

Timed Lab 4 – SQL Backup

This is a Timed Lab; this Timed Lab is worth 21 **Exam** points.

<p>For this Timed Lab, you <i>may</i> use</p> <ul style="list-style-type: none">• Course notes• Homeworks• Recitation assignments• Other course material• Any material you may find on the Internet that don't involve communicating "live" with other people.	<p>However, you <i>may not</i></p> <ul style="list-style-type: none">• Communicate with other people/students in real-time via any means. This means no Facebook, email, Piazza, IM, IRC, cell phones, Google Talk, smoke signals, etc.• Share code with other students.• Look at other students work.
--	--

The TAs will be available to answer clarifying questions about the problem, but they are not permitted to give assistance with debugging code, program logic, etc. You will have an entire recitation period to work on this assignment; this time begins *exactly* when your recitation begins and ends *exactly* when your recitation ends: No extra time will be given if you arrive late, except in highly extenuating circumstances that must be approved by Dr. Summet.

T-Square will not permit any late submissions; ensure that you submit your code to T-Square several times to prevent earning a zero due to you being unable to submit. Your TAs will give a verbal warning 10 and 5 minutes before the end of the recitation period; you should submit at these times.

In your collaboration statement, if you use code from somewhere that is *not* a class resource (i.e. not listed on the course calendar), please list where this code came from. Ensure that you fill out the header at the top of the file.

Note that you must check out with your TA before you leave the recitation room. If you do not check out with your TA or you modify your submission after you leave the recitation room, **you will receive a grade of zero on the timed lab**. No submissions will be accepted after T-Square closes the assignment (i.e. it will not let you submit).

Problem Description:

In this timed lab, you have been tasked with developing a system that will back up information from the GT chat database into XML format. The XML document you will produce will contain the information from the users table as well as the chat messages that each user has sent. Use the tables in the database that you used for Homeworks 9a and 9b.

XML Schema:

You should write out an XML file using a schema that looks like the following example. Your file will have many more user elements and many more message elements:

```
<chatdb>
  <users>
    <user un="USERNAME" password="PASSWORD" firstname="FIRSTNAME">
      <messages>
        <message id="ID">
          MESSAGE TEXT
        </message>
      </messages>
    </user>
  </users>
</chatdb>
```

You will not be required to write a GUI for this timed lab. When you run your file, you should download every user and every chat message that the user has made and create an XML file named chatdump.xml in the above format (i.e. all a user's messages are contained within the messages element for that user, and all users are contained under the users element). The ordering of attributes or the presence of indentation are not required to match exactly what is shown above. Note that you will have multiple user elements contained within the single users element, and you will have multiple message elements contained within the messages element. (Each user will have a single messages element, but the file as a whole will have one messages element per user.) If a user has a NULL firstname in the database, use "None" for their first name.

Grading:

- +2 – Connects to the database
- +3 – Downloads all users data from the database
- +3 – Downloads all message data for each user from the database
- +2 – Closes the database cursor/connection
- +2 – Creates an XML file named chatdump.xml
- +3 – XML format matches the format shown above
- +3 – All users have correct attributes and occur only once in the XML file.
- +3 – All messages are correct and contained under the correct user.