**PHP核心编程第二天—PHP的MySQL扩展**

目录

[一、 （回顾）操作MySQL数据库 4](#_Toc484794163)

[1. 访问数据库方式 4](#_Toc484794164)

[1) 命令行客户端 4](#_Toc484794165)

[2) PHP操作操作数据库 4](#_Toc484794166)

[2. PHP的MySQL数据库扩展 5](#_Toc484794167)

[1) 开启MySQL数据库扩展 5](#_Toc484794168)

[2) 操作函数参考 5](#_Toc484794169)

[二、 PHP操作数据库步骤 6](#_Toc484794170)

[1. 步骤介绍 6](#_Toc484794171)

[2. 连接数据库服务器 6](#_Toc484794172)

[1) 语法 6](#_Toc484794173)

[2) 配置连接参数进行连接 7](#_Toc484794174)

[3) 判断连接，并给出提示 7](#_Toc484794175)

[4) 屏蔽错误 7](#_Toc484794176)

[3. 设置字符集 7](#_Toc484794177)

[1) 字符集名称 8](#_Toc484794178)

[2) 设置字符集的函数mysql\_query() 8](#_Toc484794179)

[4. 选择数据库 8](#_Toc484794180)

[1) 函数一 8](#_Toc484794181)

[2) 函数二 9](#_Toc484794182)

[3) 获取错误信息的相关函数 9](#_Toc484794183)

[5. 关闭连接 9](#_Toc484794184)

[6. 应用：封装连接函数db\_connect（） 9](#_Toc484794185)

[1) 配置conf.php 9](#_Toc484794186)

[2) 定义my\_connect函数，实现连接数据库、设置字符集及选择数据库 10](#_Toc484794187)

[三、 增删改操作 10](#_Toc484794188)

[1. 插入操作 10](#_Toc484794189)

[1) 增添数据 11](#_Toc484794190)

[2) 获取自增长字段新插入行的值： 11](#_Toc484794191)

[2. 删除操作 11](#_Toc484794192)

[1) 返回增删改的受影响行数 12](#_Toc484794193)

[3. 更新操作 12](#_Toc484794194)

[4. 总结：增删改注意事项 12](#_Toc484794195)

[四、 查询操作 12](#_Toc484794196)

[1. 查询结果集 12](#_Toc484794197)

[1) 空结果集情况 13](#_Toc484794198)

[2) 判断结果集相关函数 13](#_Toc484794199)

[2. 解析结果集的三种方式 14](#_Toc484794200)

[1) mysql\_fetch\_array(resource $res[,模式]); 14](#_Toc484794201)

[2) mysql\_fetch\_row($res); 15](#_Toc484794202)

[3) mysql\_fetch\_assoc($res); 16](#_Toc484794203)

[4) 三个函数的区别与相同点（面试） 16](#_Toc484794204)

[3. 以列表形式显示结果集 16](#_Toc484794205)

[1) 指针移动的原理 16](#_Toc484794206)

[2) 遍历结果集 17](#_Toc484794207)

[3) 结果集显示在表格中 17](#_Toc484794208)

[4. （了解）动静分离 18](#_Toc484794209)

[1) 动静分离应用 18](#_Toc484794210)

[2) 流程控制语言的替代写法 19](#_Toc484794211)

[五、 PHP59文章网项目初步 19](#_Toc484794212)

[1. 页面规划 19](#_Toc484794213)

[2. 数据库、数据表创建 19](#_Toc484794214)

[六、 文章添加模块 20](#_Toc484794215)

[1. 步骤分析 20](#_Toc484794216)

[2. 设计表单 20](#_Toc484794217)

[3. 接收、验证数据 21](#_Toc484794218)

[1) 拦截非法请求 21](#_Toc484794219)

[2) 接收、检查数据 21](#_Toc484794220)

[3) 处理文件上传 21](#_Toc484794221)

[4. 连接、选择数据库 22](#_Toc484794222)

[5. 构建、执行SQL语句 22](#_Toc484794223)

[6. 结果处理 22](#_Toc484794224)

[七、 文章列表模块 23](#_Toc484794225)

[1. 设计页面 23](#_Toc484794226)

[2. 连接、选择数据库 23](#_Toc484794227)

[3. 构建、执行SQL语句 23](#_Toc484794228)

[4. 展示数据 23](#_Toc484794229)

[八、 文章详情模块 24](#_Toc484794230)

[1. 接收文章编号 24](#_Toc484794231)

[2. 连接数据库、执行SQL语句 24](#_Toc484794232)

[3. 展示数据 24](#_Toc484794233)

[九、 文章修改模块 25](#_Toc484794234)

[1. 获取文章编号 25](#_Toc484794235)

[2. 查询当前数据 25](#_Toc484794236)

[3. 显示当前数据 26](#_Toc484794237)

[4. 修改、提交数据 26](#_Toc484794238)

[5. 处理结果 26](#_Toc484794239)

[十、 文章网之删除文章 27](#_Toc484794240)

**回顾**：

表单传值：GET方式及POST方式。

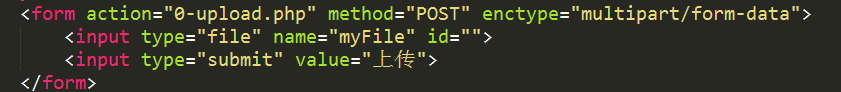
* GET方式及POST方式区别：GET传值数据上限为2kb,地址栏提交数据，安全性低。
* 常见的表单域：

单选项：type=’radio’

多选项：type=’checxbox’

