<?php

namespace Admin\Controller;

class AdminController extends BaseController {

 public function \_initialize(){

 parent::leftmenu();

 parent::\_initialize();

 }

 public function index() {

 $model = M('Admin');

 $list = $model->order('uid DESC')->select();

 $this->assign('list', $list);

 $this->display();

 }

 public function add() {

 $this->prepare();

 $this->status();

 $this->display();

 }

 public function edit() {

 $id = $\_REQUEST['id'];

 $model = M('Admin');

 $data = $model->find($id);

 $this->status();

 $this->assign('data', $data);

 $this->prepare();

 $this->display();

 }

 public function save() {

 $model = M('Admin');

 $data = $model->create();

 $data['roleids'] = implode(',', $\_POST['roleids']);

 $id = I('id');

 $status = I("status");

 $password = I('password');

 $username = I('username');

 $password = pass\_encode($username,$password);

 $cnt = $model->where("username='$username'")->count();

 if ($cnt>0)

 $this->error('用户名已存在');

 $data = array(

 'username' => $username,

 'password' => $password,

 'roleids' => $data['roleids'],

 'status'=>$status,

 'regtime' => time()

 );

 $model->add($data);

 $this->success('增加成功',U('Admin/index'));

 }

 public function update() {

 $model = M('Admin');

 $id = I('id');

 $data = $model->create();

 $data['roleids'] = implode(',', $\_POST['roleids']);

 $data['status'] = I('status');

 $password = I('password');

 if ($password) {

 $username = $model->where("uid='$id'")->getField('username');

 $password = pass\_encode($username,$password);

 $data['password'] = $password;

 } else {

 unset($data['password']);

 }

 $model->where("uid=$id")->save($data);

 $this->success('修改成功',U('Admin/index'));

 }

 public function delete() {

 $id = I('id');

 $model = M('Admin');

 $cnt = $model->count();

 if ($cnt<2)

 $this->error('至少要有一个管理员账号');

 $model->where("uid=$id AND islock=0")->delete();

 $this->success('删除成功');

 }

 public function check() {

 $username = I('param');

 $model = M('Admin');

 $cnt = $model->where("username='$username'")->count();

 if ($cnt>0)

 $data = array('info'=>'用户名已存在','status'=>'n');

 else

 $data = array('info'=>'可以使用','status'=>'y');

 $this->ajaxReturn($data);

 }

 public function prepare() {

 $roles = $this->cacheSvc->getData('RoleMlt');

 $this->assign('roles',$roles);

 }

 public function status(){

 $model = M('Arctype');

 $ksList = $model->select();

 $ksList = tree\_build($ksList,51);

 // foreach($ksList as $k=>$v){

 // $ksList[$k]['children'] = $model->field("id,title")

 // ->where(['upid'=>$v['id']])

 // ->select();

 // }

 $this->assign('ksList',$ksList);

 }

}<?php

namespace Admin\Controller;

class AdvertController extends BaseController {

 public function index() {

 $model = D('Advert');

 $where = '1=1';

 $keyword = I('keyword');

 $catid = I('catid');

 $typelist = $this->cacheSvc->getData('AdvertTypeList');

 $typemap = $this->cacheSvc->getData('AdvertTypeMlt');

 if (!empty($catid)) {

 $catid = tree\_get\_leaves($typelist, $catid);

 $where .= " AND catid IN($catid)";

 }

 if (!empty($keyword))

 $where .= " AND title LIKE '%{$keyword}%'";

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->field('id,title,code,catid,addtime,status,updatetime')

 ->where($where)

 ->order("id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 foreach($list as $k=>$v) {

 $list[$k]['catname'] = $typemap[$v['catid']]['title'];

 }

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->prepare();

 $this->assign('caption','管理广告');

 $this->display();

 }

 public function add() {

 $this->prepare();

 $data = array(

 'hash'=> uniqid2(),

 'addtime' => time(),

 'updatetime' => time(),

 'status' => 1

 );

 $this->assign('data', $data);

 $this->assign('caption','增加广告');

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Advert');

 $data = $model->find($id);

 $this->assign('data', $data);

 $this->prepare();

 $this->assign('caption', '编辑广告');

 $this->display();

 }

 public function update() {

 $model = D('Advert');

 $data = $model->create();

 if (empty($data['addtime'])) {

 $data['addtime'] = time();

 } else {

 $data['addtime'] = strtotime($data['addtime']);

 }

 $data['updatetime'] = time();

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 //更新附件信息

 $key = 'Advert/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功', U('Advert/index'));

 }

 public function delete() {

 $id = I('id');

 D('Advert')->deleteOne($id);

 $this->success('删除成功', U('Advert/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择项目');

 $model = D('Advert');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'hot':

 $model->where("id IN($ids)")->save(array('ishot'=>time()));

 break;

 case 'unhot':

 $model->where("id IN($ids)")->save(array('ishot'=>0));

 break;

 case 'delete':

 D('Advert')->deleteBatch($id);

 break;

 case 'show':

 $model->where("id IN($ids)")->save(array('status'=>1));

 break;

 case 'hide':

 $model->where("id IN($ids)")->save(array('status'=>0));

 break;

 }

 $this->success('执行成功',U('Advert/index'));

 }

 protected function prepare() {

 $tree = $this->cacheSvc->getData('AdvertTypeList');

 $this->assign('typelist', $tree);

 }

}

<?php

namespace Admin\Controller;

class AdvtypeController extends BaseController {

 public function index() {

 $code = I('get.code');

 $model = D('Advtype');

 $where['code'] = ['eq',$code];

 if($code == "index"){

 $ksid = 0;

 }else{

 $ksid = session('ksid');

 }

 $where['ksid'] = ['eq',$ksid];

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 $count = count($list);

 $this->prepare($code);

 $this->assign('code',$code);

 $this->assign('length',$count);

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->assign('caption','管理广告组');

 $this->display();

 }

 public function add() {

 $code = I('get.code');

 $data = array(

 'hash' => uniqid2(),

 'code'=>$code,

 );

 $this->assign('data', $data);

 $this->assign('caption','增加广告组');

 $this->prepare($code);

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Advtype');

 $data = $model->find($id);

 $this->assign('data', $data);

 $this->assign('caption', '编辑广告组');

 $this->prepare($data['code']);

 $this->display();

 }

 public function update() {

 $model = D('Advtype');

 $data = $model->create();

 if($data['code'] == "ksIndex"){

 $data['ksid'] = session('ksid');

 }else{

 $data['ksid'] = 0;

 }

 $data['isMobile'] = I('post.isMobile');

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 $key = 'Advtype/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->cache();

 $this->success('保存成功', U('Advtype/index',array('code'=>$data['code'])));

 }

 public function delete() {

 $id = I('id');

 $model = D('Advtype');

 $data = $model->find($id);

 $model->deleteOne($id);

 $this->cache();

 $this->success('删除成功', U('Advtype/index',array('code'=>$data['code'])));

 }

 //刷新缓存

 private function cache() {

 $this->cacheSvc->cacheData('AdvertType');

 }

 private function prepare($code) {

 if($code == "index"){

 $model = D('Arctype');

 $typelist = $model->where(['id'=>['neq',51]])->order("sortno asc")->select();

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Article/index',array('catid'=>$v['id']));

 }

 }else{

 $typelist = M('Dptarctype')->where(['ksid'=>session('ksid')])->order("sortno asc")->select();

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Dptarticle/index',array('catid'=>$v['id']));

 }

 }

 $typelist = tree\_build($typelist);

 $this->assign('typetree', $typelist);

 }

}<?php

namespace Admin\Controller;

class ArctypeController extends BaseController {

 public function index() {

 $model = D('Arctype');

 $list = $model->where(['id'=>['neq',51]])->order("sortno asc")->select();

 $list = tree\_build($list);

 $modellist = $this->cacheSvc->getData('ArchiveModelMap');

 foreach($list as $k=>$v) {

 $list[$k]['modelname'] = $modellist[$v['model']];

 }

 $this->prepare();

 $this->assign('caption', '栏目管理');

 $this->assign('list', $list);

 $this->display();

 }

 public function batch() {

 $update = $\_POST['update'];

 if(empty($update)){

 $this->error("请选择操作内容");

 }

 $ids = $\_POST['id'];

 if(empty($ids)){

 $this->error('请选择更新的项目');

 }

 $sorts = $\_POST['sortno'];

 $model = D('Arctype');

 switch($update){

 case "hidden" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('ishidden'=>1));

 }

 break;

 case "show" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('ishidden'=>0));

 }

 break;

 case "sort" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('sortno'=>$sortno));

 }

 }

 $this->cache();

 $this->success('更新成功',U('Arctype/index'));

 }

 public function add() {

 $model = D('Arctype');

 $data = array(

 'id'=>'',

 'sortno'=>0,

 'model'=>'default',

 'pagesize'=>20,

 'linktype'=>'index',

 'ishidden'=>0,

 'ismenu'=>1,

 'ishot'=>0,

 'viewnum'=>0,

 'praisenum'=>0,

 'sharenum'=>0,

 'hash'=>uniqid2()

 );

 $upid = I('pid');

 if (!empty($upid)) {

 $type = $model->where("id='$upid'")->find();

 if($type['upid'] > 0){

 $data['code'] = $type['code'];

 }else{

 $data['code'] = $type['code'] . "/index";

 }

 $data['upid'] = $upid;

 $data['model'] = $type['model'];

 $data['linktype'] = $type['linktype'];

 $data['linktpl'] = $type['linktpl'];

 $data['pagesize'] = $type['pagesize'];

 $data['ishidden'] = $type['ishidden'];

 $data['ismenu'] = $type['ismenu'];

 $data['ishot'] = $type['ishot'];

 }

 $this->assign('caption','增加栏目');

 $this->assign('data', $data);

 $this->prepare();

 $this->display('edit');

 }

 public function edit() {

 $model = D('Arctype');

 $id = I('id');

 $this->checkId($id);

 $data = $model->where("id=$id")->find();

 //显示页面

 $this->assign('caption','修改栏目');

 $this->prepare();

 $this->assign('data', $data);

 $this->display();

 }

 public function update() {

 $model = D('Arctype');

 $id = I('id');

 if ($data = $model->create()) {

 $upid = $data['upid'];

 if($upid == 0){

 $data['linktpl'] = "/" . I('code');

 }else{

 $data['linktpl'] = "/" . I('code') . "?cid={cid}";

 }

 if (empty($data['title']))

 $this->error('请填写栏目名称');

 if (empty($data['code']))

 $this->error('请填写栏目标识');//，标识用于预设模板

 if ($data['sortno']<0)

 $this->error('排序值是必须大于等于0');

 if (empty($data['ishidden']))

 $data['ishidden'] = 0;

 if (empty($data['ismenu']))

 $data['ismenu'] = 0;

 if (empty($data['ishot']))

 $data['ishot'] = 0;

 if ($upid>0) {

 $uptype = $model->where("id=$upid")->find();

 $data['level'] = $uptype['level'] + 1;

 if (empty($uptype['upids']))

 $data['upids'] = ",$upid,";

 else

 $data['upids'] = $uptype['upids'] . "$upid,";

 $model->where("id=$upid")->save(array('isleaf'=>0));

 } else {

 $data['level'] = 0;

 $data['upids'] = '';

 }

 if (empty($id)) {

 $data['isleaf'] = 1;

