1. **CI的HelloWorld！**

sublime-package

./application/controllers/hello.php

<?php

//放止用户直接通过路径来访问控制器，如果这样的话会显示找不到（封装）

if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Hello extends CI\_Controller {

public function sayhello($name,$name2){

echo $name,",Hello CI to ",$name2;

}

}

**二.CI的文本计时器demo——文本操作与调用视图操作**

**调用视图的基本格式：**

**$this->load->view('XXX');**

<?php

//./applications/controllers/hello.php

//放止用户直接通过路径来访问控制器，如果这样的话会显示找不到（封装）

if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Hello extends CI\_Controller {

public function sayhello($name,$name2){

echo $name,",Hello CI to ",$name2;

}

public function show(){

$name = "deng";

@$count = file\_get\_contents('./num.txt'); //装饰器

$count = $count ? $count:0;

$count++;

$data = array('key'=>$name,'value'=>$count);

$re = fopen('./num.txt','w');

fwrite($re, $count);

$this->load->view("testview.php",$data); //装载两个视图页面

$this->load->view("testview2.php");

}

}

**三.CI的数据库demo——对数据模型的增，删，改，查**

**数据模型——**

**1.数据模型是一个数据库类**

**2.一个模型针对一张表**

**创建一个模型必须注意——**

**必须继承数据核心类CI\_Model，同时重载父类中的构造方法**

class Model\_name extends CI\_Model

{

function \_\_construct()

{

parent::\_\_construct();

}

}

**对数据库的操作——**

**1.连接数据库**（$this->load->database());

**2.插入数据**（$this->db->insert($t\_name,$arr);)

$t\_name——你要操作的表]

$arr——你要插入的数据("key"=>value)

**3.更新数据**

$this->db->where(字段名，字段值)

$this->db->update(表名，修改值的数组)

**4.查询数据**

$this->db->where(字段名，字段值)

$this->db->select(字段)

$query = $this->db->get(表名)

return $query->result();

**5.删除数据**

$this->db->where(字段名，字段值)

$this->db->delete(表名)

./application/controllers/user.php

<?php

if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class User extends CI\_Controller {

public function insert(){

$this->load->model('test\_m');

$arr = array('usid'=>1,'uname'=>'deng1','upass'=>'1234');

$this->test\_m->user\_insert($arr);

}

public function update(){

$this->load->model('test\_m');

$arr = array('usid'=>22,'uname'=>'deng222','upass'=>'1233333');

$this->test\_m->user\_update(2,$arr);

}

public function delete(){

$this->load->model('test\_m');

$this->test\_m->user\_delete(22);

}

public function select()

{

$this->load->model('test\_m');

$arr = $this->test\_m->user\_select(1);

print\_r($arr);

echo $arr[0]->usid;

}

}

/\* End of file welcome.php \*/

/\* Location: ./application/controllers/welcome.php \*/

<?php

/\*\*

\* ./application/models/test\_m.php

\*/

class Test\_m extends CI\_Model

{

function \_\_construct()

{

parent::\_\_construct();

//connect to the database

$this->load->database();

//$this->load->insert($t\_name,$data)

}

function user\_insert($arr){

$this->db->insert('user',$arr);

}

function user\_update($id,$arr)

{

$this->db->where('usid',$id);

$this->db->update('user',$arr);

}

function user\_delete($id){

$this->db->where('usid',$id);

$this->db->delete('user');

}

function user\_select($id){

$this->db->where('usid',$id);

$this->db->select('\*');

$query = $this->db->get('user');

return $query->result();

}

}

?>

**四.CI的文件上传demo**

**1.面向过程的文件上传方法**

#/controllers/upload.php

<?php

if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Upload extends CI\_Controller {

//显示带表单的视图

public function index(){

$this->load->view('up');

}

//显示上传信息

public function up(){

if(!empty($\_POST['sub'])){

//打印变量的函数

//var\_dump($\_FILES['upfile']);

$file = $\_FILES['upfile'];

if($file['size'] >= 20000000){

echo "size no!";

}

else{

switch ($file['type']) {

case 'image/jpeg':

$br = '.jpg';

break;

default:

$br = false;

break;

}

if($br){

$time = time();

move\_uploaded\_file($file['tmp\_name'], "./upload/$time$br");

}

else{

echo "type no";

}

}

}

}

}

#/views/up.php

<html>

<!--注意相对路径的建立-->

<form action="/CI/index.php/upload/up" method="post" enctype="multipart/form-data">

<input type="file" name="upfile" />

<input type="submit" name="sub" value="提交" />

</form>

</html>

**2.面向对象的CI框架文件上传方法**

**a.定义一个数组，设置一些与上传相关的参数**

//设置上传目录，这里写./,目录要建在网站根目录，就是和application同级

//如果你要放在application目录下，可以用系统定义的路径常量APPPATH

//例如：APPPATH.'uploads/'