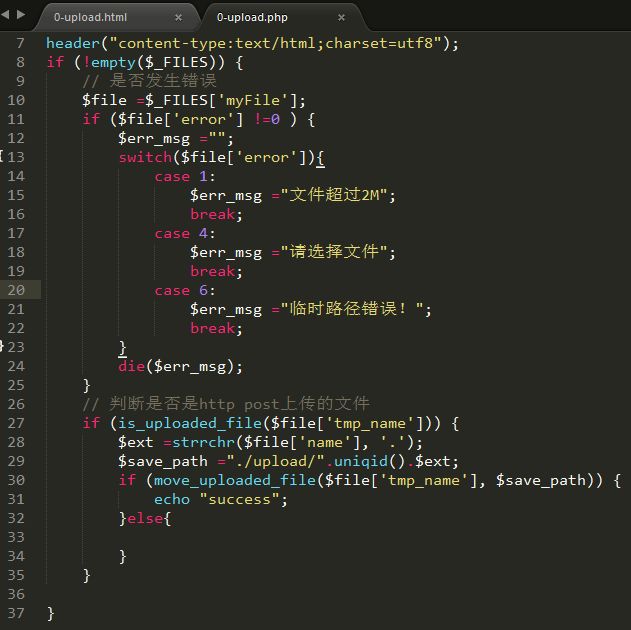
下拉菜单：select >option

* 上传文件：



* 文件上传的数据处理

1. 判断$\_FILES是否为空
2. 是否发生错误，如果有错误，给出提示。
3. 没有错误，判断是否是HTTP POST上传的文件
4. 将临时文件移动至网站的永久路径



**思考：**

* 如何建立正确的数据库连接？
* 增删改查语句执行的函数是什么？
* 增删改语句执行返回结果？查询返回的结果？
* 如何遍历结果集
* 获取结果集的行总共有几种方式？不同方式之间的区别为？

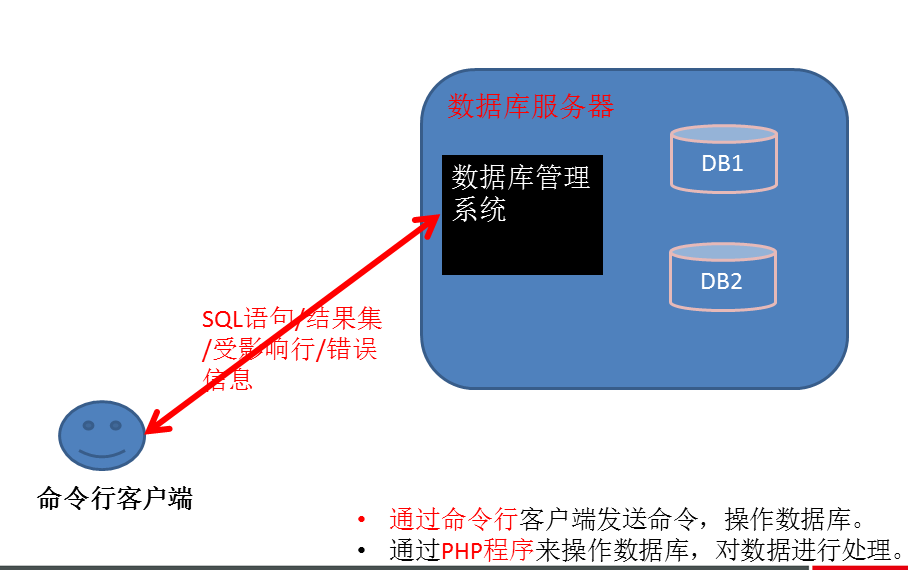
# （回顾）操作MySQL数据库



## 访问数据库方式

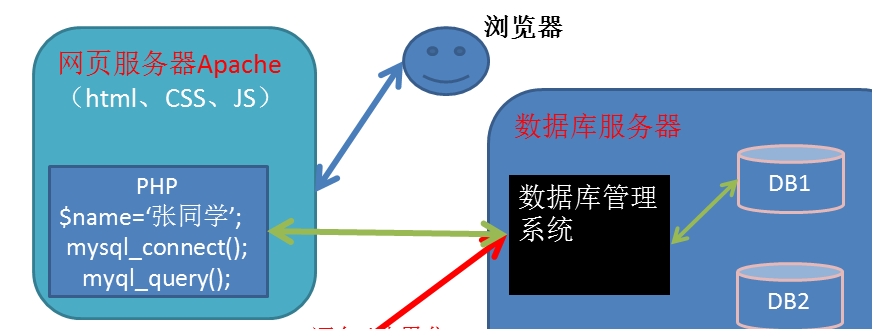
### 命令行客户端

通过客户端发送命令及接收返回的结果。



### PHP操作操作数据库

PHP通过专门的函数，建立起数据库连接。然后可以发送命令，对数据库的数据进行操作。



## PHP的MySQL数据库扩展

### 开启MySQL数据库扩展

扩展名称：php\_mysql.dll

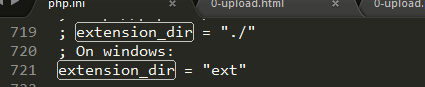
DLL文件：动态链路库文件（**Dynamic-link library**），是微软在windows操作系统中实现共享函数库的方式。程序需要时，加载dll文件到内存，进而调用其中的函数。

简单讲，DLL扩展就是封装了大量函数的函数库，可以按需调用。

**在php.ini中**

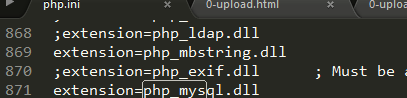
* 1. 查PHP扩展路径：

extension\_dir =”ext”

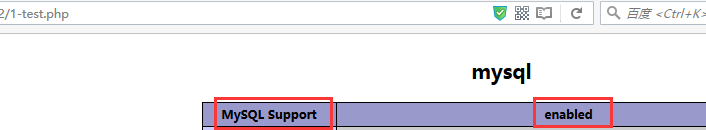


* 1. 加载扩展模块:

extension =php\_mysql.dll



* 1. 修改后，保存重启Apache。



使用mysql\_connect()函数检查加载是否成功

连接成功，返回资源型，失败返回false。

如果库函数未加载，提示“undefined function mysql\_connect”。

### 操作函数参考



# PHP操作数据库步骤

命令行操作数据库的步骤：

1. 连接数据库:参数：主机名，端口号，用户名，密码
2. 设置字符集
3. 选择数据库
4. 发送SQL
5. 检查结果

## 步骤介绍

* 1. 建立数据库连接
  2. 设置字符集：避免乱码的发生，set names gbk
  3. 选择数据库： use db\_name;
  4. 发送执行SQL语句
  5. 结果（集）、错误信息处理
  6. 断开连接:释放内存。

