

Lesson 01 Demo 08

Extracting Database Information Using Metadata

Objective: To extract information about the database in a Java program using metadata for understanding the database structure and schema. This approach allows for dynamic retrieval of database details such as tables, columns, and data types.

Tool required: Eclipse IDE

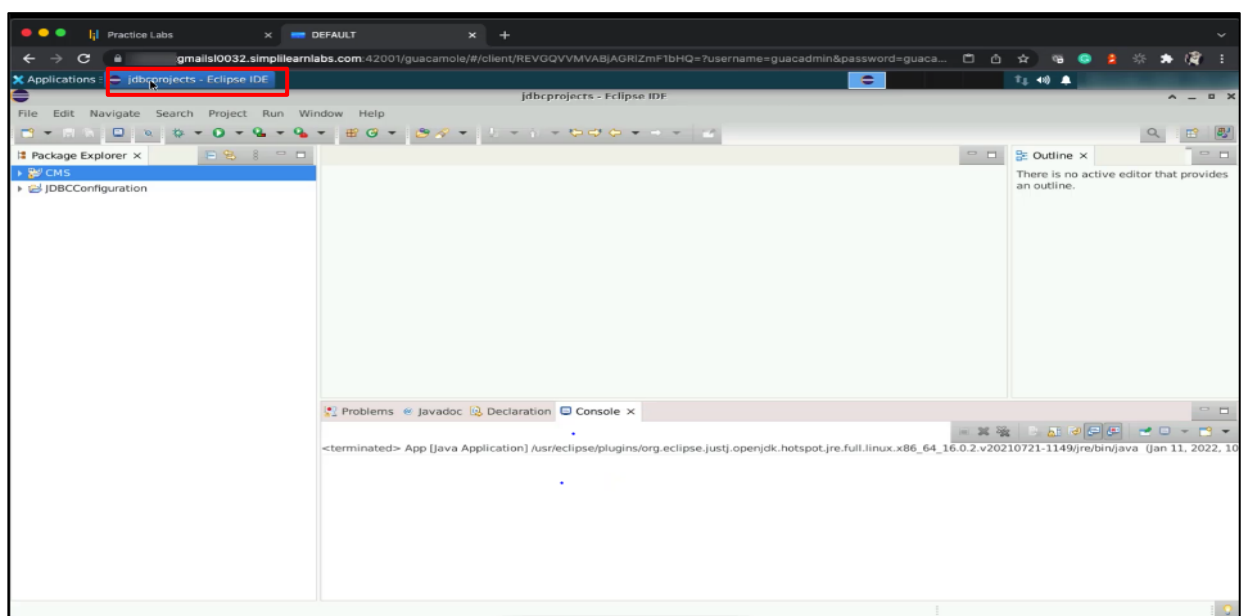
Prerequisites: None

Steps to be followed:

