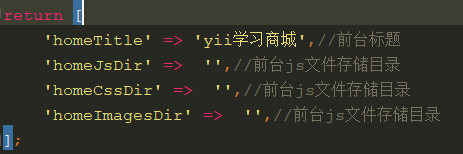
5、layout文件使用：

a）在控制器中加上以下代码 ---》public $layout = 'layout';//在类中定义一个变量，名为$layout

b）view中layouts建立对应变量名的文件，提取出公共部分添加 ----》<?php echo $content; ?>

c）控制器中加入 ----》$this->layout = 'home';

6、配置文件添加 ：在config下建立文件，返回配置数组，后params文件中的数据合并

调用方法：return \Yii::$app->params[$param]; -----》$param：数组中的键，多维使用点链接

7、css，js，image调用：<?php echo \Yii::$app->params['homeImagesDir']?>icon-cart.png

8、路由

a）默认路由：<http://yii.msshuol.cn/index.php?r=cart/index>

b）

9、去掉index.php：在入口文件写入.htaccess

RewriteEngine on

RewriteCond %{REQUEST\_FILENAME} !-d

RewriteCond %{REQUEST\_FILENAME} !-f

RewriteRule . index.php

'urlManager' => [

'enablePrettyUrl' =>true,

'showScriptName' => false,//隐藏index.php http://yii.msshuol.cn/cart/index

'rules' => [

],

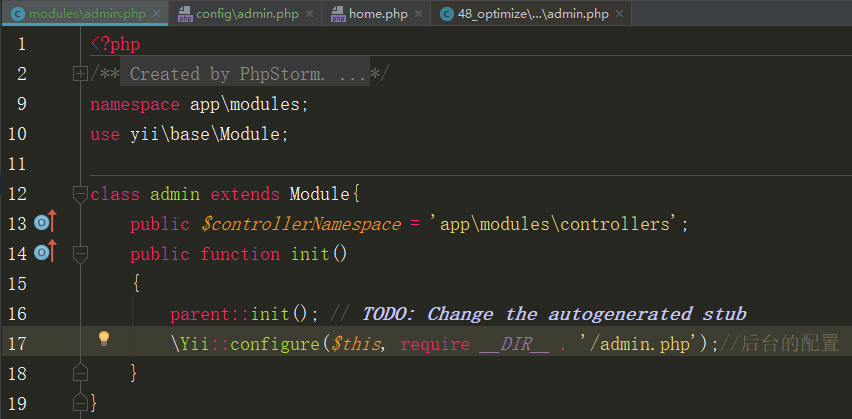
],

10、后台文件建立

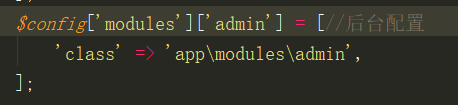
a）建立moduls模块，如果多个再用一个文件夹包裹



b）文件基础配置



c）web.php配置文件载入



11、建立链接：yii\helpers\Url::to(["manage/managers"])

12、查询sql：

$model = Admin::find(); //查询当前表的所有数据

$count = $model->count();//获取数据条数

$pageSize = Yii::$app->params['admin']['pageSize']['manager'];

$pager = new Pagination(['totalCount' => $count, 'PageSize' => $pageSize]);//设置分页

$managers = $model->offset($pager->offset)->limit($pager->limit)->asArray()->all();//获取当前页码的数据

获取数据是如果不加asArray()返回的是对象，加了是数组

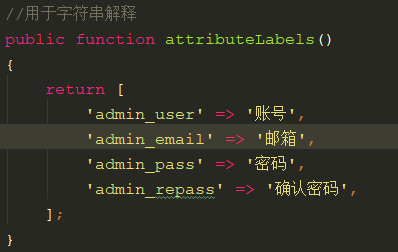
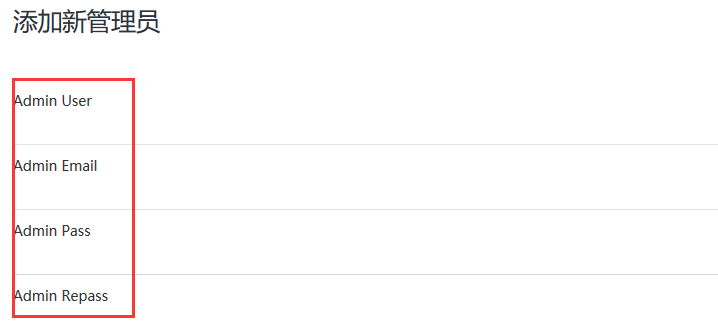
13、结合ci框架实现辅助函数载入：web.php自动引入

14、输出分页：echo yii\widgets\LinkPager::widget(['pagination' => $pager, 'prevPageLabel' => '&#8249;', 'nextPageLabel' => '&#8250;'])

