

PHP、Python、Ruby 的比较（一）

主题：数据库、文件的操作

侯惠阳（PHPer.yang）

我和大家讨论的是（PHP、Python、Ruby）这三种脚本语言各自的特性。今后每期主题不同，分别阐述他们的效率以及特点。这一期讨论的是：数据库、文件的操作。

一：PHP 部分：

A：读取数据库方面

PHP 对于数据库操作方面比较简单，可以使用 MySQL 给 PHP 写的扩展 mysql*、mysqli*，以及 PHP 官方自己写的 Mysqlnd。

这里其实我主要想说的是 mysql_connect();这个函数。注意第三个参数 new_link (bool)。如果用同样的参数第二次调用 mysql_connect()，将不会建立新连接，而将返回已经打开的连接标识。参数 new_link 改变此行为并使 mysql_connect() 总是打开新的连接，甚至当 mysql_connect() 曾在前面被用同样的参数调用过。

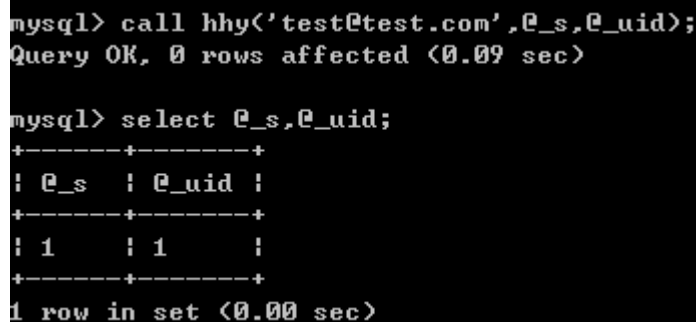
普通操作：

```
<?php
$conn = mysql_connect("localhost","root","",true);
mysql_select_db("test",$conn);
$query = mysql_query("select * from test");
while($rows = mysql_fetch_array($query)) {
    ...
}
...
```

带存储过程的操作：

Mysql

```
create procedure hhy(in yumail varchar(20),out _s int,out _uid int)
begin
    if exists(select * from user where umail = yumail) then
        set _s = 1;
        select uid into _uid from user where umail = yumail;
    else
        set _s = 0;
    end if;
end
```



```
mysql> call hhy('test@test.com',@_s,@_uid);
Query OK, 0 rows affected (0.09 sec)

mysql> select @_s,@_uid;
+-----+-----+
| @_s  | @_uid |
+-----+-----+
| 1    | 1     |
+-----+-----+
1 row in set (0.00 sec)
```

PHP 中使用：

```
mysql_query("call hhy('test@test.com',@_s,@_uid);");
mysql_query("select @_s,@_uid;");
```

...

这部分以后在详细叙述。

B: 文件操作

我这里想说的对于大文件的操作，比如一个文件有 300M。怎么读？

虽然可以在 PHP 中申请比较大的内存。

```
<?php
```

```
    ini_set('memory_limit', '300M');
```

但是如果服务器暂时不能提供很大的内存。需要怎么来优化读取大文件呢。

在统计系统的开发中，这个我和同事有一些交流和改进。我总结如下：

总结：对于 PHP 来说无论是库操作还是稳健操作都比较快。

二：Python 部分：

A: 读取数据库方面

Python 在这方面其实也比较简单。类似于 PHP。

以 Ubuntu 为例：

```
apt-get install python-mysqldb
```

代码：

```
#!/usr/bin/python
```

```
import MySQLdb
```

```
conn = MySQLdb.connect(host="localhost",user="root",passwd="",db="find")
```

```
cursor = conn.cursor()
```

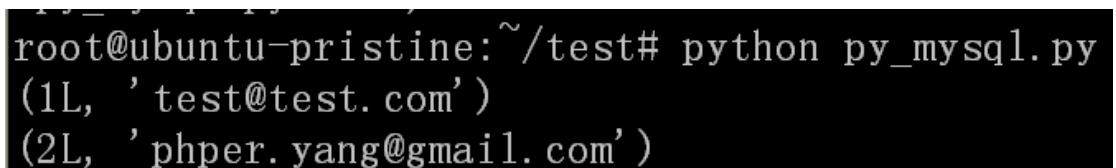
```
sel = cursor.execute('select uid,umail from user')
```

```
results = cursor.fetchall()
```

```
for i in results:
```

```
    print i;
```

```
conn.close()
```



```
root@ubuntu-pristine:~/test# python py_mysql.py
(1L, 'test@test.com')
(2L, 'phper.yang@gmail.com')
```

