



上海外国语大学  
SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

# New Media Data Analytics and Application

Lecture 4: Advanced Python Programming

Ting Wang

- File and Stream I/O
- Connect to Database
- Web Programming





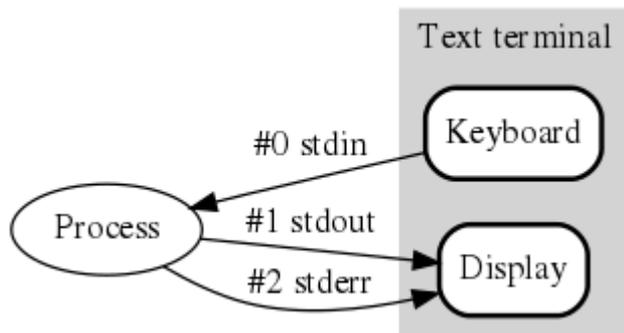
input text from web or documents and output it

# File and Stream I/O

# File and Stream I/O

## Stream 流

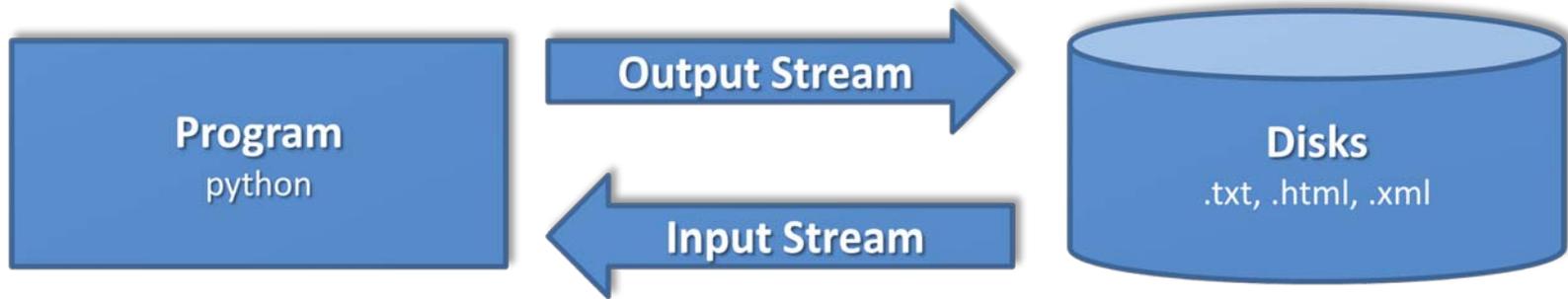
a sequence of data elements made available over time.



**Eg: The standard streams for input, output, and error.**



## *Files and Stream*



## *Read Files (Input)*

```
with open('FILE_PATH', 'r/rb') as VARIABLE_NAME  
VARIABLE_NAME.read()
```

## *Write Files (Output)*

```
with open('FILE_PATH', 'w/wb') as VARIABLE_NAME  
VARIABLE_NAME.write('TEXT')
```

*Absolute and Relative Paths* 绝对和相对路径



# File and Stream I/O

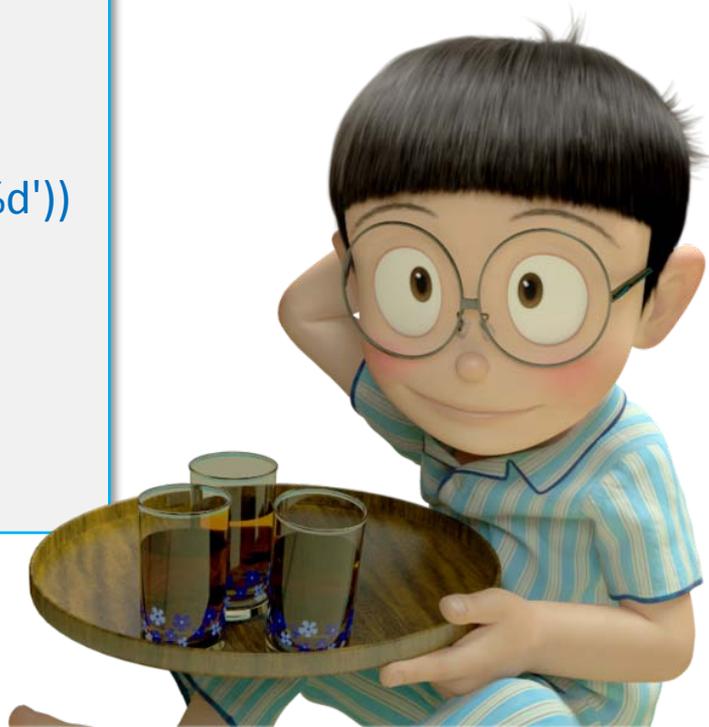
EXAMPLE 1:  
Write and Read Files



## *Example: Document Write-Save-Read*

```
import datetime
with open('test.txt', 'w') as f:
    f.write('今天是 ')
    f.write(datetime.datetime.now().strftime('%Y-%m-%d'))

with open('test.txt', 'r') as f:
    s = f.read()
    print('open for read...')
    print(s)
```



## *Example: Webpage Crawling*

```
import urllib.request
response = urllib.request.urlopen('http://www.shisu.edu.cn/about/introducing-sisu')
HTMLText = response.read()

with open('Files/shisu.html', 'wb') as f:
    f.write(HTMLText)
```





connect python with SQLite, MySQL and SQL Server

# Connect to Database

# Connect to Database

- How to use database in programming?

- Steps:

1. Connection Open
2. Cursor Creating
3. SQL Execution
4. Cursor Moving
5. Connection Close



## *SQLite*

- SQLite is a high-reliability, embedded, zero-configuration, public-domain, SQL database engine.
- SQLite is integrated in python and can be used directly.
- SQLite is a “.db” file.
- SQLite can be used with Android/iOS.
- <http://www.sqlite.org/>

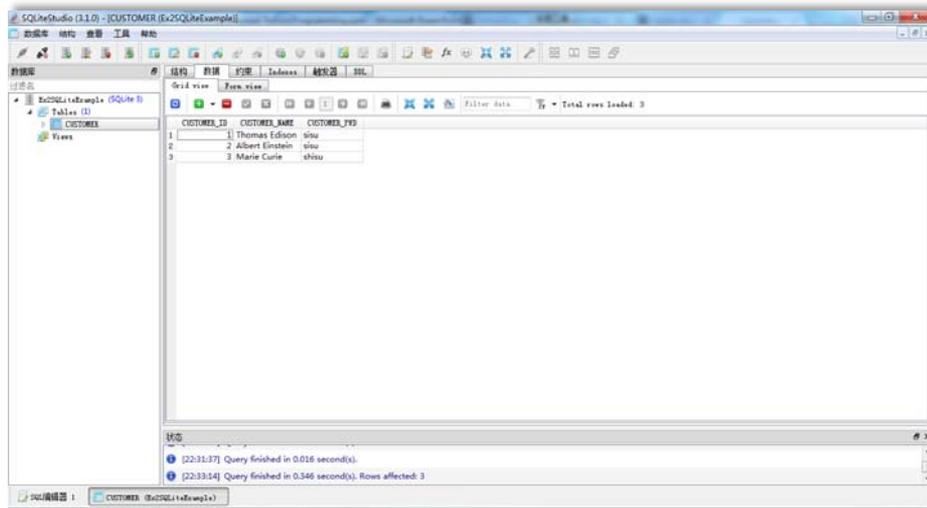


# Connect to Database

## *SQLite Studio*

a visual tool for SQLite

<http://sqlitestudio.pl/>



# Connect to Database

A world map with glowing blue grid lines and horizontal light streaks across it, set against a dark background.

EXAMPLE 2:  
SQLite and  
SQLite Studio



# Connect to Database

```
import sqlite3
#read from SQL script
with open('create.sql', 'r') as f1:
    s1 = f1.read()
# 连接到SQLite数据库
# 若文件不存在, 则自动创建
#1.Connection Open
conn=sqlite3.connect('Ex2SQLiteExample.db')
#2.Cursor Creating:
cursor = conn.cursor()

#3.SQL Execution
# 执行一条SQL语句, 创建表:
cursor.execute(s1)
```

```
# 继续执行SQL语句, 循环插入记录:
with open('insert.sql', 'r') as f2:
    #4.Cursor Moving
    #体验游标
    for line in f2.readlines():
        #执行, 游标移至当前位置
        cursor.execute(line)
        # 提交事务:
        conn.commit()
#5.Connection Close
# 关闭Cursor:
cursor.close()
# 关闭Connection:
conn.close()
```



## *Result review by SQLite Studio*

	CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_PWD
1	1	Thomas Edison	sisu
2	2	Albert Einstein	sisu
3	3	Marie Curie	shisu

*a brief introduction:*

*How to install and use SQLite Studio?*

– *Just click it!*



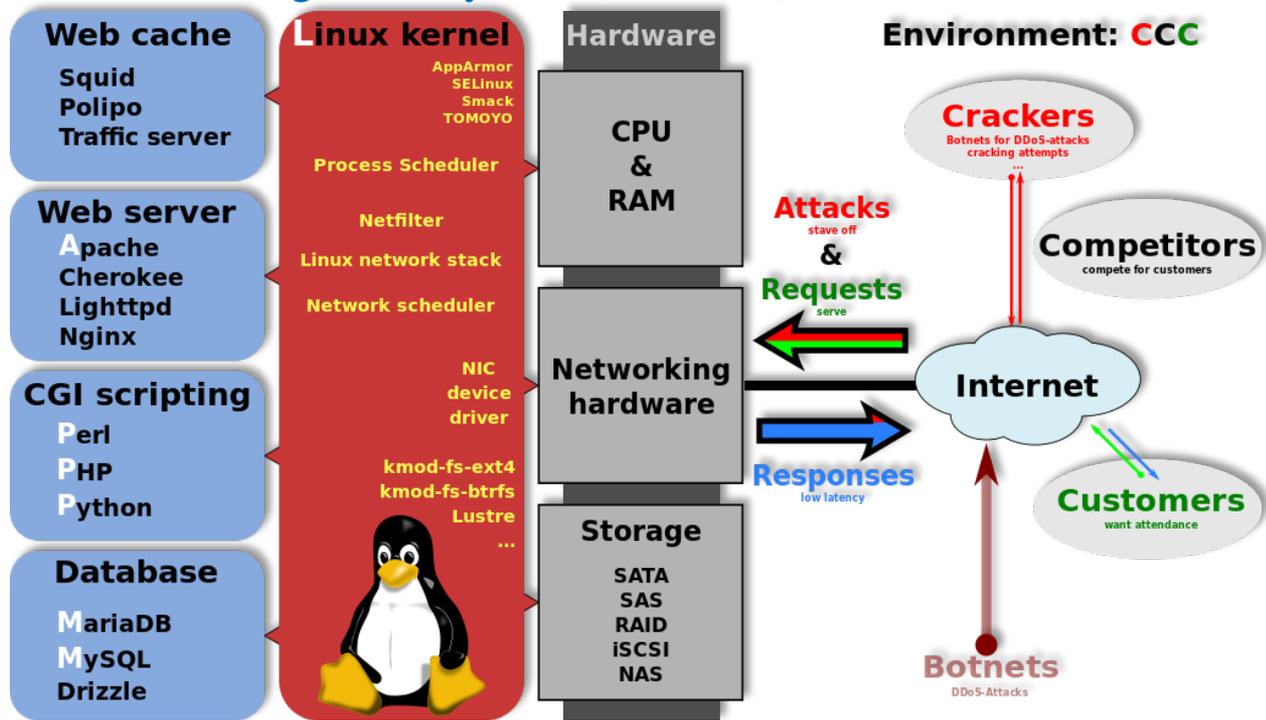
# Connect to Database

## MySQL

an open-source relational database management system (RDBMS)

