

# Relational Databases with MySQL Week 10 Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

**Instructions:** In Eclipse, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document to the repository. Additionally, push an .sql file with all your queries and your Java project code to the same repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

## Coding Steps:

In this week's coding activity, you will create a menu driven application backed by a MySQL database.

To start, choose one item that you like. It could be vehicles, sports, foods, etc....

Create a new Java project in Eclipse.

Create a SQL script in the project to create a database with one table. The table should be the item you picked.

Write a Java menu driven application that allows you to perform all four CRUD operations on your table.

Tips:

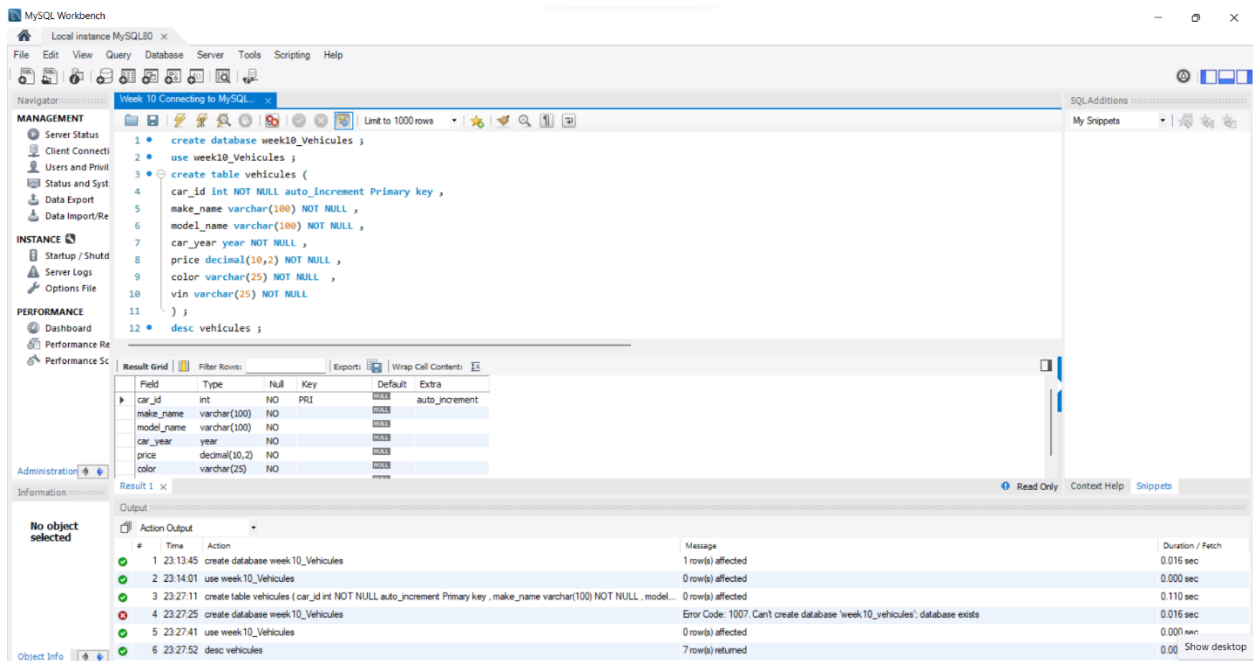
The application does not need to be as complex as the example in the video curriculum.

You need an option for each of the CRUD operations (Create, Read, Update, and Delete).

Remember that `PreparedStatement.executeQuery()` is only for Reading data and `.executeUpdate()` is used for Creating, Updating, and Deleting data.

Remember that both parameters on `PreparedStatement`s and the `ResultSet` columns are based on indexes that start with 1, not 0.

