

# PHP程序運行Python腳本 ( 接收數據及傳參 )

PHP在線 2022-08-31 00:00 發表於河南

收錄於合集

#php 5 #python 1

介紹一下，如何在php程序中運行Python腳本，  
在php中python程序的運行，主要依靠程序執行函數，  
這裡說一下三個相關函數：`exec()`，`system()` 和 `passthru()`。  
這裡主要講 `exec()` 函數，介紹使用該函數傳遞參數，  
以及如何使用python返回json數據供php使用。

## 一、`exec()` — 執行一個外部程序

```
1 exec ( string $command [, array &$output [, int &$return_var ]] ) : string
```

參數說明：

command：要執行的命令，其中包括三個子串，第一個子串為使用的當前系統的解釋器，第二個子串為所要執行腳本的位置，第三個子串為所需傳入的參數不限個數，中間用空格分隔，注意格式。子串間使用空格分割。

output：如果提供了output 參數，那麼會用命令執行的輸出填充此數組，每行輸出填充數組中的一個元素。（說明：output 中存放的並非python中return的值，並且所有return的值都不會進行保存，output 中存放的是python腳本中輸出的值，即為 `print()` 函數所輸出的所有數據）

return\_var：如果同時提供output 和return\_var 參數，命令執行後的返回狀態會被寫入到此變量。

### 1、直接運行

index.php

```
1 <?php
2 $re = exec('python ceshi.py', $out);
3 // $re = iconv('gbk', 'utf-8', $re);
4 var_dump($out);
```

```
5 echo '<br/>';  
6 var_dump($re);
```

ceshi.py

```
1 def send():  
2     data = '1,2,3,4,5'  
3     print(data)  
4 if __name__ == '__main__':  
5     send()
```

(重要說明：如果Python腳本返回的數據中含有中文，需要使用 `iconv('gbk', 'utf-8', $re)`；進行轉義)



← → ↻ ⓘ 不安全 | ceshi.test

```
array(1) { [0]=> string(9) "1,2,3,4,5" }  
string(9) "1,2,3,4,5" fengkui.net
```

## 2、傳參，接收返回數據

inde.php

```
1 $canshu1 = '这是PHP传过来的参数';  
2 $canshu2 = date('Y-m-d');  
3 $re = exec("python ceshi.py $canshu1 $canshu2");  
4 $re = iconv('gbk', 'utf-8', $re);  
5 var_dump($re);
```

test.py

```
1 import sys  
2 def send():  
3     # a1 = sys.argv[1]  
4     # a2 = sys.argv[2]  
5     re = sys.argv[1:]
```

```

6     data = '1,2,3,4,5,' + ','.join(re) # 转字符串
7     print(data)
8     if __name__ == '__main__':
9         send()

```

导入sys包，使用sys.argv[]数组获取传入参数，第一个传入参数为sys.argv[1]，第二个为sys.argv[2]以此类推，不要使用argv[0]

← → ↻ ⓘ 不安全 | ceshi.test

string(50) "1,2,3,4,5,这是PHP传过来的参数, 2020-07-04"  
fengkui.net

接收返回 json 数据：

```

1 import sys
2 import json
3 def send():
4     dict = {'id':111, 'title':'测试title'}
5     dict['data'] = sys.argv[1:]
6     jsonArr = json.dumps(dict, ensure_ascii=False)
7     print(jsonArr)
8     if __name__ == '__main__':
9         send()

```

(涉及到中文字符的时候，需要指定ensure\_ascii=False)

← → ↻ ⓘ 不安全 | ceshi.test

FeHelper

排序：默认 ☒ 升序 ☐ 降序 ☐ 乱码

```

{
  "id": 111,
  "title": "测试title",
  "data": [
    "这是PHP传过来的参数",
    "2020-07-04"
  ]
}

```

fengkui.net

## 二、system() — 执行外部程序，并且显示输出

```
1 system ( string $command [, int &$amp;return_var ] ) : string
```

同 C 版本的 `system()` 函数一样，本函数执行 `command` 参数所指定的命令，并且输出执行结果。

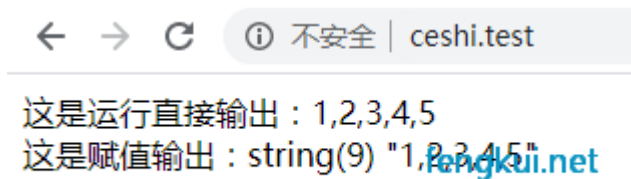
如果 PHP 运行在服务器模块中，`system()` 函数还会尝试在每行输出完毕之后，自动刷新 web 服务器的输出缓存。

如果要获取一个命令未经任何处理的 原始输出，请使用 `passthru()` 函数。

index.php

```
1 <?php
2 echo '这是运行直接输出：';
3 $re = system('python ceshi.py');
4 // $re = iconv('gbk', 'utf-8', $re);
5 echo '<br/>';
6 echo '这是赋值输出：';
7 var_dump($re);
```

这里使用最初版本的 test.py，输出效果如下：



## 三、passthru() — 执行外部程序，并且显示输出

```
1 passthru ( string $command [, int &$amp;return_var ] ) : void
```

同 `exec()` 函数类似，`passthru()` 函数 也是用来执行外部命令 (`command`) 的。当所执行的 Unix 命令输出二进制数据，并且需要直接传送到浏览器的时候，需要用此函数来替代

exec() 或 system() 函数。常用来执行诸如 pbmplus 之类的可以直接输出图像流的命令。通过设置 Content-type 为 image/gif, 然后调用 pbmplus 程序输出 gif 文件, 就可以从 PHP 脚本中直接输出图像到浏览器。

index.php

```
1 echo '这是运行直接输出：';  
2 $re = passthru('python ceshi.py');  
3 // $re = iconv('gbk', 'utf-8', $re);  
4 echo '<br/>';  
5 echo '这是赋值输出：';  
6 var_dump($re);
```

这里使用最初版本的 test.py，输出效果如下：



本文为冯奎原创文章，转载无需和我联系，但请注明来自冯奎博客fengkui.net



PHP在線

PHP在線專注於PHP編程語言學習，PHP開發經驗分享，工作問題解決以及PHP在線技...  
1篇原創內容

公眾號

收录于合集 #php 5

下一篇 · Laragon安装PHP8 · 相关新特性

喜欢此内容的人还喜欢

python进阶：fixture 二次开发



测试开发实战课



4个Linux技巧你值得拥有

入门小站



超全！Python 中常見的配置文件寫法

測試開發實戰課

