# 案例（项目）流程

## 需求

要做什么

我们：

带有登录功能的商品管理模块。

（参考成型的ecshop产品）

## 设计

怎么去做

架构

我们：

Lamp

## 编码实现

做

二次开发：

在一个成熟（常规业务逻辑实现）的产品上，完成自己项目的特定业务逻辑。

论坛（discuz，phpwind）

电商（ecshop，ecmall，shopex）

内容管理CMS（dede，wordpress）

社交（ThinkSNS，UCenter Home）

基于框架开发：

框架：规范了结构，基础功能 但 没有常规业务逻辑实现一种产品。

（ZendFramework，ThinkPHP，YII，symfony，CI 等等）

我们：

选择，基于框架开发。

基于 自定义的框架完成 业务逻辑。

## 测试

解决程序漏洞

## 发布

运营

## 维护升级

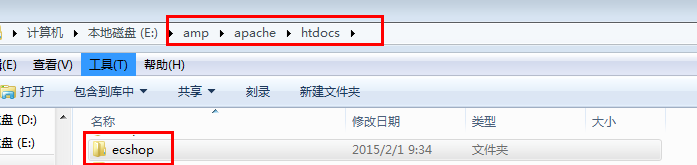
# 安装ecshop来参考

## 下载解压





## 上传到web服务器



## 通过web浏览器请求该项目

[可以设置虚拟主机]

请求后，就会自动进入安装页面，按照提示步骤安装即可!

必要环境：



如果mysql不支持，开启mysql扩展：php.ini



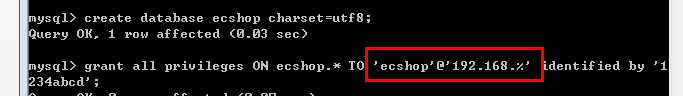
如果GD不支持，开启GD扩展：php.ini

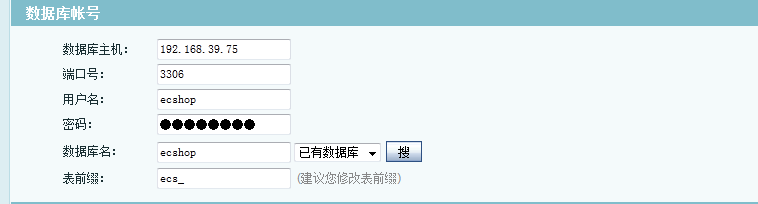


重启apache服务器！

配置数据库：

指定数据库，并设置好相应的用户权限。

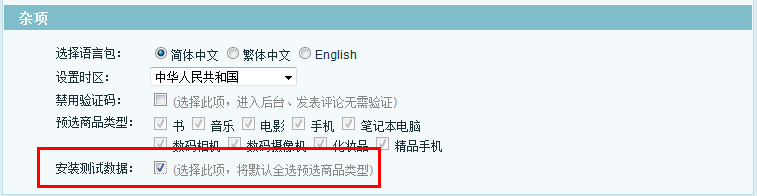




后台超级管理员帐号：



杂项，选中测试数据



测试访问：

前台：ecshop/index.php

后台：ecshop/admin/index.php

如何管理项目中大量的PHP代码(PHP文件)？

# 基本实现

功能，比赛列表，球队信息

数据表初始化：

比赛列表：

球队一 比分 球队二 比赛时间

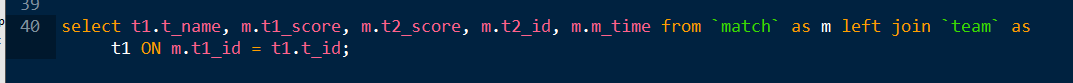
韩国 1：2 澳大利亚 15-01-31 17:00

从SQL思考，如何完成上面的查询：

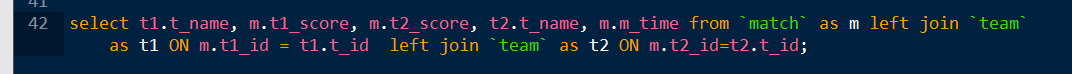
获得所有的比赛信息：

select \* from `match` ;