## Screenshots of Code:



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

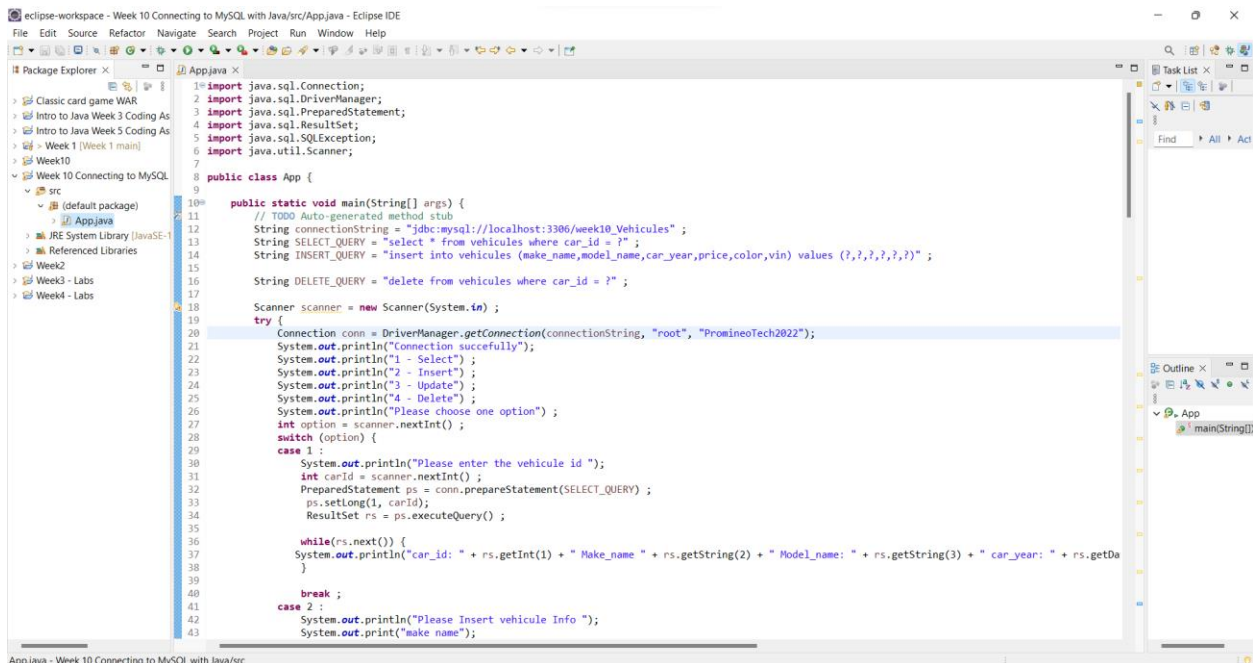
```
1 create database week10_Vehicles ;
2 use week10_Vehicles ;
3 create table vehicles (
4   car_id int NOT NULL auto_increment Primary key ,
5   make_name varchar(100) NOT NULL ,
6   model_name varchar(100) NOT NULL ,
7   car_year year NOT NULL ,
8   price decimal(10,2) NOT NULL ,
9   color varchar(25) NOT NULL ,
10  vin varchar(25) NOT NULL
11 ) ;
12 desc vehicles ;
```

The Results tab shows the output of the `desc vehicles` query:

Field	Type	Null	Key	Default	Extra
car_id	int	NO	PRI		auto_increment
make_name	varchar(100)	NO			
model_name	varchar(100)	NO			
car_year	year	NO			
price	decimal(10,2)	NO			
color	varchar(25)	NO			
vin	varchar(25)	NO			

The Output tab shows the execution log:

#	Time	Action	Message	Duration / Fetch
1	23:13:45	create database week10_Vehicles	1 row(s) affected	0.016 sec
2	23:14:01	use week10_Vehicles	0 row(s) affected	0.000 sec
3	23:27:11	create table vehicles (car_id int NOT NULL auto_increment Primary key , make_name varchar(100) NOT NULL , model...	0 row(s) affected	0.110 sec
4	23:27:25	create database week10_Vehicles	Error Code: 1007. Can't create database 'week10_vehicles'; database exists	0.016 sec
5	23:27:41	use week10_Vehicles	0 row(s) affected	0.000 sec
6	23:27:52	desc vehicles	7 row(s) returned	0.000 sec