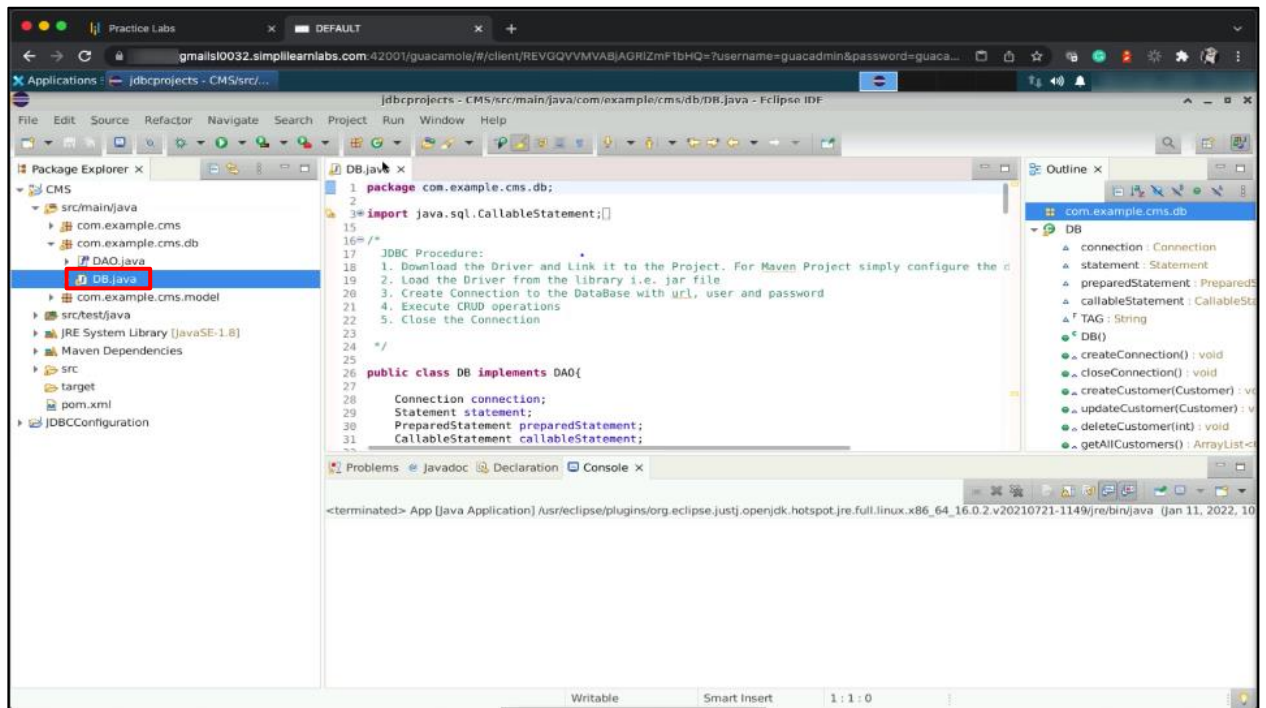
1. Extract details about the database and list of tables
2. Extract Metadata containing the list of tables