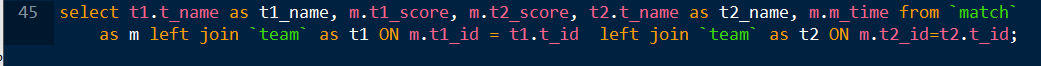
获得球队一名字：



获得球队二的名字：一个表在一次查询中可以出现多次，需要使用不同的别名进行标识！



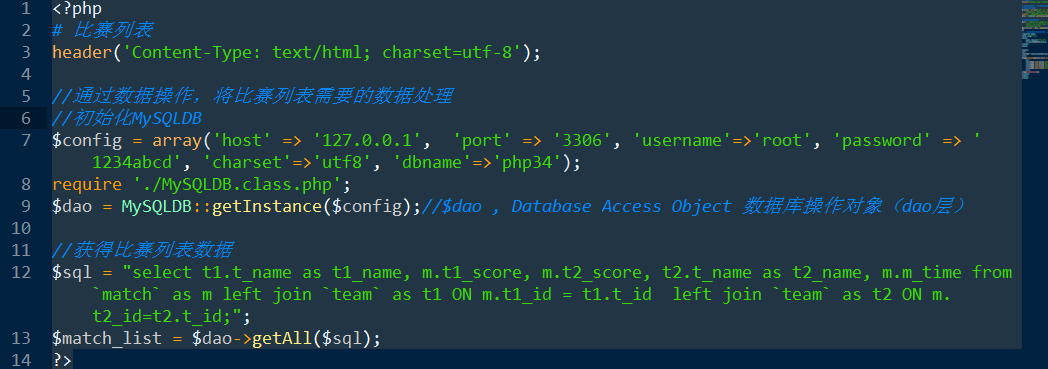
由于列名相同，需要起列别名：



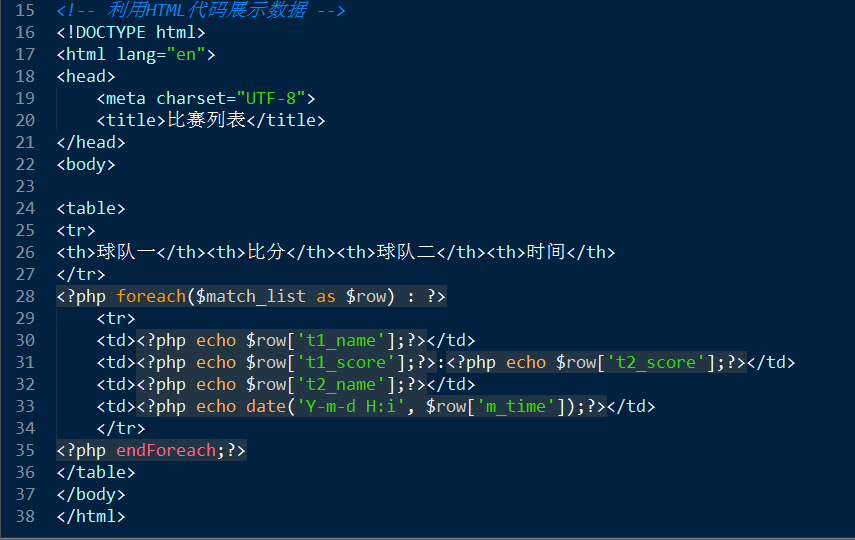


## PHP代码完成比赛列表

处理数据



展示：

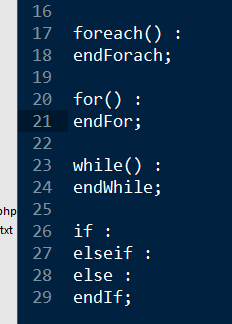


注意：

PHP逻辑最好与HTML展示分离出来

在HTML中嵌套PHP代码时，仅仅将需要PHP处理的内容，在PHP标签中输出，其他的都在PHP标签外输出。便于后期更新

模板中使用流程控制结构，建议使用 模板语法 来替代 大括号语法，包括代码段！



# 显示逻辑相分离

将功能强制分成两个部分，显示html文件，和逻辑PHP文件：

要求浏览器请求负责功能的PHP逻辑文件：

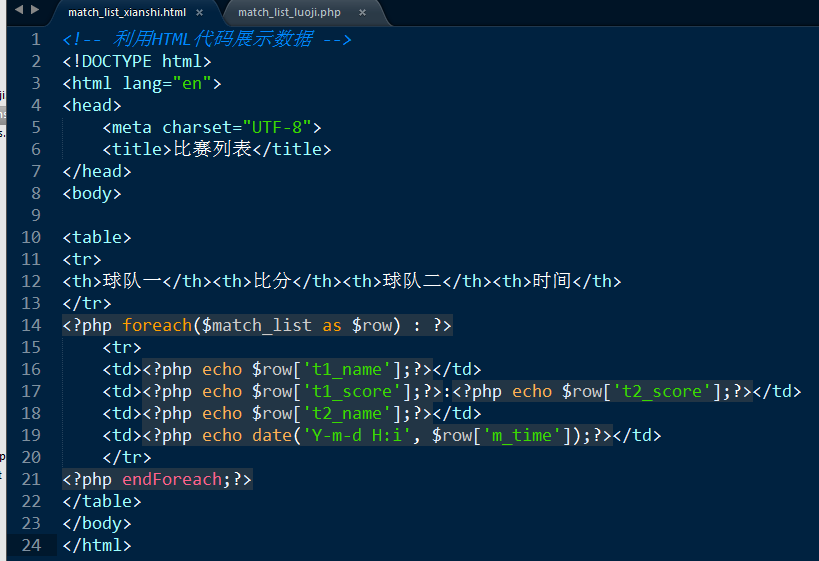
该PHP逻辑文件，对需要的显示内容进行载入。

s

逻辑PHP：



显示HTML：



浏览器请求：



好处：

为了更好的管理大量的代码：更新，重用 变得容易。