B: 文件操作

读取：

```
f = open("u.txt","r")
```

```
text = f.read()
```

```
print text
```

有木有发现和 PHP 很相似

```
<?php
```

```
    $fp = fopen('u.txt','r');
```

```
$c = fread($fp,filesize('u.txt'));  
echo $c;  
fclose($fp);
```

```
root@ubuntu-pristine:~/test# python fopen.py  
phper.yang@gmail.com  
admin@u85.us  
admin@imop.us
```

```
f = open('py.txt','w')  
f.write('phper.yang')  
f.close()
```

```
root@ubuntu-pristine:~/test# cat py.txt  
phper.yangroot@ubuntu-pristine:~/test#
```

和 PHP 的函数差不多，用法也差不多。

三：Ruby 部分：

A：读取数据库方面

Ruby 我了解有两种方式连接 mysql

使用 MySQL C 客户端 API。 <http://www.tmtm.org/en/mysql/ruby/>

Ruby Mysql DBI。 <http://rubyforge.org/frs/download.php/12368/dbi-0.1.1.tar.gz>

```
require'rubygems';  
require "mysql"  
conn = Mysql.real_connect("localhost","root","", "find")  
puts "Mysql version:" + conn.get_server_info  
res = conn.query("select uid,umail from user")  
while row = res.fetch_row do  
  print "%s,%s/n",row[0],row[1]  
end
```

B：文件操作

```
file = File.open("u.txt","r")  
while(lines = file.gets)  
  puts lines  
end
```

```
root@ubuntu-pristine: /test# /usr/bin/ruby1.8 fread.rb  
phper.yang@gmail.com  
admin@u85.us  
admin@imop.us
```

写文件:

```
file = File.new("yang.txt","w")
file.puts "phper.yang@gmail.com"
file.close
```

```
root@ubuntu-pristine:~/test# /usr/bin/ruby1.8 fwrite.rb
root@ubuntu-pristine:~/test# cat yang.txt
phper.yang@gmail.com
```

Ruby on rails 这个框架开发起来比较快速

Ruby install

gem install

gem install rails

比较快速的开发一个站点

进入项目文件夹

rails server

```
C:\Documents and Settings\ThinkPad>e:
E:\>cd rubywork/houhuiyang
E:\rubywork\houhuiyang>rails server
=> Booting WEBrick
=> Rails 3.0.7 application starting in development on http://0.0.0.0:3000
=> Call with -d to detach
=> Ctrl-C to shutdown server
[2011-07-10 13:42:24] INFO WEBrick 1.3.1
[2011-07-10 13:42:24] INFO ruby 1.9.2 (2011-02-18) [i386-mingw32]
[2011-07-10 13:42:24] INFO WEBrick::HTTPServer#start: pid=6036 port=3000
```



总结:

从皮上面看: 我们做了这三种语言在数据库 (mysql) 和文件的读写操作。我个人感觉在程序的难易程度以及书写上 PHP 比较简洁。由于 Ruby 和 Python 一样。一切皆是对象。所以其实在程序内部都是实例化的。消耗一些资源。据了解 PHP7 也将准备一切皆是对象的

改进。

从程序运行上来看：

先总结一些函数

PHP 的常用函数

```
mysql_connect();  
mysql_query();  
mysql_fetch_array();
```

```
fopen();  
fread();  
fwrite();  
等等
```

Python 的常用函数

```
open("filename","属性")  
句柄.fread(数量)  
句柄.fwrite()  
句柄.close()  
句柄.readlines()
```

```
MySQLdb.connect()  
句柄.cursor()  
cursor.execute 执行 SQL  
cursor.fetchone() 读取一行  
cursor.fetchmany(5) 读取 5 行  
cursor.fetchall() 所有  
句柄.close()
```

Ruby 的常用函数

```
File.open("filename","属性")  
句柄.gets 读取资源  
File.new() 写文件  
句柄.close
```

```
以 MYSQL API  
Mysql.real_connect()  
句柄.query()  
句柄资源.fetch_row
```

个人感觉 Python 和 Ruby 好多对象的大小写要求严格。所以必须注意。至于效率个人感觉 PHP 还是很有优势。书写简单，相应的资源也很多。下一期我们有机会讨论一下 **form 表单和抓取部分**。

个人网站

<http://www.imop.us>

<http://www.u85.us>