

附件 1：论文

《民族文汇》（教学与实践）用稿通知

马栋之，张娜 同志：

您的稿件《基于微信的继续教育信息化公共服务移动平台实践研究》经审核，符合用稿要求，将在本刊 2022 年 33 期发表。（国内刊号 CN65—1217/I，国际刊号 ISSN:1009-8798）文章刊出后，赠送当期杂志一册。

请勿一稿多投！

《民族文汇·教学与实践》编辑部



2022年11月10日

编辑部

重点提示：

1. 为了保证论文的刊发质量，我编辑部在不违背作者论文原意前提下，有权对论文进行文字方面的修改。
2. 作者文章文责自负，本编辑部不承担任何连带责任。

基于微信的继续教育信息化公共服务移动平台实践研究

山东轻工职业学院 马栋之 张娜

摘要：基于微信移动平台在继续教育信息化公共服务中的服务功能、需求分析、应用现状，提出了智慧教育是未来教育发展的演化方向。数字大脑赋能+智慧校园标准是引领教育信息化融合教育变革最后一公里的特色路径。

关键词：学习型社会；智慧教育生态；融合创新；技术赋能

微信是腾讯公司推出的一个即时通讯应用程序，随着微信功能的不断发展和完善，微信作为一个强大的云平台，为社会提供了解决方案，并以其巨大影响力、庞大用户群参与到一些社会公共事务中。本文基于微信的继续教育信息化公共服务移动平台实践与应用方面进行一定的探索，旨在教学人本化、功能集成化、学习虚拟化、实验实训可视化、管理自动化上深入地运用现代信息技术来促进继续教育改革与发展，为职业院校继续教育信息化服务提供新的思路与实践路径。

一. 微信移动平台应用于继续教育信息化公共服务功能

根据“互联网+教育”时代微信公众平台应用现状和继续教育信息化服务工作的特点，为提升职业院校继续教育信息化公共服务移动平台应用水平、服务质量、工作效率，结合微信互动性强，即时沟通，传播速度快，影响面广，形象拟人化的特点，为继续教育信息化公共服务和应用打开新空间与新视角。引入微信后，微信将成为继续教育信息化公共服务平台统一入口，微信的开发模式将系统的功能和服务引入到特定的服务器来实现。可以认为，引入微信是对已有微信移动平台系统前端的部分改造，新的继续教育信息化公共服务移动平台可以方便快捷地与现有的微信移动平台系统进行无缝衔接。

建立继续教育信息化公共服务移动平台，以“互联网+”理念整合各级各类教育管理应用系统，形成面向各种教育主题的教育数据集合^[1]。这正好符合国家教育信息化 2.0 提出的建立国家数字教育资源公共服务体系。微信本身具有便利的互动性，信息推送迅速实时更新的特性，作为参与教育的部分环节来说，微信是一个开放平台，应用开发者可通过微信开放接口接入第三方应用，还可以将应用的 LOGO 放入微信附件栏中，让微信用户方便地在会话中调用第三方应用进行。微信的用户却一定是真实的、私密的、有价值的，可随时随地提供信

息和服务，微信公众平台无论是点对点还是点对面，信息达到率极高，并实现和特定群体的文字、图片、语音的全方位沟通与互动。微信特有的对讲功能，使得社交不再限于文本传输，而是图片、文字、声音、视频的富媒体传播形式，更加便于分享用户的所见所闻。同时用户除了使用聊天功能，还可以通过微信的“朋友圈”功能，通过转载、转发及“@”功能来将内容分享给目标人群。

在线教育的载体就是微信等移动教学平台，通过这些平台的相关功能实现教育教学。现在社会面当中的教学平台有很多，而且大部分都能支持多个平台对接，比如说微信的移动教学平台，职业院校和一线老师不仅可以打造大数据个性化教学系统，为学校和老师提供家长互动、日常管理、教学备课、实现班级常态化管理等服务，而且会通过随堂测试、期中期末考试、不定期测试来了解学生的学习情况，同时还能够监督学生通过测试，这也是让家长 and 学员更加放心的一种实用的功能。对于在线教育培训机构来说，教学是很主要的，线上营销也是非常重要的，营销培训也是其中很重要的一个环节。因为要面对来自全国各地的学员，很多教育培训机构都会推出试听课程作为引流课来对外推广。为了实现良好的营销转化效果，可能需要一些比较实用的功能，比如说各大平台的流量承接，比如说可以打通各个平台，与各大平台的教学培训系统建立链接。EduSoho 教培系统这是一个可以实现营销转化效果的功能。所以对于移动教学平台的功能彰显非常高的用武之地。

二. 微信移动平台在继续教育信息化公共服务中的需求分析

1. 具有多样的传播方式及持久的影响力的“微平台”已成为人们工作、生活、学习的重要工具。利用微信移动平台应用在继续教育信息化公共服务中，开展“一师一优课，一课一名师”活动，提升教师信息技术应用能力和信息化教学能力，借助智能工具和网络手段解决课堂教学瓶颈问题，利用技术创新教学，加深对信息化与学科教学融合的深层次理解，学以致用，结合课堂教学的“真”问题，探索实践微信平台运营的“四梁八柱”，让广大师生参与建设，强化其认同感、荣誉感、责任感，探索信息技术在创新孵化、STEAM 教育、创客教育等教育教学新模式中的应用，引领学生掌握扎实的基础和知识和促进高级思维的发展与成长，逐步形成创新课程体系^[2]。

2. 践行党的二十大会议关于教育方针的精神和要义，引导树立正确的国家观、民族观、历史观、文化观，实现知识传授、价值塑造和能力培养的多元统一，结合思政教学相关的内

容和教学规划，打造分支独立的思政教育平台，把政治认同、国家意识、文化自信、人格养成等思想政治教育导向与固有的知识、技能传授有机融合，积极探索信息技术与高校思政课教学深度融合。

3. 以共享为核心，搭建微信云课堂资源库。以数据库为载体，加快平台资源的影像化、数字化和网络化。加强院校之间的交流合作，互相学习切磋，资源共享，学习和借鉴互联网上知识类账号的成功经验，与传承传统文化、弘扬社会主义价值观念的微信其他平台合作，建立在线教学联盟，整合教育资源，破除单打独斗，推动高校优质师资和资源的互联互通、共建共享，把职业院校打造成为新时代工匠的培育引领之地、成长向往之地、技能创新之地^[3]。

三. 微信移动平台在继续教育信息化公共服务中的应用现状与创新

1. 构建学习型社会的有效途径是建立数字化、网络化、个性化的教育体系，在新型的教育生态下，原本封闭的教育大门走向开放，教育得到了自我进化，人人既是教育的消费者，也可能成为教育的生产者；人人能够获取和使用知识。人人也能够共享知识，逐渐形成教育共同体，促进教育共同进步和健康、规范、有序发展，为继续教育综合改革打开了新空间与新视角。

2. 当前数字技术与教育的融合仍存在着诸多挑战，存在着信息化程度不高，教学方式交互性不足，信息化管理效果不佳等问题。要改变这一现状，教师作为教学主体，应该突破自己的思维定式，进一步提升教师的信息素养，思考如何把新技术和教学结合在一起，利用技术读懂学生，利用大数据了解学生对知识的掌握程度、学习特点、认知特征、学习偏好等，以此实现因材施教，培育教育服务新业态^[4]。

3. 打造智慧教育繁荣生态，构建基于云网融合的教学体系。智慧教育即教育信息化，智慧教育是未来教育发展的演化方向，是数字大脑赋能+智慧校园标准引领的特色路径。聚焦“应用+场景”，智能化撬动教学变革，技术赋能课堂变革，是教育信息化融合教育变革的最后一公里。打造数字观课、数据研修等应用场景。以融合创新赋能教师专业化发展，为教育插上“智慧”的翅膀，是实现教育跨越式发展的必然选择。

参考文献：

1. 王艳歌. 微信公众平台支持下的适时教学模式研究 ——以“Flash 动画制作”课程为例[J]. 办公自动化. 2018, (12).
2. 李利平. 高职教育专业教学资源库建设的改革思考[J]. 中国高教研究, 2011(6) :90-91.
3. 张秀梅. 立足一线谈微课在服装专业教学中的影响力[J]. 新课程：教研版, 2015(6) :92-93.
4. 王玥. 基于虚拟仿真实验的服装教学资源建设与应用 [J]. 教育观察, 2020, 9(21) :105-107.