The screenshot shows the Eclipse IDE with the following Java code in `App.java`:

```
1 import java.sql.Connection;
2 import java.sql.DriverManager;
3 import java.sql.PreparedStatement;
4 import java.sql.ResultSet;
5 import java.sql.SQLException;
6 import java.util.Scanner;
7
8 public class App {
9
10  public static void main(String[] args) {
11    // TODO Auto-generated method stub
12    String connectionString = "jdbc:mysql://localhost:3306/week10_Vehicles";
13    String SELECT_QUERY = "select * from vehicles where car_id = ?";
14    String INSERT_QUERY = "insert into vehicles (make_name,model_name,car_year,price,color,vin) values (?, ?, ?, ?, ?, ?)";
15    String DELETE_QUERY = "delete from vehicles where car_id = ?";
16
17    Scanner scanner = new Scanner(System.in);
18    try {
19      Connection conn = DriverManager.getConnection(connectionString, "root", "PromineoTech2022");
20      System.out.println("Connection successfully");
21      System.out.println("1 - Select");
22      System.out.println("2 - Insert");
23      System.out.println("3 - Update");
24      System.out.println("4 - Delete");
25      System.out.println("Please choose one option");
26      int option = scanner.nextInt();
27      switch (option) {
28        case 1:
29          System.out.println("Please enter the vehicle id");
30          int carId = scanner.nextInt();
31          PreparedStatement ps = conn.prepareStatement(SELECT_QUERY);
32          ps.setInt(1, carId);
33          ResultSet rs = ps.executeQuery();
34
35          while(rs.next()) {
36            System.out.println("car_id: " + rs.getInt(1) + " Make_name: " + rs.getString(2) + " Model_name: " + rs.getString(3) + " car_year: " + rs.getDa
37          }
38          break;
39        case 2:
40          System.out.println("Please Insert vehicle Info ");
41          System.out.print("make name");
42
43      }
44    } catch (SQLException e) {
45      e.printStackTrace();
46    }
47  }
48 }
```

```
eclipse-workspace - Week 10 Connecting to MySQL with Java/src/App.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer
  Classic card game WAR
  Intro to Java Week 3 Coding As
  Intro to Java Week 5 Coding As
  Week 1 [Week 1 main]
  Week 10
  Week 10 Connecting to MySQL
    src
      (default package)
        App.java
      JRE System Library [JavaSE-1]
      Referenced Libraries
  Week2
  Week3 - Labs
  Week4 - Labs

App.java
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83

case 2 :
    System.out.println("Please Insert vehicle Info ");
    System.out.print("make name");
    String name = scanner.next();
    System.out.print("model name");
    String model = scanner.next();
    System.out.print("car year");
    String car_year = scanner.next();
    System.out.print("price");
    String price = scanner.next();
    System.out.print("color");
    String color = scanner.next();
    System.out.print("vin");
    String vin = scanner.next();

    PreparedStatement ps2 = conn.prepareStatement(INSERT_QUERY);
    ps2.setString(1, name);
    ps2.setString(2, model);
    ps2.setString(3, car_year);
    ps2.setString(4, price);
    ps2.setString(5, color);
    ps2.setString(6, vin);
    int rs2 = ps2.executeUpdate();
    System.out.println("vehicle inserted");

    break;

case 3 :
    System.out.println("Please choose the vehicle id");
    int veh_id = scanner.nextInt();
    PreparedStatement ps0 = conn.prepareStatement(SELECT_QUERY);
    ps0.setLong(1, veh_id);
    ResultSet rs0 = ps0.executeQuery();
    if (rs0.next()) {
        System.out.println("No Vehicle exist");
    }
    else {
        System.out.println("Please choose the vehicle column you want to update");
        System.out.println("1 - make name");
        System.out.println("2 - model name");
        System.out.println("3 - year");
        System.out.println("4 - price");
        System.out.println("5 - car color");
    }
}
```

```
eclipse-workspace - Week 10 Connecting to MySQL with Java/src/App.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer
  Classic card game WAR
  Intro to Java Week 3 Coding As
  Intro to Java Week 5 Coding As
  Week 1 [Week 1 main]
  Week 10
  Week 10 Connecting to MySQL
    src
      (default package)
        App.java
      JRE System Library [JavaSE-1]
      Referenced Libraries
  Week2
  Week3 - Labs
  Week4 - Labs

App.java
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122

System.out.println("2 - model name");
System.out.println("3 - year");
System.out.println("4 - price");
System.out.println("5 - car color");
System.out.println("6 - VIN");
System.out.println("Please choose one option");
int option2 = scanner.nextInt();

if (option2 == 1) {
    String UPDATE_QUERY = "update vehicles set make_name = ? where car_id = ? ";
    PreparedStatement ps3 = conn.prepareStatement(UPDATE_QUERY);
    System.out.println("Please enter the new make name: ");
    String makeName = scanner.next();

    ps3.setString(1, makeName);
    ps3.setLong(2, veh_id);
    int rs3 = ps3.executeUpdate();
    System.out.println("vehicle updated");
}
else if (option2 == 2) {
    String UPDATE_QUERY = "update vehicles set model_name = ? where car_id = ? ";
    PreparedStatement ps3 = conn.prepareStatement(UPDATE_QUERY);
    System.out.println("Please enter the new model name: ");
    String modelName = scanner.next();

    ps3.setString(1, modelName);
    ps3.setLong(2, veh_id);
    int rs3 = ps3.executeUpdate();
    System.out.println("vehicle updated");
}
else if (option2 == 3) {
    String UPDATE_QUERY = "update vehicles set car_year = ? where car_id = ? ";
    PreparedStatement ps3 = conn.prepareStatement(UPDATE_QUERY);
    System.out.println("Please enter the new car year: ");
    String carYear = scanner.next();

    ps3.setString(1, carYear);
    ps3.setLong(2, veh_id);
    int rs3 = ps3.executeUpdate();
    System.out.println("vehicle updated");
}
else if (option2 == 4) {
}
```