 $model->add($data);

 } else {

 $model->save($data);

 //是否更改了上级栏目

 $oldupid = I('oldupid');

 if ($oldupid!=$upid) {

 $cnt = $model->where("upids LIKE '%,$oldupid,%'")->count();

 if ($cnt<=0) {

 $model->where("id='$oldupid'")->save(array('isleaf'=>1));

 }

 //处理子栏目

 $list = $model->where("upids LIKE '%,$id,%'")->select();

 foreach($list as $v) {

 $save = array();

 if ($oldupid==0 && $upid>0) {

 $save['level'] = array('exp','level+1');

 $save['upids'] = ",$upid".$v['upids'];

 }

 if ($oldupid>0 && $upid==0) {

 $save['level'] = array('exp','level-1');

 $save['upids'] = str\_replace(",$oldupid,", ',', $v['upids']);

 }

 if ($oldid>0 && $upid>0) {

 $save['upids'] = str\_replace(",$oldupids,", ",$upid,", $v['upids']);

 }

 $model->where("id='$v[id]'")->save($save);

 }

 }

 }

 //更新附件信息

 $key = 'Arctype/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 //缓存

 $this->cache();

 $this->success('保存成功',U('Arctype/index'));

 } else {

 $this->error($model->getError());

 }

 }

 public function delete() {

 $id = I('get.id');

 $this->checkId($id);

 $model = D('Arctype');

 $data = $model->find($id);

 //是否可以删除

 if ($data['isleaf']==0) {

 $this->error('该栏目下有子栏目，不可删除');

 }

 //如果是叶子节点，并且有内容就不可删除

 $cnt = D('Article')->where(['catid'=>['eq',$id]])->count();

 if ($cnt>0)

 $this->error('该栏目下有内容，不可删除');

 // $typemodel = $data['model'];

 // if ($typemodel && $typemodel!='default') {

 // $acrmodel = M(ucfirst($typemodel));

 // if ($arcmodel) {

 // $cnt = $arcmodel->where("catid=$id")->count();

 // if ($cnt>0)

 // $this->error('该栏目下有内容，不可删除');

 // }

 // }

 $model->deleteOne($id);

 $this->cache();

 $this->success('删除成功',U('Arctype/index'));

 }

 //刷新缓存

 private function cache() {

 $this->cacheSvc->cacheData('ArchiveType');

 }

 private function prepare() {

 // $typelist = $this->cacheSvc->getData('ArchiveTypeTree');

 $modellist = $this->cacheSvc->getData('ArchiveModelMap');

 $linktypelist = $this->cacheSvc->getData('LinkTypeMap');

 $model = D('Arctype');

 $typelist = $model->where(['id'=>['neq',51]])->order("sortno asc")->select();

 $typelist = tree\_build($typelist);

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Article/index',array('catid'=>$v['id']));

 }

 $url = U('Arctype/index');

 $this->assign('url',$url);

 $this->assign('typetree', $typelist);

 $this->assign('modellist', $modellist);

 $this->assign('linktypelist', $linktypelist);

 }

}<?php

namespace Admin\Controller;

class ArticleController extends BaseController {

 public function index() {

 $model = D('Article');

 $where = '1=1';

 $keyword = I('keyword');

 $catid = I('catid');

 $typemap = $this->cacheSvc->getData('ArchiveTypeMap');

 if (!empty($catid)) {

 $catid = $this->arctypeSvc->getLeaves($catid);

 $where .= " AND catid IN($catid)";

 }

 if (!empty($keyword))

 $where .= " AND title LIKE '%{$keyword}%'";

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("sortno asc,id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 foreach($list as $k=>$v) {

 $list[$k]['catname'] = $typemap[$v['catid']];

 }

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->prepare();

 $this->assign('caption','管理文章');

 $this->display();

 }

 public function add() {

 $this->prepare();

 $data = array(

 'hash'=> uniqid2(),

 'addtime' => time(),

 'sortno' => 0,

 'viewnum' => 0

 );

 $this->assign('data', $data);

 $this->assign('caption','增加文章');

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Article');

 $data = $model->find($id);

 if($data['catid'] == 239 || $data['catid'] == 220){

 $this->assign('css',1);

 }else{

 $this->assign('css',0);

 }

 $this->assign('data', $data);

 $this->prepare();

 $this->assign('caption', '编辑文章');

 $this->display();

 }

 public function update() {

 $model = D('Article');

 $data = $model->create();

 if ($data['catid']!=73){

 if ($data['isjz']== 1){

 $data['isjz']=0;

 }

 if ($data['ishb']== 1){

 $data['ishb']=0;

 }

 }

 if(empty($data['title'])){

 $this->error("标题不能为空");

 }

 if (empty($data['memo'])) {

 $data['memo'] = msubstr(html2text($data['content']),0,500);

 }

 if (empty($data['addtime'])) {

 $data['addtime'] = time();

 } else {

 $data['addtime'] = strtotime($data['addtime']);

 }

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 //更新附件信息

 $key = 'Article/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功', $this->getLastUrl());

 }

 public function delete() {

 $id = I('id');

 D('Article')->deleteOne($id);

 $this->success('删除成功', U('Article/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择项目');

 $model = D('Article');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'sort':

 $sorts = $\_POST['sortno'];

 if ($id) {

 foreach($id as $idss) {

 $sortno = $sorts[$idss];

 $model->where("id=$idss")->save(array('sortno'=>$sortno));

 }

 $this->success('更新成功',U('Article/index'));

 } else {

 $this->error('请选择更新的项目');

 }

 case 'hot':

 $model->where("id IN($ids)")->save(array('ishot'=>time()));

 break;

 case 'unhot':

 $model->where("id IN($ids)")->save(array('ishot'=>0));

 break;

 case 'delete':

 D('Article')->deleteBatch($id);

 break;

 }

 $this->success('执行成功',U('Article/index'));

 }

 protected function prepare() {

 $model = D('Arctype');

 $typelist = $model->where(['id'=>['neq',51]])->order("sortno asc")->select();

 $typelist = tree\_build($typelist);

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Article/index',array('catid'=>$v['id']));

 }

 $url = U('Arctype/index');

 $this->assign('url',$url);

 $this->assign('typetree', $typelist);

 }

}<?php

namespace Admin\Controller;

class ArticleController extends BaseController {

 public function index() {

 $model = D('Article');

 $where = '1=1';

 $keyword = I('keyword');

 $catid = I('catid');

 $typemap = $this->cacheSvc->getData('ArchiveTypeMap');

 if (!empty($catid)) {

 $catid = $this->arctypeSvc->getLeaves($catid);

 $where .= " AND catid IN($catid)";

 }

 if (!empty($keyword))

 $where .= " AND title LIKE '%{$keyword}%'";

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("sortno asc,id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 foreach($list as $k=>$v) {

 $list[$k]['catname'] = $typemap[$v['catid']];

 }

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->prepare();

 $this->assign('caption','管理文章');

 $this->display();

 }

 public function add() {

 $this->prepare();

 $data = array(

 'hash'=> uniqid2(),

 'addtime' => time(),

 'sortno' => 0,

 'viewnum' => 0

 );

 $this->assign('data', $data);

 $this->assign('caption','增加文章');

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Article');

 $data = $model->find($id);

 if($data['catid'] == 239 || $data['catid'] == 220){

 $this->assign('css',1);

 }else{

 $this->assign('css',0);

 }

 $this->assign('data', $data);

 $this->prepare();

 $this->assign('caption', '编辑文章');

 $this->display();

 }

 public function update() {

 $model = D('Article');

 $data = $model->create();

 if ($data['catid']!=73){

 if ($data['isjz']== 1){

 $data['isjz']=0;

 }

 if ($data['ishb']== 1){

 $data['ishb']=0;

 }

 }

 if (empty($data['memo'])) {

 $data['memo'] = msubstr(html2text($data['content']),0,500);

 }

 if (empty($data['addtime'])) {

 $data['addtime'] = time();

 } else {

 $data['addtime'] = strtotime($data['addtime']);

 }

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 //更新附件信息

 $key = 'Article/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功', $this->getLastUrl());

 }

 public function delete() {

 $id = I('id');

 D('Article')->deleteOne($id);

 $this->success('删除成功', U('Article/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择项目');

 $model = D('Article');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'sort':

 $sorts = $\_POST['sortno'];

 if ($id) {

 foreach($id as $idss) {

 $sortno = $sorts[$idss];

 $model->where("id=$idss")->save(array('sortno'=>$sortno));

 }

 $this->success('更新成功',U('Article/index'));

 } else {

 $this->error('请选择更新的项目');

 }

 case 'hot':

 $model->where("id IN($ids)")->save(array('ishot'=>time()));

 break;

 case 'unhot':

 $model->where("id IN($ids)")->save(array('ishot'=>0));

 break;

 case 'delete':

 D('Article')->deleteBatch($id);

 break;

 }

 $this->success('执行成功',U('Article/index'));

 }

 protected function prepare() {

 $model = D('Arctype');

 $typelist = $model->where(['id'=>['neq',51]])->order("sortno asc")->select();

 $typelist = tree\_build($typelist);

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Article/index',array('catid'=>$v['id']));

 }

 $url = U('Arctype/index');

 $this->assign('url',$url);

 $this->assign('typetree', $typelist);

 }

}<?php

namespace Admin\Controller;

class AuthController extends BaseController {

 public function index() {

 $this->display();

 }

 public function login() {

 $checkcode = trim($\_POST['seccode']);

 $verify = new \Think\Verify();

 if (!$verify->check($checkcode, 'admin'))

 $this->error('验证码错误');

 $username = trim($\_POST['username']);

 $password = trim($\_POST['password']);

 if (empty($username))

 $this->error('请输入用户名');

 if (empty($password))

 $this->error('请输入密码');

 $realpass = pass\_encode($username, $password);

 $model = M('Admin');

 $user = $model->field('uid,username,status')->where("username='$username' AND password='$realpass'")->find();

 if (empty($user))

 $this->error('用户名或密码错误');

 session('admin', $user);

 $this->redirect('Index/index');

 }

 public function logout() {

 session('ksid',null);

 session('admin',null);

 $this->clientRedirect('Auth/index');

 }

}

<?php

namespace Admin\Controller;

use Think\Controller;

class BaseController extends Controller {

 public $poweredby;

 protected $cacheSvc = null;

 protected $arctypeSvc = null;

 protected $memberSvc= null;

 protected function clientRedirect($url) {

 $url = U($url);

 $text = "<script>window.top.location='$url';</script>";

 die($text);

 }

 protected function checkId($id) {

 if (empty($id)) {

 $this->error('访问的页面不存在');

 }

 }

 protected function storeUrl() {

 session('lasturl',get\_url());

 }

 protected function getLastUrl() {

 $url = session('lasturl');

 if (empty($url)) {

 $url = \_\_CONTROLLER\_\_;

 }

 return $url;

 }

 protected function \_initialize() {

 //检测是否登录

 $ignoreLogin = array('Auth');

 if (!in\_array(CONTROLLER\_NAME,$ignoreLogin)) {

 $this->admin = session('admin');

 if (empty($this->admin)) {

 $this->clientRedirect('Auth/index');

 }

 }

 $siteurl = C('SITE\_URL');

 //初始化服务

 $this->cacheSvc = D('Cache','Service');