Step 1: Extract details about the database and list of tables

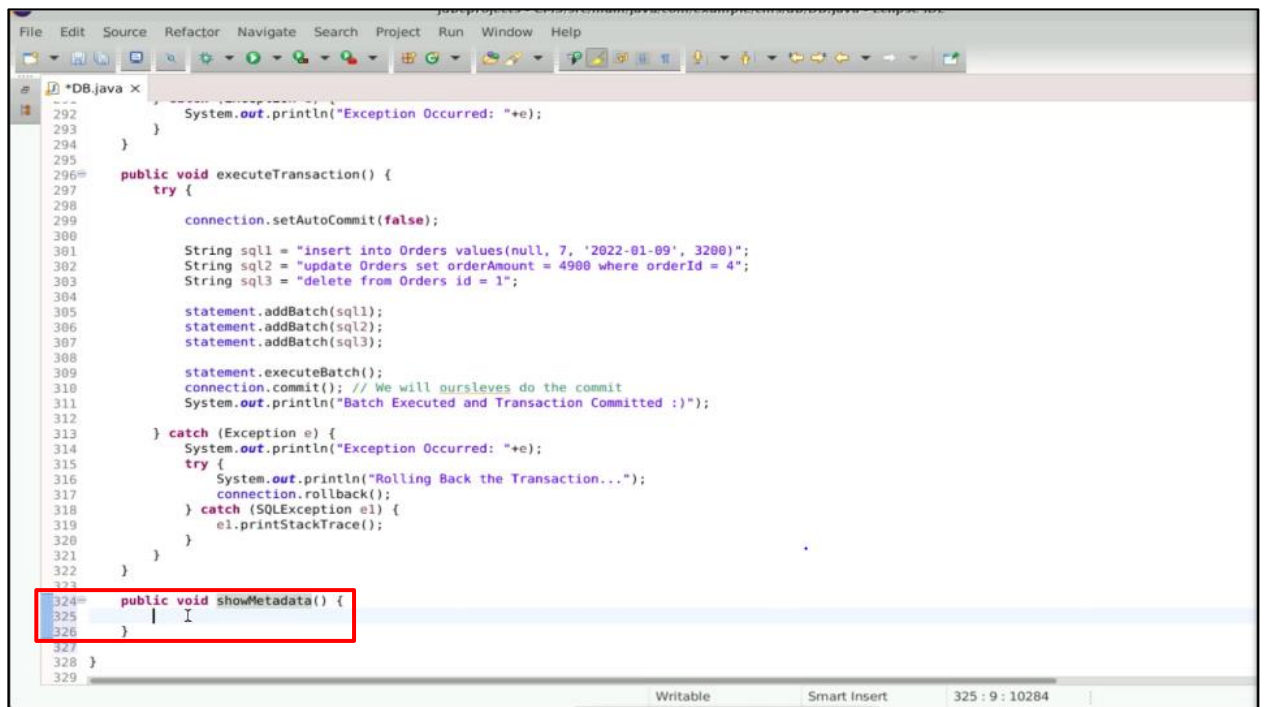
1.1 Open Eclipse IDE



1.2 Open CMS > src/main/java > com.example.cms.db > DB.java file



1.3 Create a new method showMetadata()



1.4 Write the **try-catch** block to work on the object **DatabaseMetaData** object which will provide the database information by using the **getMetaData** method

```

299
300
301     connection.setAutoCommit(false);
302     String sql1 = "Insert into Orders values(null, 7, '2022-01-09', 3200)";
303     String sql2 = "update Orders set orderAmount = 4900 where orderId = 4";
304     String sql3 = "delete from Orders id = 1";
305
306     statement.addBatch(sql1);
307     statement.addBatch(sql2);
308     statement.addBatch(sql3);
309
310     statement.executeBatch();
311     connection.commit(); // We will ourselves do the commit
312     System.out.println("Batch Executed and Transaction Committed :)");
313
314 } catch (Exception e) {
315     System.out.println("Exception Occurred: "+e);
316     try {
317         System.out.println("Rolling Back the Transaction...");
318         connection.rollback();
319     } catch (SQLException e1) {
320         e1.printStackTrace();
321     }
322 }
323
324
325 public void showMetadata() {
326     try {
327         DatabaseMetaData dbMetaData = connection.getMetaData();
328         System.out.println("Product Name: "+dbMetaData.getDatabaseProductName());
329         System.out.println("Version: "+dbMetaData.getDatabaseProductVersion()+" Minor: "+dbMetaData.getDatabaseMinorVersion()+" Major: "+dbMetaData.getDatabaseMajorVersion());
330         System.out.println("Driver Version: "+dbMetaData.getDriverVersion());
331         System.out.println("Driver Name: "+dbMetaData.getDriverName());
332         System.out.println("UserName: "+dbMetaData.getUserName());
333     } catch (Exception e) {
334         System.out.println("Exception Occurred: "+e);
335     }
336 }

```

1.5 Run the **showMetadata** method from **App.java** by adding the below line in the **App.java** file:

db.showMetadata()

```

33
34 //db.deleteCustomer(3);
35
36 //System.out.println();
37
38 ArrayList<Customer> customers = db.getAllCustomers();
39 customers.forEach( cRef -> System.out.println(cRef));
40
41 db.closeConnection();
42
43 //Scanner scanner = new Scanner(System.in);
44 //System.out.println("Enter Name: ");
45 //String name = scanner.nextLine();
46
47 //System.out.println("Enter Password: ");
48 //String password = scanner.nextLine();
49
50 //System.out.println("Enter Customer ID:");
51 //int cid = scanner.nextInt();
52
53 //scanner.close();
54
55
56 DB db = new DB();
57 db.createConnection();
58 //db.executeProcedure(name, password);
59 //db.executeProcedure(cid);
60
61 //db.executeSQLStatementsInBatch();
62
63 //db.executeTransaction();
64 db.showMetadata();
65
66 db.closeConnection();
67
68 }
69 }