### LAMP

- *Linux*
- *Apache*
- *Perl+PHP+Python*
- *MySQL*



Environment: CCC

**Crackers**

Botnets for DDoS-attacks  
cracking attempts

**Competitors**

compete for customers

**Internet**

**Customers**

want attendance

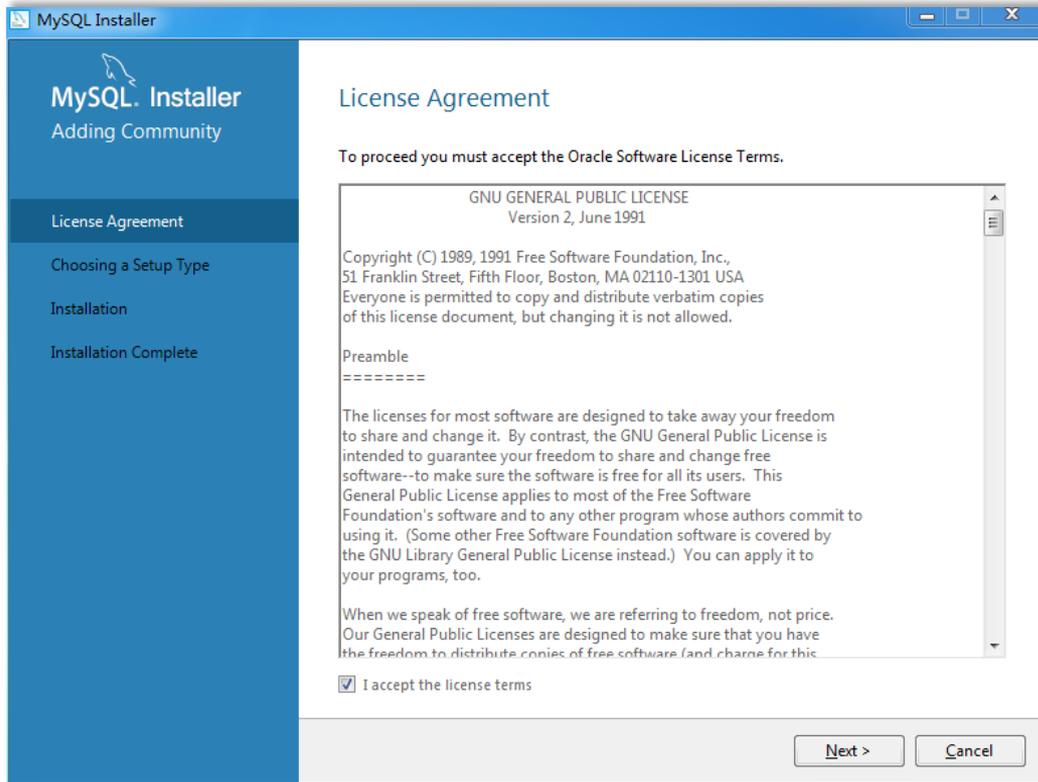
**Botnets**

DDoS-Attacks



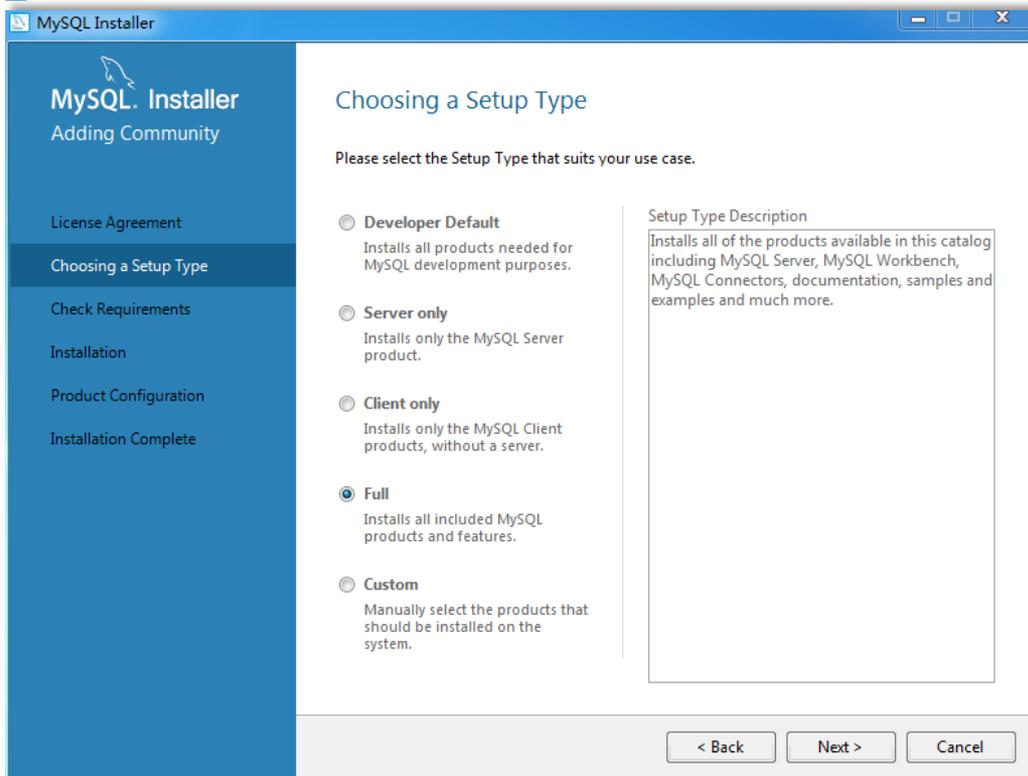
# Connect to Database

## MySQL Installation



# Connect to Database

## Choosing a Setup Type

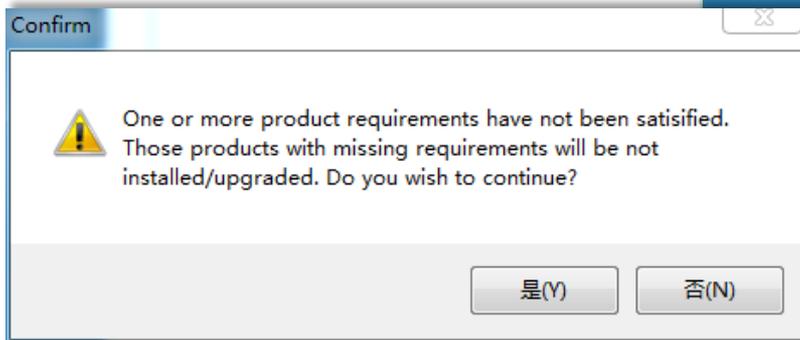
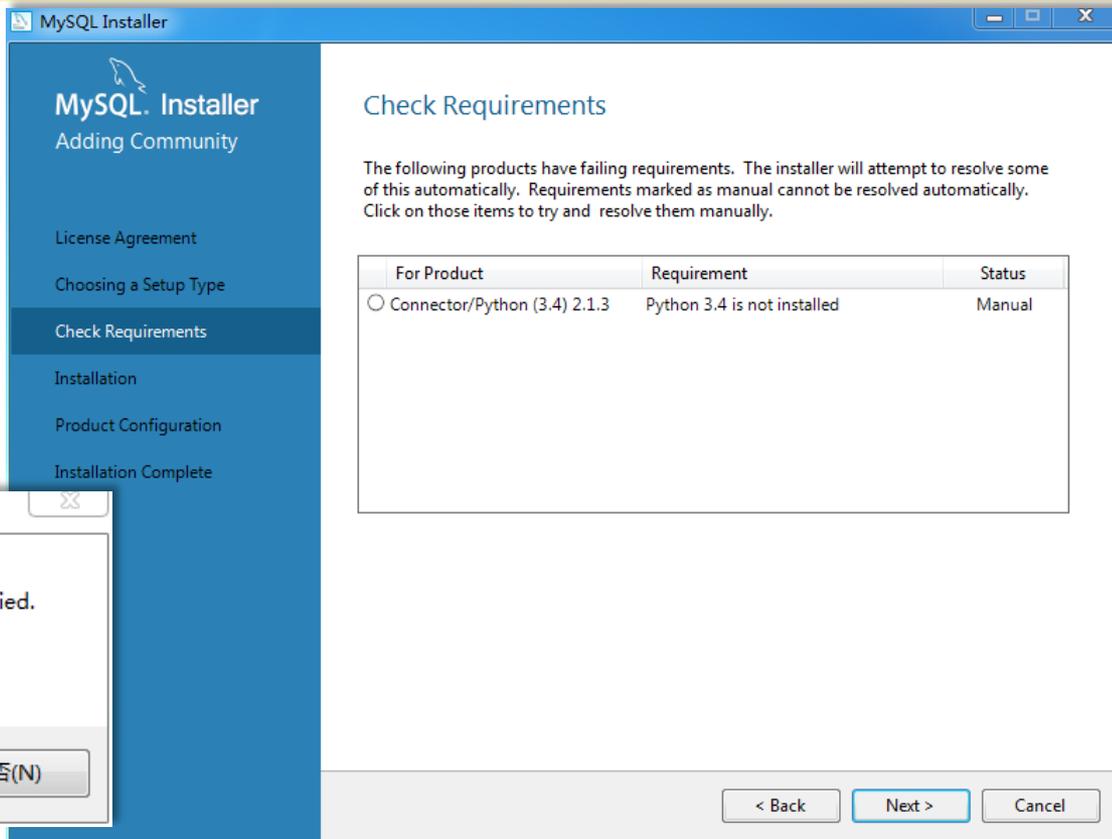


# Connect to Database

## *A Requirement:*

Python 3.5 is **NOT** officially  
Supported by MySQL

*Just ignore it!*



# Connect to Database

Connector/Python 2.1.3

Select Platform: Microsoft Windows Looking for previous GA versions?