 $this->cacheSvc->initCache();

 $this->arctypeSvc = D('Arctype','Service');

 $admin = array('uid'=>'1','username'=>'admin');

 $this->assign('pageTitle', '华易动力网站后台管理系统');

 $this->assign('admininfo', $admin);

 $this->poweredby = C('SOFT\_NAME').' '.C('SOFT\_VERSION');

 $this->assign('siteurl', $siteurl);

 //存储列表页url

 if (ACTION\_NAME == 'index') {

 $this->storeUrl();

 }

 }

 public function leftmenu(){

 $menu = load\_config(MODULE\_PATH.'Conf/menu.php');

 $this->assign('menulist',$menu['0']['submenu']);

 }

}<?php

namespace Home\Controller;

use Source\Home\Controller\Base;

class BaseDptController extends BaseController {

 protected $dptarctypeSvc = null;

 protected $page = array();

 protected $crumbs = array();

 protected $setting = array();

 protected function \_initialize() {

 $ksid = I('get.ksid');

 if(empty($ksid)){

 $ksid = session('indexKsid');

 }else{

 session('indexKsid',$ksid);

 }

 $ksidsesion = session('indexKsid');

 $this->checkId($ksid);

 //初始化工作

 $this->cacheSvc = D('Cache','Service');

 $this->dptarctypeSvc = D('Dptarctype','Service');

 $ksType = D('Arctype','Service')->getType($ksid);

 $this->assign('ksType',$ksType);

 //系统设置

 $this->setting = $this->cacheSvc->getData('SettingMap');

 C($this->setting);

 $this->assign('setting', $this->setting);

 // $\_list = M('Dptarctype')->where(['ksid'=>$ksid])->select();

 $mainNav = $this->dptarctypeSvc->getChildren(0);

 foreach($mainNav as $k=>$v){

 if($v['ishidden'] == 1){

 unset($mainNav[$k]);

 }

 }

 foreach($mainNav as $k=>$v){

 $mainNav[$k]['children'] = $this->dptarctypeSvc->getChildren($v['id']);

 foreach($mainNav['$k']['children'] as $n=>$m){

 $mainNav[$k]['children'][$n]['children'] = $this->dptarctypeSvc->getChildren($m['id']);

 }

 }

 parent::friendLink();

 //初始化页面相关

 $this->page['title'] = $this->setting['SITE\_NAME'];

 $this->page['keywords'] = $this->setting['SITE\_SEOKEY'];

 $this->page['desc'] = $this->setting['SITE\_SEODESC'];

 $this->page['channel'] = 'home';

 $this->crumbs[] = array('title'=>'科室首页','url'=>'/DptIndex/index');

 $this->assign('nav',$mainNav);

 }

}<?php

namespace Admin\Controller;

class BusinessController extends BaseController {

 public function index() {

 $pricetypemap = $this->cacheSvc->getData('Pricetypemap');

 $givetypemap = $this->cacheSvc->getData('Givetypemap');

 $model = M('Business');

 $where = '1=1';

 $xqmc = I('xqmc');

 if (!empty($xqmc))

 $where .= " AND title LIKE '%{$xqmc}%'";

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->assign('caption','管理公益活动');

 $this->display();

 }

 public function add() {

 $data = array(

 'hash'=> uniqid2(),

 'addtime' => time(),

 'sortno' => 0,

 'ishot' => 0,

 );

 $this->assign('data', $data);

 $this->prepare();

 $this->assign('caption', '增加公益活动');

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Business');

 $data = $model->find($id);

 $this->assign('data', $data);

 $this->prepare();

 $this->assign('caption', '编辑公益活动');

 $this->display();

 }

 public function update() {

 $model = D('Business');

 $data = $model->create();

 if (empty($data['addtime'])){

 $data['addtime'] = time();

 } else {

 $data['addtime'] = strtotime($data['addtime']);

 }

 $data['prices'] = ext\_implode(',', $data['prices']);

 $data['gives'] = ext\_implode(',', $data['gives']);

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 //更新附件信息

 $key = 'Business/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功', $this->getLastUrl());

 }

 public function delete() {

 $id = I('id');

 D('Business')->delete($id);

 $uid=M('member')->where("businessid = $id")->getField('uid');

 if($uid){

 M('member')->where("uid=$uid")->save(array('businessid'=>0));

 }

 $this->success('删除成功', U('Business/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择信息');

 $model = D('Business');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'hot':

 $model->where("id IN($ids)")->save(array('ishot'=>time()));

 break;

 case 'unhot':

 $model->where("id IN($ids)")->save(array('ishot'=>0));

 break;

 case 'delete':

 $model->where("id IN($ids)")->delete();

 M('member')->where("businessid IN ($ids)")->save(array('businessid'=>0));

 break;

 case 'show':

 $model->where("id IN($ids)")->save(array('status'=>1));

 break;

 case 'hidden':

 $model->where("id IN($ids)")->save(array('status'=>0));

 break;

 case 'lock':

 $model->where("id IN($ids)")->save(array('issuoding'=>1));

 break;

 case 'unlock':

 $model->where("id IN($ids)")->save(array('issuoding'=>0));

 break;

 }

 $this->success('执行成功',U('Business/index'));

 }

 protected function prepare() {

 $pricetypelist = $this->cacheSvc->getData('PricetypeList');

 $givetypelist = $this->cacheSvc->getData('GivetypeList');

 $this->assign('pricetypelist', $pricetypelist);

 $this->assign('givetypelist', $givetypelist);

 }

}<?php

namespace Admin\Controller;

class CacheController extends BaseController {

 public function index() {

 $this->assign('list', $this->cacheSvc->getConfig());

 $this->display();

 }

 public function batch() {

 $idArr = $\_POST['id'];

 if (empty($idArr))

 $this->error('请选择要刷新的缓存');

 foreach($idArr as $v) {

 $this->cacheSvc->cacheData($v);

 }

 $this->success('缓存已刷新',U('Cache/index'));

 }

 public function refresh() {

 $this->cacheSvc->initCache();

 $this->success('缓存已刷新',U('Cache/index'));

 }

}<?php

namespace Admin\Controller;

class DptarctypeController extends BaseController {

 public function index() {

 $ksid = I('get.ksid');

 if(empty($ksid)){

 $ksid = session('ksid');

 }else{

 session('ksid',$ksid);

 }

 $this->ksColumn($ksid);

 }

 public function ksColumn($ksid){

 session("ksid",$ksid);

 $ks = D('Arctype')->where("id=$ksid")->find();

 $model = D('Dptarctype');

 $list = $model->where(['ksid'=>$ksid])->order("sortno asc")->select();

 $list = tree\_build($list);

 $modellist = $this->cacheSvc->getData('ArchiveModelMap');

 foreach($list as $k=>$v) {

 $list[$k]['modelname'] = $modellist[$v['model']];

 }

 $this->prepare();

 $this->assign('kstype',$ks);

 $caption = $ks['title'] . "栏目管理";

 $this->assign('caption', $caption);

 $this->assign('list', $list);

 $this->display('index');

 }

 // 科室大分类

 public function depart(){

 $status = session('admin.status');

 if($status != 0){

 $ksid = $status;

 $this->ksColumn($ksid);

 }else{

 $model = D('Arctype');

 // 获取科室导航栏目id

 $type = D('Arctype','service')->getType("Departnav");

 // 获取科室大分类

 $list = $model->where(['upid'=>$type['id']])->select();

 foreach($list as $k=>$v){

 $list[$k]['children'] = $model->where(['upid'=>$v['id']])->select();

 }

 $this->assign('list',$list);

 $this->assign('caption', '科室管理');

 $this->display();

 }

 }

 // public function ltDepart(){

 // $upid = I('get.upid');

 // $model = D('Department');

 // $list = $model->where("upid=$upid")->select();

 // $this->assign('list',$list);

 // $this->display();

 // }

 // 科室管理

 public function manageKs(){

 $arctype = D('Arctype','service');

 $model = M('Arctype');

 $list = $model->where(['upids'=>['like',"%51%"]])->select();

 $list = tree\_build($list,51);

 $this->assign('list',$list);

 $this->display();

 }

 public function addDepart(){

 $model = D('Arctype');

 $data = array(

 'id'=>'',

 'hash'=>uniqid2(),

 'upid'=>I('upid')

 );

 $model = D('Arctype');

 // 获取科室导航栏目id

 $type = D('Arctype','service')->getType("Departnav");

 $typeList = $model->where(['upid'=>$type['id']])->select();

 $this->assign('typeList',$typeList);

 $this->assign('caption','增加科室');

 $this->assign('data', $data);

 $this->display();

 }

 public function editDepart(){

 $model = D('Arctype');

 $id = I('id');

 $this->checkId($id);

 $data = $model->where("id=$id")->find();

 //显示页面

 // 获取科室导航栏目id

 $type = D('Arctype','service')->getType("Departnav");

 $typeList = $model->where(['upid'=>$type['id']])->select();

 $this->assign('typeList',$typeList);

 $this->assign('caption','编辑科室');

 $this->assign('data', $data);

 $this->display('addDepart');

 }

 public function delDepart(){

 $id = I('id');

 $this->checkId($id);

 $model = D('Arctype');

 $list = $model->where("upid=$id")->select();

 if(!empty($list)){

 $this->error("该科室分类下还有科室,无法删除");

 }

 // 如果该科室下有栏目则不可删除

 $ksColumnList = D('Dptarctype')->where(['ksid'=>['eq',$id]])->select();

 if(!empty($ksColumnList)){

 $this->error("该科室下有栏目,无法删除");

 }

 // 删除科室链接

 D('Kslink')->where(['ksid'=>['eq',$id]])->delete();

 // 删除科室流动图片

 D('Advtype')->where(['ksid'=>['eq',$id],['code'=>"KsIndex"]])->delete();

 $model->deleteOne($id);

 // 由于科室分类放在一级网站栏目类型表中,删除成功刷新一级网站缓存

 $this->cacheTop();

 $this->success("删除成功");

 }

 // public function batchdpt(){

 // $ids = I('id');

 // $model = D('Department');

 // foreach($ids as $v){

 // $model->delete($v);

 // }

 // $this->success("删除成功");

 // }

 public function updateDepart(){

 $model = D('Arctype');

 $id = I('id');

 if ($data = $model->create()) {

 $upid = $data['upid'];

 $data['image'] = I('image');

 $data['sortno'] = I('sortno');

 if (empty($data['title']))

 $this->error('请填写栏目名称');

 if ($upid>0) {

 $uptype = $model->where("id=$upid")->find();

 $data['level'] = $uptype['level'] + 1;

 if (empty($uptype['upids']))

 $data['upids'] = ",$upid,";

 else

 $data['upids'] = $uptype['upids'] . "$upid,";

 $model->where("id=$upid")->save(array('isleaf'=>0));

 } else {

 $data['level'] = 0;

 $data['upids'] = '';

 }

 if (empty($id)) {

 $data['isleaf'] = 1;

 $model->add($data);

 } else {

 $model->save($data);

 //是否更改了上级栏目

 $oldupid = I('oldupid');

 if ($oldupid!=$upid) {

 $cnt = $model->where("upids LIKE '%,$oldupid,%'")->count();

 if ($cnt<=0) {

 $model->where("id='$oldupid'")->save(array('isleaf'=>1));