附件 2：云平台部分代码

```
<?php
namespace Admin\Controller;
class PublicController extends \Think\Controller {
    public function login($username = null, $password = null, $verify = null) {
        $username = safe ( $username, 'text' );
        $password = safe ( $password, 'text' );

        $config = S ( 'DB_CONFIG_DATA' );
        if (! $config) {
            $config = D ( 'Config' )->lists ();
            S ( 'DB_CONFIG_DATA', $config );
        }
        C ( $config );

        if (IS_POST) {

            if (C ( 'WEB_SITE_VERIFY' ) && ! check_verify ( $verify )) {
                $this->error ( '验证码输入错误! ');
            }

            $User = D ( 'Common/User' );
            $uid=$User->login ( $username, $password, 'admin_login' );
            if ($uid) {
                // 登录用户
                session('[start]');
                session('uid',$uid);

                $this->success ( '登录成功! ', U ( 'User/index' ) );
            }
            else {
                $this->error ( $User->getError () );
            }
        }
        else {
            if (is_login ()) {
                $this->redirect ( 'User/index' );
            }
            else {
                $this->display ();
            }
        }
    }
}
```

```

    }

    /* 退出登录 */
    public function logout() {
        if (is_login ()) {
            D ( 'Common/User' )->logout ();
            session ( '[destroy]' );
            $this->success ( '退出成功! ', U ( 'login' ) );
        } else {
            $this->redirect ( 'login' );
        }
    }
    public function verify() {
        $verify = new \Think\Verify ();
        $verify->entry ( 1 );
    }
}

```

<?php

```

namespace Admin\Controller;
class UserController extends AdminController {
    public function index() {
        $uid=session('uid');
        $user=M('user')->where(array('uid'=>$uid))->field('uid,trueusername,usertype')->find();
        $this->assign('user',$user);
        $usertype=$user['usertype'];
        $this->assign('usertype',$usertype);

        if (is_numeric ( $nickname )) {
            $map ['uid|nickname'] = array (
                intval ( $nickname ),
                array (
                    'like',
                    '%'. $nickname . '%'
                ),
                '_multi' => true
            );
        } else {
            $map ['nickname'] = array (
                'like',
                '%'. ( string ) $nickname . '%'
            );
        }
    }
}

```

```

$list = $this->lists ( 'User', $map );
int_to_string ( $list );
$this->assign ( '_list', $list );

$this->meta_title = '用户信息';
$this->display ();
}

//修改用户信息
public function modifyInfo(){
    if(!IS_POST){
        //需要修改的用户
        $uid=I('userid',0,intval);

        $user=M('user')->where(array('uid'=>$uid))->field('uid,trueusername,nickname,mobile')->find();

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

        //当前登录用户
        $current_userid=session('uid');

        $current_user=M('user')->where(array('uid'=>$current_userid))->field('uid,usertype')->find();
        $this->assign('current_user',$current_user);
        $this->display();
    }
    else{
        $uid=I('uid',0,intval);
        $data['uid']=I('uid',0,intval);
        $data['usertype']=I('usertype');
        $data['trueusername']=I('trueusername');
        $data['nickname']=I('nickname');
        $data['email']=I('email');

        $data['qq']=I('qq');
        $data['tel']=I('tel');
        $data['mobile']=I('mobile');

        if(!$data['trueusername']){
            $this->error('网点名称 不能为空');
        }
        if(!$data['nickname']){
            $this->error('网点登录名 不能为空');
        }
    }
}

```

```

        if(!$data['contact']){
            $this->error('联系人 不能为空');
        }
        if(!$data['mobile']){
            $this->error('手机号码 不能为空');
        }
        $res=D('User')->modifyInfoHandle($uid,$data);
        if($res['status']){
            $this->success('修改成功',U('User/index'));
        }
        else{
            $this->error('修改失败');
        }
    }
}

public function addUser() {
    if (IS_POST) {
        $post=I('post.');
        if ($post['password'] != $post['repassword']) {
            $this->error ( '密码和重复密码不一致! ');
        }

        $uid = D ( 'Common/User' )->register ( $post['truename'],
        $post['nickname'],$post['password'], $post['email'],$post['qq'],$post['mobile'] );
        if ($uid>0) {
            $this->success ( '用户添加成功! ', U ( 'User/index' ) );
        }
        else {
            $this->error ( '用户添加失败! ');
        }
    }
    else {
        $this->meta_title = '新增用户';
        $this->display ();
    }
}

public function changeStatus($method = null) {
    $id = array_unique ( ( array ) I ( 'id', 0 ) );
    if (in_array ( C ( 'USER_ADMINISTRATOR' ), $id )) {
        $this->error ( "不允许对超级管理员执行该操作!" );
    }
}

```

```

$Sid = is_array ( $Sid ) ? implode ( ',', $Sid ) : $Sid;
if ( empty ( $Sid ) ) {
    $this->error ( '请选择要操作的数据!' );
}
$map ['uid'] = array (
    'in',
    $Sid
);
switch ( strtolower ( $method ) ) {
    case 'forbiduser' :
        $this->forbid ( 'User', $map );
        break;
    case 'resumeuser' :
        $this->resume ( 'User', $map );
        break;
    case 'deleteuser' :
        $this->delete ( 'User', $map );
        break;
    case 'audit_1' :
        $this->audit ( $map, 1, $Sid );
        break;
    case 'audit_0' :
        $this->audit ( $map, 0, $Sid );
        break;
    default :
        $this->error ( '参数非法' );
}
}

private function showRegError($code = 0) {
    switch ($code) {
        case - 1 :
            $error = '用户名长度必须在 16 个字符以内! ';
            break;
        case - 2 :
            $error = '用户名被禁止注册! ';
            break;
        case - 3 :
            $error = '用户名被占用! ';
            break;
        case - 4 :
            $error = '密码长度必须在 6-30 个字符之间! ';
            break;
    }
}

```

```

        case - 5 :
            $error = '邮箱格式不正确! ';
            break;
        case - 6 :
            $error = '邮箱长度必须在 1-32 个字符之间! ';
            break;
        case - 7 :
            $error = '邮箱被禁止注册! ';
            break;
        case - 8 :
            $error = '邮箱被占用! ';
            break;
        case - 9 :
            $error = '手机格式不正确! ';
            break;
        case - 10 :
            $error = '手机被禁止注册! ';
            break;
        case - 11 :
            $error = '手机号被占用! ';
            break;
        default :
            $error = '未知错误';
    }
    return $error;
}

function menu() {
    $this->disply ();
}
}

```

```

<?php
namespace Common\Model;
use Think\Model;
class UserModel extends Model {
    protected $_validate = array (
        array (
            'nickname',
            '1,16',
            '昵称长度为 1-16 个字符',
            self::EXISTS_VALIDATE,
            'length'

```

```

    ),
    array (
        'nickname',
        ",
        '昵称被占用',
        self::EXISTS_VALIDATE,
        'unique'
    )
);

protected $_auto = array (
    array (
        'password',
        'think_weiphp_md5',
        self::MODEL_BOTH,
        'function'
    ),
    array (
        'reg_time',
        NOW_TIME,
        self::MODEL_INSERT
    ),
    array (
        'reg_ip',
        'get_client_ip',
        self::MODEL_INSERT,
        'function',
        1
    ),
    array (
        'update_time',
        NOW_TIME
    )
);

public function lists($status = 1, $order = 'uid DESC', $field = true) {
    $map = array (
        'status' => $status
    );
    return $this->field ( $field )->where ( $map )->order ( $order )->select ();
}

public function register($username,$nickname, $password, $email,$qq,$mobile) {
    $data = array (
        'nickname' => $nickname,

```

```

        'truename' => $username,
        'login_name' => $username,
        'password' => $password,
        'email' => $email,
        'qq' => $qq,
        'mobile' => $mobile,
        'is_audit' => 1
    );

    empty ( $mobile ) || $data ['mobile'] = $mobile;

    /* 添加用户 */
    if ($this->create ( $data )) {
        $uid = $this->add ();
        return $uid ? $uid : 0; // 0-未知错误, 大于 0-注册成功
    } else {
        return $this->getError (); // 错误详情见自动验证注释
    }
}

function addUser($data) {
    $data = $this->_deal_nickname ( $data );
    $res = $this->add ( $data );
    return $res;
}

public function login($username, $password, $from = 'user_login', $type = 1) {
    $username = safe ( $username, 'text' );
    $password = safe ( $password, 'text' );

    /* 获取用户数据 */
    $user = $this->field ( true )->where ( $map )->find ();

    if (! $user) {
        unset ( $map );
        $map ['login_name'] = $username;
        $user = $this->field ( true )->where ( $map )->find ();
    }

    if (is_array ( $user ) && $user ['status']) {
        if (think_weiphp_md5 ( $password ) === $user ['password']) {
            action_log ( $from, 'user', $user ['uid'], $user ['uid'] );
            // 登录成功, 返回用户 ID
            return $user ['uid'];
        }
    }
}

```

```

    }
    else {
        $this->error = '密码错误! ';
        return false;
    }
}
else {
    $this->error = '用户不存在或已被禁用! '; // 应用级别禁用
    return false;
}
}
}

```

```

public function logout() {
    $token = get_token ();
    session ( 'manager_menu_default', null );
    session ( 'mid', null );
    session ( 'user_auth', null );
    session ( 'user_auth_sign', null );
    session ( 'token', null );
    session ( 'openid_' . $token, null );
    session ( 'manager_id', null );
}

```

```

public function reg($data){
    if($data['userpwd2']!=$data['userpwd'] || "=="$data['userpwd2']){
        $info=array(
            'status'=>0,
            'info'=>'再次密码不一致， 或者根本没有输入密码',
        );
        return $info;
    }
    if(!preg_match("/^[a-zA-Z_]{4,20}$/",$data['username'])){
        $info=array(
            'status'=>0,
            'info'=>'用户名的长度应该为 4 至 20 位， 并且是英文， 可以包括——',
        );
        return $info;
    }
    if($this->where(array('nickname'=>$data['username']))->select()){
        $info=array(
            'status'=>0,
            'info'=>'这个用户名已经注册过了， 请换一个试试',
        );
    }
}

