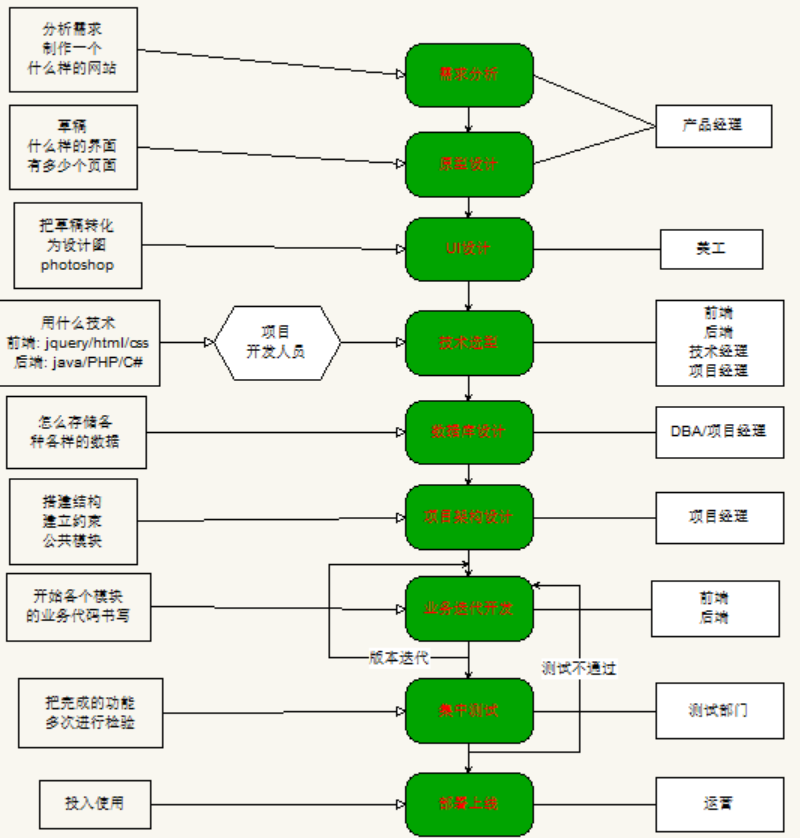
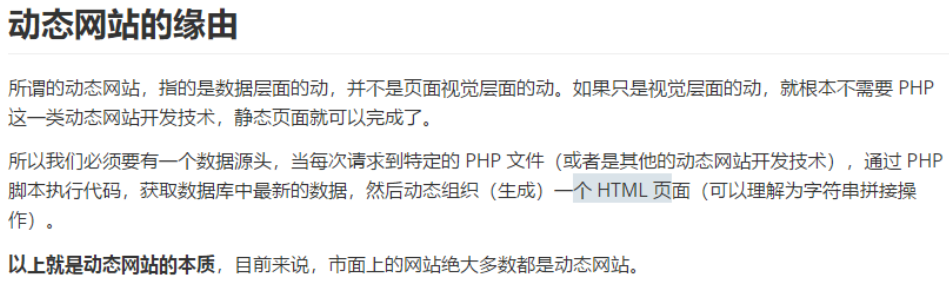
## 01-课程目标介绍



**1、项目开发的流程**





## 02-项目展示和几个概念的介绍

**本项目分为前台页面网站和后台管理系统**

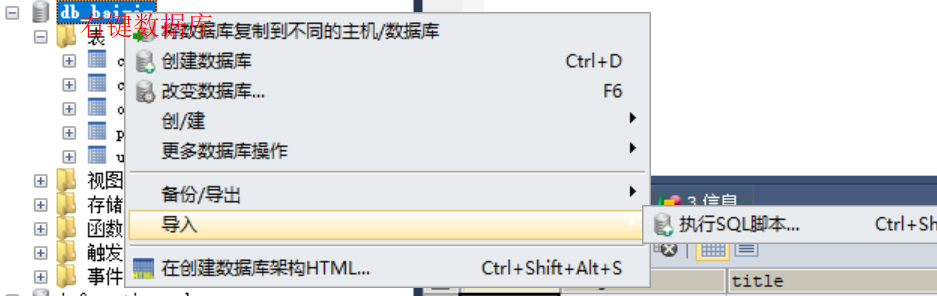


## 03-介绍数据库表结构及创建项目数据库

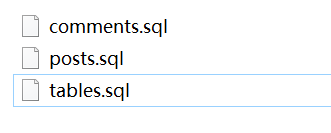
**1、需要创建数据库表有哪些？**

* 1. **选项表 (options) 用于记录网站的一些配置属性信息，如站点标题、站带你描述等**
  2. **用户表 (users) 用于记录用户信息**
  3. **文章表 (posts) 用于记录文章信息**
  4. **分类表 (categories) 用于记录文章分类信息**
  5. **评论表 (commentd) 用于记录文章评论信息**

