CI框架设置Layout布局

时间 2014-03-11 15:40:49  [CSDN博客](http://www.tuicool.com/sites/vQZRB3)

原文  [http://blog.csdn.net/wujindou/article/details/21017491](http://blog.csdn.net/wujindou/article/details/21017491?utm_source=tuicool&utm_medium=referral)

主题 [HTML](http://www.tuicool.com/topics/11060000)[PHP](http://www.tuicool.com/topics/11120000)

在使用CI的过程中，发现没有很好的布局使用，而如果每个文件都手工添加有太浪费时间和效率，回顾以前的项目记得曾经做过CI的layout的设置，

网上查找下还真找到了。。记录下使用技巧，以后有项目也可以参考使用。

首先定义全局的Controller，在项目的applcation/core目录下定义自己的Controller，不过要使用MY\_开头也就是必须命名为：MY\_Controller当然也可以自己

在config.php中修改  $config['subclass\_prefix'] = 'MY\_'; 的内容为自己项目名称，网上也有很多把layout单独开发作为类库加载的，就是单独做了

Library下的文件Layout.php 不过需要在autoload中指定自动加载，可以根据自己爱好选择。。这里只介绍自定义全局控制器方法实现.

一、在application/core目录下定义MY\_Controller内容如下：

**<?php** **if** ( ! defined('BASEPATH')) **exit**('No direct script access allowed');

**class MY\_Controller extends CI\_Controller**

**{**

**protected** $layout = 'layout/main';

**private** $js\_files = **array**();

**private** $css\_files = **array**();

**public** **function** **\_\_construct**()

{

**parent**::\_\_construct();

}

**public** **function** **add\_js**()

{

*//添加js文件*

}

**public** **function** **add\_css**(){

*//添加css文件*

}

*//file 表示是否使用渲染子视图文件，viewData表示的是子视图中渲染数据，$layout表示父视图中使用的全局数据*

**protected** **function** **render**($file = NULL, &$viewData = array(), $layoutData = array())

{

**if**( !is\_null($file) ) {

$data['content'] = $this->load->view($file, $viewData, **TRUE**);

$data['layout'] = $layoutData;

$this->load->view($this->layout, $data);

} **else** {

$this->load->view($this->layout, $viewData);

}

$viewData = **array**();

}

}

其中css\_files和js\_files用来渲染js和css文件，这里还没有实现后续需要会逐步完善。

二、在layout下建立 main.php文件，内容如下

<html>

<head>

<title>**<?php echo $layout['title'];?>**</title>

</head>

<body>

**<?php echo $content?>**

</body>

</html>

三、使用

在自定义controller中继承自MY\_Controller，添加比如下面测试代码：

**public** **function** **layout**(){

$data= **array**('user\_name'=>'张三','password'=>'密码');

$this->render('test\_layout',$data,**array**('title'=>'测试布局'));

}

这样视图会首先渲染test\_layout中的数据，然后把数据作为内容传递给父视图中。test\_layout.php代码如下

<h1>**<?php echo $user\_name;?>**</h1>

效果图：



实现其实并不复杂，关键是这样的思想和设计方法。。。