

Please consult the list of references p. 2, and then answer the following. Quiz #7, on Wednesday 29th November, will consist exclusively of questions taken from the Part I of this homework.

## Part I — Short Questions

### Question 1

What is the technology that makes it possible for a Java application to communicate with a DBMS?

### Question 2

What JDBC method do you call to get a connection to a database?

### Question 3

Briefly explain what the `next()` method from the `ResultSet` class does, and what is its return type.

### Question 4

What is the purpose of the `LIKE` operator?

### Question 5

How do you submit a `SELECT` statement to the DBMS?

### Question 6

Where is a `ResultSet` object's cursor initially pointing? How do you move the cursor forward in the result set?

### Question 7

Give three navigation methods provided by `ResultSet`.

### Question 8

Explain this JDBC URL format:

```
jdbc:mysql://localhost:3306/HW_NewDB?createDatabaseIfNotExist=true&useSSL=true
```

### Question 9

In what class is the `getColumnName()` method?

### Question 10

What is a prepared statement?



## Part II — Problems

This part will mainly help you to develop your first database program. It is important that you learn how to get a working environment for a new technology quickly, for your understanding of this lecture, for the exam, and as a CS major. I'll assume that you will have successfully completed those problems by Wednesday 29th November, so don't wait and let me know if you had difficulties solving them.

### References

- This second part is strongly inspired from [https://www.ntu.edu.sg/home/ehchua/programming/java/JDBC\\_Basic.html](https://www.ntu.edu.sg/home/ehchua/programming/java/JDBC_Basic.html). If you experience any trouble, the “Common Error Message” page, at <https://www.ntu.edu.sg/home/ehchua/programming/howto/ErrorMessage.html#JDBCErrors> might be the best place to look first.
- Section 13.3.2 of your textbook is a condensed, but good read. Many textbook on Java includes a part on Databases, cf. for instance the Chapter 16 of *Starting Out with Java: Early Objects* (5th Edition) by Tony Gaddis.

### Problem 1

Consider the code below:

```
1  CREATE DATABASE HW_ebookshop;
2
3  USE HW_ebookshop;
4
5  CREATE TABLE books (
6      id int PRIMARY KEY,
7      title varchar(50),
8      author varchar(50),
9      price float,
10     qty int
11 );
12
13 INSERT INTO books VALUES (1001, 'Java for dummies', 'Tan Ah Teck', 11.11, 11);
14 INSERT INTO books VALUES (1002, 'More Java for dummies', 'Tan Ah Teck', 22.22, 22);
15 INSERT INTO books VALUES (1003, 'More Java for more dummies', 'Mohammad Ali', 33.33, 33);
16 INSERT INTO books VALUES (1004, 'A Cup of Java', 'Kumar', 44.44, 44);
17 INSERT INTO books VALUES (1005, 'A Teaspoon of Java', 'Kevin Jones', 55.55, 55);
18
19 SELECT * FROM books;
```

Save this code as `HW_ebookshop.sql`, or extract it from the archive located at [http://spots.augusta.edu/caubert/teaching/2017/fall/csci3410/code/2017\\_11\\_15\\_FirstProg.zip](http://spots.augusta.edu/caubert/teaching/2017/fall/csci3410/code/2017_11_15_FirstProg.zip). Then, open a terminal (or command-line interpreter), navigate to the folder where you stored that file (using `cd`), and type

```
mysql -u testuser -p < HW_ebookshop.sql
```

for linux, or (something like)

```
"C:\Program Files\MySQL\MySQL Server 5.7\bin\mysql.exe" -u testuser -p <
↪ HW_ebookshop.sql
```

for Windows.

You just discovered MySQL's batch mode, that perform *series* of instructions from a file.

### Problem 2

This exercise supposes you successfully completed Problem 1. We will compile and execute your first database application, using Java and MySQL.

- (a) I will assume that you have MySQL installed and set-up as indicated in Homeworks #1 and #2.
- (b) I will assume that you have Java installed. If not, please refer to [http://spots.augusta.edu/caubert/teaching/2017/fall/csci3410/code/11\\_13\\_Intro\\_Java.zip](http://spots.augusta.edu/caubert/teaching/2017/fall/csci3410/code/11_13_Intro_Java.zip) for a simple program and the instructions to compile and execute it.
- (c) We need to set up the *driver* (or *connector*) to make the java sql API and MySQL communicate. To do so,
  - Go to <https://dev.mysql.com/downloads/connector/j/>
  - Click on “Download” in front of “Platform Independent (Architecture Independent), ZIP Archive”
  - Look for the (somewhat hidden) “No thanks, just start my download.”
  - You will download a file named “mysql-connector-java-5.1.44.zip”
  - Upon completion of the download, unzip the file, and locate the “mysql-connector-java-5.1.44-bin.jar” file.
  - Copy that file to the directory where you extracted the archive downloaded from [http://spots.augusta.edu/caubert/teaching/2017/fall/csci3410/code/2017\\_11\\_15\\_FirstProg.zip](http://spots.augusta.edu/caubert/teaching/2017/fall/csci3410/code/2017_11_15_FirstProg.zip).
- (d) Open a terminal in that same folder, and compile `FirstProg.java`, using  

```
javac FirstProg.java
```

(or an equivalent command for windows). Normally, nothing will be printed, but a `FirstProg.class` file will be created.
- (e) Now, execute that program, using  

```
java -cp .:mysql-connector-java-5.1.44-bin.jar FirstProg
```

in Linux, or  

```
java -cp .;mysql-connector-java-5.1.44-bin.jar FirstProg
```

in Windows. The `-cp` option lists the places where java should look for the class used in the program: we are explicitly asking java to use the `mysql-connector-java-5.1.44-bin.jar` executable to execute our `FirstProg` executable. Try to execute `FirstProg` without that flag, and see what happens.