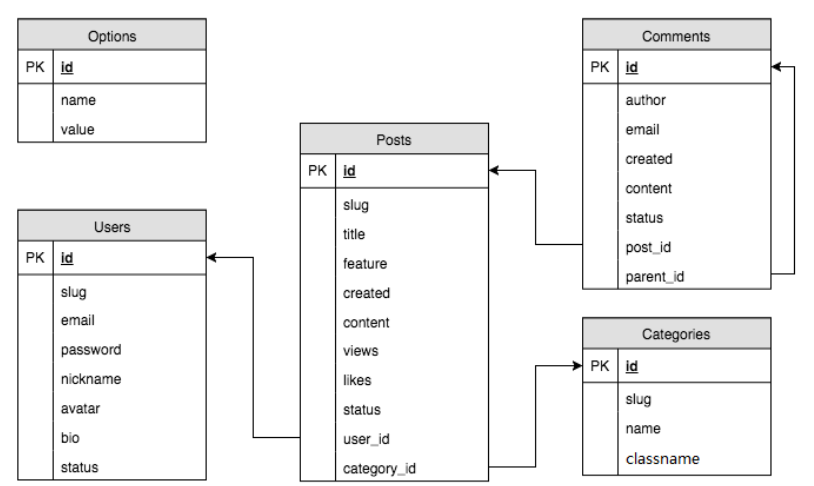
1. **如何创建数据库表**



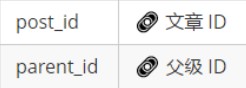
导入资料中的sql文件



## 04-介绍数据库设计



**1、需要联表查询的有**

* 1. 文章表 
  2. 评论表 

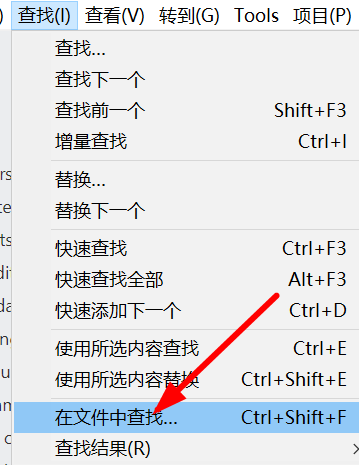
## 05-搭建项目结构和修改文件的路径

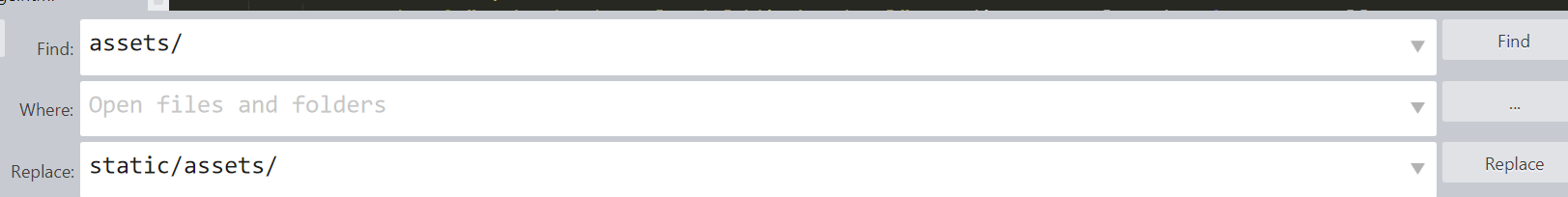
**前台是所有人看得到的**

**后台是只有管理员才可以看得到**



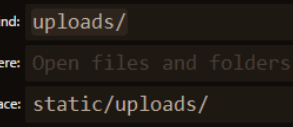
1. **路径修改步骤**
   1. 新建一个static文件
   2. 将assets和uploads剪切到static中
   3. 点击查找