<b>Windows (x86, 32-bit), MSI Installer</b> <b>Python 2.7</b> (mysql-connector-python-2.1.3-py2.7-win32.msi)	2.1.3	1.4M	<a href="#">Download</a>
<b>Windows (x86, 32-bit), MSI Installer</b> <b>Python 3.3</b> (mysql-connector-python-2.1.3-py3.3-win32.msi)	2.1.3	1.4M	<a href="#">Download</a>
<b>Windows (x86, 32-bit), MSI Installer</b> <b>Python 3.4</b> (mysql-connector-python-2.1.3-py3.4-win32.msi)	2.1.3	1.4M	<a href="#">Download</a>
<b>Windows (x86, 64-bit), MSI Installer</b> <b>Python 2.7</b> (mysql-connector-python-2.1.3-py2.7-winx64.msi)	2.1.3	1.5M	<a href="#">Download</a>
<b>Windows (x86, 64-bit), MSI Installer</b> <b>Python 3.3</b> (mysql-connector-python-2.1.3-py3.3-winx64.msi)	2.1.3	1.5M	<a href="#">Download</a>
<b>Windows (x86, 64-bit), MSI Installer</b> <b>Python 3.4</b> (mysql-connector-python-2.1.3-py3.4-winx64.msi)	2.1.3	1.5M	<a href="#">Download</a>

