Candace Samuels

Week 7: Coding Assignment

URL to GitHub Repository:

https://github.com/CSAMU4/backend_week_7_coding_assignment/tree/master

URL to Public Link of your Video:

https://screenrec.com/share/tOQjHbKAr4

Solutions:

pom.xml file

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-
4.0.0.xsd">
	<modelVersion>4.0.0</modelVersion>

	<groupId>com.promineotech</groupId>
	<artifactId>mysql-java</artifactId>
	<version>0.0.1-SNAPSHOT</version>

	<properties>
		<java.version>11</java.version>

	</properties>
	<dependencies>
		<dependency>
			<groupId>com.mysql</groupId>
			<artifactId>mysql-connector-j</artifactId>
			<version>8.0.32</version>
		</dependency>
	</dependencies>

	<build>
```

```
            <pluginManagement>
                  <plugins>
                        <plugin>
                              <groupId>org.apache.maven.plugins</groupId>
                              <artifactId>maven-compiler-plugin</artifactId>
                              <version>3.10.1</version>
                              <configuration>
                                    <source>${java.version}</source>
                                    <target>${java.version}</target>
                              </configuration>
                        </plugin>
                  </plugins>
            </pluginManagement>
      </build>
</project>
```

DbConnection.java

```
package projects.dao;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

import projects.dao.exception.DbException;

/**
 * @author Candace Samuels
 *
 */
public class DbConnection {
```

```
	private static final String SCHEMA = "projects";
	private static final String USER = "projects";
	private static final String PASSWORD = "projects";
	private static final String HOST = "localhost";
	private static final int PORT = 3306;

	public static Connection getConnection() {
		// create a URL
		String url =
String.format("jdbc:mysql://%s:%d/%s?user=%s&password=%s&useSSL=false", HOST, PORT,
SCHEMA, USER,
				PASSWORD);

		try {
			Connection conn = DriverManager.getConnection(url);
			System.out.println("Connection to schema '" + SCHEMA + "' is
successful!");
			return conn;
		} catch (SQLException e) {
			System.out.println("Unable to get connection at " + url);
			throw new DbException();
			// create an unchecked exception class from the runtime exception
superclass

		}
	}
}
```

DbException.java

```
package projects.dao.exception;

/**
```

```
 * @author Candace Samuels
 *
 */
@SuppressWarnings("serial")
public class DbException extends RuntimeException {

	/**
	 *
	 */
	public DbException() {
		// TODO Auto-generated constructor stub
	}

	/**
	 * @param message
	 */
	public DbException(String message) {
		super(message);
		// TODO Auto-generated constructor stub
	}

	/**
	 * @param cause
	 */
	public DbException(Throwable cause) {
		super(cause);
		// TODO Auto-generated constructor stub
	}

	/**
	 * @param message
	 * @param cause
	 */
```

```
	public DbException(String message, Throwable cause) {
		super(message, cause);
		// TODO Auto-generated constructor stub
	}
```

ProjectApp.java

```
package projects.dao;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

import projects.dao.exception.DbException;

/**
 * @author Candace Samuels
 *
 */
public class DbConnection {

	private static final String SCHEMA = "projects";
	private static final String USER = "projects";
	private static final String PASSWORD = "projects";
	private static final String HOST = "localhost";
	private static final int PORT = 3306;

	public static Connection getConnection() {
		// create a URL
		String url =
String.format("jdbc:mysql://%s:%d/%s?user=%s&password=%s&useSSL=false", HOST, PORT,
SCHEMA, USER,
				PASSWORD);
```

```
            try {
                  Connection conn = DriverManager.getConnection(url);
                  System.out.println("Connection to schema '" + SCHEMA + "' is
successful!");
                  return conn;
            } catch (SQLException e) {
                  System.out.println("Unable to get connection at " + url);
                  throw new DbException();
                  // create an unchecked exception class from the runtime exception
superclass


            }
      }
}
```

.gitignore

```
.settings/
/target/
```