## 模板文件 – template

HTML，负责显示部分功能文件，称之为模板文件。

指的是：结构由HTML相关代码确定，数据部分由动态脚本（PHP）来填充，这样一种混编文件称之为模板。

“混编”：模板中PHP代码，仅仅负责输出数据，而一定不要负责处理数据。

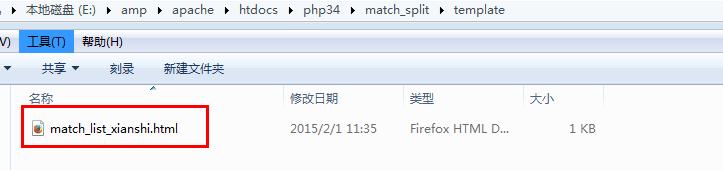
注意：浏览器应该，（不能）去请求负责展示的模板文件：

可见，需要将不能被浏览器所访问（模板）的文件隐藏起来，方案如下：

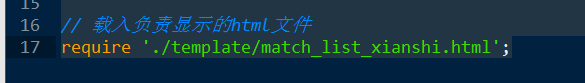
A计划：

通过web服务器对请求的控制，不允许浏览器直接访问

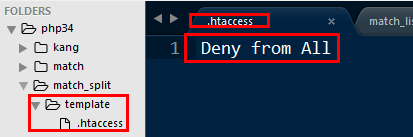
通常将不要被访问的统一管理到某个位置：



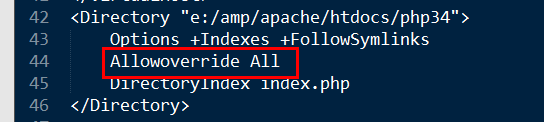
PHP代码载入正确的模板地址：



控制目录权限：通过apache的分布式配置文件完成



Tip:以上的配置，依赖于apache的主配置文件，允许目录内使用分布式配置文件.htaccess



B计划：

直接将不要被浏览器访问的文件资源，放在文档根目录之外！

因为浏览器可以请求的最上级目录，就是文档根目录了！