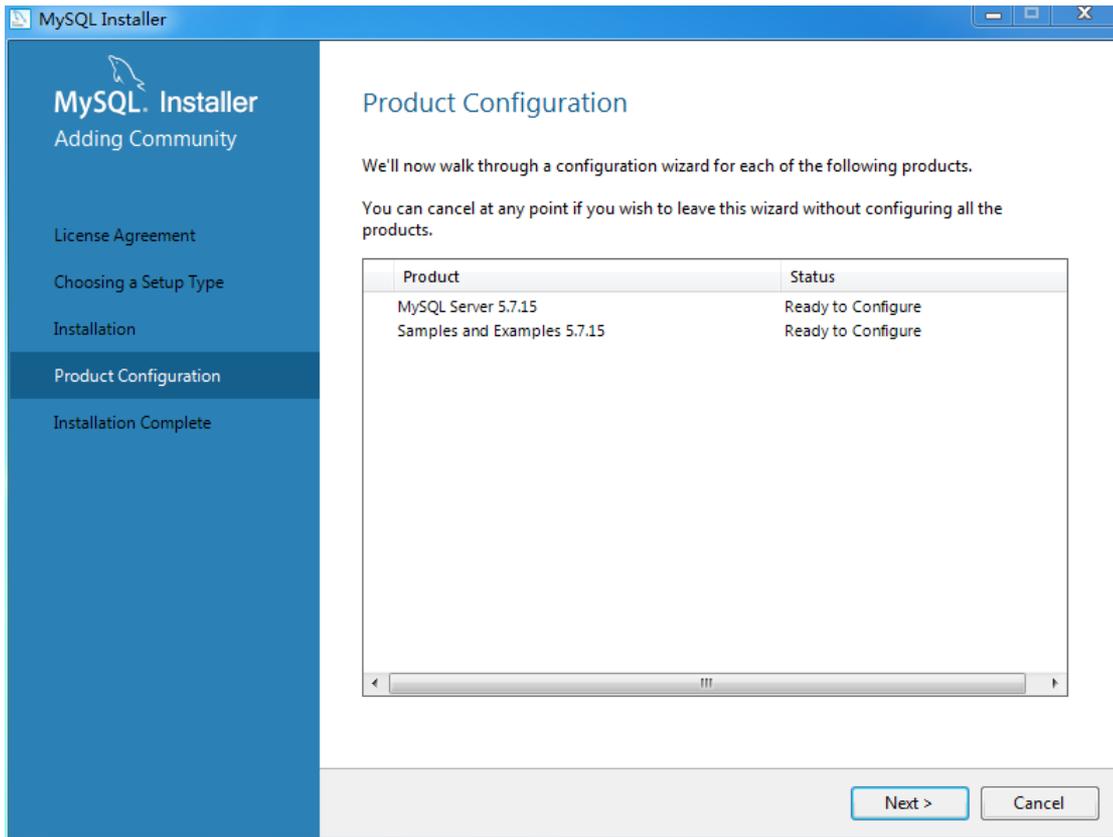
**No Python 3.5!!**

<http://dev.mysql.com/downloads/connector/python/>



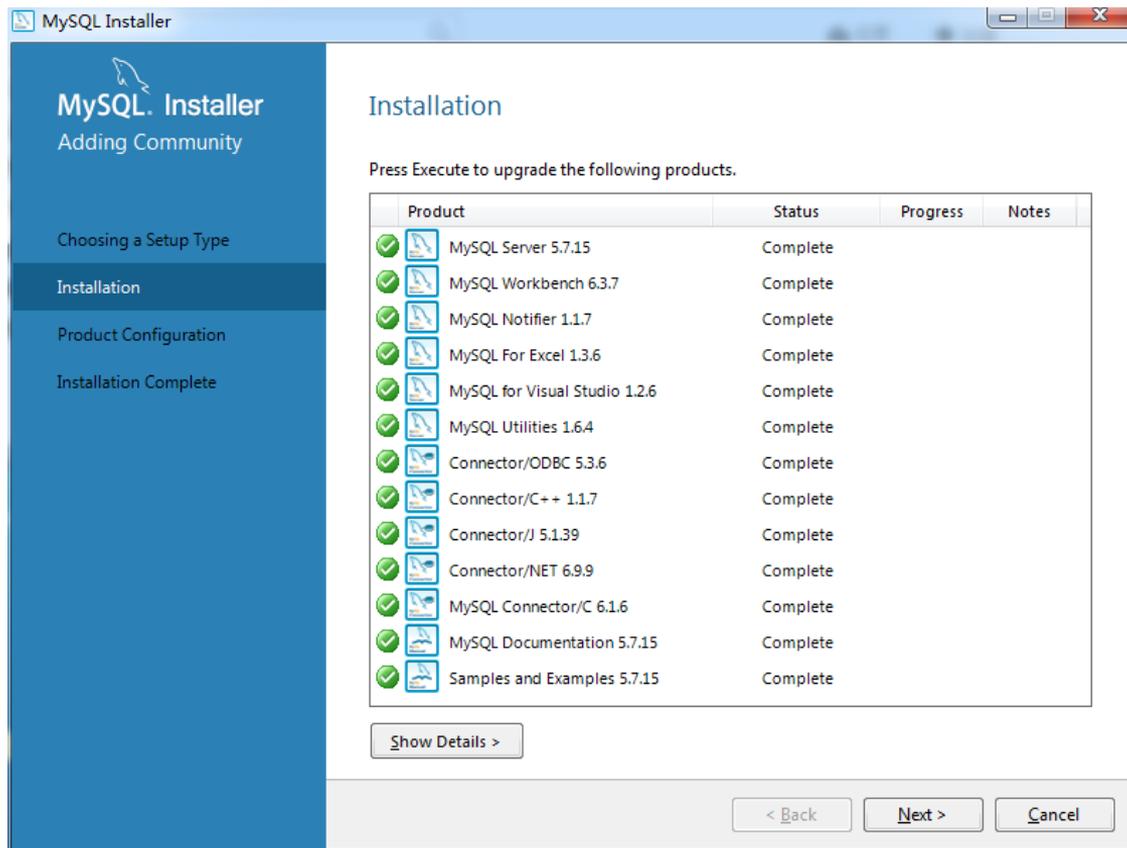
# Connect to Database

## Product Configuration



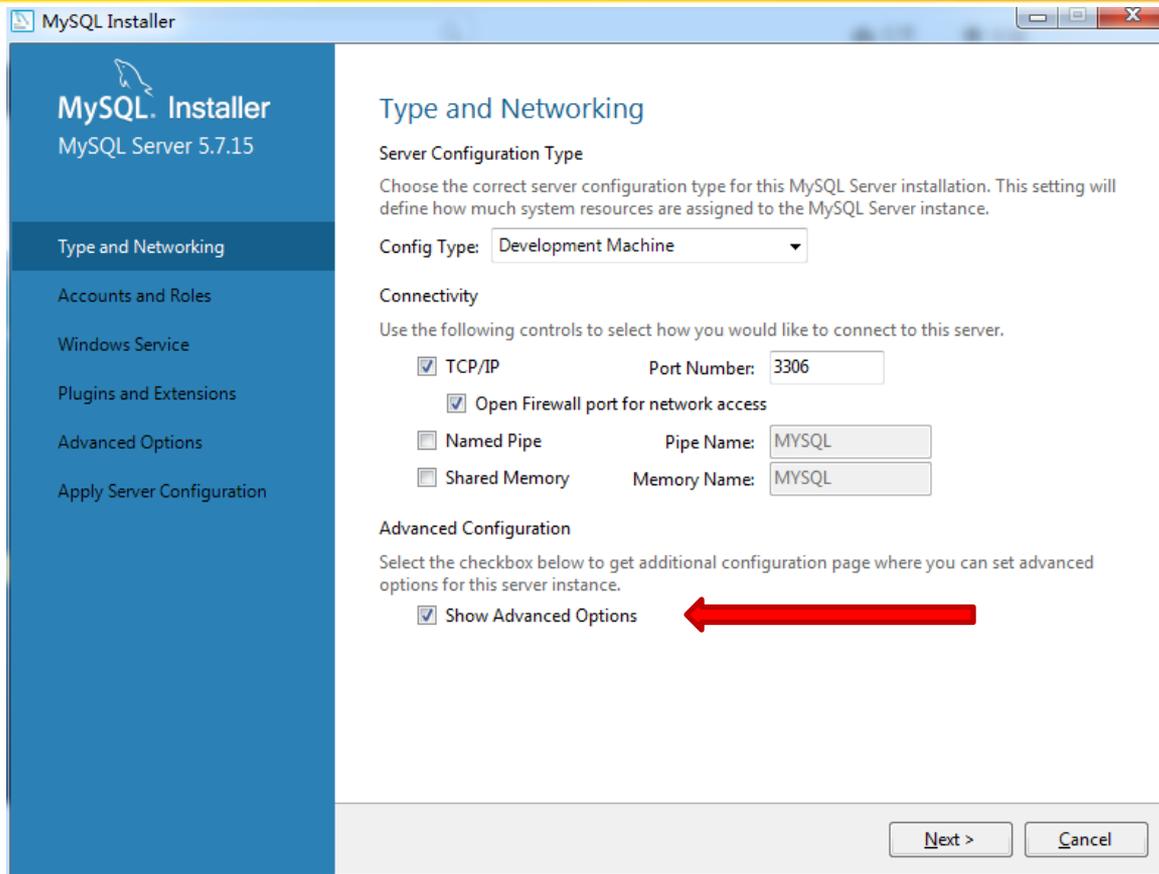
# Connect to Database

## Installation



# Connect to Database

## Show Advanced Options



# Connect to Database

## Account and Roles

MySQL Installer

MySQL. Installer  
MySQL Server 5.7.15

Type and Networking

**Accounts and Roles**

Windows Service

Plugins and Extensions

Advanced Options

Apply Server Configuration

### Accounts and Roles

**Root Account Password**  
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password Strength: **Weak**

**MySQL User Accounts**  
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

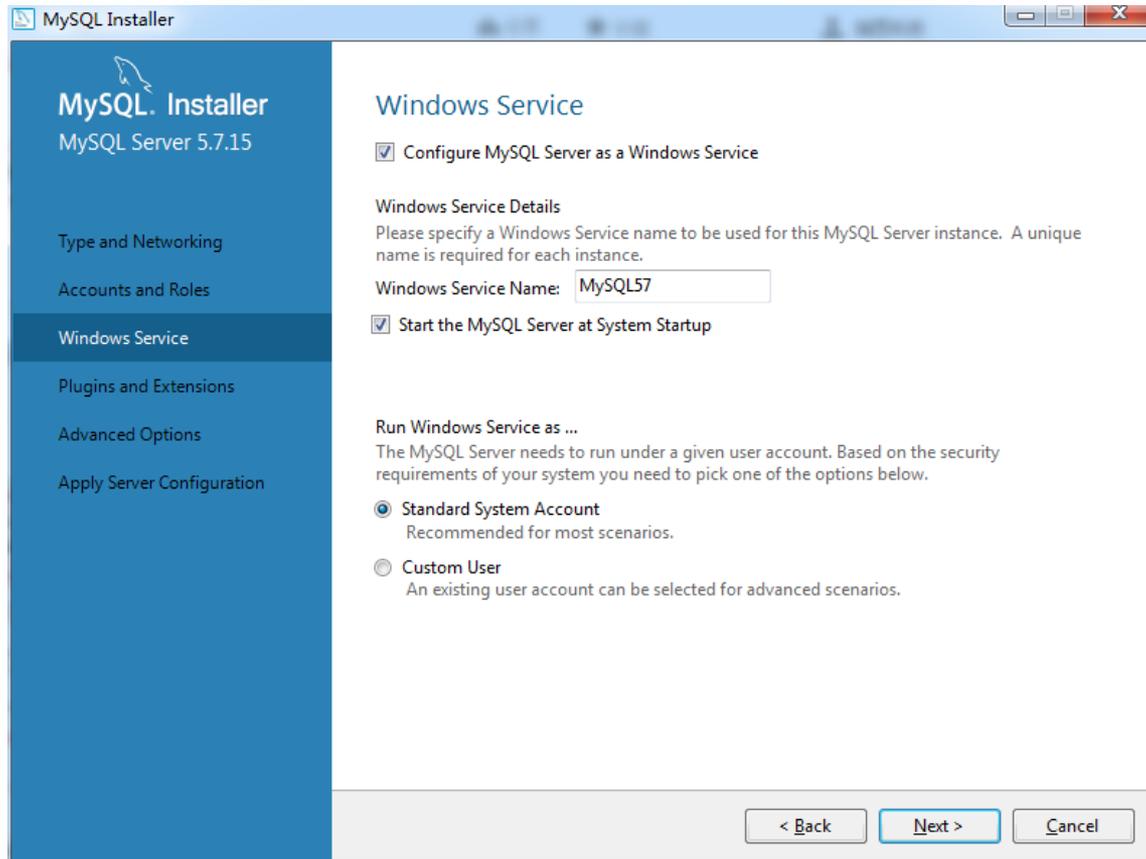
MySQL Username	Host	User Role	
MySQLAdmin	%	DB Admin	<input type="button" value="Add User"/>