## 连接数据库服务器

### 语法

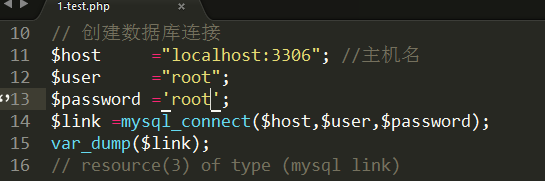
mysql\_connect(“主机名：端口号”,”用户名”,”密码”)；

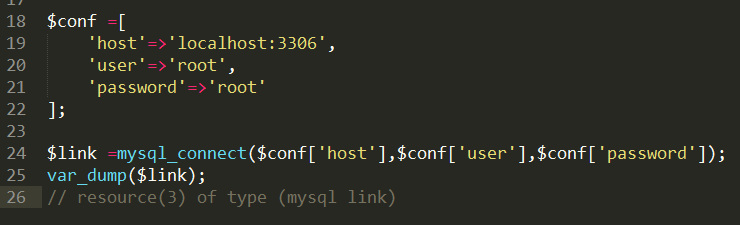
成功时，返回资源型的数据库连接。失败返回false。

资源型：是一种特殊变量，保存了到外部资源的一个引用(PHP操作的数据)。资源是通过专门的函数来建立和使用的。

资源型：给家人打电话建立的连接、打开冰箱的一刹那。

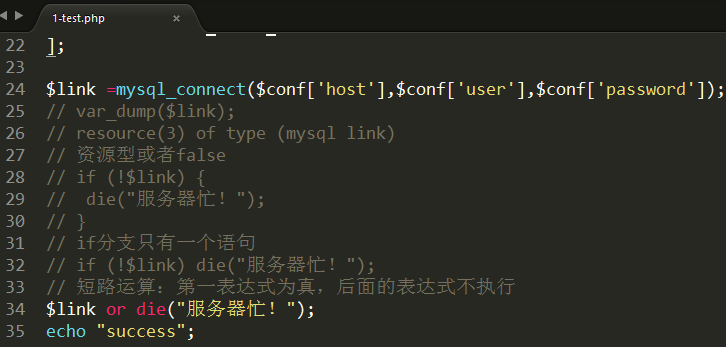
### 配置连接参数进行连接





### 判断连接，并给出提示

if语句的替换形式：短路运算。



### 屏蔽错误

使用错误抑制符@，抑制错误信息（防止网站路径暴漏）。



## 设置字符集

乱码问题的本质：字符集不一致。

### 字符集名称

数据库字符集设置遵循‘三码合一原则’：数据库、文件编码（PHP代码）及web页面显示尽量使用相同的字符集。

* PHP脚本编码为UTF8，
* 数据库服务器端默认字符集UTF8
* web显示字符集为UTF8

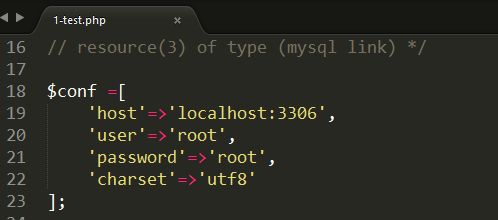
综上，与数据库交互的字符集：设置的字符集为：UTF8

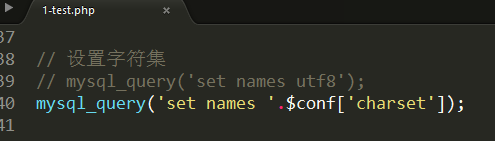
set names utf8;

### 设置字符集的函数mysql\_query()

mysql\_query($sql);

设置字符集，成功或者失败。

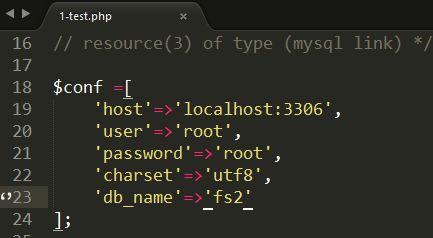




## 选择数据库

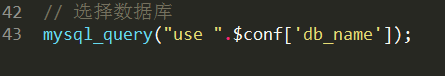
对数据进行操作，需要指定数据所在的数据库。

use db\_name;



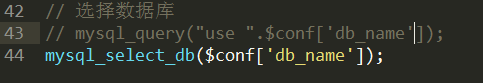
### 函数一

mysql\_query(“use $db”);



### 函数二

mysql\_select\_db(‘数据库名’);



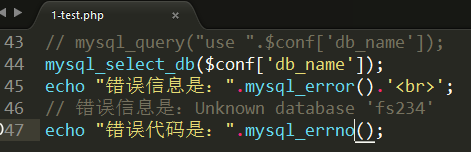
### 获取错误信息的相关函数

mysql\_error()

返回上一个SQL语句执行的错误信息。

mysql\_errno()

返回上一个SQL语句执行的错误代码。



**封装SQL语句错误提示的函数：**



## 关闭连接

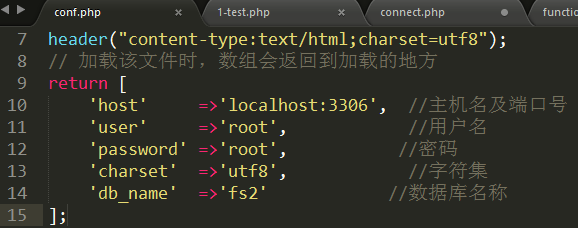
mysql\_close(resource [$res]);