```

```

    );
    return $info;
}

$data_arr['nickname']=$data['username'];
$data_arr['upwd']=$data['userpwd'];
$data_arr['last_login_ip']=$_SERVER['REMOTE_ADDR'];
$data_arr['last_login_time']=time();

if($this->add($data_arr) > 0){
    $info=array(
        'status'=>1,
        'info'=>'注册成功',
    );
    return $info;
}
else{
    $info=array(
        'status'=>0,
        'info'=>'注册失败',
    );
    return $info;
}
}

public function mylogin($user_name,$user_pwd,$user_pwd2){
    if(!$user_pwd){
        $info=array(
            'status'=>0,
            'info'=>'必须填写密码'
        );
        return $info;
    }
    if(!$user_name){
        $info=array(
            'status'=>0,
            'info'=>'必须填写用户名'
        );
        return $info;
    }

    if($user_pwd!=$user_pwd2){
        $info=array(
            'status'=>0,

```

```

        'info'=>'注册成功'
    );
    return $info;
}
$user_pwd=md5($user_pwd);
$user_info=$this->where(array('nickname'=>$user_pwd,'upwd'=>$user_pwd))->find();
}

```

//更改信息

```

public function modifyInfoHandle($uid,$data){
    if($uid==""){
        $info=array(
            'status'=>0,
            'info'=>'请选择要修改的用户'
        );
        return $info;
    }
}

```

```

if(M('user')->save($data)){
    $info=array(
        'status'=>1,
        'info'=>'修改成功'
    );
    return $info;
}

```

```

else{
    $info=array(
        'status'=>0,
        'info'=>'修改失败'
    );
    return $info;
}
}

```

```

function getUserInfoByOpenid($openid, $update = false) {
    $map ['token'] = get_token ();
    $map ['openid'] = $openid;
    $uid = M ( 'public_follow' )->where ( $map )->getField ( 'uid' );
    return $this->getUserInfo ( $uid, $update );
}

```

```

function updateInfo($uid, $save) {
    if (empty ( $uid ))

```

```

        return false;

        $save = $this->_deal_nickname ( $save );

        $map ['uid'] = $uid;
        $res = $this->where ( $map )->save ( $save );
        if ($res) {
            $this->getUserInfo ( $uid, true );
        }
        return $res;
    }

public function updateUserFields($uid, $password, $data) {
    if (empty ( $uid ) || empty ( $password ) || empty ( $data )) {
        $this->error = '参数错误! ';
        return false;
    }

    // 更新前检查用户密码
    if (! $this->verifyUser ( $uid, $password )) {
        $this->error = '验证出错： 密码不正确! ';
        return false;
    }

    $data = $this->_deal_nickname ( $data );

    // 更新用户信息
    $data = $this->create ( $data );
    if ($data) {
        $res = $this->where ( array (
            'uid' => $uid
        ))->save ( $data );
        $this->getUserInfo ( $uid, true );
        return $res;
    }
    return false;
}

//修改用户密码
public function updateUserPwd($uid,$password){
    if(empty($uid)||empty($password)){
        return false;
    }
}

```

```

$res=$this->where(array('uid'=>$uid))->setField('password',think_weiphp_md5($password))
;
    return $res;
}

protected function verifyUser($uid, $password_in) {
    $map ['uid'] = $uid;
    $password = $this->where ( $map )->getField ( 'password' );
    if (think_weiphp_md5 ( $password_in ) === $password) {
        return true;
    }
    return false;
}

function _deal_nickname($data, $type = 0) {
    if (isset ( $data ['nickname'] )) {
        $data ['nickname'] = deal_emoji ( $data ['nickname'], $type );
    }

    return $data;
}

function clear($uid) {
    $this->getUserInfo ( $uid, true );
}
}

```

```

<?php
namespace Admin\Controller;
class ConfigController extends AdminController {
    public function group() {
        $id      = I('get.id',1);
        $type    = C('CONFIG_GROUP_LIST');
        $list    = M('config')->where(array('status'=>1,'group'=>$id))->field('id,name,title,extra,value,remark,type')
->order('sort')->select();
        if($list) {
            $this->assign('list',$list);
        }
        $this->assign('id',$id);
        $this->meta_title = $type[$id].'设置';
        if ($id == 6) {
            $more_button [] = array (
                'title' => '测试邮件',

```

```

        'url' => U ( 'admin/Config/test_mail' )
    );
    $this->assign ( 'more_button', $more_button );
}
$this->display();
}
}

```

```

<?php
namespace Admin\Controller;
class UpdateController extends AdminController {
    function delcache() {
        $cache_dirs = RUNTIME_PATH;
        $this->rmdirr ( $cache_dirs );

        @mkdir ( $cache_dirs, 0777, true );
        $this->display ();
    }
}

```

```

<?php
namespace Admin\Controller;
use Think\Controller;

class CategoryController extends AdminController{
    public function index(){
        $map['status']=array('neq',-1);
        $data = $this->_get_data ( $map );

        //分页操作
        $request    = (array)I('request.');
        $total      = $data? count($data) : 1 ;
        $listRows   = C('LIST_ROWS') > 0 ? C('LIST_ROWS') :10;
        $page       = new \Think\Page($total, $listRows, $request);
        $voList     = array_slice($data, $page->firstRow, $page->listRows);
        $p          = $page->show();
        $this->assign('list_data', $voList);
        $this->assign('_page', $p? $p: "");
        // 记录当前列表页的 cookie
        Cookie('__forward__',$_SERVER['REQUEST_URI']);
        //分页操作结束
    }
}

```



```

    $this->display('index');
}

public function _get_data($map) {
    $cate=M ( 'category' );
    $list = $cate->field ( true )->where ( $map )->select ();

    $one_arr=$cate->where('level=1')->field(true)->select();
    $two_arr=$cate->where('level=2')->field(true)->select();
    $three_arr=$cate->where('level=3')->field(true)->select();
    $data=array();
    foreach($two_arr as $twokey=>$two){
        $temp_two=array();
        foreach($three_arr as $threekey=>$three){
            if($three['pid']==$two['id']){
                $temp_two[]=$three;
                unset($three_arr[$threekey]);
            }
        }
        $temp_two=array_merge($two,$temp_two);
    }
    return $data;
}

//添加
public function add(){
    if(!IS_POST){
        $id=I('id',0,intval);
        $type=I('type',0,intval);
        $level=I('level',0,intval);
        if($level==0){
            $level=1;
        }
        $cate = M('category');
        $map['type']=array('eq',$type);
        if($id>0){
            $map['id']=array('eq',$id);
            $title=$cate->where($map)->getField('title');
            $this->assign('title',$title);
        }
        if($id==0){
            $categories = $cate->where($map)->field(true)->select();
            $categories = D('Common/Tree')->toFormatTree($categories);
            $categories = array_merge(array(0=>array('id'=>0,'title_show'=>'顶级菜

```

```

单')), $categories);
        $this->assign('categories', $categories);
    }
    $this->assign('type',$type);
    $this->assign('id',$id);
    $this->assign('level',$level);
    $this->display();
}
else{
    //echo header("Content-Type: text/html; charset=utf-8");
    $data=I('post.');
    $data['subtime']=time();

    if(!$data['title']){
        $this->error('标题不能为空');
    }

    $res=D('Category')->addHandle($data);
    if($res['status']){
        if($data['type']==1){
            $this->success($res['info'],U('Category/video'));
        }
        if($data['type']==2){
            $this->success($res['info'],U('Category/wenku'));
        }
        if($data['type']==3){
            $this->success($res['info'],U('Category/test'));
        }
    }
    else{
        $this->error($res['info']);
    }
}
}
}

```

//确定几级分类

```

public function level($pid=0,$i=1){
    $cate=M('category');

    if($pid!=0){
        $i++;

        $map['id']=array('eq',$pid);
        $grand_id=$cate->where($map)->getField('pid');
    }
}

```

```

        if($grand_id!=0){
            $this->level($grand_id,$i);
        }
    }
    return $i;
}

//编辑分类
public function edit(){
    if(!IS_POST){
        $id=I('id',0,intval);
        $cate=M('category');
        $category=$cate->where(array('id'=>$id))->field(true)->find();
        $this->assign('category',$category);

        $map['id']=array('eq',$category['pid']);
        $pcate=$cate->where($map)->field(true)->find();
        $this->assign('pcate',$pcate);

        $this->display();
    }
    else{
        $id=I('id',0,intval);
        $data=I('post.');
```

//删除分类

```
public function del(){
```

```

        $id=I('id',0,intval);
        $res=D('Category')->delHandle($id);
        if($res['status']){
            $this->success($res['info'],U('Category/video'));
        }
        else{
            $this->error($res['info']);
        }
    }
}
?>