< Back    Next >    Cancel



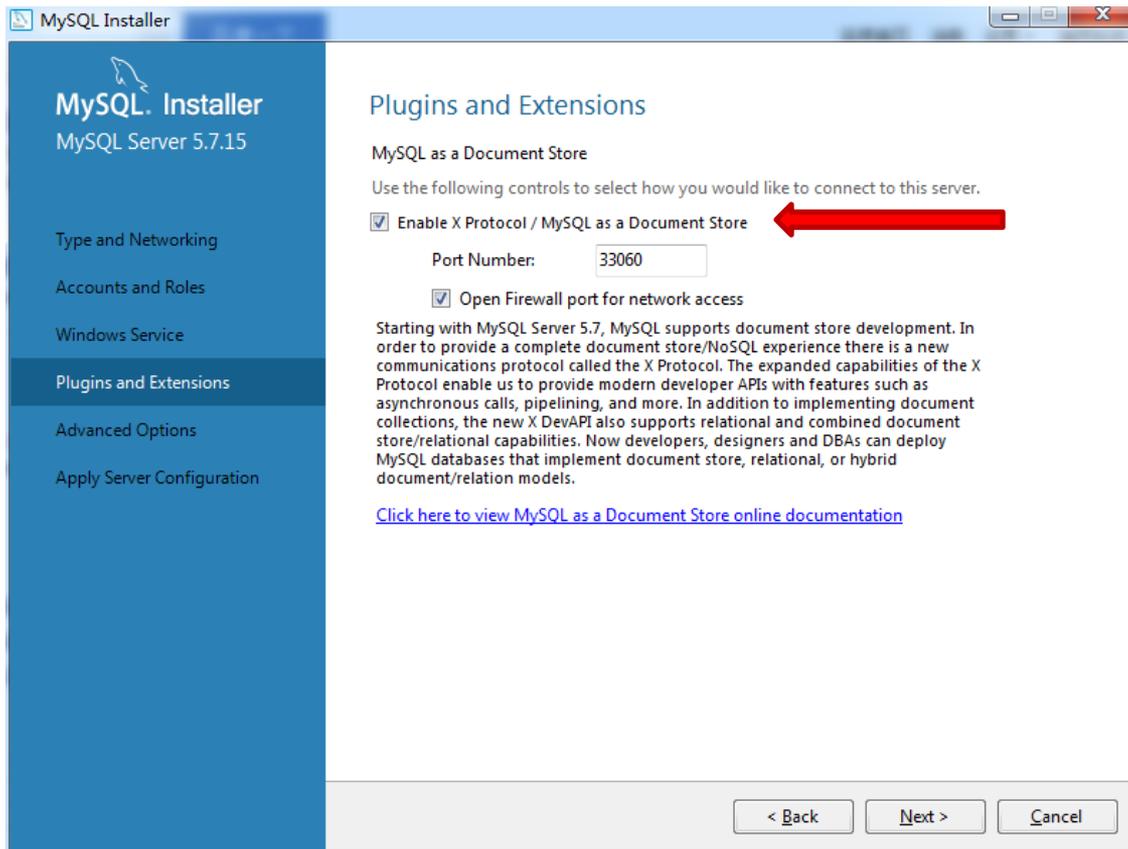
# Connect to Database

*Next*



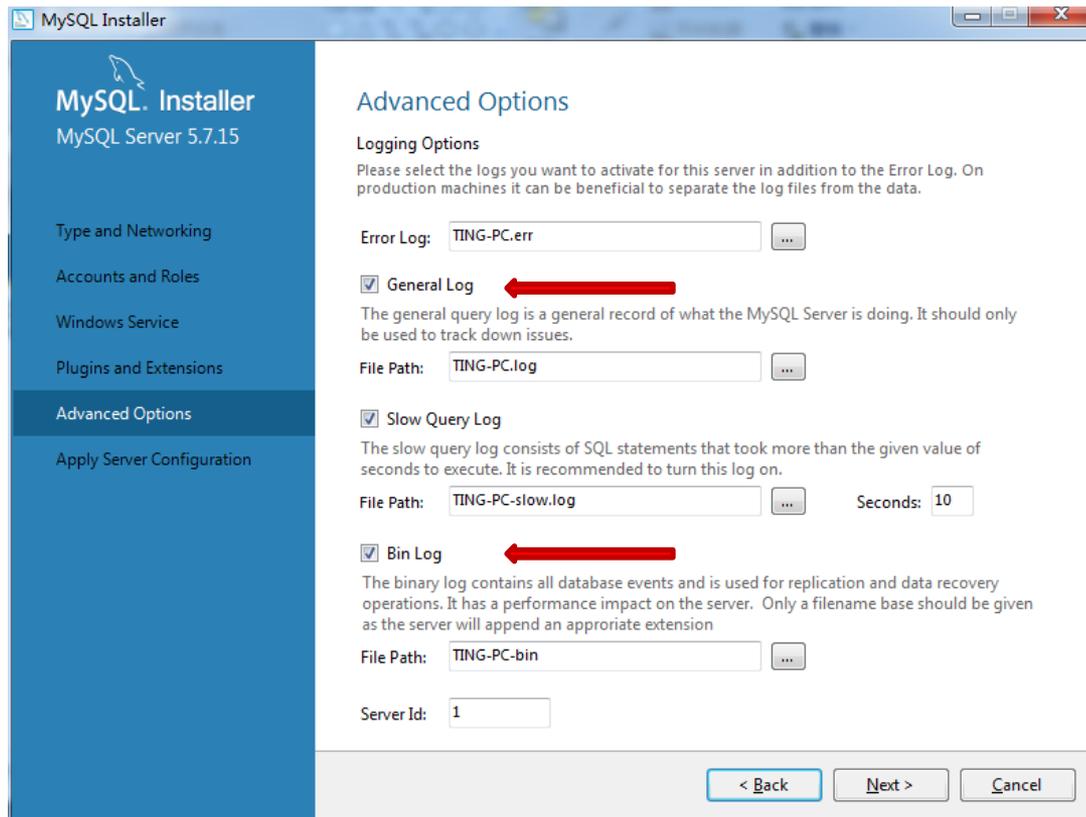
# Connect to Database

## *New MySQL Function for NoSQL*



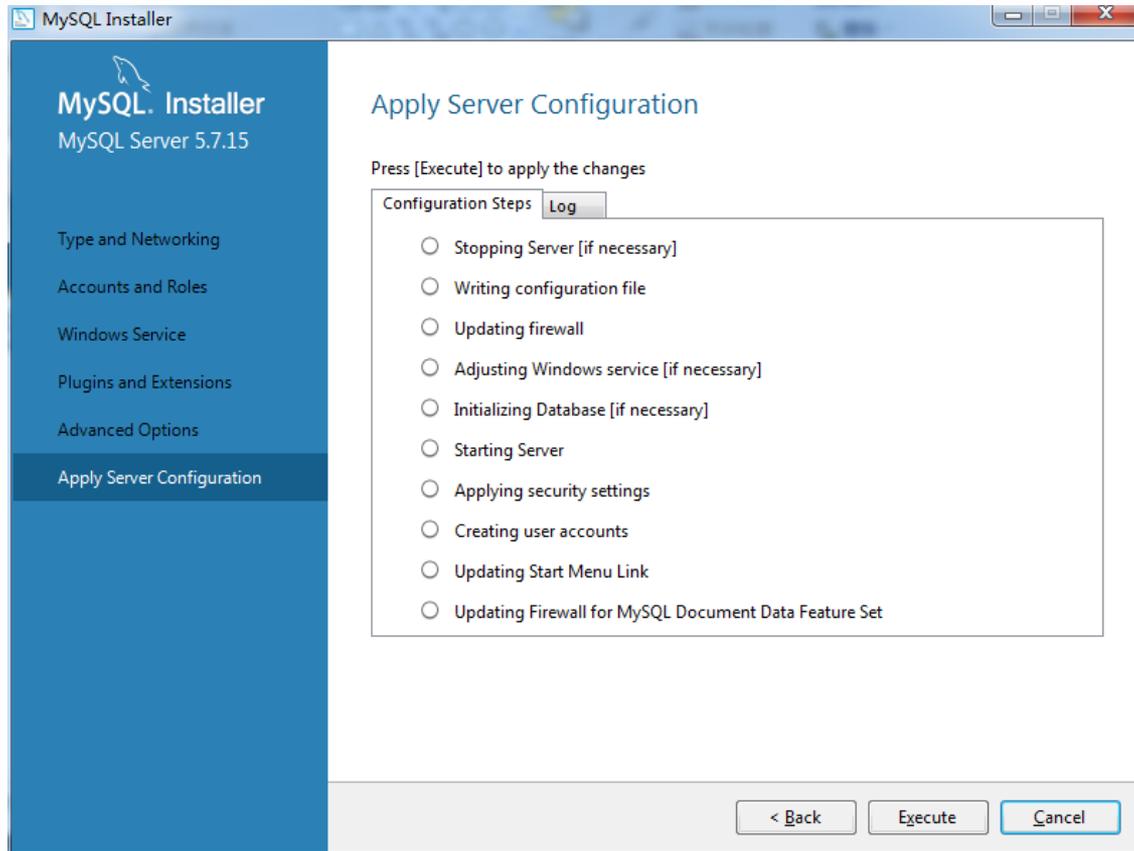
# Connect to Database

## Logs



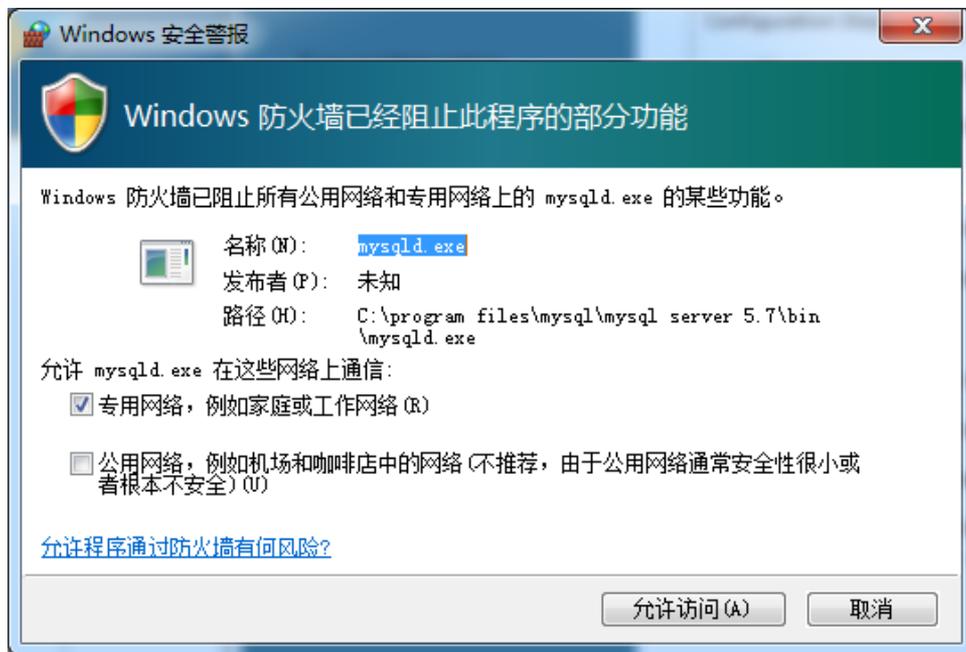
# Connect to Database

## MySQL Server Server Configuration



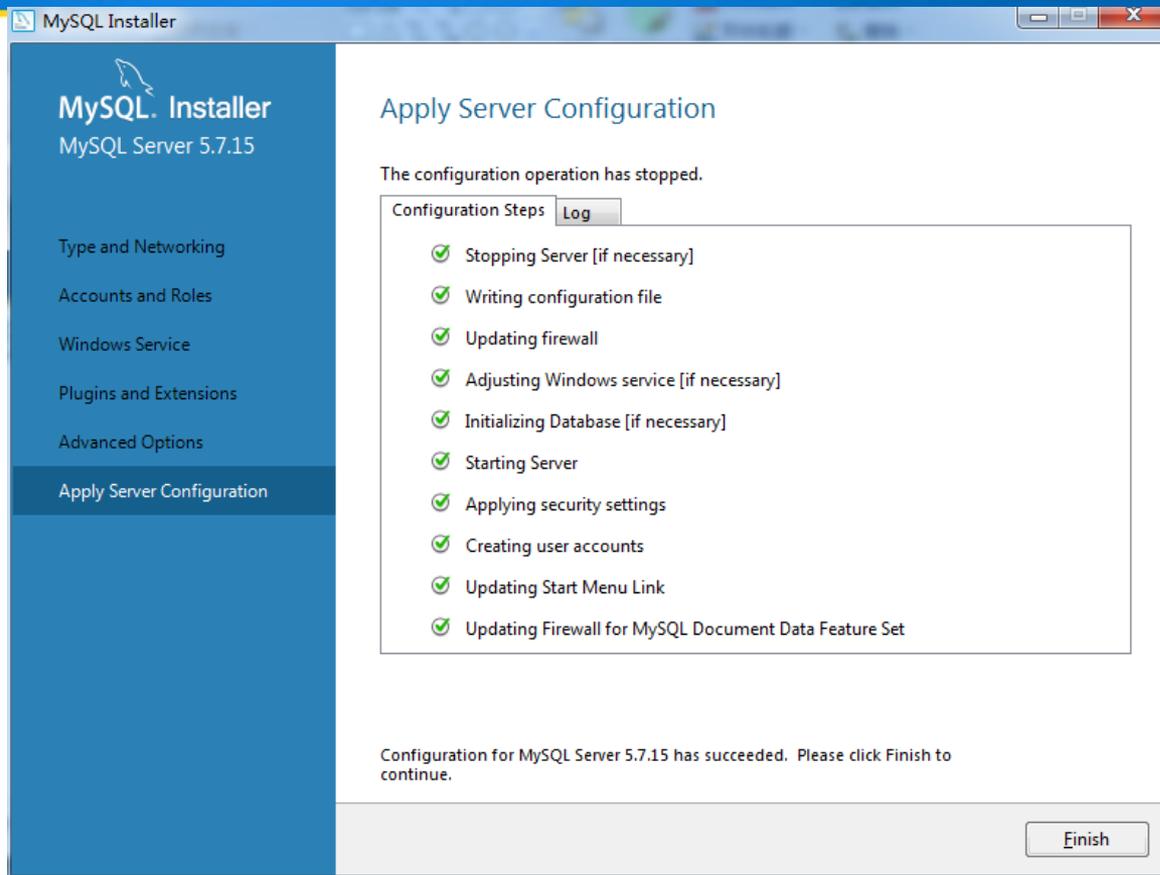
# Connect to Database

*Sometimes, maybe...if you are using Wifi*



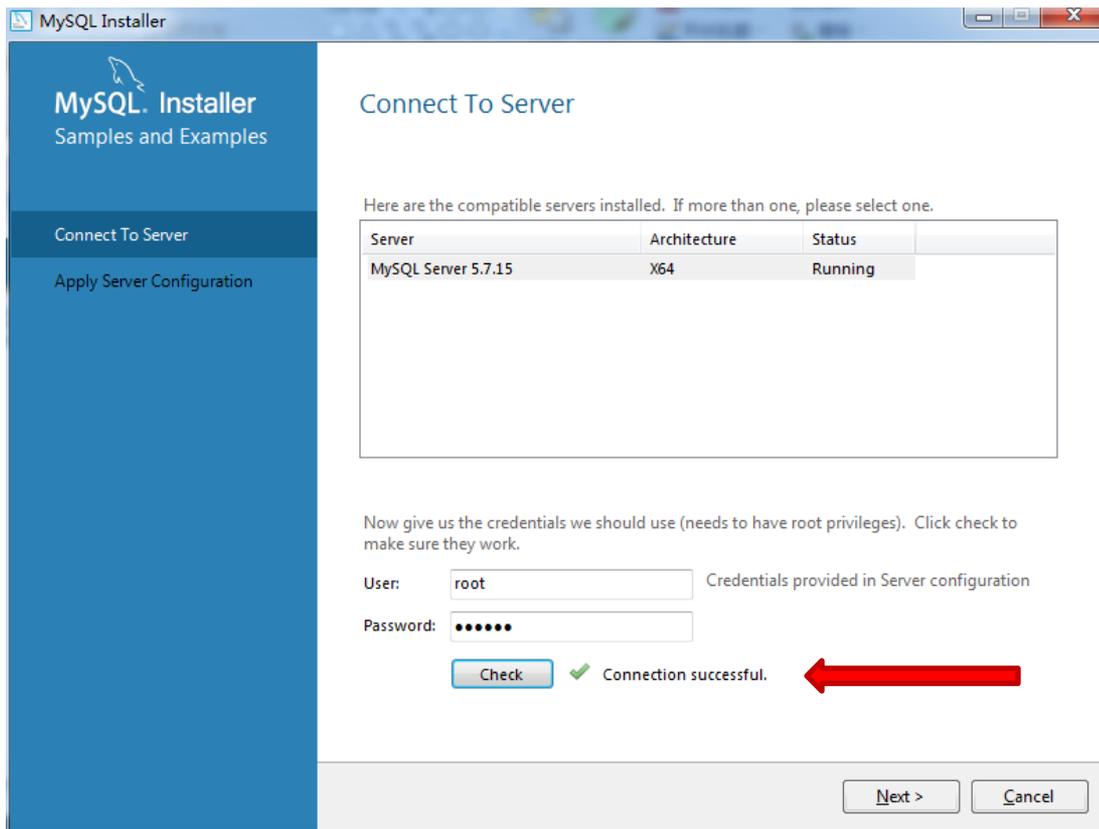
# Connect to Database

## MySQL Server Apply Server Configuration



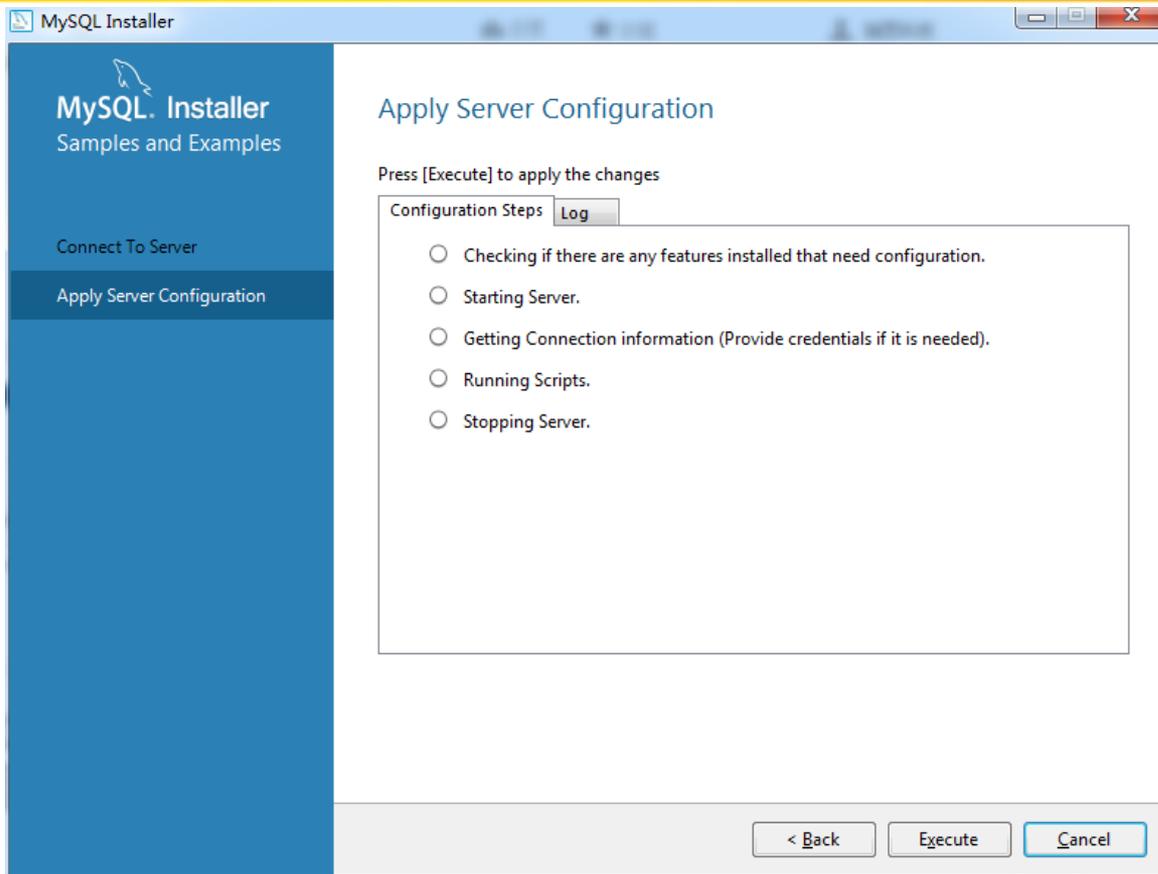
# Connect to Database

*Prepare to install  
Samples  
– Check!*



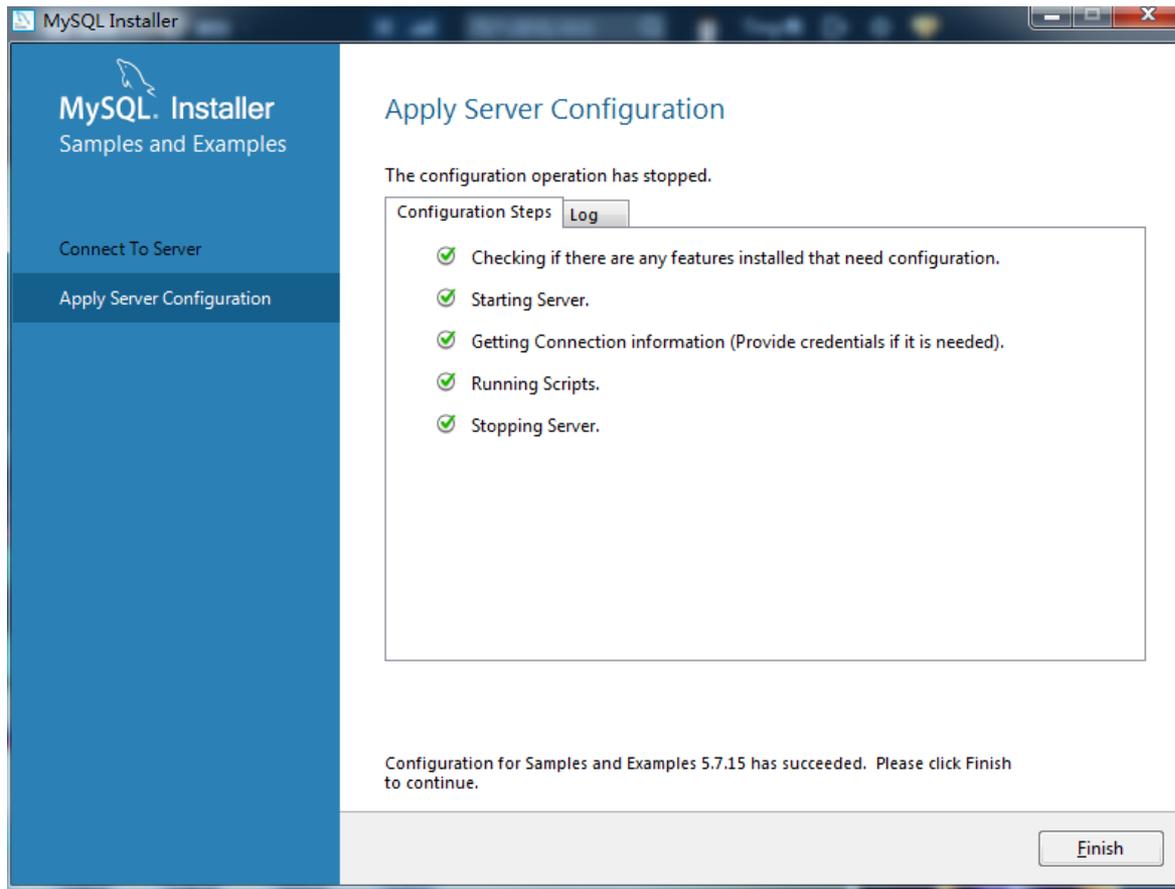
# Connect to Database

## *Samples Apply Server Configuration*



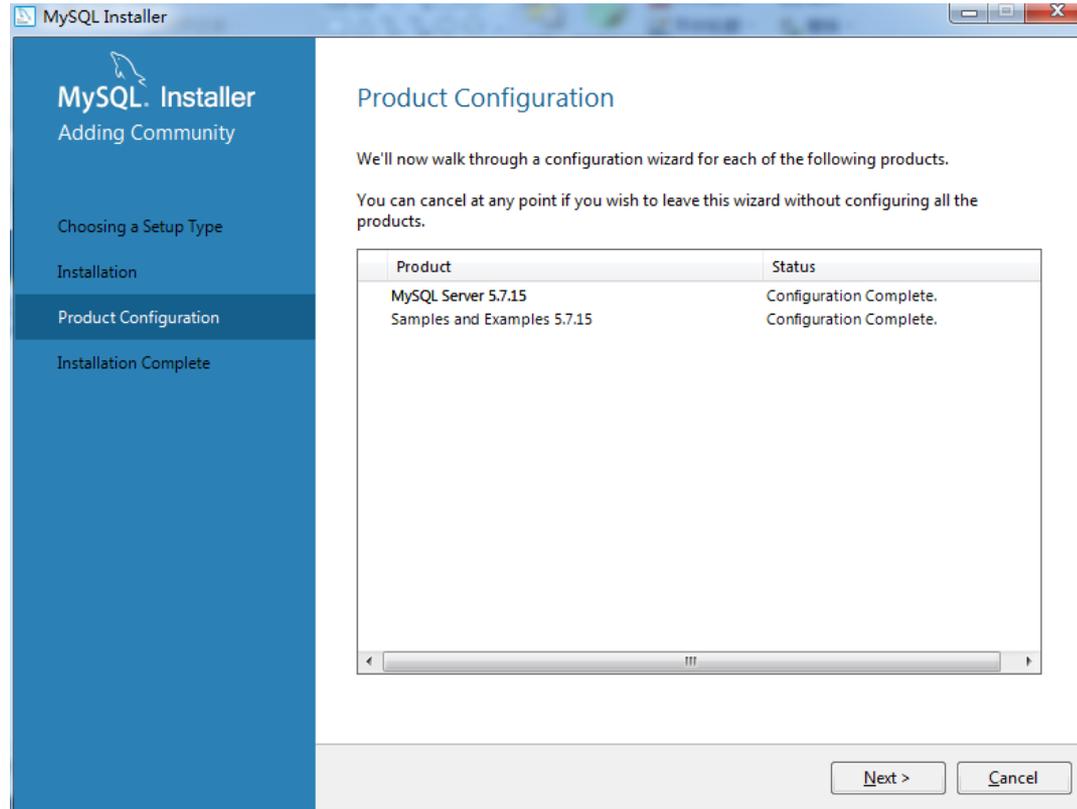
# Connect to Database

## *Finish Server Configuration*



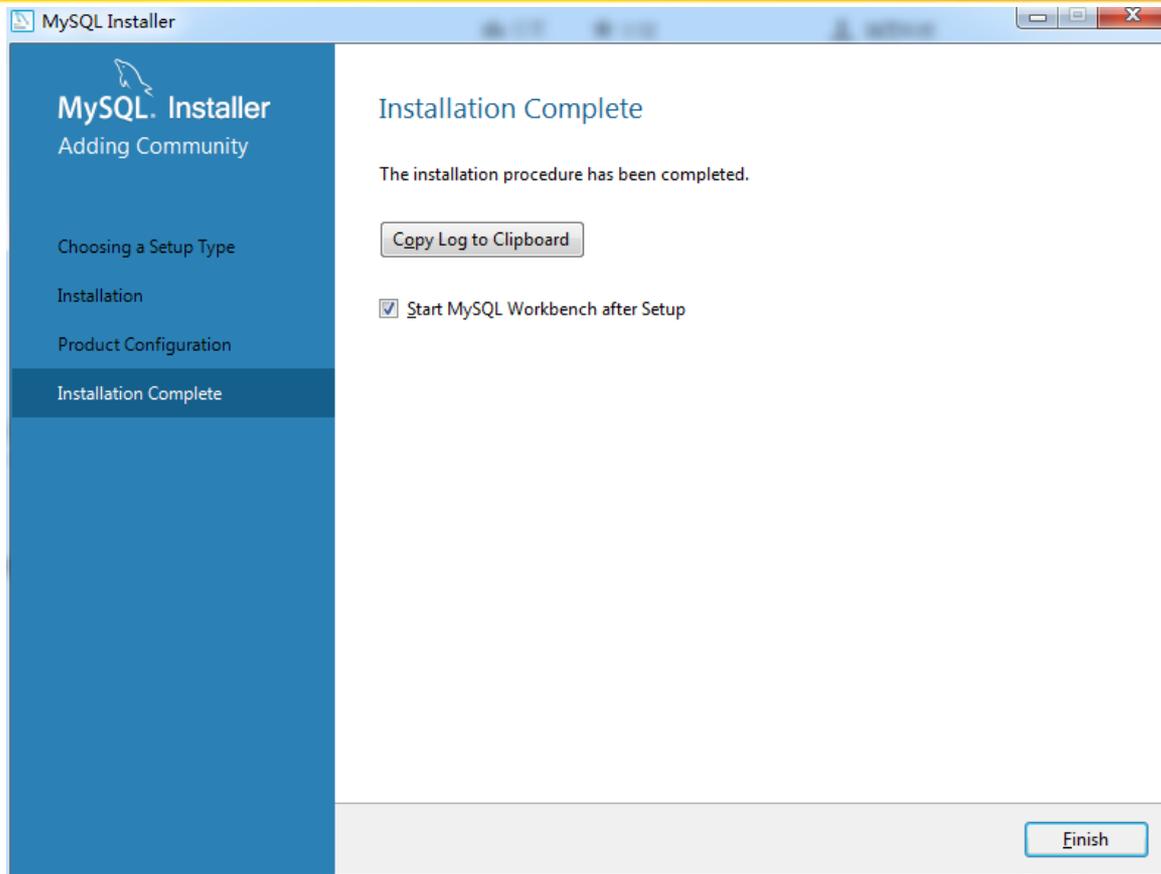
# Connect to Database

*Next*



# Connect to Database

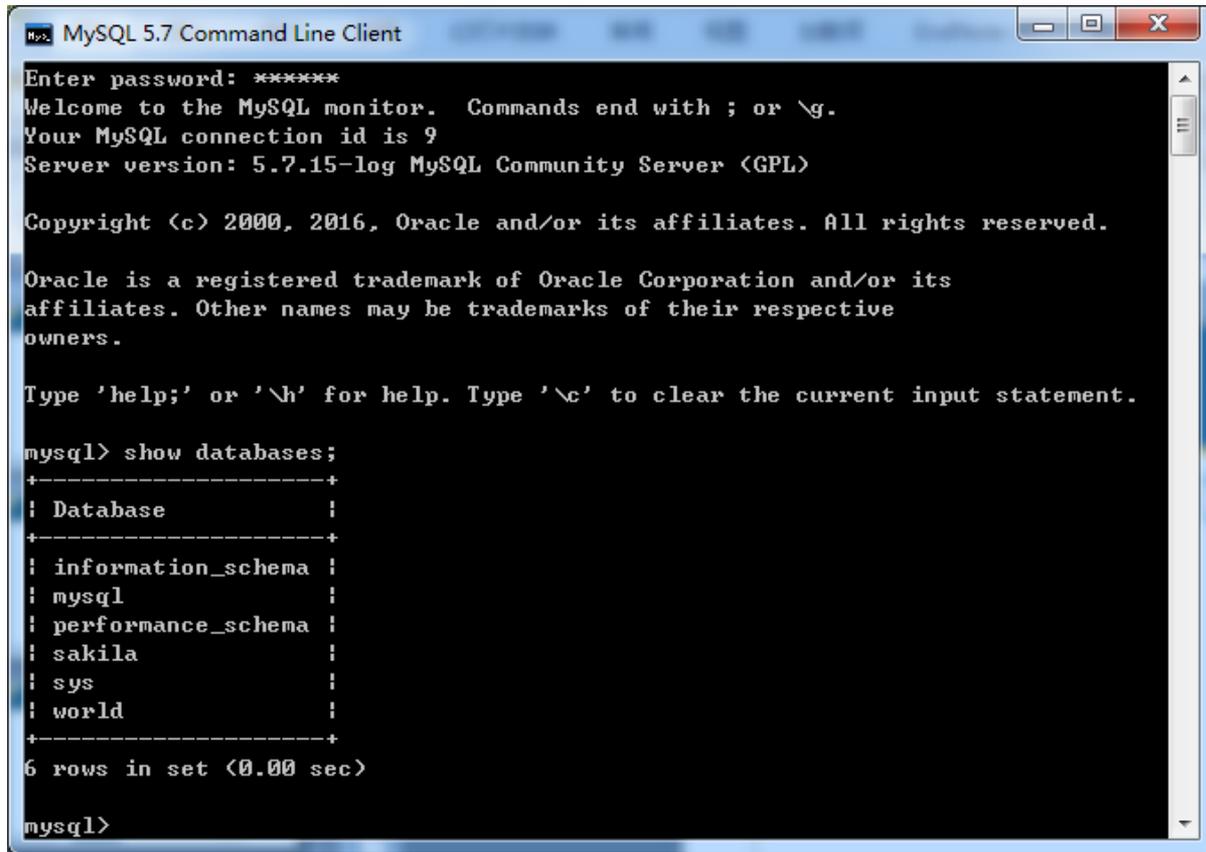
*Complete*



# Connect to Database

## Test

- *Successful?*
- *Test it!*



```
MySQL 5.7 Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 5.7.15-log MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| performance_schema |
| sakila             |
| sys                |