$config['upload\_path'] = './uploads/';

//设置允许上传的类型

$config['allowed\_types'] = 'gif|jpg|png';

$config['max\_size'] = '100';

//如果是图片还可以设置最大高度和宽度

$config['max\_height'] = 768;

$config['max\_width'] = 1024;

**b.还可以设置其他的一些额外参数，详细看用户手册**

**c.调用CI的上传通用类，并执行上传**

**//upload为调用的类名，全小写**

**$this->load->library('upload',$config);**

**//如果上传框的name写的是userfile，那就不用传参数了，如果不是，把name的值传进去**

**$this->upload->do\_upload('上传框的name');**

**d.接收出错信息或成功信息**

**//出错信息**

**$error = array('error' => $this->upload->display\_error());**

**//成功信息**

**$data = array('upload\_data' => $this->upload->data());**

<?php

if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Upload extends CI\_Controller {

//显示带表单的视图

public function index(){

$this->load->view('up');

}

//显示上传信息

public function up(){

$config['upload\_path'] = './uploads/';

$config['allowed\_types'] = 'gif|jpg|png';

$config['max\_size'] = "2000";

$this->load->library('upload',$config);

//打印成功或错误的信息

if($this->upload->do\_upload('upfile'))

{

$data = array("upload\_data" => $this->upload->data());

var\_dump($data);

}

else

{

$error = array("error" => $this->upload->display\_errors());

var\_dump($error);

}

}

}

**五.CI的登录验证demo**

**1.利用CI类实现session登录**

**a.修改配置文件（config.php）**

**$config['encryption\_key']**

**b.加载SESSION类**

**$this->load->library('session');**

**c.创建SESSION**

**$this->session->set\_userdata($array);**

**d.查看SESSION**

**$this->session->userdata(session名);**

**e.删除SESSION**

**$this->session->unset\_userdata('SESSION名');**

./application/controllers/login.php

<?php

if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Login extends CI\_Controller {

public function index()

{

$this->load->view('login.html');

}

public function checklogin(){

$this->load->model("test\_m");

$user = $this->test\_m->user\_select($\_POST['uname']);

if($user){

if($user[0]->upass == $\_POST['upass']){

echo '密码正确！';

$this->load->library('session');

$arr = array("uid" => $user[0]->usid);

$this->session->set\_userdata($arr);

echo "<br />";

echo $this->session->userdata('uid');

}

else{

echo '密码不正确！';

}

}

else{

echo '用户名不存在';

}

}

public function checksession(){

$this->load->library('session');

if($this->session->userdata('uid')){

echo '已经登录。';

}

else{

echo '没有登录。';

}

}

public function loginout(){

$this->load->library('session');

$this->session->unset\_userdata('uid');

}

}

**六.CI的分页功能demo**

**1.必须知道的一些参数**

**a.总共有多少条记录**

**b.一页要有多少条记录**

**c.总共多少页**

**d.当前页前后要显示多少个分页链接**

**2.设置一些CI分页类基本参数**

**//总条数**

**$config['total\_rows']**

**//一页显示几条**

**$config['per\_page']**

**//定义当前页的前后各有几个数字链接**

**$config['num\_links']**

**//定义没有分页参数,主URL**

**$config['base\_url']**

**3.调用CI的分页类**

**$this->load->library('pagination');**

**4.执行分页方法**

**$this->pagination->initialize($config);**

**5.输出分页链接**

**echo $this->pagination->create\_links();**

**6.查询部分数据(limit)**

**echo $this->db->limit($num,$start);  //从$start查$num条**

./applications/controllers/page.php

<?php

if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Page extends CI\_Controller {

public function user\_add(){

$this->load->model('test\_m');

for ($i = 1;$i <= 100;$i++){

$name = 'u'.$i;

$arr = array("usid"=>$i,"uname"=>$name,"upass"=>123456);

$this->test\_m->user\_insert($arr);

}

}

public function pagelist(){

$this->load->model('test\_m');

$user = $this->test\_m->user\_select\_all();

$allnum = count($user);

$pagenum = 20;

$config['total\_rows'] = $allnum;

$config['per\_page'] = $pagenum;

$config['num\_links'] = 3;

$config['base\_url'] = "/CI/index.php/page/pagelist";

$config['use\_page\_numbers'] = true;

$this->load->library('pagination');

$this->pagination->initialize($config);

var\_dump($this->pagination->create\_links());

echo $this->pagination->create\_links();

echo "<br />";

$id = $this->uri->segment(3); //获得url第三段字符

$id =$id ? $id:1;

$start = ($id - 1) \* $pagenum;

$list = $this->test\_m->user\_select\_limit($start,$pagenum);

var\_dump($list);

}

}