关闭数据库连接，释放内存。

## 应用：封装连接函数db\_connect（）

可以在任意页面调用，提高代码复用率。

### 配置conf.php





### 定义my\_connect函数，实现连接数据库、设置字符集及选择数据库





# 增删改操作

基本的数据操作：CRUD,insert/update/delete/select

## 插入操作

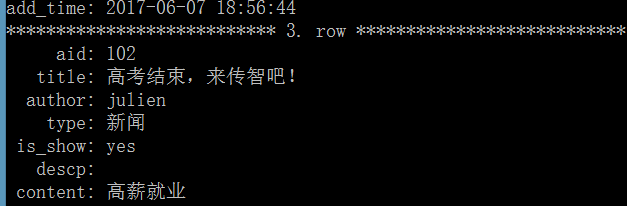
insert into 数据表 （字段列表） values (数值列表)；

### 增添数据

$sql =” insert into 数据表 （字段列表） values (数值列表)”；

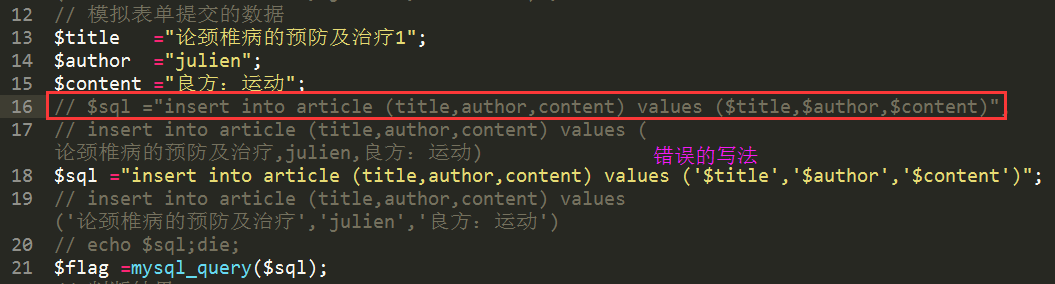
mysql\_query($sql);





**变量的情况：**

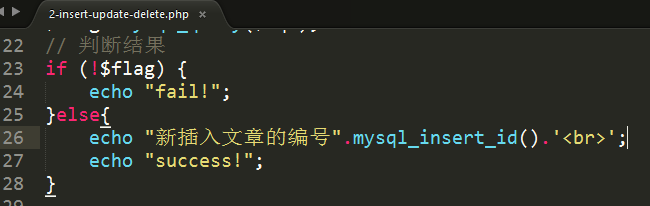
需要在数据为字符串时，在变量外添加引号。



### 获取自增长字段新插入行的值：

mysql\_insert\_id();

获取新插入行自增长字段的值。如果没有自增长字段，返回0.



## 删除操作

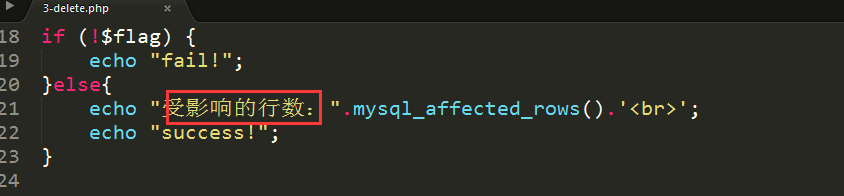
$sql =”delete from tb where条件”； //where条件不能省略

mysql\_query($sql);

### 返回增删改的受影响行数

mysql\_affetected\_rows();

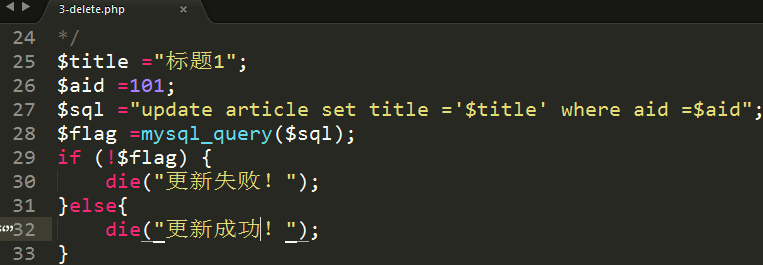
增删改操作的受影响行。



## 更新操作

$sql =”update tb set 字段1 =值1，字段2 =值2 where条件”；

Mysql\_query($sql);



## 总结：增删改注意事项

* 使用PHP发送SQL语句前，可以先打印SQL语句，检查语句的正确性。
* 使用变量拼接SQL语句时，字段为字符串类型，需要在变量的两侧使用单、双引号包裹。

# 查询操作

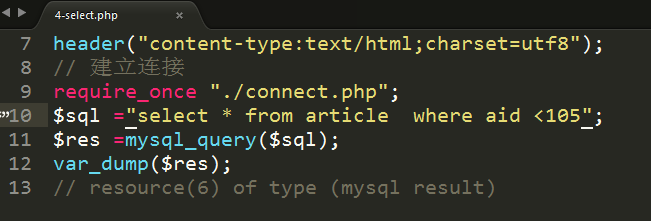
select 字段列表 from tb where 条件；

查询语句执行成功（包括空结果集）,返回资源型resource（结果集）。失败为false。

## 查询结果集

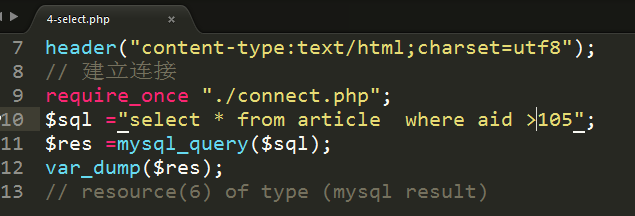
$sql =”select 字段列表 from tb where 条件”；

mysql\_query($sql);



### 空结果集情况

查询不到内容，返回空结果，但是类型是资源型。



### 判断结果集相关函数

以列表形式展示数据之前，需要筛选空结果集及非空结果集。

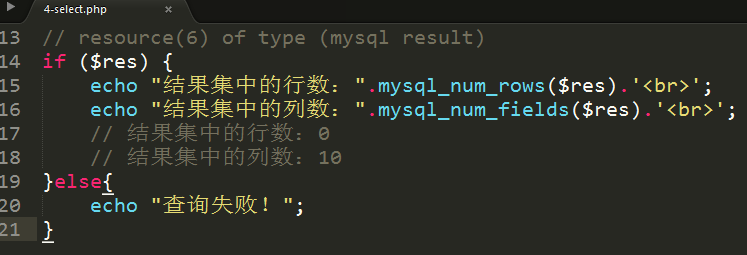
* 1. 返回查询结果集中的行数

mysql\_num\_rows($res);

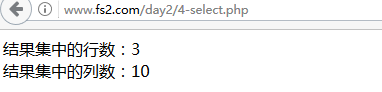
结果集的行数

②返回查询结果集中的列数

mysql\_num\_fields($res);