 }

 //处理子栏目

 $list = $model->where("upids LIKE '%,$id,%'")->select();

 foreach($list as $v) {

 $save = array();

 if ($oldupid==0 && $upid>0) {

 $save['level'] = array('exp','level+1');

 $save['upids'] = ",$upid".$v['upids'];

 }

 if ($oldupid>0 && $upid==0) {

 $save['level'] = array('exp','level-1');

 $save['upids'] = str\_replace(",$oldupid,", ',', $v['upids']);

 }

 if ($oldid>0 && $upid>0) {

 $save['upids'] = str\_replace(",$oldupids,", ",$upid,", $v['upids']);

 }

 $model->where("id='$v[id]'")->save($save);

 }

 }

 }

 //更新附件信息

 $key = 'Arctype/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 //缓存

 $this->cacheTop();

 $this->success('保存成功',U('Dptarctype/manageKs'));

 } else {

 $this->error($model->getError());

 }

 }

 // 科室批量操作

 public function batchKs(){

 $ids = I('post.id');

 if(empty($ids)){

 $this->error('请选择更新的项目');

 }

 $model = D('Arctype');

 $sorts = $\_POST['sortno'];

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('sortno'=>$sortno));

 }

 $this->cacheTop();

 $this->success('更新成功',U('Dptarctype/manageKs'));

 }

 public function batch() {

 $update = $\_POST['update'];

 $updatec = $\_POST['updatec'];

 if(empty($update) && empty($updatec)){

 $this->error("请选择操作内容");

 }

 $ids = $\_POST['id'];

 if(empty($ids)){

 $this->error('请选择更新的项目');

 }

 $sorts = $\_POST['sortno'];

 $model = D('Dptarctype');

 switch($updatec){

 case "hidden" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('isshowcon'=>1));

 }

 break;

 case "show" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('isshowcon'=>0));

 }

 break;

 }

 switch($update){

 case "hidden" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('ishidden'=>1));

 }

 break;

 case "show" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('ishidden'=>0));

 }

 break;

 case "sort" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('sortno'=>$sortno));

 }

 }

 $this->cache();

 $this->success('更新成功',U('Dptarctype/index',array('ksid'=>session('ksid'))));

 }

 public function add() {

 $model = D('Dptarctype');

 $data = array(

 'id'=>'',

 'sortno'=>0,

 'model'=>'default',

 'pagesize'=>20,

 'linktype'=>'index',

 'ishidden'=>0,

 'ismenu'=>1,

 'ishot'=>0,

 'viewnum'=>0,

 'praisenum'=>0,

 'sharenum'=>0,

 'hash'=>uniqid2()

 );

 $ksid =I('ksid');

 $upid = I('pid');

 if (!empty($upid)) {

 $type = $model->where("id='$upid'")->find();

 if($type['upid'] > 0){

 $data['code'] = $type['code'];

 }else{

 $data['code'] = $type['code'] . "/index";

 }

 $data['upid'] = $upid;

 $data['model'] = $type['model'];

 $data['linktype'] = $type['linktype'];

 $data['linktpl'] = $type['linktpl'];

 $data['pagesize'] = $type['pagesize'];

 $data['ishidden'] = $type['ishidden'];

 $data['ismenu'] = $type['ismenu'];

 $data['ishot'] = $type['ishot'];

 }

 $this->assign('ksid',$ksid);

 $this->assign('caption','增加科室栏目');

 $this->assign('data', $data);

 $this->prepare($ksid);

 $this->display('edit');

 }

 public function edit() {

 $model = D('Dptarctype');

 $id = I('id');

 $this->checkId($id);

 $data = $model->where("id=$id")->find();

 //显示页面

 $this->assign('caption','修改科室栏目');

 $this->prepare();

 $this->assign('data', $data);

 $this->display();

 }

 public function pcMobileLinks(){

 // $model = D('Arctype');

 // $typelist = $model->where(['id'=>['neq',51]])->order("sortno asc")->select();

 // $typelist = tree\_build($typelist);

 // $this->assign('typeList', $typelist);

 $this->prepare();

 $this->display();

 }

 public function kslink(){

 $ismobile = I('get.ismobile');

 $model = D('Arctype');

 $typelist = $model->where(['id'=>['neq',51]])->order("sortno asc")->select();

 $typelist = tree\_build($typelist);

 $this->assign('typeList', $typelist);

 $this->prepare();

 $data = M('Kslink')->where(['ksid'=>['eq',session('ksid')],'ismobile'=>['eq',$ismobile]])->find();

 if(empty($data)){

 $data['ismobile'] = $ismobile;

 }

 $this->assign('data',$data);

 $this->display();

 }

 public function updateKslink(){

 $model = D('Kslink');

 $id = I('id');

 $data = $model->create();

 $data['ismobile'] = I('ismobile');

 $data['title'] = I('title');

 $data['title1'] = I('title1');

 $data['title2'] = I('title2');

 $data['title3'] = I('title3');

 $data['image'] = I('image');

 $data['image1'] = I('image1');

 $data['image2'] = I('image2');

 $data['image3'] = I('image3');

 $data['link'] = I('link');

 $data['link1'] = I('link1');

 $data['link2'] = I('link2');

 $data['link3'] = I('link3');

 $data['ksid'] = session('ksid');

 if(empty($id)){

 $model->add($data);

 }else{

 $model->where(['id'=>$id])->save($data);

 }

 //更新附件信息

 $key = 'Kslink/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功');

 }

 public function update() {

 $model = D('Dptarctype');

 $id = I('id');

 if ($data = $model->create()) {

 $data['ksid'] = I('ksid');

 $data['ishidden'] = I('ishidden');

 $data['isshowcon'] = I('isshowcon');

 $data['type'] = I('type');

 $upid = $data['upid'];

 if($upid == 0){

 $data['linktpl'] = "/" . I('code');

 }else{

 $data['linktpl'] = "/" . I('code') . "?cid={cid}";

 }

 if (empty($data['title']))

 $this->error('请填写栏目名称');

 // if (empty($data['code']))

 // $this->error('请填写栏目标识');//，标识用于预设模板

 if ($data['sortno']<0)

 $this->error('排序值是必须大于等于0');

 if (empty($data['ishidden']))

 $data['ishidden'] = 0;

 if (empty($data['ismenu']))

 $data['ismenu'] = 0;

 if (empty($data['ishot']))

 $data['ishot'] = 0;

 if ($upid>0) {

 $uptype = $model->where("id=$upid")->find();

 $data['level'] = $uptype['level'] + 1;

 $data['type'] = $uptype['type'];

 $data['isshowcon'] = $uptype['isshowcon'];

 $data['ishidden'] = $uptype['ishidden'];

 if (empty($uptype['upids']))

 $data['upids'] = ",$upid,";

 else

 $data['upids'] = $uptype['upids'] . "$upid,";

 $model->where("id=$upid")->save(array('isleaf'=>0));

 } else {

 $data['level'] = 0;

 $data['upids'] = '';

 if($data['type'] == 3 || $data['type'] == 4){

 $list = M('Dptarctype')->where(['ksid'=>$data['ksid'],'upid'=>['eq',0],'id'=>['notin',$id]])->select();

 foreach($list as $k=>$v){

 if($data['type'] == 3){

 if($v['type'] == 3){

 $this->error("只能有一个专家类别");

 }

 }

 if($data['type'] == 4){

 if($v['type'] == 4){

 $this->error("只能有一个科室介绍类别");

 }

 }

 }

 }

 }

 if(empty($data['type'])){

 $this->error("请选择内容类别");

 }

 if (empty($id)) {

 $data['isleaf'] = 1;

 $model->add($data);

 } else {

 $model->save($data);

 //是否更改了上级栏目

 $oldupid = I('oldupid');

 if ($oldupid!=$upid) {

 $cnt = $model->where("upids LIKE '%,$oldupid,%'")->count();

 if ($cnt<=0) {

 $model->where("id='$oldupid'")->save(array('isleaf'=>1));

 }

 //处理子栏目

 $list = $model->where("upids LIKE '%,$id,%'")->select();

 foreach($list as $v) {

 $save = array();

 if ($oldupid==0 && $upid>0) {

 $save['level'] = array('exp','level+1');

 $save['upids'] = ",$upid".$v['upids'];

 }

 if ($oldupid>0 && $upid==0) {

 $save['level'] = array('exp','level-1');

 $save['upids'] = str\_replace(",$oldupid,", ',', $v['upids']);

 }

 if ($oldid>0 && $upid>0) {

 $save['upids'] = str\_replace(",$oldupids,", ",$upid,", $v['upids']);

 }

 $model->where("id='$v[id]'")->save($save);

 }

 }

 }

 //更新附件信息

 $key = 'Dptarctype/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 //缓存

 $this->cache();

 $this->success('保存成功',U('Dptarctype/index',array('ksid'=>$data['ksid'])));

 } else {

 $this->error($model->getError());

 }

 }

 public function delete() {

 $id = I('get.id');

 $this->checkId($id);

 $model = D('Dptarctype');

 $data = $model->find($id);

 //是否可以删除

 if ($data['isleaf']==0) {

 $this->error('该栏目下有子栏目，不可删除');

 }

 //如果是叶子节点，并且有内容就不可删除

 $cnt = D('Dptarticle')->where(['catid'=>['eq',$id]])->count();

 if ($cnt>0)

 $this->error('该栏目下有内容，不可删除');

 // $typemodel = $data['model'];

 // if ($typemodel && $typemodel!='default') {

 // $acrmodel = M(ucfirst($typemodel));

 // if ($arcmodel) {

 // $cnt = $arcmodel->where("catid=$id")->count();

 // if ($cnt>0)

 // $this->error('该栏目下有内容，不可删除');

 // }

 // }

 $model->deleteOne($id);

 $this->cache();

 $this->success('删除成功',U('Dptarctype/index',array('ksid'=>session('ksid'))));

 }

 //刷新缓存

 private function cache() {

 $this->cacheSvc->cacheData('DptArchiveType');

 }

 //刷新一级网站缓存

 private function cacheTop() {

 $this->cacheSvc->cacheData('ArchiveType');

 }

 private function prepare() {

 // $typelist = $this->cacheSvc->getData('DptArchiveTypeTree');

 $typelist = M('Dptarctype')->where(['ksid'=>session('ksid')])->order("sortno asc")->select();

 $typelist = tree\_build($typelist);

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Dptarticle/index',array('catid'=>$v['id']));

 }

 $modellist = $this->cacheSvc->getData('ArchiveModelMap');

 $linktypelist = $this->cacheSvc->getData('LinkTypeMap');

 $url = U('Dptarctype/index');

 $this->assign('url',$url);

 $this->assign('typetree', $typelist);

 $this->assign('modellist', $modellist);

 $this->assign('linktypelist', $linktypelist);

 }

 // private function getDptType(){

 // $dptTypelist = $this->cacheSvc->getData('DptArchiveTypeTree');

 // }

}<?php

namespace Admin\Controller;

class DptarticleController extends BaseController {

 public function index() {

 $model = D('Dptarticle');

 $ksid = session('ksid');

 $keyword = I('keyword');

 $catid = I('catid');

 $this->dptarctypeSvc = D('Dptarctype','Service');

 if(!empty($catid)){

 $type = $this->dptarctypeSvc->getType($catid);

 if($type['type'] == 3){

 if($type['upid'] == 0){

 $subtitle = [210,211,212];

