

SCC201 Databases: Assessed Coursework

Write a program (in Java and using JDBC) that takes a database and produces a textual “backup” of the contents.

This textual backup should consist of SQL statements that will recreate the contents of the original database i.e. CREATE TABLE and INSERT .. INTO instructions. Ideally, you should be able to recreate the original database by using the text files your program produces as input to SQLite.

Your program must NOT use the system schema tables found in SQLite; any access to schema information must use the appropriate JDBC methods. The JDBC documentation is linked from the course Moodle page, but another good starting place to find out how to access metadata via JDBC is to google “JDBC metadata”.

Milestones

- a) A single text file containing all the INSERT..INTO statements required.

```
INSERT INTO projects VALUES( COMIC, COMIC, ESPRIT, 100000 );
```

This will not work, as the text field values are not quoted in primes.

- b) A single text file containing all the correct INSERT..INTO statements required.

```
INSERT INTO projects VALUES( 'COMIC', 'COMIC', 'ESPRIT', 100000 );
```

To test this, we will provide you with the CREATE TABLE statements required to create the tables that your text file will document.

- c) A single text file as in (b) above, but also contains at the start the CREATE TABLE statements that create the tables that your text file will document. (But without the primary and foreign keys being indicated).

```
CREATE TABLE give_course(  
    s_id VARCHAR(4),  
    c_id VARCHAR(3)  
);
```

- d) A single text file as in (c) above, but the CREATE TABLE statements include indicators of primary keys.

```
CREATE TABLE give_course(  
    s_id VARCHAR(4),  
    c_id VARCHAR(3),  
    PRIMARY KEY (s_id, c_id )  
);
```

Continues on next page...

- e) A single text file as in (d) above, but including foreign keys.

```
CREATE TABLE give_course(  
    s_id VARCHAR(4),  
    c_id VARCHAR(3),  
    PRIMARY KEY(s_id, c_id),  
    FOREIGN KEY (s_id) REFERENCES staff(s_id),  
    FOREIGN KEY (c_id) REFERENCES courses(c_id)  
);
```

NOTE when executing the CREATE TABLE statements they have to be run in a correct order.

- f) The database metadata also includes details of the indexes present in the database. Add code to establish what indexes are present and include CREATE INDEX statements in your backup to recreate these. You should have a set of statements of the form:

```
CREATE INDEX PRIMARY ON planets (planet_id ASC);
```

- g) If the database structure is modified after creation it is possible for tables to be returned in an order that would break key constraints – i.e. a table depends on a yet to be created table. Tables need sorting based on foreign keys.

As with (e) but with code to ensure the CREATE TABLE statements are in the ‘correct’ order.

Your solution will be tested against the databases you have been given **and at least one other database** – the expectation is that your solution is complete/ correct enough, at the stage you have reached, to **work with any database**. Marks will be awarded for how far you got with your solution, how well it addresses the issues for each stage, correct and well-presented, indented output and, for your approach. Some additional marks may be available for additional features – a simple example might be inclusion of DROP TABLE statements that prepare the database for creating new tables if some already exist, additional flags like NOT NULL on key attributes (there are a few possible), including comments indicating driver and database version information, etc. Marks may be deducted for any concerns relating to output or code quality... structure, efficiency, commenting, etc.

Submission

If you get past milestone (a), you must create a batch file containing all the instructions necessary for running the files created as output by your program to create and populate the database copy.

Submission (to Moodle) Checklist (for the milestone you have reached)

1. All your Java source files.
2. All the output files created by your code (for the milestone you have reached).
3. The batch files required to create the database copies.

To gain marks you must submit to Moodle and demonstrate your solution as submitted to Moodle. Failure to submit, to demo, or demoing code different to that submitted will result in a mark of zero.