15、表单初始化插件：yii\bootstrap\ActiveForm

16、表单提交按钮：yii\helpers\Html

17、表单出现英文需要在对应的模型中加入解释函数：attributeLabels



18、控制器处理数据返回一次性数据：Yii::$app->session->setFlash('info', '添加成功');

19、一次性数据回调：

if(Yii::$app->session->hasFlash('info')){

echo Yii::$app->session->getFlash('info');

}

20、url添加参数：yii\helpers\Url::to(['managers/del','id' => $value['id']])

21、重新跳转url：$this->redirect(['manage/managers']);

22、物理删除：$model->deleteAll('id = :id',[':id' => $id])

23、逻辑删除

24、邮键发送：

$token = create\_token\_find\_pass($data['Admin']['admin\_user'],$time);

$mailer = Yii::$app->mailer->compose('seekpass',['admin\_user' => $data['Admin']['admin\_user'], 'time' => $time, $token => $token]);

$mailer->setFrom(Yii::$app->params['senderEmail']);

$mailer->setTo($data['Admin']['admin\_email']);

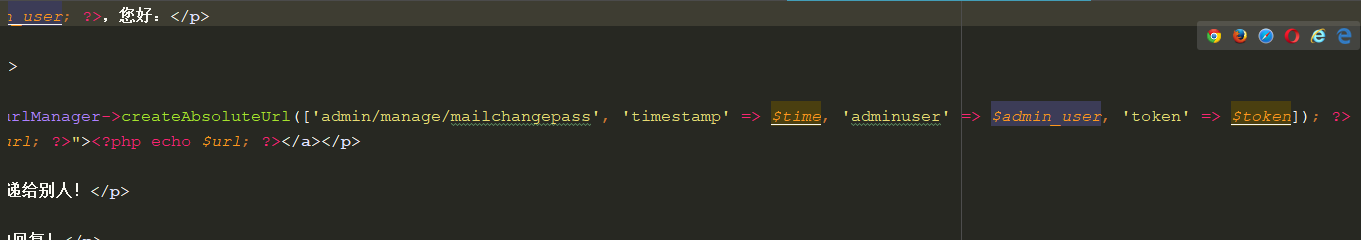
$mailer->setSubject('找回密码');

if($mailer->send()){

return true;

}

对应mail文件夹需要建立文件 模板对应compose中的seekpass

 seekpass.php中的数据是compose传如的，没有控制器

报错误：Swift\_RfcComplianceException Address in mailbox given [1078962834] does not comply with RFC 2822, 3.6.2.

yikjceymfmdihhff

qknoewoffrvmgahh

ibvmprlpzauqhead

web.php也需要配置

'mailer' => [

'class' => 'yii\swiftmailer\Mailer',

// send all mails to a file by default. You have to set

// 'useFileTransport' to false and configure a transport

// for the mailer to send real emails.

'useFileTransport' => true,

'transport' => [

'class' => 'Swift\_SmtpTransport',

'host' => 'smtp.qq.com',//'smtp.163.com',

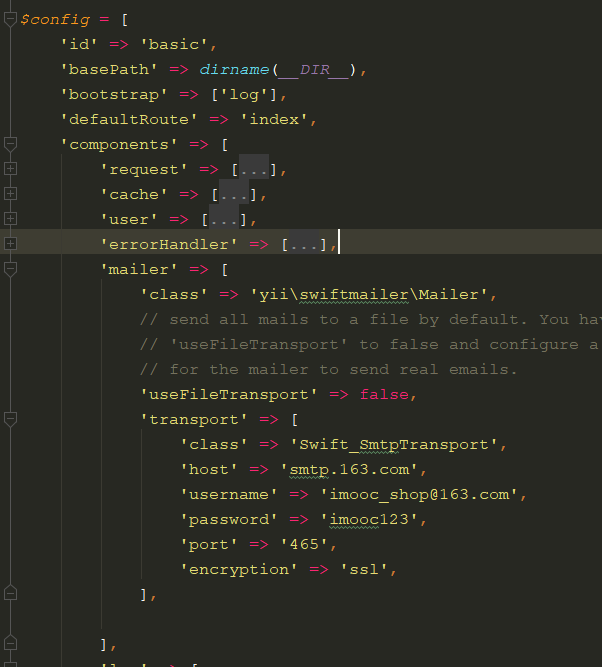
'username' => '1078962834@qq.com',//账号

'password' => 'yikjceymfmdihhff',//授权码

'port' => '465',

'encryption' => 'ssl',

],

],

25、关联表操作：hasOne()、hasMany()

controller调用：$model = User::find()->joinWith('profile');

model定义：

public function getProfile()

{

return $this->hasOne(Profile::className(), ['userid' => 'userid']);

}

Profile::className()：关联表class

['userid' => 'userid']：键的关联

$model = User::find()->where(User::tableName().'.del = 0')->joinWith('profile'); ----》加条件搜索