 }else{

 if($type['title'] == "正高职称"){

 $subtitle = [210];

 }elseif($type['title'] == "副高职称"){

 $subtitle = [211];

 }else{

 $subtitle = [212];

 }

 }

 $where['subtitle'] = ['in',$subtitle];

 }else{

 $catid = $this->dptarctypeSvc->getLeaves($catid);

 $where['catid'] = ['in',$catid];

 }

 }

 $typemap = $this->cacheSvc->getData('DptArchiveTypeMap');

 if (!empty($keyword))

 $where['title'] = ['like',"%$keyword%"];

 $where['ksid'] = ['like',"%$ksid%"];

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("sortno asc,id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 foreach($list as $k=>$v) {

 $list[$k]['catname'] = $typemap[$v['catid']];

 }

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->prepare();

 $this->assign('caption','管理科室文章');

 $this->assign('ksid',$ksid);

 $this->display();

 }

 public function add() {

 $this->prepare();

 $data = array(

 'hash'=> uniqid2(),

 'addtime' => time(),

 'sortno' => 0,

 'viewnum' => 0

 );

 $this->ks();

 $this->assign('data', $data);

 $this->assign('caption','增加科室文章');

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Dptarticle');

 $data = $model->find($id);

 $type = M('Dptarctype')->find($data['catid']);

 $kss = explode(",",$data['ksid']);

 foreach($kss as $k){

 if(!empty($k)){

 $ksIds[$i]['id'] = $k;

 $i ++;

 }

 }

 $this->assign('ksIds',$ksIds);

 $this->assign('type',$type);

 $this->assign('data', $data);

 $this->prepare($ksid);

 $this->ks();

 $this->assign('caption', '编辑科室文章');

 $this->display();

 }

 public function update() {

 $model = D('Dptarticle');

 $data = $model->create();

 $catid = I('post.catid');

 $type = M('Dptarctype')->where(['id'=>$catid])->find();

 if($type['type'] == 3){

 $data['subtitle'] = I('post.subtitle');

 $data['image3'] = I('post.image3');

 $data['image4'] = I('post.image4');

 $data['type'] = $type['type'] ;

 $data['ismove'] = I('post.ismove');

 $ksIds = I('post.ks');

 $nowKsId = session('ksid');

 if($data['ismove'] == 0){

 $bj = 0;

 foreach($ksIds as $k){

 $data['ksid'] = $data['ksid'] . "," . $k;

 if($k == $nowKsId){

 $bj = 1;

 }

 }

 if($bj != 1){

 $data['ksid'] = $nowKsId . $data['ksid'];

 }

 if($data['subtitle'] == ""){

 $this->error("请选择职级");

 }

 if($data['ksid'] == ","){

 $this->error("请选择科室");

 }

 }else{

 $ksMove = I("post.ksMove");

 if(empty($ksMove)){

 $this->error("请选择移动的科室");

 }

 $data['ksid'] = $ksMove;

 $code['title'] = $type['title'];

 $code['ksid'] = $ksMove;

 $code['type'] = 3;

 $ksMoveType = M("Dptarctype")->where($code)->find();

 if(empty($ksMoveType)){

 $this->error("目标科室没有该栏目,请到该科室下添加该栏目");

 }

 $data['catid'] = $ksMoveType['id'];

 }

 }else{

 $data['ksid'] = session('ksid');

 $data['type'] = $type['type'] ;

 $data['subtitle'] = '';

 }

 if(empty($data['title'])){

 $this->error("请填写标题或名称");

 }

 if($data['catid'] == 0){

 $this->error("请选择栏目");

 }

 if ($data['catid']!=73){

 if ($data['isjz']== 1){

 $data['isjz']=0;

 }

 if ($data['ishb']== 1){

 $data['ishb']=0;

 }

 }

 if (empty($data['memo'])) {

 $data['memo'] = msubstr(html2text($data['content']),0,500);

 }

 if (empty($data['addtime'])) {

 $data['addtime'] = time();

 } else {

 $data['addtime'] = strtotime($data['addtime']);

 }

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 //更新附件信息

 $key = 'Dptarticle/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功', $this->getLastUrl());

 }

 public function delete() {

 $id = I('id');

 D('Dptarticle')->deleteOne($id);

 $this->success('删除成功', U('Dptarticle/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择项目');

 $model = D('Dptarticle');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'sort':

 $sorts = $\_POST['sortno'];

 if ($id) {

 foreach($id as $idss) {

 $sortno = $sorts[$idss];

 $model->where("id=$idss")->save(array('sortno'=>$sortno));

 }

 $this->success('更新成功',U('Dptarticle/index'));

 } else {

 $this->error('请选择更新的项目');

 }

 case 'hidden':

 $model->where("id IN($ids)")->save(array('ishidden'=>1));

 break;

 case 'show':

 $model->where("id IN($ids)")->save(array('ishidden'=>0));

 break;

 case 'hot':

 $model->where("id IN($ids)")->save(array('ishot'=>time()));

 break;

 case 'unhot':

 $model->where("id IN($ids)")->save(array('ishot'=>0));

 break;

 case 'delete':

 D('Dptarticle')->deleteBatch($id);

 break;

 }

 $this->success('执行成功',U('Dptarticle/index'));

 }

 protected function prepare() {

 // $tree = $this->cacheSvc->getData('DptArchiveTypeTree');

 $ksid = session("ksid");

 $typelist = M('Dptarctype')->where(['ksid'=>$ksid])->order("sortno asc")->select();

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Dptarticle/index',array('catid'=>$v['id']));

 }

 $typelist = tree\_build($typelist);

 $url = U('Dptarctype/index');

 $this->assign('url',$url);

 $this->assign('typetree', $typelist);

 }

 protected function ks(){

 $model = M('Arctype');

 $ksList = $model->select();

 $ksList = tree\_build($ksList,51);

 $this->assign('ksList',$ksList);

 }

}<?php

namespace Home\Controller;

class DptIndexController extends BaseDptController{

 public function index(){

 $ksid = session('indexKsid');

 $this->checkId($ksid);

 $model = M('Dptarticle');

 // banner

 $list = M('Advtype')->where(['ksid'=>$ksid])->find();

 $this->assign('banner',$list);

 unset($list);

 // 科室介绍

 $type = $this->dptarctypeSvc->getType("DptIntro");

 $catid = $this->dptarctypeSvc->getLeaves($type['id']);

 $code = [

 'catid'=>['eq',$catid],

 'ksid'=>session("indexKsid")

 ];

 $list = $model->where($code)

 ->field("id,title,addtime,memo")

 ->find();

 $this->assign('intro',$list);

 unset($list,$catid,$type,$code);

 // 科室动态

 $type = $this->dptarctypeSvc->getType("DptTrends");

 $catid = $this->dptarctypeSvc->getLeaves($type['id']);

 $code = [

 'catid'=>['in',$catid],

 'ksid'=>session("indexKsid")

 ];

 $list = $model->where($code)

 ->order("addtime desc")

 ->limit(7)

 ->field("id,title,addtime")

 ->select();

 $this->assign('trendsList',$list);

 unset($list,$catid,$type,$code);

 // 科研研讨

 $type = $this->dptarctypeSvc->getType("DptDiscuss");

 $catid = $this->dptarctypeSvc->getLeaves($type['id']);

 $code = [

 'catid'=>['in',$catid],

 'ksid'=>session("indexKsid")

 ];

 $list = $model->where($code)

 ->order("addtime desc")

 ->limit(7)

 ->field("id,title,addtime,catid")

 ->select();

 $this->assign('discussList',$list);

 unset($list,$catid,$type,$code);

 // 特色诊疗

 $type = $this->dptarctypeSvc->getType("DptDiagnosis");

 $catid = $this->dptarctypeSvc->getLeaves($type['id']);

 $code = [

 'catid'=>['in',$catid],

 'ksid'=>session("indexKsid")

 ];

 $list = $model->where(['catid'=>['in',$catid]])

 ->order("addtime desc")

 ->limit(7)

 ->field("id,title,addtime")

 ->select();

 // 特色诊疗推荐

 $data = $model->where(['catid'=>['in',$catid],['ishot'=>['gt',0]]])

 ->order('addtime desc')

 ->find();

 $this->assign('hotdiagnosisList',$data);

 $this->assign('diagnosisList',$list);

 unset($list,$catid,$type,$data);

 // 团队协作

 $type = $this->dptarctypeSvc->getType("DptTeam");

 $catid =

 $children = $this->dptarctypeSvc->getChildren($type['id']);

 foreach($children as $k=>$v){

 if($v['title'] == "团队协作"){

 $catidXz = $v['id'];

 }

 if($v['title'] == "专家团队"){

 $catidEx = $v['id'];

 }

 }

 $list = $model->where(['catid'=>['in',$catidXz]])

 ->order("addtime desc")

 ->limit(7)

 ->field("id,title,addtime")

 ->select();

 // 团队协作推荐

 $data = $model->where(['catid'=>['in',$catidXz],['ishot'=>['gt',0]]])

 ->order('addtime desc')

 ->find();

 $this->assign('hotteamCooList',$data);

 $this->assign('teamCooList',$list);

 unset($list,$catid,$type);

 // 专家介绍

 // $type = $this->dptarctypeSvc->getType("DptTeam/index");

 $list = M('Dptarticle')->where(['catid'=>['in',$catidEx]])

 ->order("id desc")

 ->select();

 $this->assign('expertList',$list);

 unset($list,$catid,$type,$code);

 $this->setTitle($this->setting['SITE\_SEOTITLE']);

 $this->display();

 }

}<?php

namespace Admin\Controller;

class FeedbackController extends BaseController {

 public function index() {

 $model = D('feedback');

 $where = '1=1 ';

 $username = I('username');

 $status = I('status');

 if($username){

 $where .= " AND username LIKE '%{$username}%'";

 $map['username']=$username;

 }

 if($status){

 $where .= " AND status='$status' ";

 $map['status']=I('status');

 }

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 50);

 foreach($map as $key=>$val) {

 $pager->parameter[$key] = urlencode($val);

 }

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->assign('seldata', $seldata);

 $this->assign('caption','合作伙伴管理');

 $this->display();

 }

 public function edit() {

 $id = I('id');

 $model = D('feedback');

 $data = $model->find($id);

 $data['mobile']=M('member')->where('uid =\''.$data['uid'].'\'')->getField('mobile');

 $data['imagess']= explode(",",$data['images']) ;// 分割字符串

 $this->assign('data', $data);

 $this->assign('caption', '编辑咨询留言信息');

 $this->display();

 }

 public function update() {

 $model = D('feedback');

 $data = $model->create();

 if (empty($data['retime'])) {

 $data['retime'] = time();

 } else {

 $data['retime'] = strtotime($data['retime']);

 }

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 $this->success('保存成功', $this->getLastUrl());

 }

 public function delete() {

 $id = I('id');

 D('feedback')->deleteOne($id);

 $this->success('删除成功', U('feedback/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择合作伙伴信息');

 $model = D('feedback');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'delete':

 D('feedback')->deleteBatch($id);

 break;

 }

 $this->success('执行成功',U('feedback/index'));

 }

}<?php

namespace Admin\Controller;

class FriendlinkController extends BaseController {

 public function index() {

 $model = D('Friendlink');

 $count = $model->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->field('\*')