| world              |
+-----+
6 rows in set (0.00 sec)

mysql>
```



# Connect to Database

## *Install a third-party module for the connection between Python and MySQL: Preparation*

### *Python **setuptools**: A Preparation*

If you have **NO** Python *setuptools*, Install it in the first place!

If you have **already installed** Python *setuptools*, ignore this page and turn to the next!

*Step1: Download the file from:* [https://bootstrap.pypa.io/ez\\_setup.py](https://bootstrap.pypa.io/ez_setup.py)

*Step2: Install ez\_setup.py with the command in Windows CMD:* `python ez_setup.py`



# Connect to Database

*pymysql, a third-party module for the connection between Python and MySQL*

*Step1:Download pymysql, <https://github.com/PyMySQL/PyMySQL>*

*Step2:Install pymysql with the command in Windows CMD: python setup.py install*



# Connect to Database

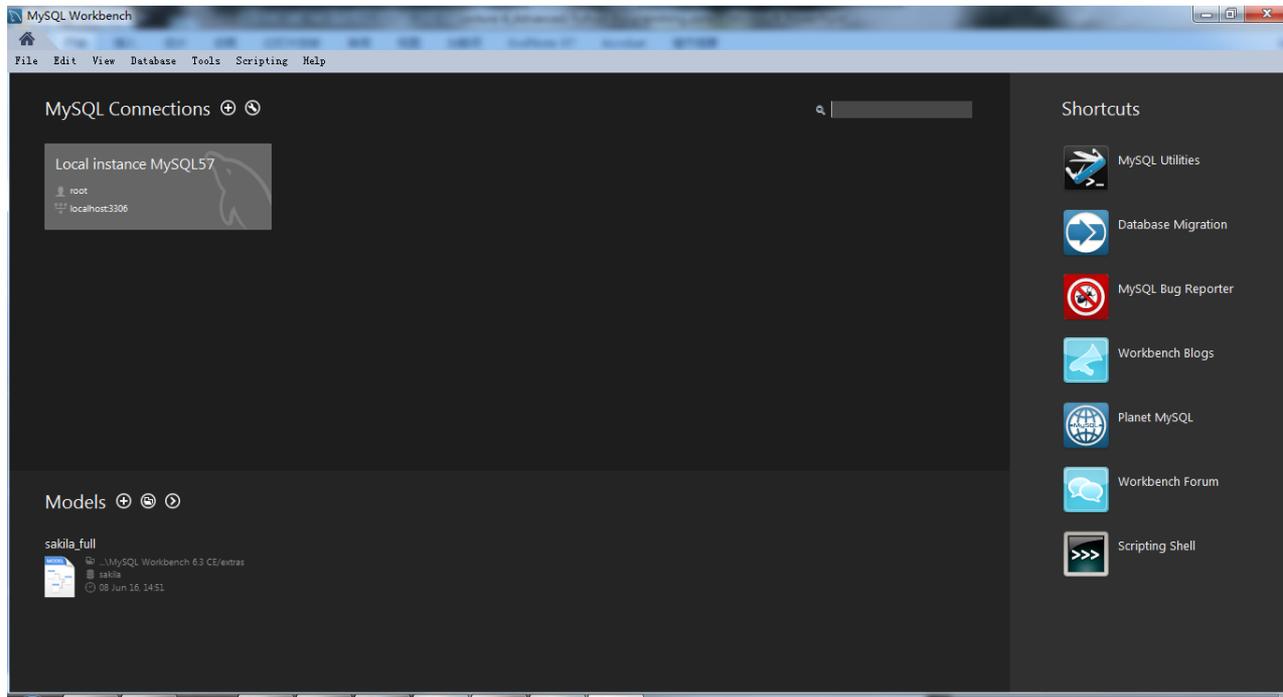
## *Tips*

Do **NOT** forget to restart your IDE after you install some new third-party modules.



# Connect to Database

## *Start MySQL Workbench*



# Connect to Database



EXAMPLE 3:  
python - MySQL



# Connect to Database

## Code

```
import pymysql

# 连接到MySQL数据库
#1.Connection Open
conn = pymysql.connect(user='root',
password='123456', database='login')
#2.Cursor Creating:
cursor = conn.cursor()

#3.SQL Execution