26、select下拉选项：$form->field($model,'parentid')->dropDownList($list);

27、但选项：$form->field($model, 'ishot')->radioList([0 => '不热卖', 1 => '热卖'], ['class' => 'span8']);

28、规则rules:

a）最小值：[['price','saleprice'], 'number', 'min' => 0.01, 'message' => '价格必须是数字']

b）安全不需要验证：[['issale','ishot', 'pics', 'istui'],'safe']

27、资源管理组建

a）资源：css等一类

b）资源包：管理资源的一个类，渲染时，将需要的资源加入资源包，注册资源渲染

c）资源位置：web目录下，不是在web/asset下

d）相关属性：

basePath：资源文件或web可以访问的目录

baseUrl：web访问资源的url

css、js：资源包文件数组

depends：该资源包依赖的其他资源包

jsOption：加载js时的选项

cssOption：加载css的选项

按需加载：

registerJsFile、registerCssFile、registerJs、registerJs

sourcePath：其他位置存放资源\tmp\css，外目录

publishOption：站外资源配置

e）页面引用

**use** app\assets\AppAsset;  
AppAsset::*register*($this);

$this->head()

$this->beginPage();

$this->beginBody();

$this->endBody();

$this->endPage()

f）无法加载bootstrap，加载其他依赖**yii\bootstrap\BootstrapPluginAsset**

29、小部件使用（NavBar、Nav渲染页面）

1）引入Nav和Navbar;

2）Navbar需要有开始标签和结束标签；

NavBar::begin([  
 'options' => [  
 'class' => 'top-bar animate-dropdown'  
 ]  
]);

中间输出内容

NavBar::end();

3）在开始标签和结束标签的中间需要使用：

Nav::widget([

'options' => [],//可以写样式和其他的配置之类的

'items' => [

['label' => '导航菜单的名称'，'url' =>['/member/auth']],

['label' => '导航菜单的名称'，'url' =>['/member/auth']],

],

])

30、面包屑导航（Breadcrumbs）

1）引入面包屑小插件

use yii\widgets\Breadcrumbs;

2）在显示的面包屑的地方的写法

layout.php布局页面

echo Breadcrumbs::widget([

//首页

'homeLink' => ['label' => '首页', 'url' => ['/admin/default/index']],

//如果其他页面有设置params这个参数中的breadcrumbs,那就显示里面的内容，注意里面是个数组

'links' => isset($this->params['breadcrumbs']) ? $this->params['breadcrumbs']:[],

//面包屑显示的模板

'itemTemplate' => "<li><b>{link}</b></li>&gt;",

]);

3.具体的内容文件中

add.php

$this->params['breadcrumbs'][] = ['label' => '分类管理','url'=>['/admin/category/list']];

$this->params['breadcrumbs'][] = ['label' => '添加分类',];

31、资源压缩使用（根据环境）components中添加

'assetManager' =>[  
 'class' => 'yii\web\AssetManager',  
 'bundles' => [  
 'yii\web\JqueryAsset' => [  
 'js' => [  
 YII\_ENV\_DEV ? 'jquery.js' : 'jquery.min.js'  
 ]  
 ],  
 'yii\bootstrap\BootstrapAsset' => [  
 'css' => [  
 YII\_ENV\_DEV ? 'css/bootstrap.css' : 'css/bootstrap.min.css'  
 ]  
 ],  
 'yii\bootstrap\BootstrapPluginAsset' => [  
 'js' => [  
 YII\_ENV\_DEV ? 'js/bootstrap.js' : 'css/bootstrap.min.js'  
 ]  
 ]  
 ]  
]

32、用户认证体系

1）认证：监狱用户身份的过程。通常使用一个标识符或一个加密令牌鉴别用户身份

2）认证框架注意点

a）设置用户组件：yii\web\User ----》管理认证状态

identity：Yii::$app->user->identity ----》当前用户身份实例，未认证为null

id：Yii::$app->user->id ----》当前用户id，未认证null

isGuest：Yii::$app->user->isGuest ---》 浦安端当前用户是否是游客

相关方法：

login：将当前用户的身份登记到yii\web\user

登陆相关：enableSession、enableAutoLogin

logout：注销用户，启用session时有意义logout（true/false）

b）创建类实现接口：yii\web\IdentityInterface ---》继承实现、

3）认证接口（yii\web\IdentityInterface）：

yii\web\IdentityInterface::findidentity()

yii\web\IdentityInterface::findidentityByAccessToken()

yii\web\IdentityInterface::fgetId()

yii\web\IdentityInterface::getAuthKey()

yii\web\IdentityInterface::validateAuthKey()

4）配置位置：web.php中components中

'user' => [  
 'identityClass' => 'app\models\User',  
 'enableAutoLogin' => true,  
],

