Connecting Python to a Database Server

We have seen in the lecture that it is possible to access a MySQL database from within a Python program. This is done using Oracle's '`mysql.connector`' module, a platform-independent module for connecting Python programs to MySQL databases. (However, in some of the teaching classrooms it's possible that an older module called '`MySQLdb`' has been installed instead.)

This exercise merely confirms that one of these Python-MySQL connectors has been successfully installed on the computer you're using.

1. Checking to see if a connector is installed

Firstly, open a Python shell and type the following statement.

```
>>> from mysql.connector import *
```

If the module is not installed correctly you will get an error similar to the following.

```
>>> from mysql.connector import *
   Traceback (most recent call last):
     File "<stdin>", line 1, in ?
   ImportError: No module named mysql.connector
```

If this happens in the S-Block or F-Block computer labs try the following statement, to see if the older connector is installed.

```
>>> from MySQLdb import *
```

If the error occurs in one of the Collaborative Learning Spaces it is likely that the particular version of Python you are using is not the one with the connector installed. Some computers in the Collaborative Learning Spaces have two versions of Python installed, and only one of these will have a connector module. Therefore, you will need to open the other Python installation via the Windows "all programs" menu.

2. Installing a connector, if necessary

If you cannot find any Python-MySQL Connector at all, you will need to install it yourself. Instructions for installation of this module are included in the *IFB104 Software Handbook*, and you can also find the module online easily:

`http://dev.mysql.com/downloads/connector/python/`

3. Using the connector in your code

Whichever Python-SQL connector you end up with, you will need to ensure that an appropriate import statement is included at the beginning of your Python program, either

```
from mysql.connector import *
```

or

```
from MySQLdb import *
```

4. Testing the connector

Python scripts that access MySQL through either of these interfaces will, in general, do the following steps:

1. Import the connector module;
2. Create a connection to a MySQL database on a server;
3. Get a "cursor" pointing into the database of interest;
4. Send SQL statements and queries to the database management system and retrieve the results; and
5. Close the connection to the database to unlock it.

A simple Python program, `connect_to_database.py`, has been provided to show how to connect to a MySQL database and retrieve data from it. Run this code to ensure that you can successfully connect to the database. Please note the following points:

- To access the database server we have provided you will need your username and password credentials, which you will have received by email in Week 5.
- Note that this test program assumes the existence of the Airline database from last week's workshop. Before running the Python program you may need to re-create the Airline database by importing the 'dump' script file `airline.sql` into the database first, via a user interface such as *phpMyAdmin*.
- Before the Python program can be run you will need to enter your personal database access credentials into the call to the `connect` function. Failure to do so will result in a '`Programming Error: Access denied`' message.