```

```

<?php
namespace Admin\Model;
use Think\Model;
class CategoryModel extends Model {
    //新增 Category 处理
    public function addHandle($data){
        $id=$this->add($data);
        if($id){
            $info=array(
                'status'=>1,
                'info'=>'添加成功'
                //'classid'=>$class_id
            );
            return $info;
        }
        else{
            $info=array(
                'status'=>0,
                'info'=>'添加失败'
            );
            return $info;
        }
    }
}

```

```

//编辑处理
public function editHandle($id,$data){
    if($id==0 || $id==""){
        $info=array(
            'status'=>0,
            'info'=>'请选择要操作的项目'
        );
    }
}

```

```

        );
    }
    if($this->where(array('id'=>$id))->save($data)){
        $info=array(
            'status'=>1,
            'info'=>'修改成功'
        );
        return $info;
    }
    else{
        $info=array(
            'status'=>0,
            'info'=>'修改失败'
        );
        return $info;
    }
}

//删除
public function delHandle($id){
    if($id==0 || $id==""){
        $info=array(
            'status'=>0,
            'info'=>'请选择要操作的项目'
        );
    }

    if($this->where(array('id'=>$id))->setField('status',-1)){
        $info=array(
            'status'=>1,
            'info'=>'删除成功'
        );
        return $info;
    }
    else{
        $info=array(
            'status'=>0,
            'info'=>'删除失败'
        );
        return $info;
    }
}
}
}

```

```

<?php
namespace Home\Controller;
class IndexController extends HomeController {
    // 系统首页
    public function index() {
        //session('uid',null);
        $uid=session('uid');
        $map_u['status']=array('eq',1);
        $map_u['uid']=array('eq',$uid);
        $user=M('front_user')->where($map_u->field(true)->find());
        $this->assign('user',$user);

        //1、视频 2、文本 3、考试
        $map['type']=array('eq',1);
        $map['status']=array('neq',-1);

        $cate=M('category');
        $list = $cate->field(true)->where($map)->select();

        $map['level']=array('eq',1);
        $one_arr=$cate->where($map)->field(true)->select();
        $this->assign('one_arr',$one_arr);

        $map['level']=array('eq',2);
        $two_arr=$cate->where($map)->field(true)->select();
        $this->assign('two_arr',$two_arr);

        $map['level']=array('eq',3);
        $three_arr=$cate->where($map)->field(true)->select();
        $this->assign('three_arr',$three_arr);

        unset($map['level']);
        //视频分类
        $cate=M('category');
        $map['pid']=array('eq',0);
        $map['type']=array('eq',1);
        $map['status']=array('eq',1);
        $video_cate=$cate->where($map)->field(true)->select();
        $this->assign('video_cate',$video_cate);

        $vi=M('video');
        $map1['status']=array('eq',1);
        $video=$vi->where($map1)->field(true)->select();
    }
}

```

```

$this->assign('video',$video);
if(!empty($uid)){
    $map_uid['uid']=array('eq',$uid);
    $frontUser=M('front_user')->where($map_uid)->field(true)->find();
    $this->assign('frontUser',$frontUser);
}

//手机端视频分类
$map_m['status']=array('eq',1);
$video_m=$vi->where($map_m)->field(true)->limit(8)->select();
$this->assign('video_m',$video_m);
//手机端幻灯片
$map_banner['status']=array('eq',1);
$banner=M('banner')->where($map_banner)->field(true)->limit(5)->select();
$this->assign('banner',$banner);

//统计网站的访问量
$webclicknum = M("webclicknum")->find(1);
$webclicknum['clicknum']++;//访问量加一
$webclicknum['clicktime'] = time();
M("webclicknum")->save($webclicknum);
//实例化公告表对象
$notice = D("Notice");
$this->assign("notice",$notice->where("status=1")->order("subtime
desc")->limit("6")->select());

$shotuser = M('wenku')->query("select uid,count(*) num from wp_wenku group by uid
order by num desc limit 8");

foreach ($shotuser as &$shot) {
    $userob = M("front_user")->field("nickname,headimgurl")->find($shot['uid']);
    //查询用户的粉丝
    $fansnum = M("Friend")->where("uid={$shot['uid']}")->count();
    $shot['fansnum'] = $fansnum;
    $shot['nickname'] = $userob['nickname'];
    $shot['picture'] = $userob['picture'];
}
$this->assign("hotuser",$shotuser);//将活跃用户数组赋值给模板
//将结果集输出到模板中
$this->assign("wenku",$lib);

$this->display();
}

```

```

public function wenku(){
    $map['status']=array('eq',1);

    $wenku=M('wenku')->where($map)->field(true)->order('subtime DESC')->select();
    foreach($wenku as $k=>$v){
        $wenku[$k]['size']=sprintf("%.2f", $v['size']/(1024*1024));
    }

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

    $this->display();
}

```

```

public function wenku_view(){
    //手机端
    $id=I('id',0,intval);
    if($id==0){
        $this->error('系统错误');
    }
    $wen=M('wenku');
    $map['id']=array('eq',$id);
    $wenku=$wen->where($map)->field(true)->find();
    $wenku['size']=sprintf("%.2f", $wenku['size']/(1024*1024));
    $this->assign('wenku',$wenku);

    //手机端的猜你喜欢
    $cate_id=$wenku['category_id'];
    unset($map);
    $map['category_id']=array('eq',$cate_id);
    $recommend_wenku=$wen->where($map)->field(true)->limit(5)->select();
    $this->assign('recommend_wenku',$recommend_wenku);
    $this->display();
}

```

```

public function wenku_cate(){

    //手机端
    $map['type']=array('eq',2);
    $map['status']=array('neq',-1);
    $cate=M('category');
    $map['level']=array('eq',1);
    $one_arr=$cate->where($map)->field(true)->select();
    $this->assign('one_arr',$one_arr);
}

```

```

$map['level']=array('eq',2);
$two_arr=$cate->where($map)->field(true)->select();
$wen=M('wenku');
foreach($two_arr as $k=>$v){
    $count=0;
    $all_sonCate=array_filter(explode(',',$v['id'].",".getSonCateByCateId($v['id'])));
    foreach($all_sonCate as $key=>$value){
        $map['category_id']=array('eq',$value);
        $map['status']=array('eq',1);
        $temp=$wen->where($map)->count();
        $count=$count+$temp;
    }
    $two_arr[$k]['count']=$count;
}

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

$this->display();
}

```

```

public function wenku_user(){
    $map['status']=array('eq',1);
    $wenku=M('wenku')->where($map)->field(true)->select();
    foreach($wenku as $k=>$v){
        $wenku[$k]['size']=sprintf("%.2f", $v['size']/(1024*1024));
    }
    $this->assign('wenku',$wenku);

    $this->display();
}

```

```

public function allVideo(){
    $map['status']=array('eq',1);
    $video=M('video')->where($map)->field(true)->select();
    $this->assign('video',$video);

    //一、二级分类
    $map_c['type']=array('eq',1);
    $map_c['status']=array('neq',-1);
    //$data = $this->_get_data ( $map );
    $cate=M ( 'category' );
    $list = $cate->field ( true )->where ( $map_c )->select ();

    $one_arr=$cate->where('level=1')->field(true)->select();
}

```

```

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

$this->display();
}

public function vi_of_cate(){
    //一、二级分类
    $map_c['type']=array('eq',1);
    $map_c['status']=array('neq',-1);
    //$data = $this->_get_data ( $map );
    $cate=M ( 'category' );
    $list = $cate->field ( true )->where ( $map_c )->select ();

    $one_arr=$cate->where('level=1')->field(true)->select();
    $this->assign('one_arr',$one_arr);

    //获取分类下的所有视频
    $cate_id=I('cate_id',0,intval);
    if($cate_id==0){
        $this->error('系统错误');
    }
    $all_cate=explode(',',getSonCateByCateId($cate_id));

    $map['status']=array('eq',1);
    $cate_id_of_video=M('video')->where($map)->field(true)->select();

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

        if(in_array($v['category_id'],$all_cate)){
            continue;
        }
        else{
            unset($cate_id_of_video[$k]);
        }
    }
    $this->assign('cate_id_of_video',$cate_id_of_video);
    $this->display();
}

//定义视频播放的方法
public function video(){
    //一、二级分类
    $map_c['type']=array('eq',1);
    $map_c['status']=array('neq',-1);

```

```

//$data = $this->_get_data ( $map );
$scate=M ( 'category' );
$list = $scate->field ( true )->where ( $map_c )->select ();

$one_arr=$scate->where('level=1')->where ( $map_c )->field(true)->select();
$this->assign('one_arr',$one_arr);

$two_arr=$scate->where('level=2')->where ( $map_c )->field(true)->select();
$this->assign('two_arr',$two_arr);

$scat_id=I('cat_id',0,intval);
if($scat_id==0){
    $this->error('系统错误');
}
$map['status']=array('eq',1);
$map['id']=array('eq',$scat_id);
$vi=M('video');
$video=$vi->where($map)->field(true)->select();
$this->assign('video',$video);

$this->display();
}

