Querying a Customer Database

As an example of using a "real" database, in this exercise you will run some queries on a large table of survey results. To do this exercise you will need to know your password for accessing your personal database on the MySQL server via the *phpMyAdmin* interface at the following address.

`http://131.181.70.168/phpmyadmin`

A. Import the Customer Data

Accompanying these instructions you will find an SQL script `customers.sql`. Connect to your personal SQL database using the *phpMyAdmin* interface and run this script via the **Import** tab. This script will create a large table called `customers` in your database. When you import the script make sure that you have highlighted your database in the left-hand column so that the system knows which database to put the table in. (Depending on the privileges you have been granted on the MySQL server, you may only be able to see one database schema.)

B. Browse the Customer Data

Click on the `customers` table in the left-hand column to open the *phpMyAdmin* interface's **Browser** tab and use it to examine the contents of the table. This table contains (anonymised) information describing over 3,000 people who completed a survey. Notice that the interface limits the number of rows shown, although you can change the limits if you want to see more rows. Familiarise yourself with the kind of data stored in the table.

C. Query the Customer Data

Now imagine that you want to find out some things about the people in the `customers` table. Given its large size, it would obviously be impractical to browse through it manually. Therefore, you will need to write some SQL queries (**`select`** statements) to solve the following problems.

Ensure that you have highlighted the `customers` table in the left-hand column and have selected the *phpMyAdmin* interface's **Browser** (or **SQL**) tab. Notice that when the interface displays the contents of the table it also shows you the **`select`** query sent to the database server to retrieve the data. You can now choose to 'edit' this statement to create new queries, which you can execute by pressing the 'Go' button. Do this to find answers to the following problems.

1. Show all details of customers who are 60 or older.
2. How many customers in the survey have a doctorate? (You can answer this question just by asking for all customers with this qualification and noting the number of rows returned. Alternatively, you can apply SQL's **`count`** function to the list of columns to be returned, in which case the result produced will be the number of rows.)
3. Show all details of customers who are male and divorced.
4. Show just the ages of male, divorced customers, ordered from youngest to oldest.
5. Show the average age of male, divorced customers. (You can do this by applying SQL's **`avg`** function to the column returned by your previous query.)

6. Show the education level and number of cars of all customers with either a Master or Doctorate degree, ordered by number of cars from highest to lowest.
7. Show all details of the female customers who have a doctorate and have never married or have three or more cars.
8. Show the customer ids and marital status of all customers with six or more cars.
9. Choose one of the customers returned by the previous query and show all the details for just that person.