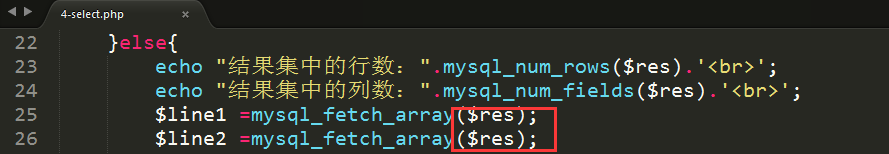


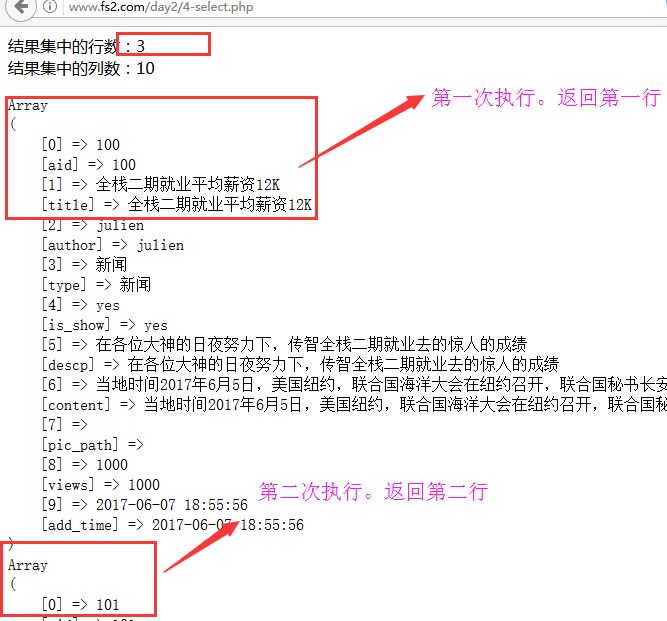


## 解析结果集的三种方式

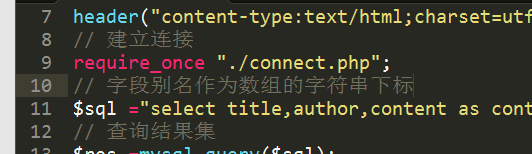
查询结果集数据类型为资源型。

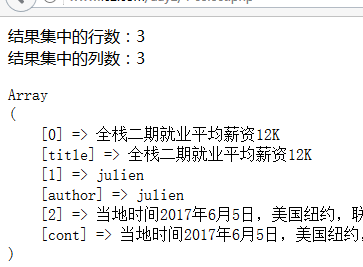
### mysql\_fetch\_array(resource $res[,模式]);





* 返回结果集中的一行，以混合数组形式返回。既有数字下标，又有字符下标（字段名称）。
* 使用查询字段的名称作为字符下标。字段列表中的别名字段同样显示（字段别名不使用汉字）。





* 函数的参数必须为查询结果集
* 每次执行移动一次指针，返回一行数据。
* 模式包括：MYSQL\_NUM（索引数组）,MYSQL\_ASSOC（关联数组）,MYSQL\_BOTH（混合数组）三种。

**指针移动的效果**：

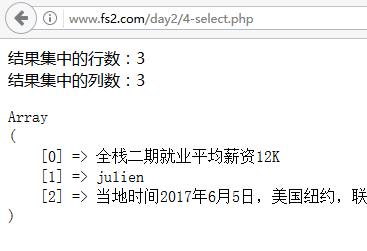


获取结果集中的全部行，使用循环语法。

### mysql\_fetch\_row($res);

* 返回为索引数组。





* 字段重复问题：mysql\_fetch\_row()可以解决下标重复相互元素覆盖的问题

$sql =”selec \* from tb1 join tb2 on tb1.id =tb2.id”;

### mysql\_fetch\_assoc($res);

返回关联数组，下标为查询字段的名称。



### 三个函数的区别与相同点（面试）

1. **相同点**：

每次执行，取出结果集中的一行（以数组的形式），并且将指针移动一次。指针非法时，返回false。

1. **不同点**：

* mysql\_fetch\_row()：返回索引数组，可以处理连接查询字段重名的情况
* mysql\_fetch\_assoc()：返回关联数组。相同字段名称，只出现一次，后者覆盖前者。可以设置别名。
* mysql\_fetch\_array()：