5）相关属性

identity：Yii::$app->user->identity -----》当前用户身份实例，未认证的未null

id：Yii::$app->user->id ----》当前用户id，无为null

isGuest：Yii::$app->user->isGuest -----》判断是否是游客（未认证）

6）相关方法

login：将当前用户身份登记到yii\web\User enableSession、enableAutoLogin

loginout：注销用户，启动session有效 logout(true)、logout(false)

7）实现步骤

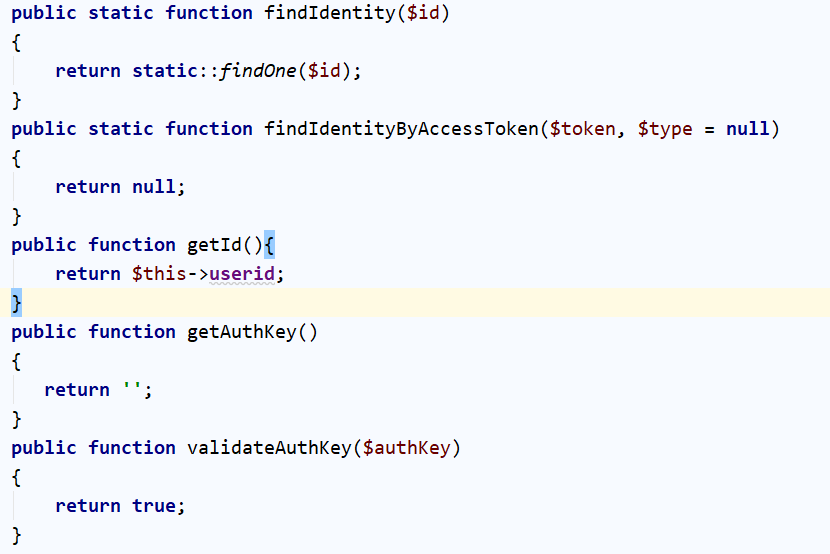
a）web.php 进行配置

**'user'** => [  
 **'identityClass'** => **'app\models\User'**,  
 **'enableAutoLogin'** => **true**,  
],

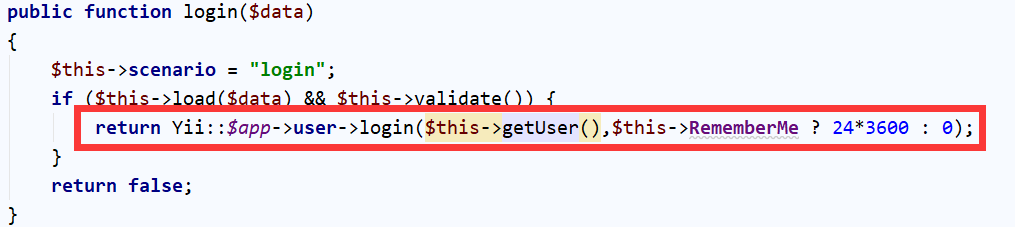
b）用户登陆模型继承

**implements** \yii\web\IdentityInterface

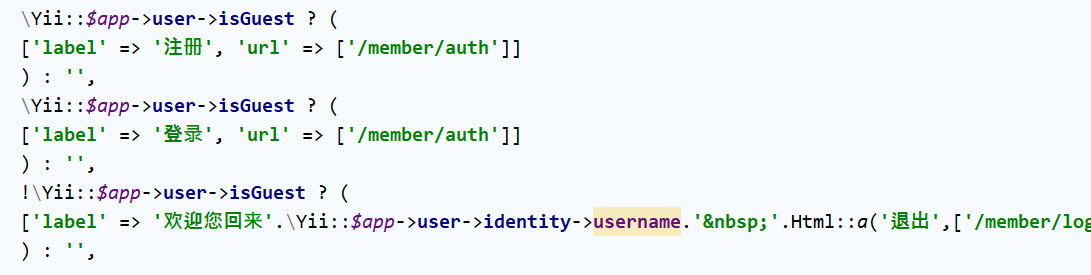
实现方法



c）登陆接口



d）view部分使用isGuest判断



f）前后台json分离

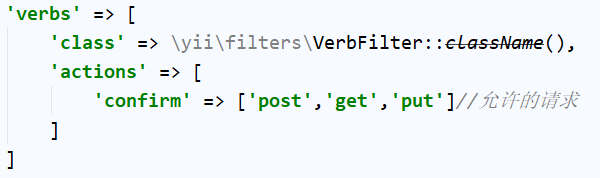


9）、行为过滤器，在控制器中定义behaviors方法



10）、访问过滤器，behavior中添加数组





11）、密码操作：bcrypt

Yii::$app->getSecurity()->generatePasswordHash() 密码加密

Yii::$app->getSecurity()->validatePassword(); 密码解密

33、rbac

4吗