```

1.6 Run the code. In the output, see all the attributes, like Product Name and Version printed using the try-catch block.

```
<terminated> App [Java Application] /usr/eclipse/plugins/org.eclipse.justi.openjdk.hotspot.jre.full.linux.x86_64_16.0.2.v20210721-1149/jre/bin/java (Jan 11, 2022, 10:34:42 AM -
[DB] Driver Loaded
[DB] Connection Created
Product Name: MySQL
Version: 8.0.27-0ubuntu0.20.04.1 Minor: 0 Major: 8
Driver Version: mysql-connector-java-8.0.27 (Revision: e920b979015ae7117d60d72bcc8f077a839cd791)
Driver Name: MySQL Connector/J
UserName: john@localhost
[DB] Connection Closed. Close Status: true
```

1.7 Log in into **mysql** by opening the terminal

```
erishantgmail@ip-172-31-17-157: ~
erishantgmail@ip-172-31-17-157:~$ mysql -u john -p
Enter password:
```

1.8 Type the command **use estore;** to change the database

```
erishantgmail@ip-172-31-17-157: ~
erishantgmail@ip-172-31-17-157:~$ mysql -u john -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 20
Server version: 8.0.27-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use estore;
```

1.9 Write the **show tables;** command to list the tables in the **estore** database

```

erishantgmail@ip-172-31-17-157: ~$ mysql -u john -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 20
Server version: 8.0.27-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use estore;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;

```

```

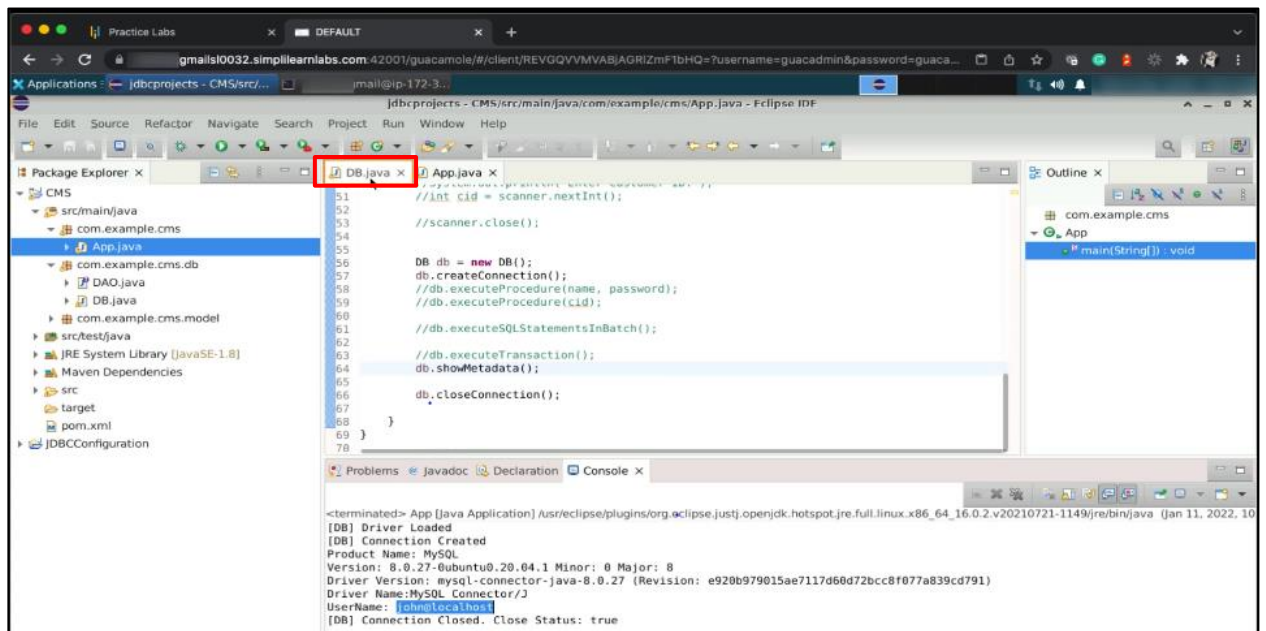
mysql> show tables;
+-----+
| Tables_in_estore |
+-----+
| Customer          |
| Employees         |
| Orders            |
| User              |
+-----+
4 rows in set (0.00 sec)

mysql>