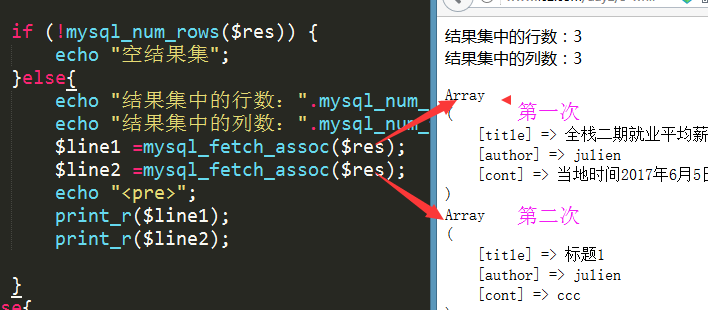
返回混合数组。

可以设置第二个参数，作为返回值的模式：MYSQL\_NUM（索引）,MYSQL\_ASSOC（关联）,MYSQL\_BOTH（混合）。较其他函数灵活。

## 以列表形式显示结果集

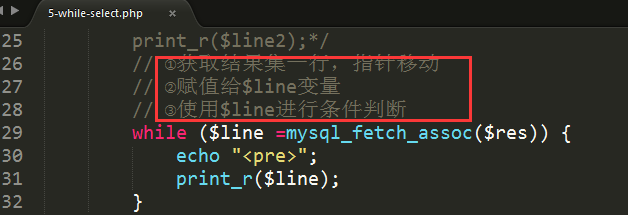
### 指针移动的原理

解析结果集的函数共同点：执行一次，读取一行，指针移动一次。



### 遍历结果集

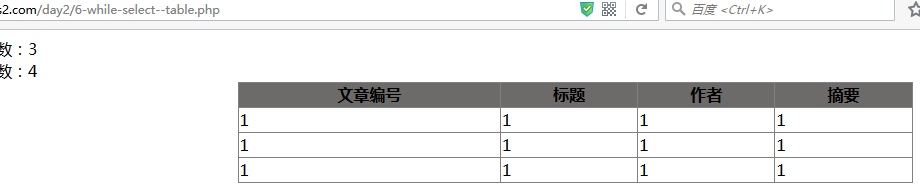
每次执行指针自动移动。指针非法，返回false。使用while循环获取每一行。



### 结果集显示在表格中



**先使用固定的数据填充表格的行：**



**使用正确的数据填充：**





## （了解）动静分离

### 动静分离应用

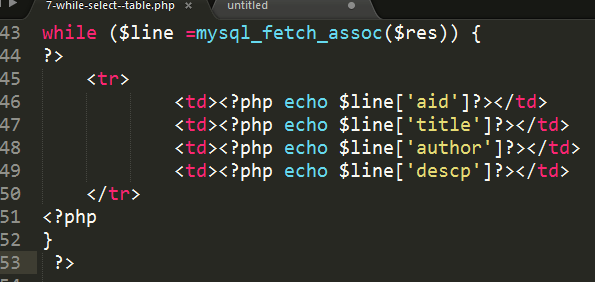
分离静态代码（HTML,CSS,JS）及动态代码（PHP），便于网站升级维护。

第一步:





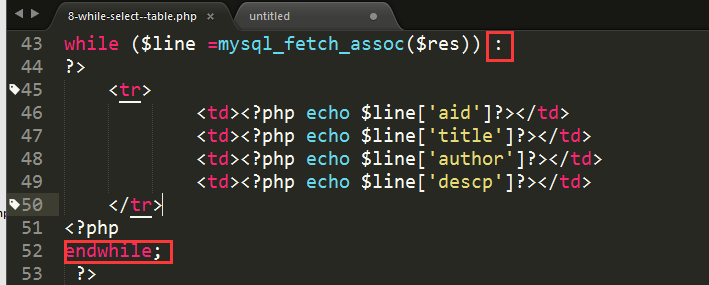
第二步：



### 流程控制语言的替代写法

将while,if及for的左大括号使用“：”替换，

右大括号分别使用endwhie;,endif;,endfor;替换



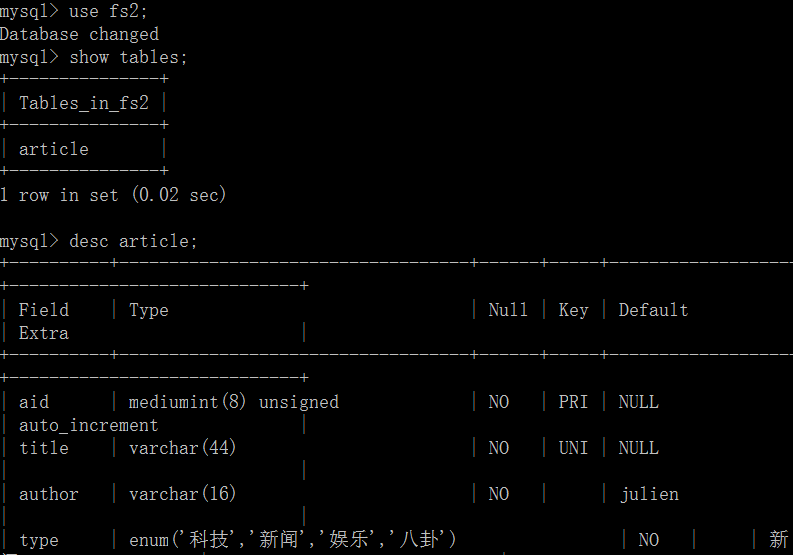
# PHP59文章网项目初步

核心逻辑：增删改查

## 页面规划

* 首页：index.php,列表页（展示全部的数据）
* 添加模块：表单add.php 数据处理insert.php
* 删除模块：delete.php 只需要文章编号
* 修改模块：表单页面edit.php (查询文章的原始数据—展示在表单中) 专门处理修改逻辑：update.php
* 查询模块：detail.php 显示文章的具体信息

## 数据库、数据表创建



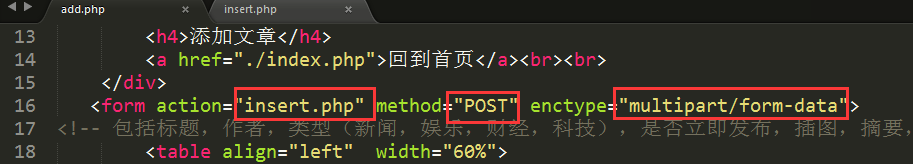
# 文章添加模块

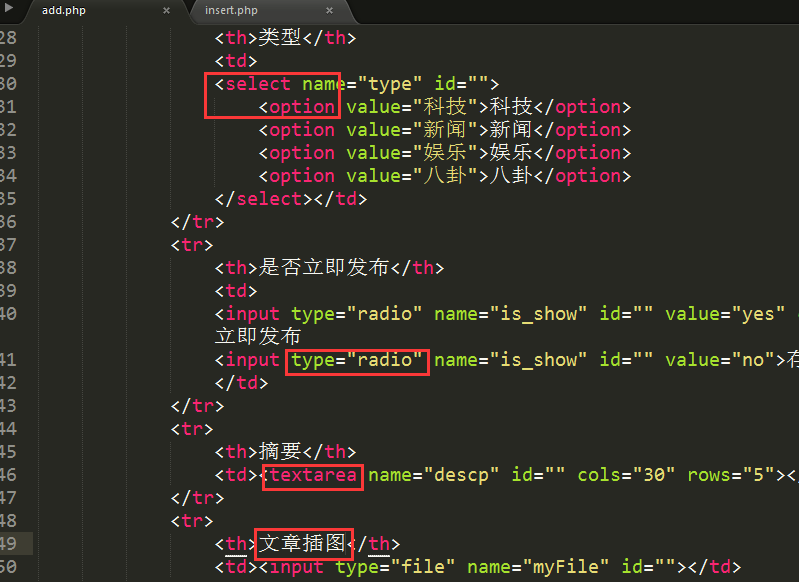
## 步骤分析

* 1. 数据库连接
  2. 表单设计
  3. 接收并检查数据的合法性及有效性(表单提交的数据、重要的信息[标题、内容]不能为空)
  4. 构建执行SQL语句
  5. 结果处理
  6. 关闭连接