```

```

public function play(){
    //一、二级分类
    $map_c['type']=array('eq',1);
    $map_c['status']=array('neq',-1);
    //$data = $this->_get_data ( $map );
    $scate=M ( 'category' );
    $list = $scate->field ( true )->where ( $map_c )->select ();

    $one_arr=$scate->where('level=1')->where ( $map_c )->field(true)->select();
    $this->assign('one_arr',$one_arr);

    $two_arr=$scate->where('level=2')->where ( $map_c )->field(true)->select();
    $this->assign('two_arr',$two_arr);

    $id=I('id',0,intval);
    if($id==0){
        $this->error('系统错误');
    }
    $vi=M('video');
    $map['id']=array('eq',$id);
}

```

```

$map['status']=array('eq',1);
$video=$vi->where($map)->field(true)->find();
$this->assign('video',$video);

//视频列表
unset($map);
$map['video_id']=array('eq',$id);
$chap=M('chapter');
$chapters=$chap->where($map)->field(true)->select();
$this->assign('chapters',$chapters);

$first_chapter=$chap->where($map)->field(true)->find();
$this->assign('first_chapter',$first_chapter);

//手机端的推荐课程
$cate_id=$video['category_id'];
unset($map);
$map['category_id']=array('eq',$cate_id);
$recommend_video=$vi->where($map)->field(true)->select();
$this->assign('recommend_video',$recommend_video);
$this->display();
}

public function user(){

    $this->display();
}

public function _get_data($map){
    $cate=M ( 'category' );
    $list = $cate->field ( true )->where ( $map )->select ();

    $one_arr=$cate->where('level=1')->field(true)->select();
    $two_arr=$cate->where('level=2')->field(true)->select();
    $three_arr=$cate->where('level=3')->field(true)->select();
    $data=array();
    foreach($two_arr as $twokey=>$twov){
        $temp_two=array();
        foreach($three_arr as $threekey=>$threev){
            if($threev['pid']==$twov['id']){
                $temp_two[]=$threev;
                unset($three_arr[$threekey]);
            }
        }
    }
}

```

```

        $temp_two=array_merge($twov,$temp_two);

    }
    return $data;
}

function videoPlay(){
    $uid=session('uid');
    $video_id=I('vid',0,intval);
    $chapter_id=I('cid',0,intval);
    if($video_id<=0 or $chapter_id<=0){
        $this->error('系统错误');
    }

    $map['video_id']=array('eq',$video_id);
    $map['id']=array('eq',$chapter_id);
    $chap=M('chapter');
    $chapter=$chap->where($map)->field(true)->find();
    $this->assign('chapter',$chapter);

    //执行视频浏览量的加一
    $chap->where($map)->setInc('clicknum');

    $vid=M('video');
    $map_v['id']=array('eq',$video_id);
    $vid->where($map_v)->setInc('clicknum');

    //获得该课程下的章节列表
    $map_list['video_id']=array('eq',$video_id);
    $chapter_list=$chap->where($map_list)->field('id,chapter_title,video_url')->select();
    $this->assign('chapter_list',$chapter_list);

    $this->display();
}

function noticeDetail(){
    $id=I('id',0,intval);
    if($id==0){
        $this->error('请正确选择');
    }
    $notice=M('notice')->where(array('id'=>$id))->field(true)->find();
    //dump($notice);die;
    $this->assign('notice',$notice);
}

```

```
    $this->display();  
}
```

//查询所有的类别

```
public function catIndex(){  
    $model = M("Category");  
    //查询出所有的一级类别  
    $list = $model->where("pid = 0")->select();  
    //定义存放响应子类的类别 id 数组  
    $sids = array();  
    foreach ($list as $v) {  
        $type = $model->field("id")->where("pid={$v['id']}")->select();  
        foreach ($type as $k) {  
            $sids[] = $k['id'];  
        }  
    }  
    //查询子类类别名  
    $where['id'] = array("in",$sids);  
    $zelist = $model->where($where)->select();  
    //赋值模板  
    $this->assign("list",$list);  
    $this->assign("zelist",$zelist);  
    //颜色随机  
    $color = "#".rand(180,255).rand(180,255);  
    $this->assign("color",$color);  
    $this->display();  
}
```

```
public function typeSelect(){  
    //设置默认选中的选项  
    //判断是否有类别 id 的传值  
    if($_REQUEST['id']){  
        $id=$_REQUEST['id'];  
    }else{  
        $id='0';  
    }  
    $mm=M('Cat');  
    //查询数据库表中所有类型 order by concat(path,id) 按照类别的层次进行查询  
    $res=$mm->field('id,name,path')->order("concat(path,id)")->select();  
  
    $cust_ids[] = 0;//定义存放类别 id 的数组
```

```

$cust_names[] = "..";//定义存放类别信息的数组
foreach($res as $vo){
    //分割查出的每条路径为数组
    $path = explode(",",$vo['path']);
    //通过数组元素的个数分别传入不同的类别名 达到级别的分级显示
    switch (count($path)) {
        case '1':
            $vo['name'] = $vo['name'];
            break;
        case '2':
            $vo['name'] = "-".$vo['name'];
            break;
        case '3':
            $vo['name'] = "---".$vo['name'];
            break;
    }
    $cust_ids[] = $vo['id'];//对应的类别 id 存进数组中
    $cust_names[]=$vo['name'];//把对应的类型名存进数组
}
//把所有类别信息的 id 数组以及类别名数组赋值给模板
$this->assign('cust_ids',$cust_ids);
$this->assign('cust_names',$cust_names);

$this->assign('mySelect',$id);//设置默认选中的 option 的下标值 id
}
//根据传递的 id 或 pid 搜索资源
public function show(){
    R("Video/index");
    R("Wenku/index");
    R("Message/index2");
    //分类名
    if(!$_GET['q']){
        if($_GET['id']){
            $res=M('Cat')->find($_GET['id']);
        }else{
            $res=M('Cat')->find($_GET['pid']);
        }
        $this->assign('cname',$res['name']);
    }
    $this->display();
}

public function wenkuIndex(){
    //实例化表对象

```

```

$model = M("Wenku");

//=====封装搜索条件=====
//封装搜索条件
if (!empty($_GET['q'])) {
    $where['title'] = array("like", "%".$_GET['q']."%");
    $this->assign("title", "搜索结果");
}

//封装文库类型条件
if (!empty($_GET['pid'])) {
    $type['id'] = array("eq", $_GET['pid']);
    $type['pid'] = array("eq", $_GET['pid']);
    $type['_logic'] = "OR";
    //查询出类型表中符合条件的所有类型的 id
    $ty = M("Category")->field("id")->where($type)->select();
    $tid = array();//存放符合条件的类型的 id
    foreach ($ty as $t) {
        $tid[] = $t['id'];
    }
    if (count($tid)>0) {
        //把符合条件的所有 id 存进搜索条件中
        $where['tid'] = array("in", $tid);
    }
    //查询下当前类别名
    $typename = M("Category")->field("name")->find($_GET['pid']);
    $this->assign("title", $typename["name"]);
}

//拼装搜索用户上传的资源
if (!empty($_GET['uid'])) {
    $where['uid'] = array("eq", $_GET['uid']);
    $this->assign("title", "上传列表");
}

//拼装用户收藏的资源条件
if (!empty($_GET['collid'])) {
    $coll = M("collect");
    $lids
    $coll->where("uid={$_SESSION[C('USER_AUTH_KEY')]['id']}")->field("lid")->select();
    $sids = array();
    foreach ($lids as $v) {
        $sids[] = $v['lid'];
    }
    $where['id'] = array("in", $sids);
}

```

```

        $this->assign("title","收藏一览");
    }

    //封装所有显示的文档
    $where['status'] = array("eq","2");
    $where['title'] = array("like","%{"$_GET['q']}%");
    //=====封装搜索条件=====

    //调用查询语句
    $result = $model->where($where)->order("clicknum desc")->select();

    //查询上传用户名
    //遍历上面查询出的资源信息结果 对结果集进行所需信息的追加
    foreach ($result as &$vo) {
        $ob = D("front_user")->field("username,headimgurl")->find($vo['uid']);
        $vo['nickname'] = $ob['nickname'];
        $vo['headimgurl'] = $ob['headimgurl'];
        //组装列表中的资源名
        $ext = array_pop(explode("/", $vo['type']));
        $vo['libname'] = $vo['title'].".$ext;
    }

    //向模板中输出查询信息和分页信息
    //$this->assign("page",$);
    $this->assign("list",$result);

    $this->assign("list2",$result);

    //最新资源列表
    $newlist = $model->where("status=2")->order("subtime desc")->limit(16)->select();
    $this->assign("newlist",$newlist);

    //=====
    //执行资源分类的查询
    $type = M("Category")->where("pid=0")->limit(6)->select();
    $this->assign("typelist",$type);
    //=====
    if($_SESSION[C('USER_AUTH_KEY')]){
        //调用贴吧中的查询 hfnum()方法
        R("Message/hfnum");
    }
    //输出模板
    if(!$_GET['r']){

```

```

        if($_GET['q'] && isset($_GET['pid'])){
            if($_GET['pid']=='0' && $_GET['q']){
                $this->display();
            }
        }else if(!$_GET['pid'] && !$_GET['q']){
            $this->display();
        }else{
            if($_GET['pid']){
                $this->display();
            }
        }
    }
}