The screenshot shows the Eclipse IDE with the file 'App.java' open. The code is part of a program that connects to a MySQL database and allows users to update vehicle information. The code includes a switch statement with cases for updating price, color, and VIN. It uses PreparedStatement and Scanner classes. The Package Explorer on the left shows the project structure, including 'src' and 'App.java'. The Outline view on the right shows the 'mainString()' method.

```
118 ps3.setLong(2, veh_id);
119 int rs3 = ps3.executeUpdate();
120 System.out.println("vehicle updated ");
121 }
122 else if (option2 == 4) {
123 String UPDATE_QUERY = "update vehicles set price = ? where car_id = ? ";
124 PreparedStatement ps3 = conn.prepareStatement(UPDATE_QUERY);
125 System.out.println("Please enter the new car price: ");
126 String carPrice = scanner.next();
127 ps3.setString(1, carPrice);
128 ps3.setLong(2, veh_id);
129 int rs3 = ps3.executeUpdate();
130 System.out.println("vehicle updated ");
131 }
132 else if (option2 == 5) {
133 String UPDATE_QUERY = "update vehicles set color = ? where car_id = ? ";
134 PreparedStatement ps3 = conn.prepareStatement(UPDATE_QUERY);
135 System.out.println("Please enter the new car color: ");
136 String carColor = scanner.next();
137 ps3.setString(1, carColor);
138 ps3.setLong(2, veh_id);
139 int rs3 = ps3.executeUpdate();
140 System.out.println("vehicle updated ");
141 }
142 else if (option2 == 6) {
143 String UPDATE_QUERY = "update vehicles set vin = ? where car_id = ? ";
144 PreparedStatement ps3 = conn.prepareStatement(UPDATE_QUERY);
145 System.out.println("Please enter the new car VIN: ");
146 String carVin = scanner.next();
147 ps3.setString(1, carVin);
148 ps3.setLong(2, veh_id);
149 int rs3 = ps3.executeUpdate();
150 System.out.println("vehicle updated ");
151 }
152 else {
153 System.out.println("Please enter a valide choice ");
154 }
155 break;
156 case 4 :
157 }
158 }
159 }
160 }
```

The screenshot shows the Eclipse IDE with the file 'App.java' open. The code continues from the previous screenshot, adding a case for deleting a vehicle. It uses PreparedStatement and Scanner classes. The Package Explorer on the left shows the project structure, including 'src' and 'App.java'. The Outline view on the right shows the 'mainString()' method.

```
143 }
144 else if (option2 == 6) {
145 String UPDATE_QUERY = "update vehicles set vin = ? where car_id = ? ";
146 PreparedStatement ps3 = conn.prepareStatement(UPDATE_QUERY);
147 System.out.println("Please enter the new car VIN: ");
148 String carVin = scanner.next();
149 ps3.setString(1, carVin);
150 ps3.setLong(2, veh_id);
151 int rs3 = ps3.executeUpdate();
152 System.out.println("vehicle updated ");
153 }
154 else {
155 System.out.println("Please enter a valide choice ");
156 }
157 break;
158 case 4 :
159 System.out.println("Please enter the vehicle id you want to delete");
160 int id = scanner.nextInt();
161 PreparedStatement ps5 = conn.prepareStatement(SELECT_QUERY);
162 ps5.setLong(1, id);
163 ResultSet rs5 = ps5.executeQuery();
164 if (!rs5.next()) {
165 System.out.println("No Vehicle exist");
166 }
167 else {
168 PreparedStatement ps4 = conn.prepareStatement(DELETE_QUERY);
169 ps4.setLong(1, id);
170 int rs4 = ps4.executeUpdate();
171 System.out.println("vehicle deleted ");
172 }
173 break;
174 default :
175 System.out.println("Please enter one option between 1 to 4");
176 }
177 }
178 }
179 }
180 }
181 } catch (SQLException e) {
182 // TODO Auto-generated catch block
183 e.printStackTrace();
184 }
185 }
```

## Screenshots of Running Application:

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator: Week 10 Connecting to MySQL

MANAGEMENT

- Server Status
- Client Connecti
- Users and Privi
- Status and Syst
- Data Export
- Data Import/Re

INSTANCE

- Startup / Shuld
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Re
- Performance Sc