```

Step 2: Extract Metadata containing the list of tables

2.1 Open Eclipse IDE and go to the **DB.java** file to see the metadata containing the list of tables



2.2 Create a projection array for extracting a table

```

322     }
323 }
324
325 public void showMetadata() {
326     try {
327         DatabaseMetaData dbMetaData = connection.getMetaData();
328         System.out.println("Product Name: "+dbMetaData.getDatabaseProductName());
329         System.out.println("Version: "+dbMetaData.getDatabaseProductVersion()+" Minor: "+
330             dbMetaData.getDatabaseProductVersion());
331         System.out.println("Driver Version: "+dbMetaData.getDriverVersion());
332         System.out.println("Driver Name: "+dbMetaData.getDriverName());
333         System.out.println("UserName: "+dbMetaData.getUserName());
334
335         String projection[] = {"TABLE"};
336     } catch (Exception e) {
337         System.out.println("Exception Occurred: "+e);
338     }
339 }
340
341

```

```

<terminated> App [Java Application] /usr/eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.linux.x86_64.16.0.2.v20210721-1149/jre/bin/java (Jan 11, 2022, 10
[DB] Driver Loaded
[DB] Connection Created
Product Name: MySQL
Version: 8.0.27-0ubuntu0.20.04.1 Minor: 0 Major: 8
Driver Version: mysql-connector-java-8.0.27 (Revision: e920b979015ae7117d60d72bcc8f077a839cd791)
Driver Name: MySQL Connector/J
UserName: john@localhost
[DB] Connection Closed. Close Status: true

