Summary

My main aim during this project was to show my ability to handle and manage the .data files while sticking to the specification provided by the 'clients brief'. My first action was to plan out how I would implement the project and to develop the JUnit test case for the fileReader module (which I later extended to include the method).

Early on I decided to split the project into two classes, fileReader for handling the .data files formatting of strings so that they could be used within the DBAccess class as SQL queries. For the DBAccess I used MySQL to create the database then to populate the table with data I used the formatted strings from fileReader as well as the formatString method to provide extra formatting to hand persons.data.

Another aim of mine was so that project could manage the current specification from the client but to leave room for it to be modified and expanded should the needs arise. One way in which I did this was to use and ArrayList so that should the need arise I would be able to add much more lines from the .data file due to the ability of the ArrayList to expand and reduce.

Other areas it would be possible to expand on would be to allow users to lnput the file in which they wanted to upload to the database (as long as it fits the formatting set out in the two .data files), more database queries in from the DBAccess class which can be called for access.

What is in the zip file.

In the zip file I have exported the whole netbeans project for you to see

- It contains a separate file to explain the SQL code outside of the JDBC statements as it make it clear what is going on
- The two classes which work in the background managing the files (fileReader) and the Database (DBAccess).
- Third class which runs the program, it creates an instance of the DBAcces and runs some of the methods in it
- There is also a JUnit test in their for the fileReader to make sure it throws the error correctly and that it formats the String in the desired way so that it can be used as SQL