# 执行SQL语句，循环插入记录:
```

```
with open('insertMySQL.sql', 'r') as f2:
```

```
    #4.Cursor Moving
```

```
    #体验游标
```

```
    for line in f2.readlines():
```

```
        #执行, 游标移至当前位置
```

```
        cursor.execute(line)
```

```
        # 提交事务:
```

```
        conn.commit()
```

```
#5.Connection Close
```

```
# 关闭Cursor:
```

```
cursor.close()
```

```
# 关闭Connection:
```

```
conn.close()
```



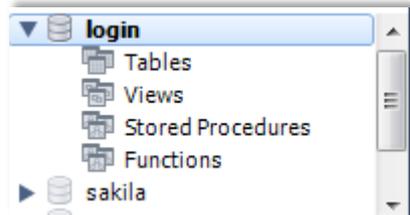
# Connect to Database

## Steps

1. Create a new schema



2. Set as a default schema (right click)

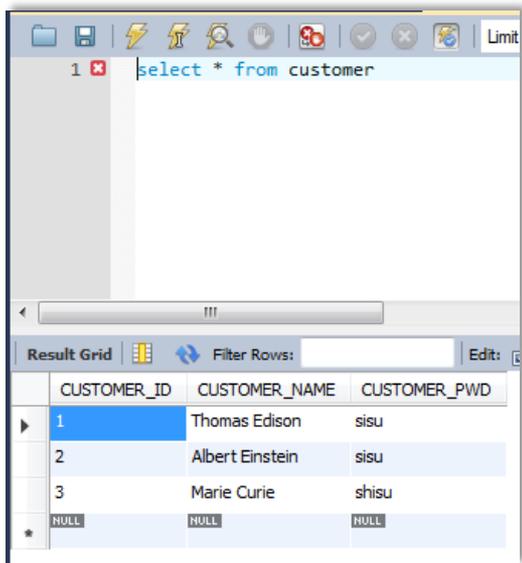


3. Run CreateMySQL.sql



4. Run Python code

5. Select \* from ...



The screenshot shows a SQL query editor with the query `select * from customer` and a result grid below it. The result grid has columns for `CUSTOMER_ID`, `CUSTOMER_NAME`, and `CUSTOMER_PWD`. The data is as follows:

CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_PWD
1	Thomas Edison	sisu
2	Albert Einstein	sisu
3	Marie Curie	shisu
NULL	NULL	NULL

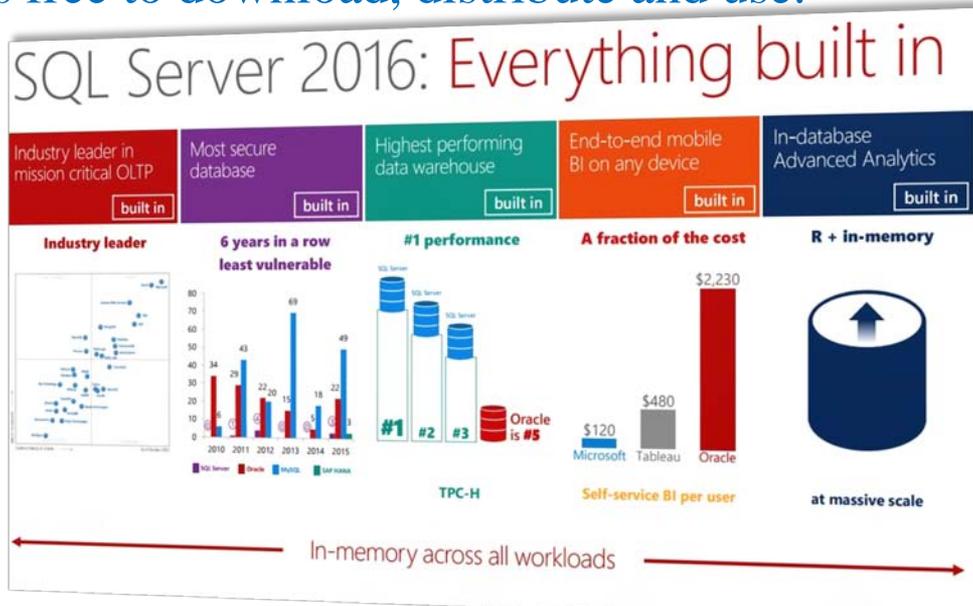


# Connect to Database

## *Microsoft SQL Server*

a relational database management system developed by Microsoft

Microsoft SQL Server Express is a version of Microsoft SQL Server relational database management system that is free to download, distribute and use.