```

2.3 Create a result set using the ResultSet object to filter the data

```

322     }
323 }
324
325 public void showMetadata() {
326     try {
327         DatabaseMetaData dbMetaData = connection.getMetaData();
328         System.out.println("Product Name: "+dbMetaData.getDatabaseProductName());
329         System.out.println("Version: "+dbMetaData.getDatabaseProductVersion()+" Minor: "+
330             dbMetaData.getDatabaseProductVersion());
331         System.out.println("Driver Version: "+dbMetaData.getDriverVersion());
332         System.out.println("Driver Name: "+dbMetaData.getDriverName());
333         System.out.println("UserName: "+dbMetaData.getUserName());
334
335         String projection[] = {"TABLE"};
336         ResultSet set = dbMetaData.getTables(null, null, null, projection);
337     } catch (Exception e) {
338         System.out.println("Exception Occurred: "+e);
339     }
340
341

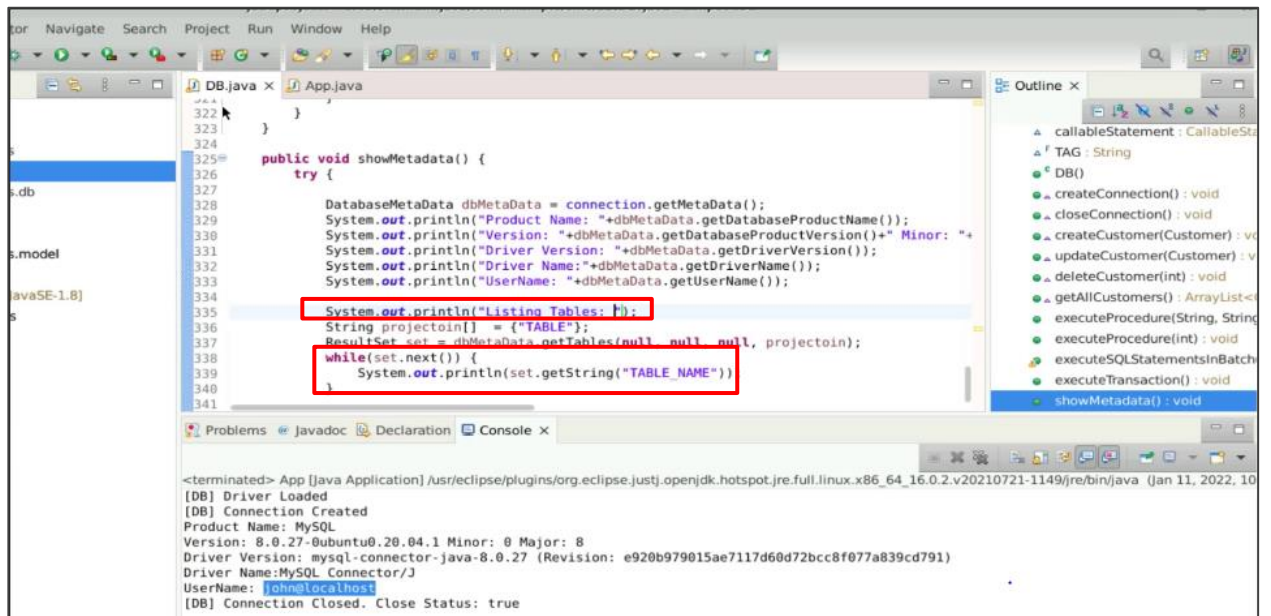
```

```

<terminated> App [Java Application] /usr/eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.linux.x86_64.16.0.2.v20210721-1149/jre/bin/java (Jan 11, 2022, 10
[DB] Driver Loaded
[DB] Connection Created
Product Name: MySQL
Version: 8.0.27-0ubuntu0.20.04.1 Minor: 0 Major: 8
Driver Version: mysql-connector-java-8.0.27 (Revision: e920b979015ae7117d60d72bcc8f077a839cd791)

```

2.4 Loop through the result set and add the print statement



The screenshot shows the Eclipse IDE with a Java project. The main editor displays the `DB.java` file, which contains a `showMetadata()` method. The code is as follows:

```

322 }
323 }
324
325 public void showMetadata() {
326     try {
327
328         DatabaseMetaData dbMetaData = connection.getMetaData();
329         System.out.println("Product Name: "+dbMetaData.getDatabaseProductName());
330         System.out.println("Version: "+dbMetaData.getDatabaseProductVersion()+" Minor: "+
331         System.out.println("Driver Version: "+dbMetaData.getDriverVersion());
332         System.out.println("Driver Name: "+dbMetaData.getDriverName());
333         System.out.println("UserName: "+dbMetaData.getUserName());
334
335         System.out.println("Listing Tables: ");
336         String projectoin[] = {"TABLE"};
337         ResultSet set = dbMetaData.getTables(null, null, null, projectoin);
338         while(set.next()) {
339             System.out.println(set.getString("TABLE_NAME"))
340         }
341     }
342 }

