03/14/2022 Homework

1. Statement vs PreparedStatement vs CallableStatement
2. Setup jdbc in your local laptop and run the CRUD operation.

**Statement vs. PreparedStatement vs. CallableStatement**

There are 3 execute methods:

1. boolean execute(String SQL): returns a boolean value of true if a ResultSet object can be retrieve; otherwise, it returns false. Use this method to execute SQL DDL statements or when you need to use truly dynamic SQL.
2. int executeUpdate(String SQL): returns the number of rows affected by the execution of the SQL statement. Use this for which you expect to get a number of rows affected, such as INSERT, UPDATE, or DELETE statements.
3. ResultSet executeQuery(String SQL): returns a ResultSet object. Use this method when you expect to get a result set, as you would with a SELECT statement.

Statement: This is used to execute a static SQL queries. It does not pass parameters to DQL queries at runtime. Use this when you are using static SQL statements at runtime.

1. Less readable
2. Vulnerable to SQL injection
3. JDBC passes the query with inline values to the database so there is no query optimization, and the database engine must ensure all the checks
4. The query will not appear as the same to the database and it will prevent cache usage
5. Batch updates need to be executed separately
6. Suitable for DDL queries like CREATE, ALTER, and DROP
7. Can’t be used for storing and retrieving files and arrays

PreparedStatement: This is used to execute dynamic or parameterized SQL queries. It extends the Statement interface. It has methods to bind various object types, including files and arrays. You must bind all the parameters before executing, or you will receive an SQL Exception. Use this when you plan to use the SQL statements many times.

1. Code is easier to understand
2. It protects against SQL injection
3. Uses pre-compilation. It will check the cache before pre-compiling the query. If it is not cached, the database engine will save it for the next usage
4. Speeds up the communication between the database and the JVM through non-SQL binary protocol
5. Provides a batch execution during a single database connection
6. Provides an easy way to store and retrieve files by using BLOB and CLOB data types
7. Implements methods like getMetadata() that contain information about the returned result

We use setXXX(//number of position, //value ) to set values to place holders where XXX is the type.

CallableStatement: This is used to execute stored procedures, cursors, and Functions. It extends PreparedStatement. Use this when you want to access the database stored procedures that are already compiled and stored in server. You can pass 3 types of parameters to stored procedures:

IN – used to pass the values to stored precedures

OUT – used to hold the result returned by the stored procedures

INOUT – acts as both IN and OUT parameters

Before calling the stored procedure, use registerOutParameter() to register OUT parameter.

We use setter() to pass IN parameters.

We use getter() to retrieve the result.

1. Performance is high as it calls the stored procedures which are already compiled and stored in the database server