ASSIGNMENT

1. How do you set up the class path to be able to use the sql-connector jar?

First, second click on the project you are going to be using. Then down to Build Path -> Configure Build Path. A window will pop up and you want to make sure you select the Libraries tab, then click the Add External JARs button on the right. Browse through and find where you stored your jar file, once selected, hit Apply and Close. There should now see Referenced Libraries dropdown below your project and the sql-connector jar should be right there.

2. What is a JDBC driver?

A JDBC driver is used to establish the connection to the database. There are different types of drivers that can be used to connect to many different types of database like MySQL and PostgreSQL.

3. What does the DriverManger class do?

Selects the driver needed to connect to the database and can then create and return a connection to that database.

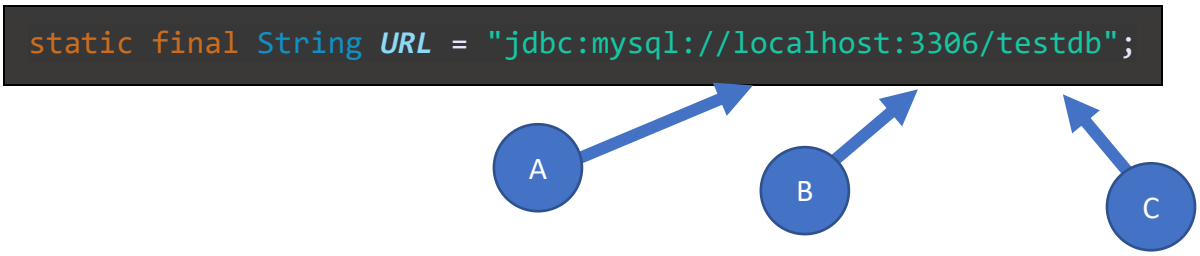
4. When creating the connection to our SQL database why must we surround this code in a try-catch block?

If there is an issue connecting to the database, whether because there is a bad connection to the server or the password given is not correct, there will be an SQLException thrown. Since there is a possibility of this exception being thrown, we should surround this code in a try-catch block.

5. What must you do at the end of your program once you are finished using your connection?

Once finished with the connection, you should always make sure to close it.

6. What do arrows A, B, and C point to in the url?



A: Hostname

B: Port number

C: Database name

7. Can you connect to your database with the URL = “jdbc:mysql:/testdb”?

Yes, if the database, testdb, is already on your localhost on port 3306.

8. What are the 4 main steps to retrieving information from a database using JDBC?

One: Establish a connection

Two: Create a statement

Three: Execute the statement with an SQL command

Four: Use the Result/ResultSet returned from the statement

9. Can you execute a query using a Statement object if you close your connection?

No, once the connection, there is no way to reach the database to execute your statement.

10. What kinds of SQL DML statements can you run using the executeUpdate() method?

You can execute update, delete and insert statements using this method.

11. What is a ResultSet?

A ResultSet is an object that contains the results of an executed query. Data records can be stored and pulled from this ResultSet.

12. Can you still use a ResultSet if you close your Statement and Connection?

No, a ResultSet is tied to a Statement, which is in turn tied to a Connection. Cutting one link will make you unable to use your ResultSet.

13. The ResultSet get methods retrieve column data, what two types of parameters can you pass to these methods to get your column values?

You can either pass a String that contains the name of the column you want or an int that indicates the column number (this column number starts at 1 with the first column in a table).

14. What kinds of SQL statements can you pass along to the executeQuery(), executeUpdate(), and execute() command?

executeQuery(): select statements

executeUpdat(): insert, update, and delete

execute(): any SQL statement

15. What 2 design methods should you try to use when creating a connection manager class?

Use Static Factory Method (creating an object) and Singleton (create one instance of a class).

16. What is a properties file? How is it useful when creating a connection manager?

A properties file holds information to use in parts of your code. Can be used to set variables. You can use it to store your URL, Username, and Password for your connection so its not displayed in your code.

17. What is a RowSet?

Rowset inherits from ResultSet, it a more flexible class that can be used to retrieve data from the database. Unlike ResultSet, it does not need a Statement or PreparedStatement and has it's own scrollability and updatability.

18. What kind of information can we get from ResultSetMetadata?

You can obtain information on the data structure returned by the ResultSet, most commonly, the column information for tables.

19. What kind of information can we get from DatabaseMetadata?

You can obtain information on the different databases in your connection as well as information on the connection.

ASSIGNMENT

Option 1:

Create a database management system that allows users to login and manage all their databases so they can insert, update, delete, and view the data in their tables. The requirements are below:

- There should be some sort of UI where the user can login (can be console, using JSP, swing, etc.) where the user gives their username, password, and database they wish to access (we are assuming they will be using localhost and port 3306.
- Once they login, they will see a menu that lists the tables in the database. They can choose the table they wish to view.
- When they choose the table, depending on how you wish to implement this, they will have the choice to insert, update, delete, and view data from the table.
- Once the user is done, make sure there is a way for them to logout and close their connection.

BONUS: When the user logs in, instead of providing the database at login, only have them put in their username and password. After logging in, the user will see a menu of all their databases, they can then select which database they wish to work with.

Option 2:

Create a UI where users can register their child for Little League. The registration page can be displayed in Angular, Swing, or on the console. Please see the UML provided and run the script to create the database. There is already data in this database that already has some registered players, please look through the database to understand the data being collected based on one registered player. The requirements for this project are below:

- The registration page will ask for parent's information as well as the player's information for the new 2020 season. Though the guardian for the player may put in their information, they can include information for another guardian.

- Assume that all players registering on your page for this season are all new players and thus, are not already in the database.
- When registering a new player, you must be able to generate a player number (number assigned to players on a team) and make sure it does not conflict with anyone else's number on the team. The player number CANNOT be chosen by the user on the registration page.
- Players can choose from a list of schools already in the database on the registration page. There should also be an option for a player to select a home school option. In that case, the player record will return a null value for school_id.
- You must use JDBC to connect to your database and execute your queries using prepared statements.

If you finish the requirements above, try adding the following features as a bonus:

- If a player's school is not listed in the choices given, the player has the option to add their school and update the school table.
- Create an option for users to create an account so they can register their kids for little league.
    - Create a table called account which has these fields: account_id, username, password, guardian_id. The guardian_id should be a foreign key to the guardian table.
    - Once logged in, guardians can register multiple kids.