```

//定义视频被赞的方法

```

public function favo(){
    //实例化资源表对象
    $model = D("Video");
    $zan = M("Zan");//实例化赞表对象
    //查询相应的资源
    $model->find($_POST['id']);
    //封装要添加的数据
    $map['uid'] = $_SESSION[C('USER_AUTH_KEY')]['id'];
    $map['vid'] = $_POST['id'];

    //封装搜索条件
    $where['uid'] = array("eq",$_SESSION[C('USER_AUTH_KEY')]['id']);
    $where['vid'] = array("eq",$_POST['id']);

    //查询该用户是否赞过这个资源
    $his = $zan->where($where)->find();
    if ($his) {
        $model->favonum -= 1;//让资源的被赞量减一
        $model->save();
        $zan->where($where)->delete();//删除赞数据表中的数据
        echo "no";die;
    }else{
        $model->favonum += 1;//赞数加一
        $model->save();
        //往赞表中存进资源和用户的 id
        $zan->add($map);//保存数据
        echo "yes";die;
    }
}
}

```

```

//定义视频被收藏的方法
public function collect(){
    //实例化收藏表对象
    $collect = M("Collect");

    //封装搜索要添加的数据
    $map['uid'] = $_SESSION[C('USER_AUTH_KEY')]['id'];
    $map['vid'] = $_POST['id'];

    //封装搜索条件
    $where['uid'] = array("eq",$_SESSION[C('USER_AUTH_KEY')]['id']);
    $where['vid'] = array("eq",$_POST['id']);

    //查询该用户是否收藏过这个视频
    $coll = $collect->where($where)->find();
    if ($coll) {
        $collect->where($where)->delete();//删除收藏数据表中的数据
        echo "no";die;
    }
    else{
        //往收藏表中存进视频和用户的 id
        $collect->add($map);//保存数据
        echo "yes";die;
    }
}

```

```

//定义添加笔记的方法
public function addNote(){
    $model = D("Note");
    $list = $model->create();
    if ($model->add($list)) {
        $this->success("创建成功！ ",U("Video/detail?{$list['vid']}"));
    }else{
        $this->error("创建失败！ ");
    }
}

```

```

//定义资源信息查询方法 详情页
public function wenku_detail(){
    //实例化
    $model = D("Wenku");
    $vo = $model->find($_GET['id']);
}

```

```

$model->clicknum += 1;//执行资源的浏览量加一的操作
$model->save();

//查询资源表中 uid 对应的用户名追加到资源结果集中
$user = D("front_user")->find($vo['uid']);
$vo['nickname'] = $user['nickname'];
$vo['picture'] = $user['picture'];

//执行资源类别名搜索追加到结果集
$type = M("Category")->field("title")->find($vo['tid']);
$vo['typename'] = $type['name'];

//组装资源 swf 格式的文件名
$randname = substr($vo['file_name'],0,strpos($vo['file_name'],"."));
$vo['swfname'] = $randname.".swf";//将文件名追加进结果数组

$this->assign("vo",$vo);
$type = array_pop(explode("/", $vo['type']));
$this->assign("type", $type);

//调用 CommentActin 类中的显示评论 index()方法 通过 R 方法
R("Libcom/index",array($_GET['id']));

//实例化赞表和收藏表对象
$zan = M("Zan");
$collect = M("Collect");
//封装搜索条件
$where['uid'] = array("eq",$uid);
$where['lid'] = array("eq",$_GET['id']);
//查询该用户是否赞过这个资源
$iszanzan = $zan->where($where)->find();
$coll = $collect->where($where)->find();
$collnum = $collect->where($where)->count();//查询被收藏数
$this->assign("iszanzan",$iszanzan);
$this->assign("collect",$coll);
$this->assign("collnum",$collnum);
//你可能还喜欢
$stulist = $model->where("status=2")->order("favonum desc")->limit(10)->select();
$this->assign("tulist",$stulist);

$this->display();
}

```

```

/* 电脑前端用户注册页面 */
public function register() {
    if (IS_POST) { // 注册用户
        $username=I('username');
        $email=I('email');
        $password=I('password');
        $repassword=I('repassword');
        /* 测试用户名 */
        if (empty ( $username)) {
            $this->error ( '用户名不能为空! ');
        }
        else if (strlen ( $username ) > 16) {
            $this->error ( '用户名长度必须在 16 个字符以内! ');
        }

        /* 检测密码 */
        if (strlen ( $password ) < 6 || strlen ( $password ) > 30) {
            $this->error ( '密码长度必须在 6-30 个字符之间! ');
        }
        if ($password != $repassword) {
            $this->error ( '密码和重复密码不一致! ');
        }

        $come_from=3;

        /* 调用注册接口注册用户 */
        $uid = D ( 'FrontUser' )->register ( $username, $password, $email);

        if (0 < $uid) {
            // 注册成功
            $user ['uid'] = $uid;
            //$user ['mobile'] = $mobile;
            //注册之前所在的页面
            $url = Cookie ( '__forward__' );
            if (!$url) {
                $url = U ( 'Home/Index/index' );
            }
            D ( 'FrontUser' )->autoLogin ( $user );
            if($uid>0){
                $this->success ( '恭喜, 注册成功! ', $url );
            }
            if($uid==0){
                $this->success('恭喜, 注册成功',$url);
            }
        }
    }
}

```

```

    }
    else { // 注册失败，显示错误信息
        $this->error ( $this->showRegError ( $uid ) );
    }
}
else { // 显示注册表单
    $this->display ();
}
}

/* 登录页面 */
public function login() {
    if (IS_POST) {
        $username=I('username');
        $password=I('password');

        $dao = D ( 'FrontUser' );
        $uid = $dao->login ( $username, $password );
        if (! $uid) { // 登录失败
            $this->error ( $dao->getError () );
            exit ();
        }

        session('[start]');
        session('uid',$uid);
        $this->success ( '登录成功! ', U ( 'Index/index' ));
    }
    else {
        $this->display ( );
    }
}

// 清缓存
function generalClean() {
    // getModelByName_comment
    $key = I ( 'key' );
    dump ( $key );
    $d = S ( $key );
    dump ( $d );
    if (isset ( $_GET ['do_clean'] ))
        dump ( S ( $key, null ) );
}

//跳转页面

```

```

function jump(){
    $this->display ();
}
}

```

```

<?php
namespace Admin\Controller;
use Think\Controller;

class VideoController extends AdminController{
    public function index(){
        $map['status']=array('eq',1);
        $video=M('video')->where($map)->field(true)->order('id desc')->select();
        $user=M('user');
        $cate=M('category');
        foreach($video as $key=>$value){

$usename=$user->where(array('uid'=>$value['uid']))->field('nickname')->find();
            $video[$key]['username']=$usename['nickname'];
        }

        //分页操作
        $request    =    (array)I('request.');
        $total      =    $video? count($video) : 1 ;
        $listRows   =    C('LIST_ROWS') > 0 ? C('LIST_ROWS') :2;
        $page       =    new \Think\Page($total, $listRows, $request);
        $voList     =    array_slice($video, $page->firstRow, $page->listRows);
        $p          =    $page->show();
        $this->assign('video', $voList);
        $this->assign('_page', $p? $p: "");
        // 记录当前列表页的 cookie
        Cookie('__forward__',$_SERVER['REQUEST_URI']);
        //分页操作结束

        $this->display();
    }

function _get_data($map) {
    $list = M ('category')->field ( true )->where ( $map )->select ();

    // 取一级菜单
    foreach ( $list as $k => $vo ) {
        if ($vo ['pid'] != 0)

```



```

else{
    //echo header("Content-Type: text/html; charset=utf-8");
    $data=I('post.');
    $data['subtime']=time();

    if($data['type']!=1){
        $this->error('系统出错， 请选择类型');
    }
    if(!$data['video_title']){
        $this->error('标题不能为空');
    }
    $map['id']=array('eq',$data['category_id']);
    $category_title=M('category')->where($map)->getField('title');
    $data['category_title']=$category_title;

    $parent_cate_id=getParentCate($data['category_id']);
    if($parent_cate_id==0){
        //取$cid
        $root_cate_id=$data['category_id'];
    }
    else{
        $grand_cate_id=getParentCate($parent_cate_id);
        if($grand_cate_id==0){
            $root_cate_id=$parent_cate_id;
        }
        else{
            $root_cate_id=$grand_cate_id;
        }
    }
    $data['root_cate_id']=$root_cate_id;

    $res=D('Video')->addHandle($data);
    if($res['status']){
        $this->success($res['info'],U('Video/index'));
    }
    else{
        $this->error($res['info']);
    }
}
}
}