 ->order("id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->assign('caption','管理链接');

 $this->display();

 }

 public function add() {

 $data = array(

 'hash'=> uniqid2(),

 'addtime' => time(),

 );

 $this->assign('data', $data);

 $this->assign('caption','增加链接');

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Friendlink');

 $data = $model->find($id);

 $this->assign('data', $data);

 $this->assign('caption', '编辑链接');

 $this->display();

 }

 public function update() {

 $model = D('Friendlink');

 $data = $model->create();

 if (empty($data['addtime'])) {

 $data['addtime'] = time();

 } else {

 $data['addtime'] = strtotime($data['addtime']);

 }

 if (empty($data['id'])) {

 $data['status'] = 1;

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 //更新附件信息

 $key = 'Friendlink/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功', U('Friendlink/index'));

 }

 public function delete() {

 $id = I('id');

 D('Friendlink')->deleteOne($id);

 $this->success('删除成功', U('Friendlink/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择项目');

 $model = D('Friendlink');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'delete':

 D('Friendlink')->deleteBatch($id);

 break;

 case 'show':

 $model->where("id IN($ids)")->save(array('status'=>1));

 break;

 case 'hide':

 $model->where("id IN($ids)")->save(array('status'=>0));

 break;

 }

 $this->success('执行成功',U('Friendlink/index'));

 }

}<?php

namespace Admin\Controller;

class GivetypeController extends BaseController {

 public function index() {

 $model = D('givetype');

 $list = $model->select();

 $this->assign('caption', '赠品设置');

 $this->assign('list', $list);

 $this->display();

 }

 public function batch() {

 $ids = $\_POST['id'];

 $sorts = $\_POST['sortno'];

 $model = D('givetype');

 if ($ids) {

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('sortno'=>$sortno));

 }

 $this->success('更新成功',U('givetype/index'));

 } else {

 $this->error('请选择要更新的选项');

 }

 }

 public function add() {

 $model = D('givetype');

 $data = array(

 'id'=>'',

 'sortno'=>0

 );

 $this->assign('caption','增加赠品');

 $this->assign('data', $data);

 $this->display('edit');

 }

 public function edit() {

 $model = D('givetype');

 $id = I('id');

 $this->checkId($id);

 $data = $model->where("id=$id")->find();

 //显示页面

 $this->assign('caption','修改赠品');

 $this->assign('data', $data);

 $this->display();

 }

 public function update() {

 $model = M('givetype');

 $id = I('id');

 if ($data = $model->create()) {

 if (empty($id)) {

 $model->add($data);

 } else {

 $model->save($data);

 }

 //缓存

 $this->cache();

 $this->success('保存成功',U('givetype/index'));

 }

 }

 public function delete() {

 $id = I('get.id');

 $this->checkId($id);

 $model = D('givetype');

 $data = $model->find($id);

 //是否可以删除

 if (empty($data)){

 $this->error('该信息已经删除');

 }

 $model->delete($id);

 //缓存

 $this->cache();

 $this->success('删除成功',U('givetype/index'));

 }

 //刷新缓存

 private function cache() {

 $this->cacheSvc->cacheData('Givetype');

 }

}<?php

namespace Admin\Controller;

class HomeController extends BaseController {

 public function \_initialize(){

 parent::leftmenu();

 parent::\_initialize();

 }

 public function index() {

 $ip = $\_SERVER["SERVER\_ADDR"];

 if (empty($ip))

 $ip = gethostbyname($\_SERVER['SERVER\_NAME']);

 $host = empty($\_SERVER['HTTP\_HOST'])?$\_SERVER['HTTP\_HOST']:$\_SERVER['SERVER\_NAME'];

 $data = array(

 'os'=> PHP\_OS,

 'server' => $\_SERVER['SERVER\_SOFTWARE'],

 'host' => $host . '(' . $ip . ')',

 );

 $this->assign('caption','管理首页');

 $this->assign('data', $data);

 $this->display();

 }

}<?php

namespace Admin\Controller;

class IndexController extends BaseController {

 public function index() {

 $menus = load\_config(MODULE\_PATH.'Conf/menu.php');

 $status = session('admin.status');

 if($status != 0){

 $menu[] = $menus[2];

 }else{

 $menu = $menus;

 }

 $admin = session('admin');

 $this->assign('menu', $menu);

 $this->assign('status',$status);

 $this->assign('admininfo', $admin);

 $this->display();

 }

 // private function prepare() {

 // $infoList = $this->cacheSvc->getData('ArchiveTypeTree');

 // $ptyList = $this->cacheSvc->getData('PtyArchiveTypeTree');

 // $dptList = $this->cacheSvc->getData('DptArchiveTypeTree');

 // $kstypeList = D('Department')->where("upid=0")->select();

 // $this->assign('infoList', $infoList);

 // $this->assign('ptyList', $ptyList);

 // $this->assign('dptList', $dptList);

 // $this->assign('ksList', $kstypeList);

 // }

}

<?php

namespace Admin\Controller;

class IndexlinkController extends BaseController {

 // 首页按钮链接

 public function index(){

 $listPc = M('indexlink')->where(['isPc'=>['eq',1]])->order("sortno asc")->select();

 $listMobile = M('indexlink')->where(['isPc'=>['eq',0]])->order("sortno asc")->select();

 $this->prepare();

 $this->assign('listPc',$listPc);

 $this->assign('listMobile',$listMobile);

 $this->display();

 }

 public function add(){

 $data = [

 'id'=>'',

 'sortno'=>0,

 'hash'=>uniqid2()

 ];

 $this->prepare();

 $this->assign('caption','添加链接');

 $this->assign('data',$data);

 $this->display();

 }

 public function edit(){

 $id = I('get.id');

 $this->checkId($id);

 $data = M('Indexlink')->find($id);

 $this->prepare();

 $this->assign('caption','修改链接');

 $this->assign('data',$data);

 $this->display('add');

 }

 public function update(){

 $model = D('Indexlink');

 $id = I('post.id');

 if($data = $model->create()){

 if (empty($data['title'])){

 $this->error('请填写链接名称');

 }

 if (empty($data['link'])){

 $this->error('请填写链接地址');//，标识用于预设模板

 }

 if ($data['sortno']<0){

 $this->error('排序值是必须大于等于0');

 }

 $data['isPc'] = I('post.isPc');

 if(empty($id)){

 $model->add($data);

 }else{

 $model->save($data);

 }

 //更新附件信息

 $key = 'Indexlink/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功',U('Indexlink/index'));

 }else{

 $this->error($model->getError());

 }

 }

 public function delete(){

 $id = I('get.id');

 $this->checkId($id);

 D('Indexlink')->delete($id);

 $this->success('删除成功',U('Indexlink/index'));

 }

 public function batch(){

 $ids = I("post.id");

 if(empty($ids)){

 $this->error('请选择更新的项目');

 }

 $sorts = I('post.sortno');

 foreach($ids as $id){

 $sortno = $sorts[$id];

 D('Indexlink')->where(['id'=>['eq',$id]])->save(['sortno'=>$sortno]);

 }

 $this->success('更新成功',U("Indexlink/index"));

 }

 public function prepare(){

 $model = D('Arctype');

 $typelist = $model->where(['id'=>['neq',51]])->order("sortno asc")->select();

 $typelist = tree\_build($typelist);

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Article/index',array('catid'=>$v['id']));

 }

 $url = U('Arctype/index');

 $this->assign('url',$url);

 $this->assign('typetree', $typelist);

 }

}<?php

namespace Admin\Controller;

class MemberController extends BaseController {

 public function index() {

 $model = M('Member');

 $statuslist = $this->cacheSvc->getData('UserStatusList');

 $teastatus=array('未审核','已审核');

 $isindex=array('','推荐');

 $where = '1=1';

 $nickname = I('nickname');

 $usertype = I('usertype');

 if ($nickname != '') {

 $where .= " AND nickname like '%$nickname%'";

 }

 if ($usertype != '') {

 $where .= " AND usertype=$usertype";

 }

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("uid DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 foreach($list as $k=>$v) {

 if($v['businessid']){

 $list[$k]['sdxqmc']=M('business')->where('id=\''.$v['businessid'].'\'')->getField('xqmc');

 }

 }

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->display();

 }

 public function add() {

 $this->display();

 }

 public function edit() {

 $id = $\_REQUEST['id'];

 $model = M('Member');

 $data = $model->find($id);

 $usertype=array('','普通会员','理财师');

 $data['usertype']=$usertype[$data['usertype']];

 $data['usercat']=explode(';',$data['usercat']);

 $data['nlq']=explode(';',$data['nlq']);

 $this->assign('data', $data);

 $this->display();

 }

 public function save() {

 $model = M('Member');

 $data = $model->create();

 $id = I('id');

 $password = I('password');

 $username = I('username');

 $password = pass\_encode($username,$password);

 $cnt = $model->where("username='$username'")->count();

 if ($cnt>0)

 $this->error('用户名已存在');

 $data['password'] = $password;

 $data['addtime'] = time();

 $data['status'] = 1;

 $model->add($data);

 $this->success('增加成功',U('Member/index'));

 }

 public function update() {

 $model = M('Member');

 $id = I('id');

 $info= $model->find($id);

 $data = $model->create();

 if($info['usertype']==2)

 {

 if(empty($info['nickname']))

 {

 $this->error('理财师姓名必须填写');

 }

 if ($data['isindex']==1)

 {

 echo $data['isindex'];

 $data['teastatus']=1;

 }

 }

 $password = I('password');

 $info = $model->where("uid='$id'")->find();

 if (!empty($password)) {

 $username = $info['username'];

 $password = pass\_encode($username,$password);

 $data['password'] = $password;

 } else {

 unset($data['password']);

 }

 $status = I('status');

 $data['status'] = $status;

 if (!empty($id))

 $model->where("uid=$id")->save($data);

 else

 $model->add($data);

 $this->success('修改成功',U('Member/index'));

 }

 public function delete() {

 $id = I('id');

 $model = M('Member');

 $userinfo=$model->find($id);

 $model->where("uid=$id")->delete();

 M('Connect')->where("userid=$id")->delete();

 $this->success('删除成功');

 }

 public function batch() {

 $id = $\_POST['id'];

 $ids = implode(',', $id);

 $op = $\_POST['op'];

 $model = M('Member');

 switch($op) {

 case 'islahei':

 $model->where("islahei=0 and uid IN($ids)")->save(array('islahei'=>1));

 break;

 case 'noislahei':

 $model->where("islahei=1 and uid IN($ids)")->save(array('islahei'=>0));

 break;

 case 'delete':

 //如果发布过文章，不能删除

 $model->where("uid IN($ids)")->delete();

 break;

 }

 $this->success('操作成功',U('Member/index'));

 }

 public function check() {

 $username = I('param');

 $model = M('Member');

 $cnt = $model->where("username='$username'")->count();

 if ($cnt>0)

 $data = array('info'=>'用户名已存在','status'=>'n');

 else

 $data = array('info'=>'可以使用','status'=>'y');

 $this->ajaxReturn($data);

 }

}

<?php

namespace Admin\Controller;

use Think\Controller;

class MiscController extends Controller {

 public function index() {

 $this->show('');

 }

 public function verify() {

 $verify = new \Think\Verify();

 $verify->fontttf = '4.ttf';

 $verify->length = 4;

