

## 使用MySQL

1079次阅读

---

MySQL是Web世界中使用的最广泛的数据库服务器。SQLite的特点是轻量级、可嵌入，但不能承受高并发访问，适合桌面和移动应用。而MySQL是为服务器端设计的数据库，能承受高并发访问，同时占用的内存也远远大于SQLite。

此外，MySQL内部有多种数据库引擎，最常用的引擎是支持数据库事务的InnoDB。

## 安装MySQL

可以直接从MySQL官方网站下载最新的[Community Server 5.6.x](#)版本。MySQL是跨平台的，选择对应的平台下载安装文件，安装即可。

安装时，MySQL会提示输入root用户的口令，请务必记清楚。如果怕记不住，就把口令设置为password。

在Windows上，安装时请选择UTF-8编码，以便正确地处理中文。

在Mac或Linux上，需要编辑MySQL的配置文件，把数据库默认的编码全部改为UTF-8。MySQL的配置文件默认存放在/etc/my.cnf或者/etc/mysql/my.cnf：

```
[client]
default-character-set = utf8

[mysqld]
default-storage-engine = INNODB
character-set-server = utf8
collation-server = utf8_general_ci
```

重启MySQL后，可以通过MySQL的客户端命令行检查编码：

```
$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor...
...
```

```
mysql> show variables like '%char%';
```

Variable_name	Value
character_set_client	utf8
character_set_connection	utf8
character_set_database	utf8
character_set_filesystem	binary
character_set_results	utf8
character_set_server	utf8
character_set_system	utf8
character_sets_dir	/usr/local/mysql-5.1.65-osx10.6-x86_64/share/charsets/

```
8 rows in set (0.00 sec)
```

看到utf8字样就表示编码设置正确。

## 安装MySQL驱动

由于MySQL服务器以独立的进程运行，并通过网络对外服务，所以，需要支持Python的MySQL驱

动来连接到MySQL服务器。

目前，有两个MySQL驱动：

- `mysql-connector-python`：是MySQL官方的纯Python驱动；
- `MySQL-python`：是封装了MySQL C驱动的Python驱动。

可以把两个都装上，使用的时候再决定用哪个：

```
$ easy_install mysql-connector-python
$ easy_install MySQL-python
```

我们以`mysql-connector-python`为例，演示如何连接到MySQL服务器的`test`数据库：

```
# 导入MySQL驱动：
>>> import mysql.connector
# 注意把password设为你的root口令：
>>> conn = mysql.connector.connect(user='root', password='password', database='test', use_unicode=True)
>>> cursor = conn.cursor()
# 创建user表：
>>> cursor.execute('create table user (id varchar(20) primary key, name varchar(20))')
# 插入一行记录，注意MySQL的占位符是%s：
>>> cursor.execute('insert into user (id, name) values (%s, %s)', ['1', 'Michael'])
>>> cursor.rowcount
1
# 提交事务：
>>> conn.commit()
>>> cursor.close()
# 运行查询：
>>> cursor = conn.cursor()
>>> cursor.execute('select * from user where id = %s', '1')
>>> values = cursor.fetchall()
>>> values
[(u'1', u'Michael')]
# 关闭Cursor和Connection：
>>> cursor.close()
True
>>> conn.close()
```

由于Python的DB-API定义都是通用的，所以，操作MySQL的数据库代码和SQLite类似。

## 小结

- MySQL的SQL占位符是%s；
  - 通常我们在连接MySQL时传入`use_unicode=True`，让MySQL的DB-API始终返回Unicode。
-