# Connect to Database

## *pymssql, a third-party module for connecting Python and Microsoft SQL Server*

*Step1: Download pymssql*, <https://pypi.python.org/pypi/pymssql>

```
pymssql-2.1.3-cp35-cp35m-win32.whl (md5)
```

```
pymssql-2.1.3-cp35-cp35m-win_amd64.whl (md5)
```

```
pymssql-2.1.3.tar.gz (md5)
```

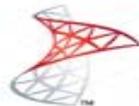
*Step2: Install pymssql with the command in Windows CMD:*

```
pip install pymssql-2.1.3-cp35-cp35m-win_amd64.whl
```

<https://pypi.python.org/pypi/pip>



# Connect to Database



Microsoft®

SQL Server™

FURTHER. FORWARD.

EXAMPLE 4:  
python - MS SQL



上海外国语大学

SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

# Connect to Database

```
import pymssql

# 连接到MSSQL数据库
#1.Connection Open
conn = pymssql.connect(user='sa',
password='123456', database='login')
#2.Cursor Creating:
cursor = conn.cursor()
#3.SQL Execution
# 执行SQL语句，循环插入记录:
with open('createMSSQL.sql', 'r') as f1:
    s1 = f1.read()
    # 执行一条SQL语句，创建表:
    cursor.execute(s1)
```

```
with open('insertMSSQL.sql', 'r') as f2:
    #4.Cursor Moving
    #体验游标
    for line in f2.readlines():
        #执行, 游标移至当前位置
        cursor.execute(line)
        # 提交事务:
        conn.commit()
#5.Connection Close
# 关闭Cursor:
cursor.close()
# 关闭Connection:
conn.close()
```





my first website using web framework

# Web Programming

## *Python Web Framework*

- Flask
- Django
- Web.py
- Tornado
- Bottle



## *Install Flask: pip install flask*



```
D:\SISU\新媒体数据分析与应用\Lecture 4\Flask>pip install flask
Collecting flask
  Using cached Flask-0.11.1-py2.py3-none-any.whl
Collecting Jinja2>=2.4 (from flask)
  Using cached Jinja2-2.8-py2.py3-none-any.whl
Collecting click>=2.0 (from flask)
  Using cached click-6.6.tar.gz
Collecting Werkzeug>=0.7 (from flask)
  Using cached Werkzeug-0.11.11-py2.py3-none-any.whl
Collecting itsdangerous>=0.21 (from flask)
  Using cached itsdangerous-0.24.tar.gz
Collecting MarkupSafe (from Jinja2>=2.4->flask)
  Using cached MarkupSafe-0.23.tar.gz
Installing collected packages: MarkupSafe, Jinja2, click, Werkzeug, itsdangerous, flask
  Running setup.py install for MarkupSafe ... done
  Running setup.py install for click ... done
  Running setup.py install for itsdangerous ... done
Successfully installed Jinja2-2.8 MarkupSafe-0.23 Werkzeug-0.11.11 click-6.6 flask-0.11.1 itsdangerous-0.24
```

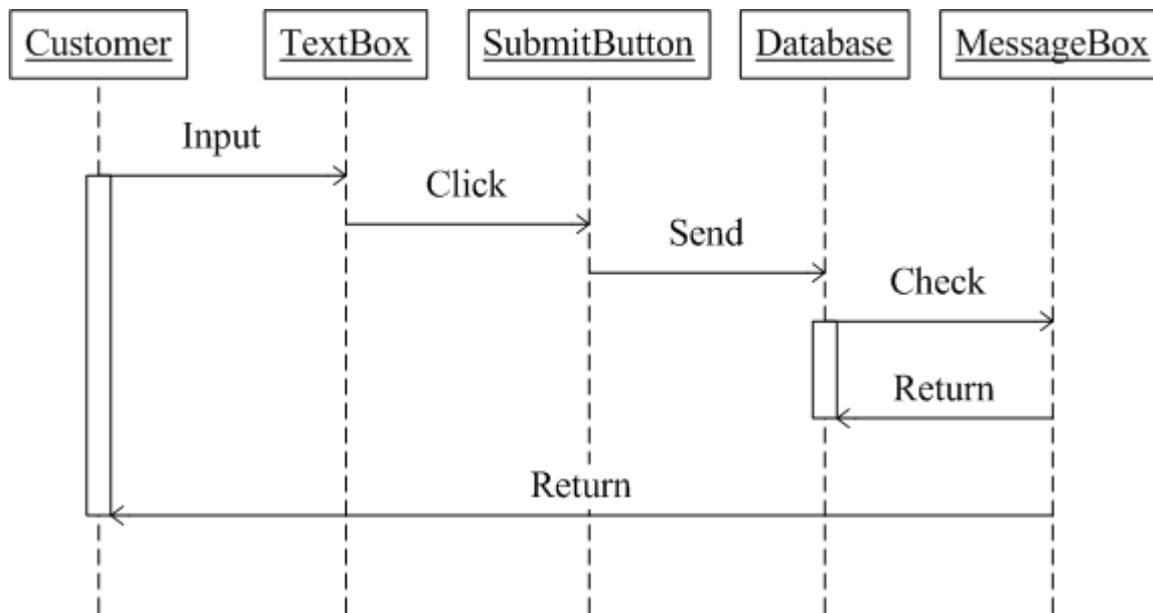
## EXAMPLE 5: Flask Login



Adam .

Sign-in options

## Login



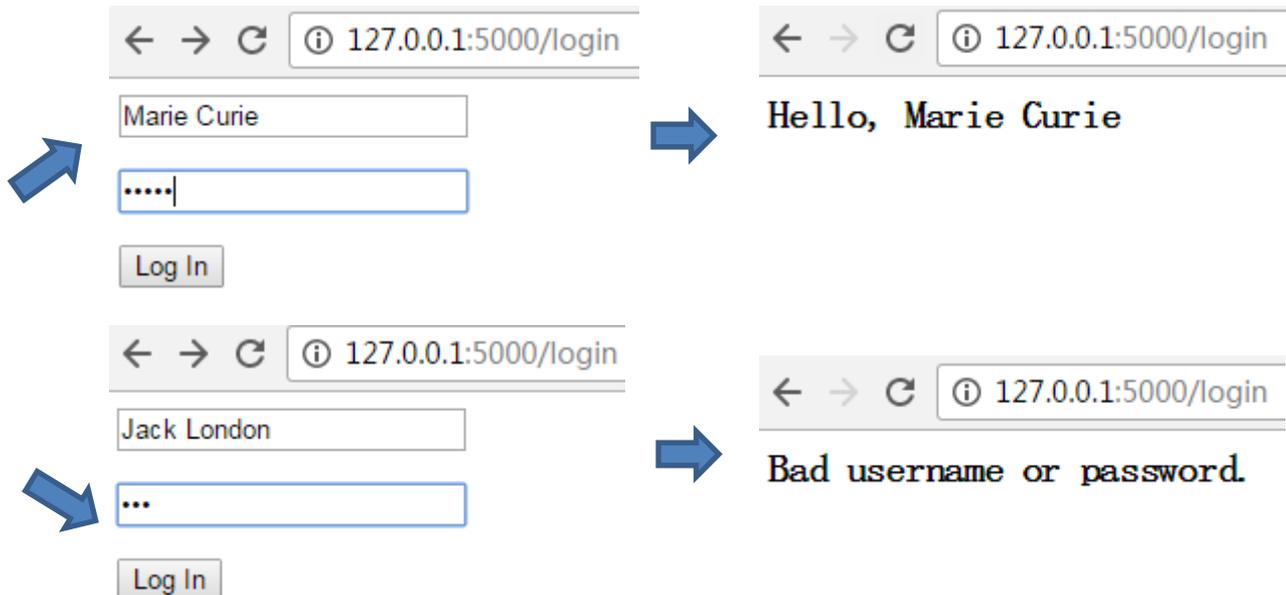
## Testing

```
D:\SISU\新媒体数据分析与应用\Lecture 4>python login.py
* Running on http://127.0.0.1:5000/ <Press CTRL+C to quit>
```

← → ↻ ⓘ 127.0.0.1:5000

Home

[Login](#)





上海外国语大学  
SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

# Reference

## *Django*

- <http://www.ziqiangxuetang.com/django/django-tutorial.html>

## *Python 操作MySQL 数据库9 个实用实例*

- <http://www.jb51.net/article/76231.htm>





上海外国语大学  
SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

# Home Work

*Design a simple web site and connect your databases to this web site (deadline: Oct.12)*

- It should contain:
  - Source code, including Registration and Login
  - Databases
  - A Report (**very important!!**)
- You may use any web frameworks you like.



上海外国语大学  
SHANGHAI INTERNATIONAL STUDIES UNIVERSITY

Next Time

Next Time

**IT WILL BE YOUR TIME**  
**IT WILL BE YOUR TURN**  
**OCTOBER 12, 2016**

# Next Time

1. 4 groups, 20min for each, where 15min for presentation, and 5min for questions;
2. The speaker should use a PPT to show their business plan;
3. Only one speaker in the stage of presentation, but his/her team member can help them in the stage of questions.



## Tips: How to write a business plan in ten pages?

1. What is the problem?
2. How to solve it?
3. Who is your customer?
4. How big is your market?
5. Your core competence  
(monopoly or sales? If no,  
then technology)
6. Technical methodology
7. Your plan and schedule
8. How to get money?
9. Your team
10. Investment you have got

## *Reference to Business Plan*

- [http://blog.sina.com.cn/s/blog\\_4a01359b0102wnbm.html](http://blog.sina.com.cn/s/blog_4a01359b0102wnbm.html)
- <http://tech2ipo.com/54752>





# The End of Lecture 4

Thank You

<http://www.wangting.ac.cn>

