University of Stirling
Computing Science and Mathematics
CSCU9YW - Web Services

Assignment – Music Catalogue

Assignment Outline

The deadline for handing this assignment is **11am, Monday, 25th March 2019**. Separately, you will be asked to demonstrate your solution to the given problem. Demonstrations will take place in the lab session on Monday (25th March) and Thursday (28th March). For the submission you should provide your program code (WSDL file, service implementation class and client) as appendixes to a report document (about 5 pages plus cover sheet with your student number) discussing the problem, any assumptions you made, and **your solution**. The report should include appropriate screen shots of your application. The document also should provide details as to how complete your solution is, and if applicable, any special cases when your program is not working correctly. Please **do not simply repeat the assignment problem –** we are familiar with it.

You should submit your work ***to Canvas***. It will consist of a single document including your report and the contents of the source files. *You are expected to demonstrate your solution*, so please do test out the final version. Make sure that what you submit does work in some fashion. You can delete or comment out incomplete code before submission. After submission, you should ***leave your files*** in the folder untouched, until you are notified of your grade for this assignment.

It is important that your program code is thoroughly and intelligently commented. You will lose marks if the code is not sufficiently and usefully commented! Remember, don't state the obvious. Good comments explain the *use* of variables, and *why* certain values are assigned. Don't say: 'set variable i incremented by 1', but something like: "one additional option". For this assignment, the link between the WSDL definition and your code is especially important and should be commented appropriately.

In short, your assignment should consist of:

- a cover sheet giving your student number
- a report of about 5 pages including screenshots of your application
- a printout of your program code including comments.

Plagiarism

Work which is submitted for assessment must be your own work. Students are encouraged to share ideas. However, each student must individually code their own assignments. Source code will be checked and tested to verify individual work. All students should note that the University has a formal policy on plagiarism which can be found at http://www.stir.ac.uk/academicpolicy/handbook/assessment/#q-8.

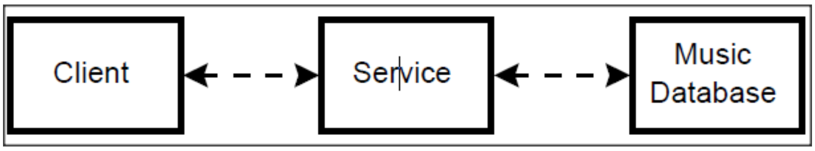
Plagiarism means presenting the work of others as though it were your own. The University takes a very serious view of plagiarism, and the penalties can be severe. Specific guidance in relation to Computing Science assignments may be found in the Computing Science Student Handbook.

We check submissions carefully for evidence of plagiarism, and pursue the cases found. Penalties range from a reduced grade, through a grade 5 for the module, to being required to withdraw from studies.

Assignment Problem

The aim of this assignment is to write a client and service that provide information from a music catalogue. Some aspects of this assignment have already been written. Your task is to add the web service code. Copy the assignment starter files from the Canvas pages.

The Music Album Service



The system architecture is shown above. The client can remotely enquire about music items by composer/artist name or by disc number. The server responds to queries with reference to the music database. The client-server and server-database connections use network links, so all three systems can be distributed. However, for the assignment you will run both client and server on your local system; the music database is supplied for you and runs on a divisional system (MySQL server mysql0).

The ‘music’ database holds information about discs in a table with the following structure:

```
CREATE TABLE music  ( id int NOT NULL,
                      disc int,
                      track int,
                      composer VARCHAR(20),   // e.g. Verdi, Mozart
                      work VARCHAR(20),        //e.g. Nabucco, Aida
                      title VARCHAR(20),        // e.g. Overture
                      PRIMARY KEY (id));
```

As the service uses a MySQL database, you will need to copy the file mysql-connector-java-5.0.8-bin.jar into the lib directory under your tomcat installation on H: (Tomcat will need to be restarted after you copied this file).

Create a new Eclipse project MusicService. Start by completing Service.wsdl using the comments in the file. When you think the WSDL is sufficiently complete, try building the service using the wsdl2java tool. If you get errors from wsd2java, fix the WSDL and try again. Once the WSDL compiles cleanly, look at the generated service skeleton for the operations/methods that it contains. Use the provided MusicService.java as a start to code the service. This has the database interface already coded. However, web service specific code (linking with the WSDL) has been left out. Complete these operations/methods using the comments provided as a guide. If Eclipse does not find the database classes, you will need to include the mysql-connector jar into your build path. Once you code compiles and deploys correctly, turn your attention to the client.

Create a new Eclipse project for the client. Use the provided client.java file. Again, some parts of the client which are not linked to web services specifically have already been written. However, the web service methods are to be added as described in the comments. At this point, try compiling and running the client. If things are working correctly, you should be able to enter a composer/artist’s name and click Check. This should display information about matching tracks in the catalogue. It should also be possible to search using a disc number such as 3. Appropriate handling of error conditions will attract marks.

Assessment Criteria

In this assignment we shall be assessing your work with respect to various criteria, the most important of which are:

- Correctness of operation
- Appropriate use of programming constructs
- Intelligent code comments
- Clear and comprehensive report
- Consistency, legibility and tidiness of program layout

The marks for the assignment are split between the WSDL (30%), the service (20%), client (20%), code comments (10%) and the report (20%).

Handing In

Hand-in will be on Monday, 25th March 2019 at 11am. After hand-in you will be **required** to demo your solution. Demonstrations will take place in the lab sessions on Monday (25th March) and Thursday 28th March. Your submission should include a single document including the report and all of your code,. The title page should include your student number **but not your name**. It is also a good idea to note your student number on each sheet you hand-in. Demos of your solution should not take longer than 10min.

Late submission

If you cannot meet the assignment hand in deadline and have good cause, please see Dr Mario Kolberg to explain the situation and ask for an extension. Coursework will be accepted up to **seven days** after the hand in deadline (or expiry of any agreed extension), with the mark being lowered by three points per day. After seven days the work will be deemed a non-submission and will receive an X (no grade).

Backups: You are advised to make backup copies of your work regularly.