```

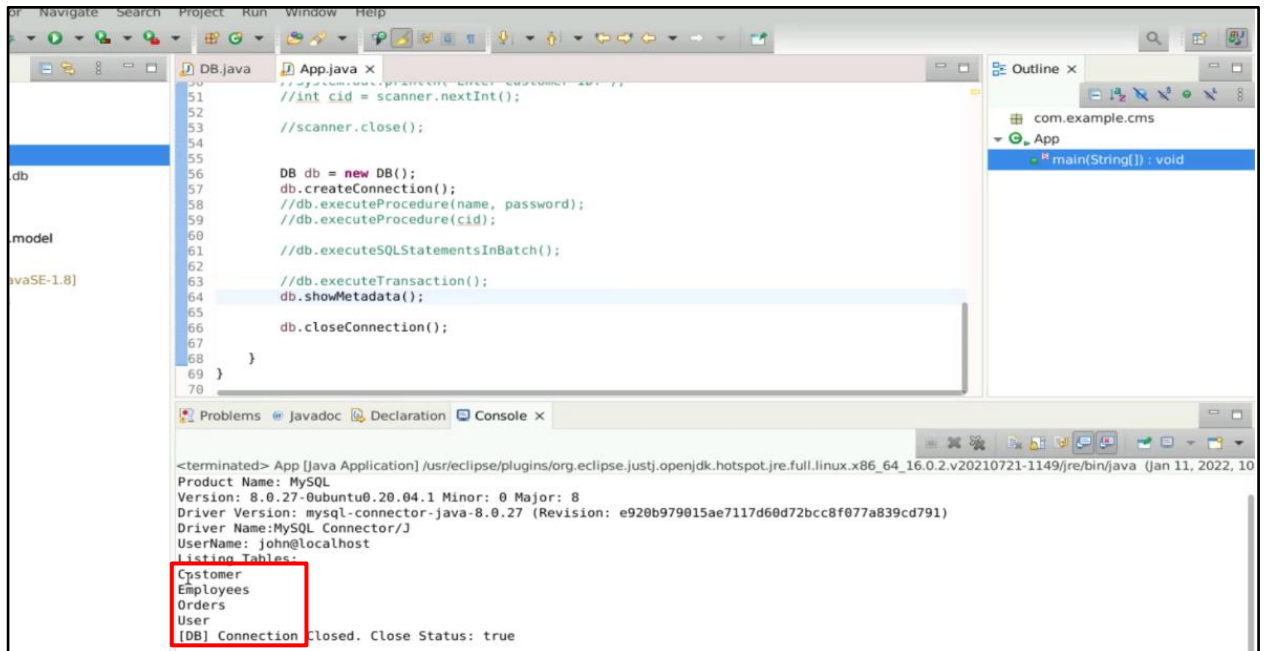
The console output at the bottom shows the execution results:

```

<terminated> App [Java Application] /usr/eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.linux.x86_64.16.0.2.v20210721-1149/jre/bin/java (Jan 11, 2022, 10
[DB] Driver Loaded
[DB] Connection Created
Product Name: MySQL
Version: 8.0.27-0ubuntu0.20.04.1 Minor: 0 Major: 8
Driver Version: mysql-connector-java-8.0.27 (Revision: e920b979015ae7117d60d72bcc8f077a839cd791)
Driver Name: MySQL Connector/J
UserName: john@localhost
[DB] Connection Closed. Close Status: true

```


2.5 Run the program from the **App.java** file. See the output similar to the **show tables;** command in the console.



```

DB.java
App.java x
//int cid = scanner.nextInt();
//scanner.close();
DB db = new DB();
db.createConnection();
//db.executeProcedure(name, password);
//db.executeProcedure(cid);
//db.executeSQLStatementsInBatch();
//db.executeTransaction();
db.showMetadata();
db.closeConnection();
}
}

Problems Javadoc Declaration Console x
<terminated> App [java Application] /usr/eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.linux.x86_64.16.0.2.v20210721-1149/jre/bin/java (Jan 11, 2022, 10
Product Name: MySQL
Version: 8.0.27-0ubuntu0.20.04.1 Minor: 0 Major: 8
Driver Version: mysql-connector-java-8.0.27 (Revision: e920b979015ae7117d60d72bcc8f077a839cd791)
Driver Name:MySQL Connector/J
UserName: john@localhost
Listing Tables:
Customer
Employees
Orders
User
[DB] Connection Closed. Close Status: true
  
```

By following these steps, you have successfully extracted information about the database in a Java program using metadata for understanding the database structure and schema. This approach allows for dynamic retrieval of database details such as tables, columns, and data types.