 $verify->fontSize = 18;

 $verify->entry('admin');

 }

 public function upload() {

 $updir = C('UPLOAD\_PATH');

 $upurl = C('UPLOAD\_URL');

 $dir = empty($\_REQUEST['dir'])?$\_REQUEST['dir']:'image';

 $upload = new \Think\Upload();// 实例化上传类

 $upload->maxSize = 31457280 \* 2 ;// 设置附件上传大小

 $upload->exts = array('jpg', 'gif', 'png', 'jpeg','doc','docx','xsl','xslx','pdf');// 设置附件上传类型

 $upload->rootPath = $updir;

 $upload->savePath = "$dir/"; // 设置附件上传目录

 $upload->subName = array('date','Ym');

 $info = $upload->uploadOne($\_FILES['imgFile']);

 if(!$info) {// 上传错误提示错误信息

 $data = array('error'=>1,'message'=>$upload->getError());

 }else{// 上传成功

 $url = "$upurl/".$info['savepath'].$info['savename'];

 $data = array('error'=>0,'url'=>$url);

 $att = array(

 'hash' => $\_REQUEST['hash'],

 'name' => $info['name'],

 'savename' => $info['savename'],

 'savepath' => $info['savepath'],

 'ext' => $info['ext'],

 'size' => $info['size'],

 'addtime'=> time()

 );

 M('Attachment')->add($att);

 }

 header('Content-Type:text/html; charset=utf-8');

 die(json\_encode($data));

 }

}<?php

namespace Admin\Controller;

class OrderController extends BaseController {

 public function index() {

 $model = M('Order');

 $where='1=1';

 $status = I('status');

 $orderno= I('orderno');

 $username= I('username');

 if ($status != '') {

 $where .= " AND status=$status";

 }

 if ($orderno != '') {

 $where .= " AND orderno LIKE '%{$orderno}%'";

 }

 if ($username != '') {

 $where .= " AND username LIKE '%{$username}%'";

 }

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->assign('caption','管理订单信息');

 $this->display();

 }

 public function edit() {

 $id = I('id');

 $model = D('Order');

 $data = $model->find($id);

 $this->assign('data', $data);

 $this->assign('caption', '订单详细信息');

 $this->display();

 }

 public function update() {

 $model = D('Order');

 $id = I('id');

 $content = I('content');

 $data=array(

 'id'=>$id,

 'content'=>$content,

 'status'=>1,

 'endtime'=>time()

 );

 $model->save($data);

 $this->success('保存成功', U('Order/index'));

 }

 public function delete() {

 $id = I('id');

 D('Order')->delete($id);

 $this->success('删除成功', U('Order/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择订单');

 $model = D('Order');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'delete':

 $model->where(" id IN($ids) ")->delete();

 break;

 }

 $this->success('执行成功',U('Order/index'));

 }

}

<?php

namespace Admin\Controller;

class PricetypeController extends BaseController {

 public function index() {

 //$typemap = $this->cacheSvc->getData('PriceTypemap');

 $model = D('pricetype');

 $list = $model->select();

 $this->assign('caption', '钱数设置');

 $this->assign('list', $list);

 $this->display();

 }

 public function batch() {

 $ids = $\_POST['id'];

 $sorts = $\_POST['sortno'];

 $model = D('pricetype');

 if ($ids) {

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('sortno'=>$sortno));

 }

 $this->success('更新成功',U('pricetype/index'));

 } else {

 $this->error('请选择要更新的选项');

 }

 }

 public function add() {

 $model = D('pricetype');

 $data = array(

 'id'=>'',

 'sortno'=>0

 );

 $this->assign('caption','增加钱数');

 $this->assign('data', $data);

 $this->display('edit');

 }

 public function edit() {

 $model = D('pricetype');

 $id = I('id');

 $this->checkId($id);

 $data = $model->where("id=$id")->find();

 //显示页面

 $this->assign('caption','修改钱数');

 $this->assign('data', $data);

 $this->display();

 }

 public function update() {

 $model = M('pricetype');

 $id = I('id');

 if ($data = $model->create()) {

 if (empty($id)) {

 $model->add($data);

 } else {

 $model->save($data);

 }

 //缓存

 $this->cache();

 $this->success('保存成功',U('pricetype/index'));

 }

 }

 public function delete() {

 $id = I('get.id');

 $this->checkId($id);

 $model = D('pricetype');

 $data = $model->find($id);

 //是否可以删除

 if (empty($data)){

 $this->error('该信息已经删除');

 }

 $model->delete($id);

 //缓存

 $this->cache();

 $this->success('删除成功',U('pricetype/index'));

 }

 //刷新缓存

 private function cache() {

 $this->cacheSvc->cacheData('PriceType');

 }

}<?php

namespace Admin\Controller;

class ProductController extends BaseController {

 public function index() {

 $model = D('Product');

 $where = '1=1';

 $keyword = I('keyword');

 $username = I('username');

 $catid = I('catid');

 $typemap = $this->cacheSvc->getData('ArchiveTypeMap');

 if (!empty($catid)) {

 $catid = $this->arctypeSvc->getLeaves($catid);

 $where .= " AND catid IN($catid)";

 }

 if (!empty($keyword))

 $where .= " AND proname LIKE '%{$keyword}%'";

 if (!empty($username))

 $where .= " AND username LIKE '%{$username}%'";

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->field('id,proname,catid,addtime,state,isok,username')

 ->where($where)

 ->order("id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 foreach($list as $k=>$v) {

 $list[$k]['catname'] = $typemap[$v['catid']];

 }

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->prepare();

 $this->assign('caption','管理产品');

 $this->display();

 }

 public function edit() {

 $id = I('id');

 $model = D('Product');

 if(!empty($id)){

 $data = $model->find($id);

 $this->assign('data', $data);

 $this->assign('caption', '编辑产品');

 }else{

 $data = array(

 'addtime' => time(),

 'price' => 0,

 'ishot' => 0

 );

 $this->assign('caption', '添加产品');

 }

 $this->prepare();

 $this->display();

 }

 public function update() {

 $model = D('Product');

 $data = $model->create();

 $relation = $\_POST['relation'];

 $relation = ext\_implode(',', $relation);

 $data['relation'] = $relation;

 if (empty($data['memo'])) {

 $data['memo'] = msubstr(html2text($data['content']),0,500);

 }

 if (empty($data['addtime'])) {

 $data['addtime'] = time();

 } else {

 $data['addtime'] = strtotime($data['addtime']);

 }

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 //更新附件信息

 $key = 'Product/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功', $this->getLastUrl());

 }

 public function delete() {

 $id = I('id');

 D('product')->delete($id);

 $this->success('删除成功', U('product/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择项目');

 $model = D('product');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'good':

 $model->where("id IN($ids)")->save(array('isgood'=>time()));

 break;

 case 'ungood':

 $model->where("id IN($ids)")->save(array('isgood'=>0));

 break;

 case 'hot':

 $model->where("id IN($ids)")->save(array('ishot'=>time()));

 break;

 case 'unhot':

 $model->where("id IN($ids)")->save(array('ishot'=>0));

 break;

 case 'delete':

 $ids=explode(',',$ids);

 foreach($ids as $delid) {

 $model->delete($delid);

 D('product')->where("id=$delid")->delete();

 }

 break;

 case 'show':

 $model->where("id IN($ids)")->save(array('status'=>1));

 break;

 case 'hide':

 $model->where("id IN($ids)")->save(array('status'=>0));

 break;

 }

 $this->success('执行成功',U('product/index'));

 }

 protected function prepare() {

 $tree = $this->cacheSvc->getData('ArchiveTypeTree');

 $this->assign('typetree', $tree);

 }

}<?php

namespace Admin\Controller;

class PtyarctypeController extends BaseController {

 public function index() {

 $model = D('Ptyarctype');

 $list = $model->select();

 $list = tree\_build($list);

 $modellist = $this->cacheSvc->getData('PtyarchiveModelMap');

 foreach($list as $k=>$v) {

 $list[$k]['modelname'] = $modellist[$v['model']];

 }

 $this->prepare();

 $this->assign('caption', '党群栏目管理');

 $this->assign('list', $list);

 $this->display();

 }

 public function batch() {

 $update = $\_POST['update'];

 if(empty($update)){

 $this->error("请选择操作内容");

 }

 $ids = $\_POST['id'];

 if(empty($ids)){

 $this->error('请选择更新的项目');

 }

 $sorts = $\_POST['sortno'];

 $model = D('Ptyarctype');

 switch($update){

 case "hidden" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('ishidden'=>1));

 }

 break;

 case "show" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('ishidden'=>0));

 }

 break;

 case "sort" :

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('sortno'=>$sortno));

 }

 }

 $this->success('更新成功',U('Ptyarctype/index'));

 }

 public function add() {

 $model = D('Ptyarctype');

 $data = array(

 'id'=>'',

 'sortno'=>0,

 'model'=>'default',

 'pagesize'=>20,

 'linktype'=>'index',

 'ishidden'=>0,

 'ismenu'=>1,

 'ishot'=>0,

 'viewnum'=>0,

 'praisenum'=>0,

 'sharenum'=>0,

 'hash'=>uniqid2()

 );

 $upid = I('pid');

 if (!empty($upid)) {

 $type = $model->where("id='$upid'")->find();

 if($type['upid'] > 0){

 $data['code'] = $type['code'];

 }else{

 $data['code'] = $type['code'] . "/index";

 }

 $data['upid'] = $upid;

 $data['model'] = $type['model'];

 $data['linktype'] = $type['linktype'];

 $data['linktpl'] = $type['linktpl'];

 $data['pagesize'] = $type['pagesize'];

 $data['ishidden'] = $type['ishidden'];

 $data['ismenu'] = $type['ismenu'];

 $data['ishot'] = $type['ishot'];

 }

 $this->assign('caption','增加党群栏目');

 $this->assign('data', $data);

 $this->prepare();

 $this->display('edit');

 }

 public function edit() {

 $model = D('Ptyarctype');

 $id = I('id');

 $this->checkId($id);

 $data = $model->where("id=$id")->find();

 //显示页面

 $this->assign('caption','修改党群栏目');

 $this->prepare();

 $this->assign('data', $data);

 $this->display();

 }

 public function update() {

 $model = D('Ptyarctype');

 $id = I('id');

 if ($data = $model->create()) {

 $upid = $data['upid'];

 if($upid == 0){

 $data['linktpl'] = "/" . I('code');

 }else{

 $data['linktpl'] = "/" . I('code') . "?cid={cid}";

 }

 if (empty($data['title']))

 $this->error('请填写栏目名称');

 if (empty($data['code']))

 $this->error('请填写栏目标识');//，标识用于预设模板

 if ($data['sortno']<0)

 $this->error('排序值是必须大于等于0');

 if (empty($data['ishidden']))

 $data['ishidden'] = 0;

 if (empty($data['ismenu']))

 $data['ismenu'] = 0;

 if (empty($data['ishot']))

 $data['ishot'] = 0;

 if ($upid>0) {

 $uptype = $model->where("id=$upid")->find();

 $data['level'] = $uptype['level'] + 1;

 if (empty($uptype['upids']))

 $data['upids'] = ",$upid,";

 else

 $data['upids'] = $uptype['upids'] . "$upid,";

 $model->where("id=$upid")->save(array('isleaf'=>0));