## 设计表单

包括标题，作者，类型（新闻，娱乐，财经，科技），是否立即发布，插图，摘要，内容

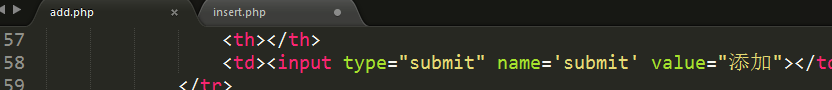




## 接收、验证数据

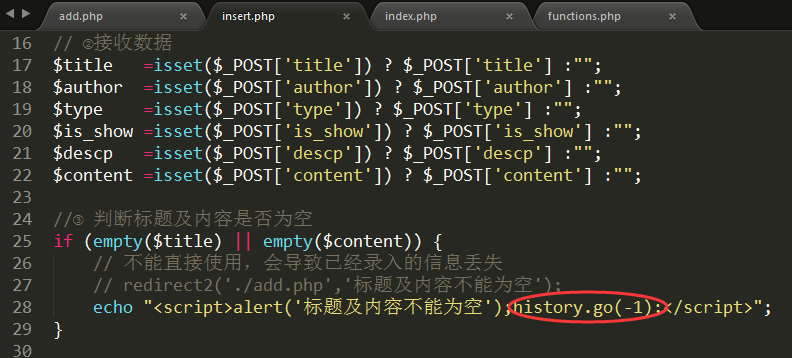
### 拦截非法请求

没有提交，不允许代码继续执行。





### 接收、检查数据

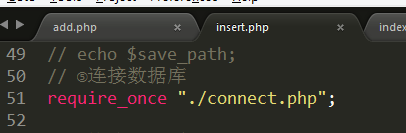


### 处理文件上传

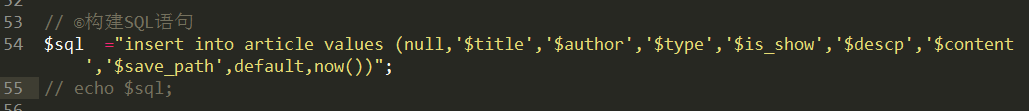


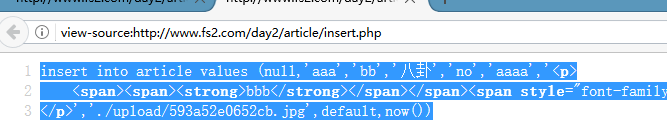
## 连接、选择数据库

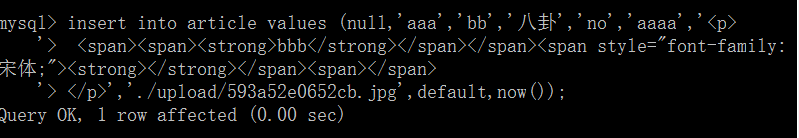
尽量放在靠后的位置。



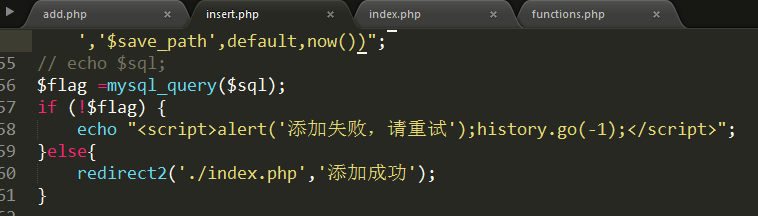
## 构建、执行SQL语句







## 结果处理

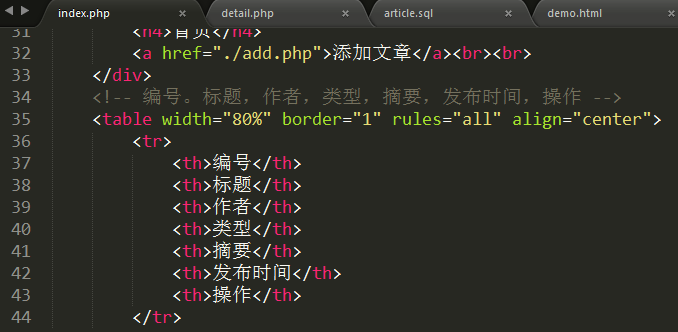


# 文章列表模块

核心：查询指定文章信息，展示在列表中。

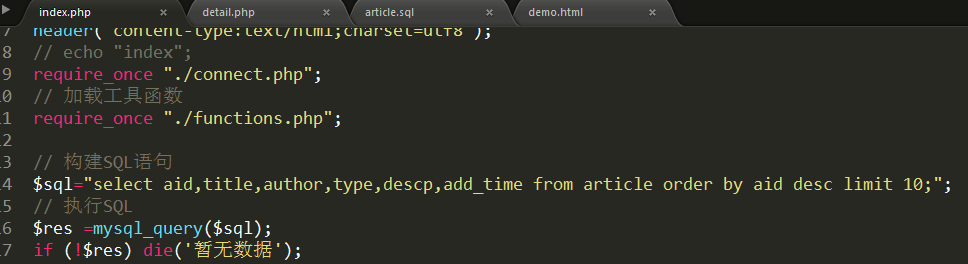
## 设计页面

表格的形式展示：



## 连接、选择数据库

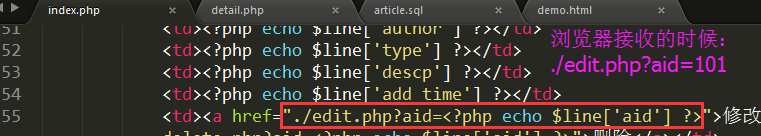
## 构建、执行SQL语句

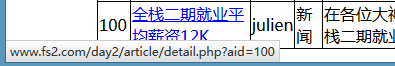


## 展示数据

需要完成三个连接：

* 标题链接：包含aid参数