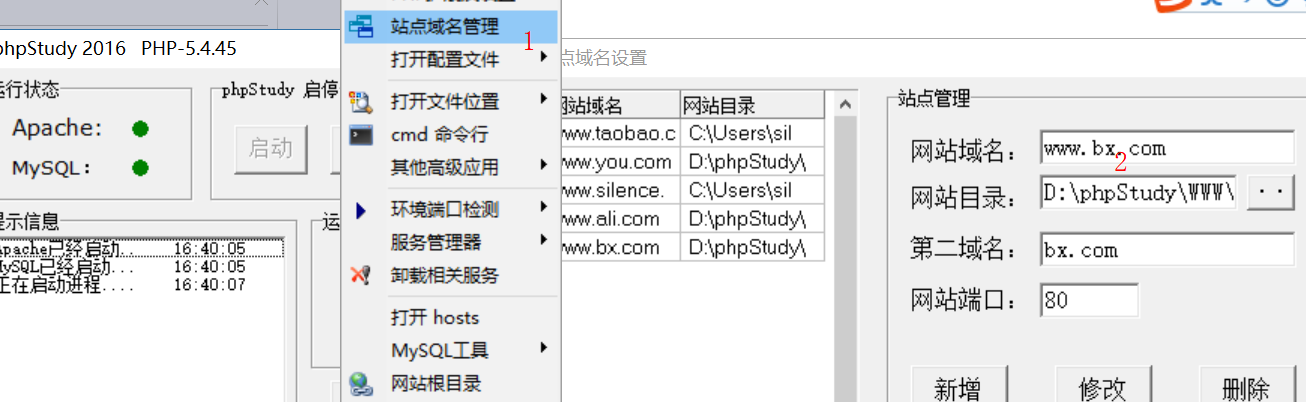
再点击**保存全部**即可全部变成 static/assets/

* 1. 再把uploads改成static下面的uploads即： static/uploads

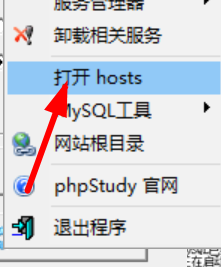
快捷键：

## 06-部署项目和修改静态页面为动态页面

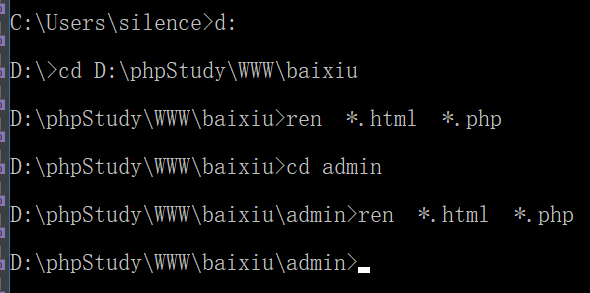
1. **如何配置虚拟主机**
   1. **第一步**



* 1. **第二步**

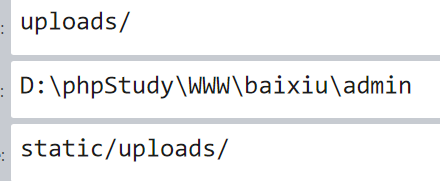
1. **如何将.html文件转变成.php文件**
   1. **cmd命令打开黑窗口**
   2. **找到项目所在的路径，注意盘符，**
      1. **比如打开的时候是C盘，项目在D盘 ，需要输入命令D：回车**
      2. **cd 加项目路径**
   3. **在输入 ren \*.html \*.php**

****

1. **将链接跳转地址.html改成 .php**

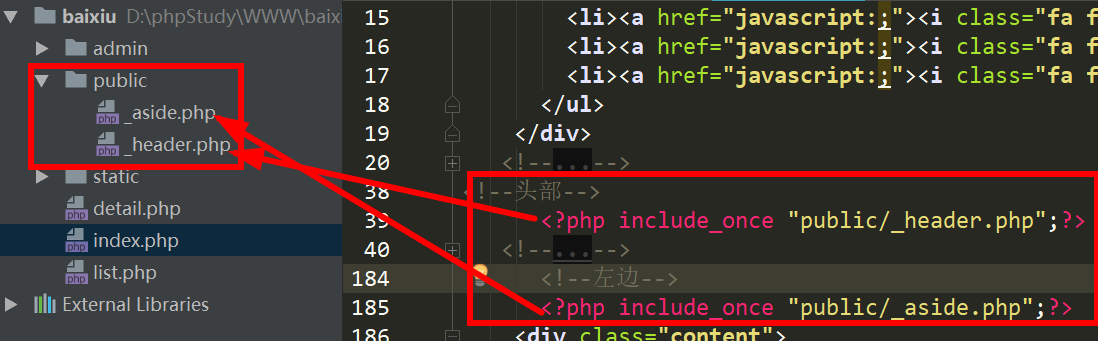


1. **用户图像地址的更改**



## 07-抽离公共模块

1. **如何提取公共部分**
   1. **把公共的header、aside标签中的代码分别提取放入public目录中： \_header.php和\_aside.php**
   2. **Include\_once “引入文件的路径”**