```

```

//编辑视频课程
public function edit(){

```

```

if(!IS_POST){
    $id=I('id',0,intval);
    $this->video=M('video')->where(array('id'=>$id))->field(true)->find();
    $categories = M('category')->field(true)->select();
    $categories = D('Common/Tree')->toFormatTree($categories);
    $categories = array_merge(array(0=>array('id'=>0,'title_show'=>'顶级菜单')),
$categories);

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

    $this->display();
}
else{
    $id=I('id',0,intval);
    $data=I('post.');
```

```

    if(!$data['video_title']){
        $this->error('标题不能为空');
    }
    $map['id']=array('eq',$data['category_id']);
    $category_title=M('category')->where($map)->getField('title');
    $data['category_title']=$category_title;

    $parent_cate_id=getParentCate($data['category_id']);
    if($parent_cate_id==0){
        //取$id
        $root_cate_id=$data['category_id'];
    }
    else{
        $grand_cate_id=getParentCate($parent_cate_id);
        if($grand_cate_id==0){
            $root_cate_id=$parent_cate_id;
        }
        else{
            $root_cate_id=$grand_cate_id;
        }
    }
    $data['root_cate_id']=$root_cate_id;
    //dump($data);die;
    $res=D('Video')->editHandle($id,$data);
    if($res['status']){
        $this->success($res['info'],U('Video/index'));
    }
    else{
        $this->error($res['info']);
    }
}

```

```

    }
}
}

//删除
public function del(){
    //删除时要先检查该视频课程下有没有具体的章节，如果有的话，则不能删除。
    $id=I('id',0,intval);
    $map['video_id']=array('eq',$id);
    $map['status']=array('eq',1);
    $chapter_num=M('chapter')->where($map)->count();
    if($chapter_num!=0){
        $this->error('请删除该视频课程下的具体章节后再删除该视频课程');
    }
    else{
        $res=D('Video')->delHandle($id);
        if($res['status']){
            $this->success($res['info'],U('Video/index'));
        }
        else{
            $this->error($res['info']);
        }
    }
}
}

```

//删除多个项目

```

public function delMultiType(){
    $id=array_unique((array)I('id'));
    $id = is_array($id) ? implode(',',$id) : $id;
    if ( empty($id) ) {
        $this->error('请选择要操作的数据!');
    }
    $map['id'] = array('in',$id);
    $result=M('newstype')->where($map)->delete();
    if($result){
        $this->success('删除成功');
    }
    else{
        $this->error('删除失败');
    }
}
}

```

//图片上传

```

public function uploadify_pic()

```

```

{
    if (!empty($_FILES)) {
        //图片上传设置
        $config = array(
            'maxSize' => 3145728,
            'savePath' => "",
            'saveName' => array('uniqid',''),
            'exts' => array('jpg', 'gif', 'png', 'jpeg'),
            'autoSub' => true,
            'subName' => array('date','Ymd'),
        );
        $upload = new \Think\Upload($config);// 实例化上传类
        $images = $upload->upload();

        //判断是否有图
        if($images){
            $info=$images['Filedata']['savepath'].$images['Filedata']['savename'];
            //返回文件地址和名给 JS 作回调用
            echo $info;
            return $info;
        }
        else{
            //$this->error($upload->getError());//获取失败信
        }
    }
}

//视频上传
public function uploadify_video()
{
    if (!empty($_FILES)) {
        //视频上传设置
        $config = array(
            'rootPath' => './Uploads/video/',
            'savePath' => date('Ymd'),
            'maxSize' => 2145728000,//200M
            'saveName' => array('uniqid',''),
            'exts' => array('mp4', 'avi', 'mov', 'rmvb','wmv'),
            'autoSub' => true,
            'subName' => 'a',
        );
        $upload = new \Think\Upload($config);// 实例化上传类
        $video = $upload->upload();
    }
}

```

```

        //判断是否有图
        if($video){
            $info=$video['Filedata']['savepath'].$video['Filedata']['savename'];
            //返回文件地址和名给 JS 作回调用
            echo $info;
            return $info;
        }
        else{
            //$this->error($upload->getError());//获取失败信
        }
    }
}

//删除多个项目
public function delMulti(){
    $id=array_unique((array)I('id'));
    $id = is_array($id) ? implode(',',$id) : $id;
    if ( empty($id) ) {
        $this->error('请选择要操作的数据!');
    }
    $map['id'] = array('in',$id);
    $result=M('news')->where($map)->delete();
    if($result){
        $this->success('删除成功');
    }
    else{
        $this->error('删除失败');
    }
}
}
?>

```

```

<?php
namespace Admin\Controller;
class NoticeController extends AdminController {
    function index() {
        $notice = M ( 'Notice' )->where(array('status'=>1))->field(true)->select();

        //分页操作
        $request    =    (array)I('request.');
        $total      =    $notice? count($notice) : 1 ;
        $listRows   =    C('LIST_ROWS') > 0 ? C('LIST_ROWS') :7;
        $page       =    new \Think\Page($total, $listRows, $request);
    }
}

```

```

    $voList    =    array_slice($notice, $page->firstRow, $page->listRows);
    $p        =    $page->show();
    $this->assign('_list', $voList);
    $this->assign('_page', $p? $p: "");

    Cookie('__forward__',$_SERVER['REQUEST_URI']);
    //分页操作结束

    $this->display ();
}

/**
 * 新增公告
 */
public function add(){
    if(IS_POST){
        $_POST['subtime'] = time();
        $Model = D('Notice');
        $data = $Model->create();
        if($data){
            $id = $Model->add();
            if($id){
                $this->success('新增成功', U('index'));
            }
            else {
                $this->error('新增失败');
            }
        }
        else {
            $this->error($Model->getError());
        }
    }
    else {
        $this->display();
    }
}

public function edit(){
    if(!IS_POST){
        $id=I('id',0,intval);
        $this->notice=M('Notice')->where(array('id'=>$id))->field(true)->find();

        $this->display();
    }
}

```

```

    }
    else{
        $id=I('id',0,intval);
        if(!$ _POST['title']){
            $this->error('标题不能为空');
        }
        $res=D('Notice')->editHandle($id,$ _POST);
        if($res['status']){
            $this->success($res['info'],U('Notice/index'));
        }
        else {
            $this->error($res['info']);
        }
    }
}

```

```

public function del(){
    $id=I('id');
    $res=D('Notice')->delHandle($id);
    if($res['status']){
        $this->success($res['info'],U('Notice/index'));
    }
    else{
        $this->error($res['info']);
    }
}

```

//删除多个项目

```

public function delMulti(){
    $id=array_unique((array)I('id'));
    $id = is_array($id) ? implode(',',$id) : $id;
    if ( empty($id) ) {
        $this->error('请选择要操作的数据!');
    }
    $map['id'] = array('in',$id);
    $result=M('Notice')->where($map)->delete();
    if($result){
        $this->success('删除成功');
    }
    else{
        $this->error('删除失败');
    }
}
}

```

```

}
<?php
namespace Admin\Controller;
use Think\Controller;

class BannerController extends AdminController{
    public function index(){
        $this->banner=M('banner')->select();
        $this->display();
    }

    //添加 banner
    public function add(){
        if(!IS_POST){
            $this->display();
        }
        else{
            echo header("Content-Type: text/html; charset=utf-8");
            $data=I('post.');
            $data['subtime']=time();
            if(!$data['title']){
                $this->error('标题不能为空');
            }

            $res=D('Banner')->addHandle($data);
            if($res['status']){
                $this->success($res['info'],U('Banner/index'));
            }
            else{
                $this->error($res['info']);
            }
        }
    }

    //编辑
    public function edit(){
        if(!IS_POST){
            $id=I('id',0,intval);
            $this->banner=M('banner')->where(array('id'=>$id))->find();
            $this->display();
        }
        else{
            $id=I('id',0,intval);
            $data=I('post.');

```

```

        if(!$data['title']){
            $this->error('标题不能为空');
        }

        $res=D('Banner')->editHandle($id,$data);
        if($res['status']){
            $this->success($res['info'],U('Banner/index'));
        }
        else{
            $this->error($res['info']);
        }
    }
}

```

//删除

```

public function del(){
    $id=I('id',0,intval);
    $res=D('Banner')->delHandle($id);
    if($res['status']){
        $this->success($res['info'],U('Banner/index'));
    }
    else{
        $this->error($res['info']);
    }
}

```

//删除多个项目

```

public function delMulti(){
    $id=array_unique((array)I('id'));
    $id = is_array($id) ? implode(',',$id) : $id;
    if ( empty($id) ) {
        $this->error('请选择要操作的数据!');
    }
    $map['id'] = array('in',$id);
    $result=M('banner')->where($map)->delete();
    if($result){
        $this->success('删除成功');
    }
    else{
        $this->error('删除失败');
    }
}

```

```

//图片上传
public function uploadify()
{
    if (!empty($_FILES)) {
        //图片上传设置
        $config = array(
            'maxSize' => 3145728,
            'savePath' => "",
            'saveName' => array('uniqid',"),
            'exts' => array('jpg', 'gif', 'png', 'jpeg'),
            'autoSub' => true,
            'subName' => array('date','Ymd'),
        );
        $upload = new \Think\Upload($config);// 实例化上传类
        $images = $upload->upload();

        //判断是否有图
        if($images){
            $info=$images['Filedata']['savepath'].$images['Filedata']['savename'];
            //返回文件地址和名给 JS 作回调用
            echo $info;
            return $info;
        }
        else{
            $this->error($upload->getError());//获取失败信
        }
    }
}
?>