* 修改链接：包含aid参数
* 删除链接：包含aid参数。

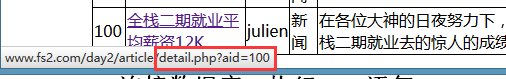




# 文章详情模块

## 接收文章编号

GET方式提交，$\_GET数组接收。



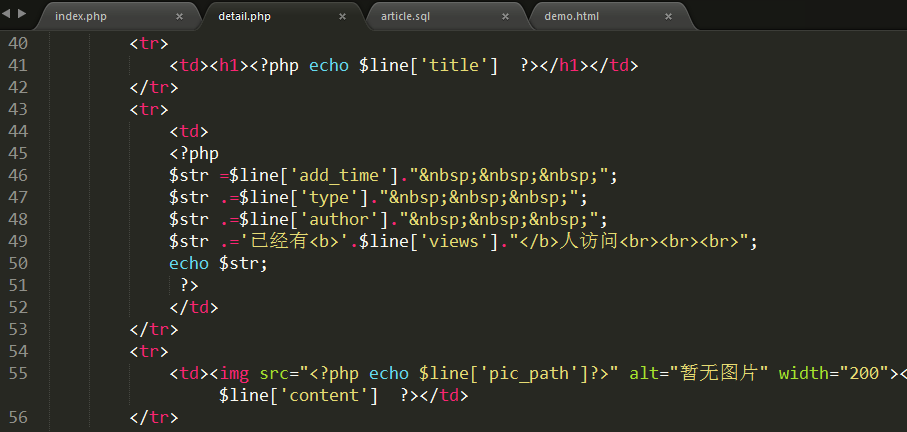
## 连接数据库、执行SQL语句



## 展示数据

展示布局分成三部分：使用三行一列的表格展示。



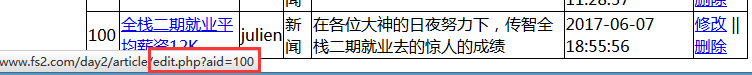


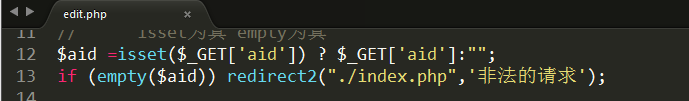


# 文章修改模块

难度在于：获取文章的数据---展示在表单（使用PHP语句输出值HTML对应位置）---用户修改后处理

## 获取文章编号





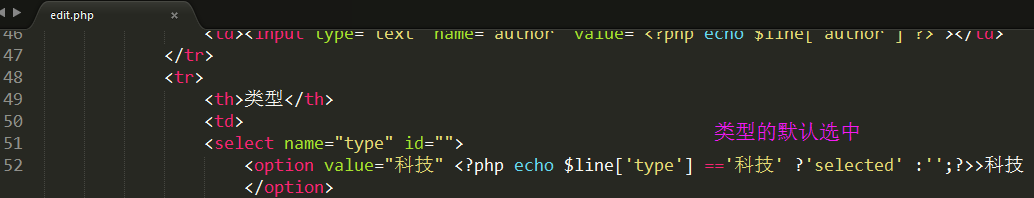
## 查询当前数据

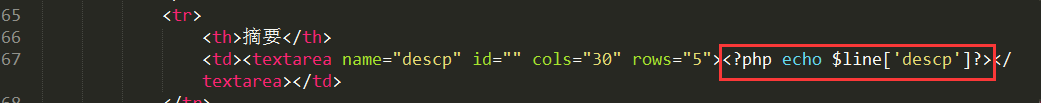
按照文章编号，查询文章的原始数据，作为后续显示的源数据。



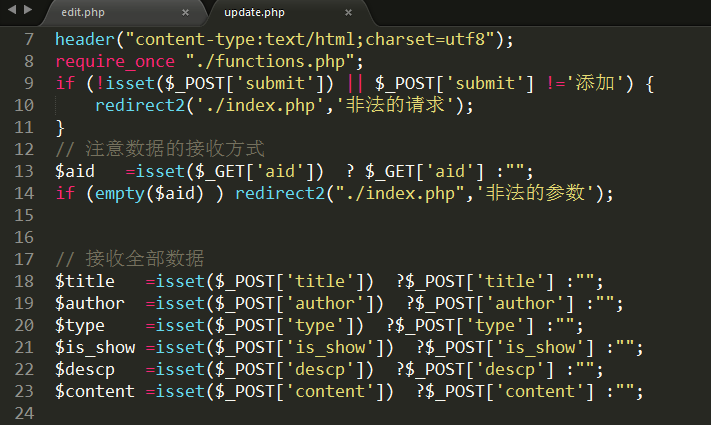
## 显示当前数据

将文章的原始数据显示在表单的指定位置，供后续修改操作使用。

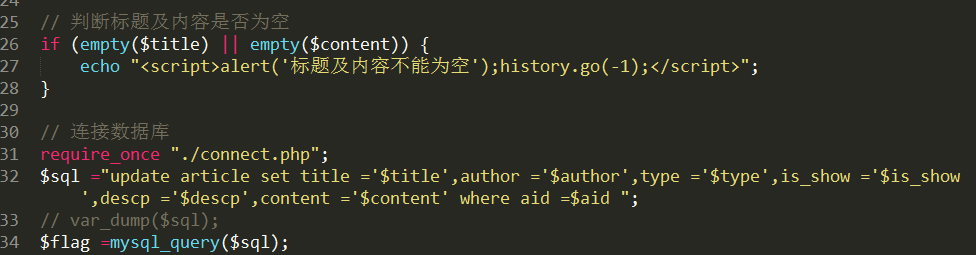




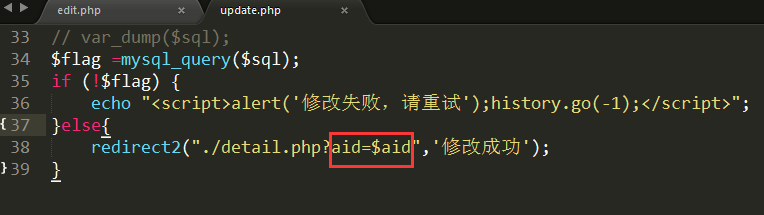
## 修改、提交数据



## 处理结果



跳转至详情页面，需要传递文章编号。



# 文章网之删除文章

接收aid🡪执行删除—返回至首页