## 08-动态生成分类导航

**1、配置文件**



// include:1.如果文件载入失败，也不会影响后续代码的执行 2.如果重复载入文件那么被载入的文件的代码会真正的重复执行

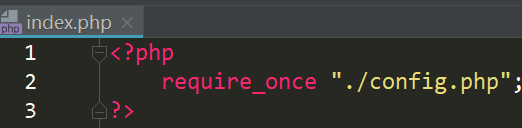
// include\_once:1.如果文件载入失败，也不会影响后续代码的执行 2.如果重复载入文件，最终也只会载入一次

// require：1.如果文件载入失败，那么后续不再执行 2.如果重复载入文件那么被载入的文件的代码会真正的重复执行

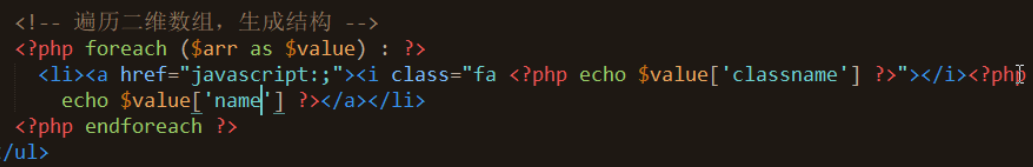
// require\_once:1.如果文件载入失败，那么后续不再执行 2.如果重复载入文件，最终也只会载入一次

**2、步骤分析：**

1. 设置配置文件config.php
2. 在index.php中导入config.php
3. 在公共头部文件中\_header.php中操作以下
   1. 连接数据库
   2. 查询分类数据
   3. 遍历循环数据得到一个二维数组



1. 渲染数据
   1. 遍历得到的数组 **(注意foreach语法:以：foreach:开头 endforeach结尾)**



1. **为什么在\_header.php中导入数据库连接时可以获取主机，密码等信息？**

**因为在index.php中先导入了config.php ,而在index.php文件的下面又引入了\_header.php，所以引入 config.php在前，引入\_header.php在后故可以获取数据信息**

## 09-最新发布功能完成

**查询sql语句**

select p.title,p.feature,p.created,p.content,p.views,p.likes,c.`name`,u.`nickname` from posts p

LEFT JOIN categories c ON c.`id`=p.`category\_id`

LEFT JOIN users u ON u.id=p.`user\_id`

WHERE p.`category\_id`!=1

ORDER BY p.`created` DESC

LIMIT 5

**注意：p , c 是一个别名方便找字段，可随意起名**

**1、多表数据库查询 思路分析：**

1. 查询文章表中的 p.title,p.feature,p.created,p.content,p.views,p.likes和 分类表中的名称和用户表中的昵称 分别给每个表起一个别名 posts p , categories c

2、关联通过文章表和分类表中的id来查询 c.`id`=p.`category\_id`

3、关联通过用户id和文章id来进行查询 u.id=p.`user\_id` ,users u

4、条件查询分类id不等于1 即不查询名称为未分类的

5、按照创建的时间来查询 时间为倒序，即距离我们最近的显示在最上面

5、限制条数为5条 LIMIT 5

**2、评论条数的查询**

**查询comments表中id的个数，通过**post\_id=p.id来查询，一篇文章对应一个评论，查询后的数量用字段commentsCount表示



SELECT p.title,p.feature,p.created,p.content,p.views,p.likes,c.`name`,u.`nickname`,

(SELECT COUNT(id) FROM comments WHERE post\_id=p.id) AS commentsCount

FROM posts p

LEFT JOIN categories c ON c.`id`=p.`category\_id`

LEFT JOIN users u ON u.id=p.`user\_id`

WHERE p.`category\_id`!=1

ORDER BY p.`created` DESC

LIMIT 5