 } else {

 $data['level'] = 0;

 $data['upids'] = '';

 }

 if (empty($id)) {

 $data['isleaf'] = 1;

 $model->add($data);

 } else {

 $model->save($data);

 //是否更改了上级栏目

 $oldupid = I('oldupid');

 if ($oldupid!=$upid) {

 $cnt = $model->where("upids LIKE '%,$oldupid,%'")->count();

 if ($cnt<=0) {

 $model->where("id='$oldupid'")->save(array('isleaf'=>1));

 }

 //处理子栏目

 $list = $model->where("upids LIKE '%,$id,%'")->select();

 foreach($list as $v) {

 $save = array();

 if ($oldupid==0 && $upid>0) {

 $save['level'] = array('exp','level+1');

 $save['upids'] = ",$upid".$v['upids'];

 }

 if ($oldupid>0 && $upid==0) {

 $save['level'] = array('exp','level-1');

 $save['upids'] = str\_replace(",$oldupid,", ',', $v['upids']);

 }

 if ($oldid>0 && $upid>0) {

 $save['upids'] = str\_replace(",$oldupids,", ",$upid,", $v['upids']);

 }

 $model->where("id='$v[id]'")->save($save);

 }

 }

 }

 //更新附件信息

 $key = 'Ptyarctype/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 //缓存

 $this->cache();

 $this->success('保存成功',U('Ptyarctype/index'));

 } else {

 $this->error($model->getError());

 }

 }

 public function delete() {

 $id = I('get.id');

 $this->checkId($id);

 $model = D('Ptyarctype');

 $data = $model->find($id);

 //是否可以删除

 if ($data['isleaf']==0) {

 $this->error('该栏目下有子栏目，不可删除');

 }

 //如果是叶子节点，并且有内容就不可删除

 $typemodel = $data['model'];

 if ($typemodel && $typemodel!='default') {

 $acrmodel = M(ucfirst($typemodel));

 if ($arcmodel) {

 $cnt = $arcmodel->where("catid=$id")->count();

 if ($cnt>0)

 $this->error('该栏目下有内容，不可删除');

 }

 }

 $model->deleteOne($id);

 $this->cache();

 $this->success('删除成功',U('Ptyarctype/index'));

 }

 //刷新缓存

 private function cache() {

 $this->cacheSvc->cacheData('PtyArchiveType');

 }

 private function prepare() {

 $typelist = $this->cacheSvc->getData('PtyArchiveTypeTree');

 $modellist = $this->cacheSvc->getData('ArchiveModelMap');

 $linktypelist = $this->cacheSvc->getData('LinkTypeMap');

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Ptyarticle/index',array('catid'=>$v['id']));

 }

 $url = U('Ptyarctype/index');

 $this->assign('url',$url);

 $this->assign('typetree', $typelist);

 $this->assign('modellist', $modellist);

 $this->assign('linktypelist', $linktypelist);

 }

}<?php

namespace Admin\Controller;

class PtyarticleController extends BaseController {

 public function index() {

 $model = D('Ptyarticle');

 $where = '1=1';

 $keyword = I('keyword');

 $catid = I('catid');

 $this->dptarctypeSvc = D('Ptyarctype','Service');

 $typemap = $this->cacheSvc->getData('PtyArchiveTypeMap');

 if (!empty($catid)) {

 $catid = $this->dptarctypeSvc->getLeaves($catid);

 $where .= " AND catid IN($catid)";

 }

 if (!empty($keyword))

 $where .= " AND title LIKE '%{$keyword}%'";

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("sortno asc,id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 foreach($list as $k=>$v) {

 $list[$k]['catname'] = $typemap[$v['catid']];

 }

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->prepare();

 $this->assign('caption','管理党群文章');

 $this->display();

 }

 public function add() {

 $this->prepare();

 $data = array(

 'hash'=> uniqid2(),

 'addtime' => time(),

 'sortno' => 0,

 'viewnum' => 0

 );

 $this->assign('data', $data);

 $this->assign('caption','增加党群文章');

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Ptyarticle');

 $data = $model->find($id);

 $this->assign('data', $data);

 $this->prepare();

 $this->assign('caption', '编辑党群文章');

 $this->display();

 }

 public function update() {

 $model = D('Ptyarticle');

 $data = $model->create();

 if ($data['catid']!=73){

 if ($data['isjz']== 1){

 $data['isjz']=0;

 }

 if ($data['ishb']== 1){

 $data['ishb']=0;

 }

 }

 if (empty($data['memo'])) {

 $data['memo'] = msubstr(html2text($data['content']),0,500);

 }

 if (empty($data['addtime'])) {

 $data['addtime'] = time();

 } else {

 $data['addtime'] = strtotime($data['addtime']);

 }

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 //更新附件信息

 $key = 'Ptyarticle/'.$data['id'];

 D('Attachment')->attachTo($data['hash'],$key);

 $this->success('保存成功', $this->getLastUrl());

 }

 public function delete() {

 $id = I('id');

 D('Ptyarticle')->deleteOne($id);

 $this->success('删除成功', U('Ptyarticle/index'));

 }

 public function batch() {

 $id = $\_POST['id'];

 if (empty($id))

 $this->error('请选择项目');

 $model = D('Ptyarticle');

 $ids = implode($id, ',');

 $op = $\_POST['op'];

 switch($op) {

 case 'sort':

 $sorts = $\_POST['sortno'];

 if ($id) {

 foreach($id as $idss) {

 $sortno = $sorts[$idss];

 $model->where("id=$idss")->save(array('sortno'=>$sortno));

 }

 $this->success('更新成功',U('Ptyarticle/index'));

 } else {

 $this->error('请选择更新的项目');

 }

 case 'hot':

 $model->where("id IN($ids)")->save(array('ishot'=>time()));

 break;

 case 'unhot':

 $model->where("id IN($ids)")->save(array('ishot'=>0));

 break;

 case 'delete':

 D('Ptyarticle')->deleteBatch($id);

 break;

 }

 $this->success('执行成功',U('Ptyarticle/index'));

 }

 protected function prepare() {

 $typelist = $this->cacheSvc->getData('PtyArchiveTypeTree');

 foreach($typelist as $k=>$v){

 $typelist[$k]['url'] = U('Ptyarticle/index',array('catid'=>$v['id']));

 }

 $url = U('Ptyarctype/index');

 $this->assign('url',$url);

 $this->assign('typetree', $typelist);

 }

}<?php

namespace Admin\Controller;

class RoleController extends BaseController {

 public function index() {

 $model = D('Role');

 $where = '1=1';

 $count = $model->where($where)->count();

 $pager = new \Think\Page($count, 20);

 $multi = $pager->show();

 $list = $model->where($where)

 ->order("id DESC")

 ->limit($pager->firstRow.','.$pager->listRows)

 ->select();

 $this->assign('multi', $multi);

 $this->assign('list', $list);

 $this->assign('caption','管理角色');

 $this->display();

 }

 public function add() {

 $data = array(

 'hash' => uniqid2(),

 'sortno' => 0

 );

 $this->assign('data', $data);

 $this->assign('caption','增加角色');

 $this->prepare();

 $this->display('edit');

 }

 public function edit() {

 $id = I('id');

 $model = D('Role');

 $data = $model->find($id);

 $this->assign('data', $data);

 $this->assign('caption', '编辑角色');

 $this->prepare();

 $this->display();

 }

 public function update() {

 $model = D('Role');

 $ruleall = I('ruleall');

 if (empty($ruleall))

 $\_POST['rules'] = implode(',', $\_POST['rules']);

 else

 $\_POST['rules'] = $ruleall;

 $data = $model->create();

 if (empty($data['id'])) {

 $data['id'] = $model->add($data);

 } else {

 $model->save($data);

 }

 $this->cache();

 $this->success('保存成功', U('Role/index'));

 }

 public function delete() {

 $id = I('id');

 D('Role')->deleteOne($id);

 $this->cache();

 $this->success('删除成功', U('Role/index'));

 }

 //刷新缓存

 private function cache() {

 $this->cacheSvc->cacheData('Role');

 }

 private function prepare() {

 $rules = load\_config(MODULE\_PATH.'Conf/rule.php');

 $this->assign('rules', $rules);

 unset($rules);

 }

}<?php

namespace Admin\Controller;

class ServiceController extends BaseController {

 public function index() {

 $model = D('service');

 $list = $model->select();

 $this->assign('caption', '志愿服务设置');

 $this->assign('list', $list);

 $this->display();

 }

 public function batch() {

 $ids = $\_POST['id'];

 $sorts = $\_POST['sortno'];

 $model = D('service');

 if ($ids) {

 foreach($ids as $id) {

 $sortno = $sorts[$id];

 $model->where("id=$id")->save(array('sortno'=>$sortno));

 }

 $this->success('更新成功',U('service/index'));

 } else {

 $this->error('请选择要更新的选项');

 }

 }

 public function add() {

 $model = D('service');

 $data = array(

 'id'=>'',

 'sortno'=>0

 );

 $this->assign('caption','增加志愿服务');

 $this->assign('data', $data);

 $this->display('edit');

 }

 public function edit() {

 $model = D('service');

 $id = I('id');

 $this->checkId($id);

 $data = $model->where("id=$id")->find();

 //显示页面

 $this->assign('caption','修改志愿服务');

 $this->assign('data', $data);

 $this->display();

 }

 public function update() {

 $model = M('service');

 $id = I('id');

 if ($data = $model->create()) {

 if (empty($id)) {

 $model->add($data);

 } else {

 $model->save($data);

 }

 //缓存

 $this->cache();

 $this->success('保存成功',U('service/index'));

 }

 }

 public function delete() {

 $id = I('get.id');

 $this->checkId($id);

 $model = D('service');

 $data = $model->find($id);

 //是否可以删除

 if (empty($data)){

 $this->error('该信息已经删除');

 }

 $model->delete($id);

 //缓存

 $this->cache();

 $this->success('删除成功',U('service/index'));

 }

 //刷新缓存

 private function cache() {

 $this->cacheSvc->cacheData('service');

 }

}<?php

namespace Admin\Controller;

class SettingController extends BaseController {

 public function \_initialize(){

 parent::leftmenu();

 parent::\_initialize();

 }

 private function edit() {

 $typeModel = M('Settype');

 $types = $typeModel->select();

 $type = I('type');

 if (empty($type))

 $type="base";

 $model = M('Setting');

 $list = $model->where("catid='$type'")->order('sortno')->select();

 //$feee= '网站底部右侧信息';

 //print\_r ($list);

 //$model->where("var='SITE\_CLOSEREASON'")->save(array('title'=>$feee));

 $this->assign('type', $type);

 $this->assign('types', $types);

 $this->assign('list', $list);

 $this->display('index');

 }

 public function \_empty($name) {

 if (empty($name) || $name=='index')

 $name = 'base';

 $this->edit($name);

 }

 public function update() {

 $type = $\_POST['type'];

 $model = M('setting');

 $list = $model->where("catid='$type'")->getField('var', true);

 foreach($list as $v) {

 $val = trim($\_POST[$v]);

 if ($val != '') {

 $model->where("var='$v'")->save(array('val'=>$val));

 }

 }

 $this->cache();

 $this->success('保存成功',U('Setting/index',array('type'=>$type)));

 }

 private function cache() {

 $this->cacheSvc->cacheData('Setting');

 }

}