使用PHP统计PHP的代码行数

# 开发环境

1、环境搭建：Windows 7+Apache 2.4.18+MySQL 5.7.11+PHP 7.1.0。

2、文本编辑器：Sublime 3。

# 二、主要技术

本案例主要使用PHP中的文件操作函数以及正则表达式来实现统计PHP项目中有效代码行数的功能。

# 三、效果图展示

实现效果图，如图1所示。

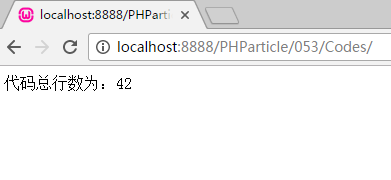


图1 效果图（统计有效代码行数）

# 四、具体步骤

## 4.1、创建PHP文件

创建一个名为“index.php”的PHP文件用来实现统计有效代码行数的功能。

## 4.2、统计代码有效行数

### （1）创建统计代码有效行数的函数

function code\_linenum($path, $i) {

......

}

### （2）判断是否是文件夹

if (!is\_dir($path)) {

return false;

}

### （3）匹配指定模式的文件名或目录

$files = glob($path . '/\*');

### （4）遍历文件或目录

if ($files) {

foreach ($files as $file) {

...... // 统计代码行数

}

}

### （5）判断给定文件名是否是一个目录

if (is\_dir($file)) {

// 递归调用code\_linenum()函数

code\_linenum($file, $i);

}

### （6）统计代码行数

$buffer = '';

// 以只读的方式打开文件

$handle = @fopen($file, 'r');

// 如果打开成功

if ($handle) {

// 检测是否已到达文件末尾

while(!feof($handle)) {

// 从文件指针中读取一行

$buffer = fgets($handle,4096);

// 移除字符串两侧的空格和换行、回车符

$buffer = trim($buffer);

if (!empty($buffer)) {

// 匹配是否是有效代码

$comments = array();

$comments[0] = '';

$comments[0] .=preg\_match("/^[\s]\*$/",

$buffer)?preg\_match("/^[\s]\*$/",

$buffer):''; // 匹配空行

$comments[0] .=preg\_match("/^[\s]\*\/\//",

$buffer)?preg\_match("/^[\s]\*\/\//",

$buffer):''; // 匹配两杠注释

$comments[0] .=preg\_match("/^[\s]\*(\/\\*).

\*(\\*\/)[\s]\*$/",$buffer)?preg\_match("/^[\s]\*

(\/\\*).\*(\\*\/)[\s]\*$/",$buffer):''; //匹配单行注释

$comments[0] .=preg\_match("/^[\s]\*(\/\\*).

\*/",$buffer)?preg\_match("/^[\s]\*(\/\\*).

\*/",$buffer):''; // 匹配多行注释

if (empty($comments[0])) {

global $i;

$i++; // 代码行数递增

}

}

}

fclose($handle); // 关闭文件

### （7）返回代码行数

return $i; // 返回代码行数

## 4.3、输出代码总行数

//调用函数，统计给定目录下的代码行数

global $i;

$linenums = code\_linenum('../Codes' ,$i);

echo '代码总行数为：' . $linenums; // 输出统计得出的代码函数

# 五、网页测试

开启Apache服务后，打开浏览器，在浏览器地址栏中输入文件地址，例如输入：“localhost:8888/PHParticle/053/Codes/”，可以看到程序运行结果如图2所示。

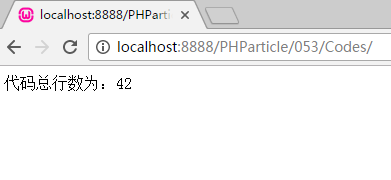


图2 案例结果图（统计有效代码行数）