```

```

<extend name="Public/base" />
<block name="style">
    <title>{:C('WEB_SITE_TITLE')}--用户管理</title>
</block>

<block name="body">
    <div class="container-fluid animated fadeInRight">
        <!-- BEGIN PAGE HEADER-->
        <div class="row-fluid margin-top-10">
            <!-- BEGIN PAGE TITLE & BREADCRUMB-->
            <a href="{:U('add')}" class="btn green" target-form="ids"><i
class="icon-plus"></i> 新增</a>

```



```

<!-- BEGIN PAGE CONTENT-->
<div class="row-fluid margin-top-10 ">
  <div class="span12">
    <!-- BEGIN ACCORDION PORTLET-->
    <div class="portlet box blue">
      <div class="portlet-title">
        <div class="caption"><i class="icon-reorder"></i> 新增用户</div>
        <div class="tools">
          <a href="javascript:;" class="collapse"></a>
          <a href="javascript:;" class="remove"></a>
        </div>
      </div>
      <div class="portlet-body form">

        <!-- BEGIN FORM-->
        <form action="{U('User/addUser')}}" method="post" class="form-horizontal">
          <div class="row-fluid">
            <div class="span10">
              <div class="control-group">
                <label class="control-label"> 登 录 名 <font
color="#FF0000">*</font></label>
                <div class="controls">
                  <input type="text" class="span11 m-wrap"
name="nickname"/>
                  <span class="help-inline">
                    英文、数字或下划线
                  </span>
                </div>
              </div>
            </div>
            <div class="span2">
              <div class="control-group">
                <label class="control-label"> 真 实 姓 名 <font
color="#FF0000">&nbsp;*</font></label>
                <div class="controls">
                  <div class="input-icon ">
                    <input type="text" class="span11 m-wrap"
name="truename"/>
                    <span class="help-inline">
                      汉字、数字或下划线
                    </span>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </form>
      </div>
    </div>
  </div>
</div>

```

```

        </div>
    </div>
</div>

<div>
    <div class="row-fluid">
        <div class="span10">
            <div class="control-group">
                <label class="control-label"> 密 码 <font
color="#FF0000">*</font></label>
                <div class="controls">
                    <div class="input-icon">
                        <input type="password" class="span11 m-wrap"
name="password"/>
                        <span class="help-inline">密码不能少于 6 位</span>
                    </div>
                </div>
            </div>
        </div>
    </div>

    <div class="row-fluid">
        <div class="span10">
            <div class="control-group">
                <label class="control-label"> 确 认 密 码 <font
color="#FF0000">&nbsp;*</font></label>
                <div class="controls">
                    <input type="password" class="span11 m-wrap"
name="repassword"/>
                </div>
            </div>
        </div>
    </div>

    <div class="row-fluid">
        <div class="span10">
            <div class="control-group">
                <label class="control-label">邮箱</label>
                <div class="controls">
                    <div class="input-icon">
                        <input type="text" class="span11 m-wrap"
name="email"/>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

        </div>
        </div>

    </div>

    <div class="row-fluid">
        <div class="span10">
            <div class="control-group">
                <label                                class="control-label">qq<font
color="#FF0000">&nbsp;*</font></label>
                <div class="controls">
                    <div class="input-icon">
                        <input type="text" class="span11 m-wrap" name="qq"/>
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div class="row-fluid">
        <div class="span10">
            <div class="control-group">
                <label    class="control-label">    电    话    <font
color="#FF0000">&nbsp;*</font></label>
                <div class="controls">
                    <input    type="text"    class="span11    m-wrap"
name="mobile"/>
                </div>
            </div>
        </div>
    </div>

    <div class="form-actions">
        <button    id="submit"                type="submit"    class="btn    blue"
target-form="form-horizontal">确定</button>
        <button                type="button"                class="btn"
onclick="javascript:history.back(-1);return false;">返回</button>
    </div>
</form>

<!-- END FORM-->
</div>
</div>
<!-- END ACCORDION PORTLET-->
</div>

```

```

    </div>
  </div>
</block>

<block name="script">
  <script type="text/javascript">
//导航高亮
    $(function(){
      var $window = $(window), $subnav = $("#subnav"), url;
      /* 左边菜单高亮 */
      url = "/index.php?s=/Admin/User/index.html";
      $subnav.find("a[href$=" + url + "]").parent().addClass("active");
      $subnav.find("a[href$=" + url + "]").parent().parent().parent().addClass("active");
    });
  </script>
</block>

<extend name="Public/base"/>
<block name="body">
  <div class="container-fluid animated fadeInRight">
    <!-- BEGIN PAGE CONTENT-->
    <div class="row-fluid margin-top-10 ">
      <div class="span12">
        <!-- BEGIN ACCORDION PORTLET-->
        <div class="portlet box blue">
          <div class="portlet-title">
            <div class="caption"><i class="icon-reorder"></i> 修改系统管理员信息</div>
            <div class="tools">
              <a href="javascript:;" class="collapse"></a>
              <a href="javascript:;" class="remove"></a>
            </div>
          </div>
          <div class="portlet-body form">

            <!-- BEGIN FORM-->
            <form action="{U('User/addUser')}}" method="post" class="form-horizontal">
              <div class="row-fluid">
                <div class="span10">
                  <div class="control-group">
                    <label class="control-label"> 登 录 名 <font
color="#FF0000">*</font></label>
                    <div class="controls">
                      <input type="text" class="span11 m-wrap" name="nickname"
value="{User.nickname}">

```

```

        <span class="help-inline">
            英文、数字或下划线
        </span>
    </div>
</div>
<div class="row-fluid">
    <div class="span10">
        <div class="control-group">
            <label class="control-label"> 真实姓名 <font
color="#FF0000">&nbsp;*</font></label>
            <div class="controls">
                <div class="input-icon ">
                    <input type="text" class="span11 m-wrap"
name="truename" value="{ $user.truename }">
                    <span class="help-inline">
                        汉字、数字或下划线
                    </span>
                </div>
            </div>
        </div>
    </div>
</div>
<div class="row-fluid">
    <div class="span10">
        <div class="control-group">
            <label class="control-label">邮箱</label>
            <div class="controls">
                <div class="input-icon">
                    <input type="text" class="span11 m-wrap" name="email"
value="{ $user.email }">
                </div>
            </div>
        </div>
    </div>
</div>
<div class="row-fluid">
    <div class="span10">
        <div class="control-group">

```



```

$(function(){
    var $window = $(window), $subnav = $("#subnav"), url;
    /* 左边菜单高亮 */
    url = "/index.php?s=/Admin/User/index.html";
    $subnav.find("a[href$=\"" + url + "\"]').parent().addClass("active");
    $subnav.find("a[href$=\"" + url + "\"]').parent().parent().parent().addClass("active");
});
</script>
</block>

<extend name="Public/base" />

<block name="body">
    <div class="container-fluid animated fadeInRight">
        <!-- BEGIN PAGE CONTENT-->
        <div class="row-fluid margin-top-10 ">
            <div class="span12">
                <!-- BEGIN ACCORDION PORTLET-->
                <div class="portlet box blue">
                    <div class="portlet-title">
                        <div class="caption"><i class="icon-reorder"></i> 修改密码</div>
                        <div class="tools">
                            <a href="javascript:;" class="collapse"></a>
                            <a href="javascript:;" class="remove"></a>
                        </div>
                    </div>
                    <div class="portlet-body form">

                        <!-- BEGIN FORM-->
                        <form          action="{:U('User/modifyPwdHandle')}"          method="post"
class="form-horizontal">
                            <div class="control-group">
                                <label class="control-label">网点名称</label>
                                <div class="controls">
                                    <div class="input-icon left">
                                        <i class="icon-user"></i>
                                        <input type="text" name="truename" class="span6 m-wrap"
value="{ $editing_user.truename }" readonly="readonly" />
                                    </div>
                                </div>
                            </div>
                            <div class="control-group">
                                <label class="control-label">新密码</label>

```

```

        <div class="controls">
            <div class="input-icon left">
                <i class="icon-lock"></i>
                <input type="password" name="password" class="span6 m-wrap "
placeholder="请填写密码" />
            </div>
        </div>
    </div>
    <div class="control-group">
        <label class="control-label">确认密码</label>
        <div class="controls">
            <div class="input-icon left">
                <i class="icon-lock"></i>
                <input name="repassword" type="password" class="span6 m-wrap "
placeholder="请确认密码"/>
            </div>
        </div>
    </div>
    <div class="form-actions">
        <input type="hidden" name="uid" value="{Suid}"/>
        <button id="submit" type="submit" class="btn blue ajax-post"
target-form="form-horizontal">确定</button>
        <button type="button" class="btn" onclick="javascript:history.back(-1);return
false;">返回</button>
    </div>
</form>

<!-- END FORM-->
</div>
</div>
<!-- END ACCORDION PORTLET-->
</div>
</div>
</div>
</block>

```

```

<block name="script">
    <script type="text/javascript">
//导航高亮
$(function(){
    var $window = $(window), $subnav = $("#subnav"), url;
    /* 左边菜单高亮 */
    url = "/index.php?s=/Admin/User/index.html";
    $subnav.find("a[href$=" + url + "']").parent().addClass("active");

```

```
$subnav.find("a[href$=" + url + "']").parent().parent().parent().addClass("active");
});
</script>
</block>
```