SQL Editor

```

1 create database week10_Vehicles ;
2 use week10_Vehicles ;
3 create table vehicles (
4   car_id int NOT NULL auto_increment Primary key ,
5   make_name varchar(100) NOT NULL ,
6   model_name varchar(100) NOT NULL ,
7   car_year year NOT NULL ,
8   price decimal(10,2) NOT NULL ,
9   color varchar(25) NOT NULL ,
10  vin varchar(25) NOT NULL
11 );
12 desc vehicles ;

```

Result Grid

Field	Type	Null	Key	Default	Extra
car_id	int	NO	PRI		auto_increment
make_name	varchar(100)	NO			
model_name	varchar(100)	NO			
car_year	year	NO			
price	decimal(10,2)	NO			
color	varchar(25)	NO			

Output

#	Time	Action	Message	Duration / Fetch
1	23:13:45	create database week10_Vehicles	1 row(s) affected	0.016 sec
2	23:14:01	use week10_Vehicles	0 row(s) affected	0.000 sec
3	23:27:11	create table vehicles ( car_id int NOT NULL auto_increment Primary key , make_name varchar(100) NOT NULL , model...	0 row(s) affected	0.110 sec
4	23:27:25	create database week10_Vehicles	Error Code: 1007. Can't create database 'week10_vehicles'; database exists	0.016 sec
5	23:27:41	use week10_Vehicles	0 row(s) affected	0.000 sec
6	23:27:52	desc vehicles	7 row(s) returned	0.000 sec

Object Info: No object selected

eclipse-workspace - Week 10 Connecting to MySQL with Java/src/App.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- src
  - App.java

App.java

```

35
36 while(rs.next()) {
37   System.out.println("car_id: " + rs.getInt(1) + " Make_name " + rs.getString(2) + " Model_name: " + rs.getString(3) + " car_year: " + rs.getDa
38 }
39
40 break ;
41 case 2 :
42   System.out.println("Please Insert vehicle Info ");

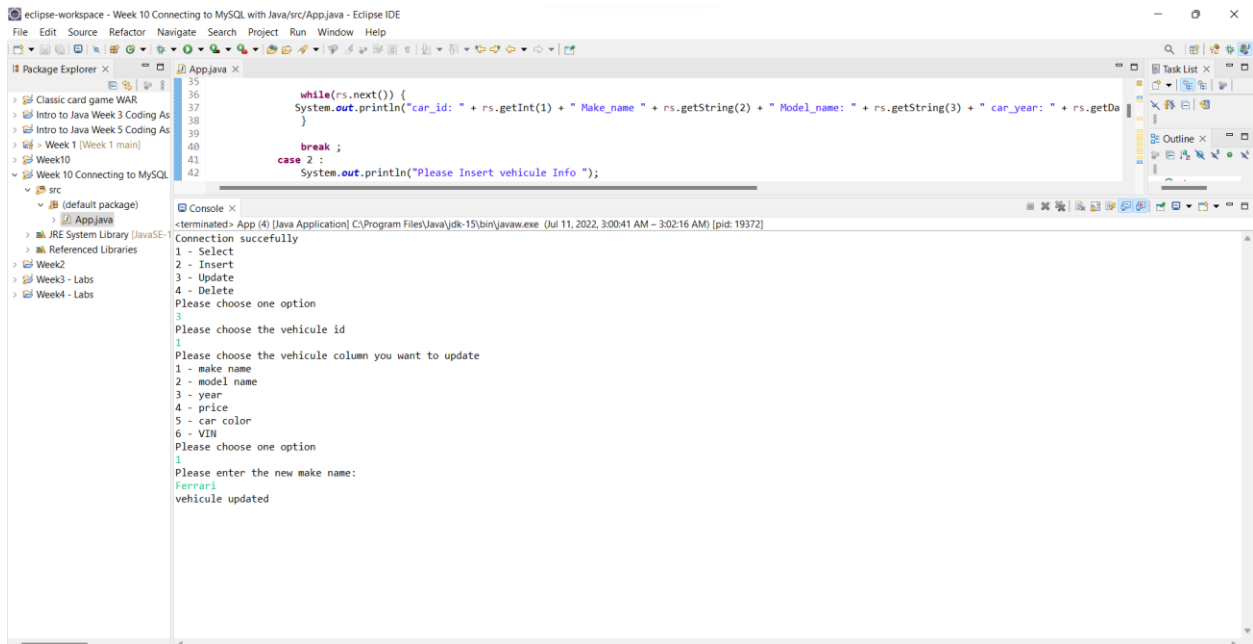
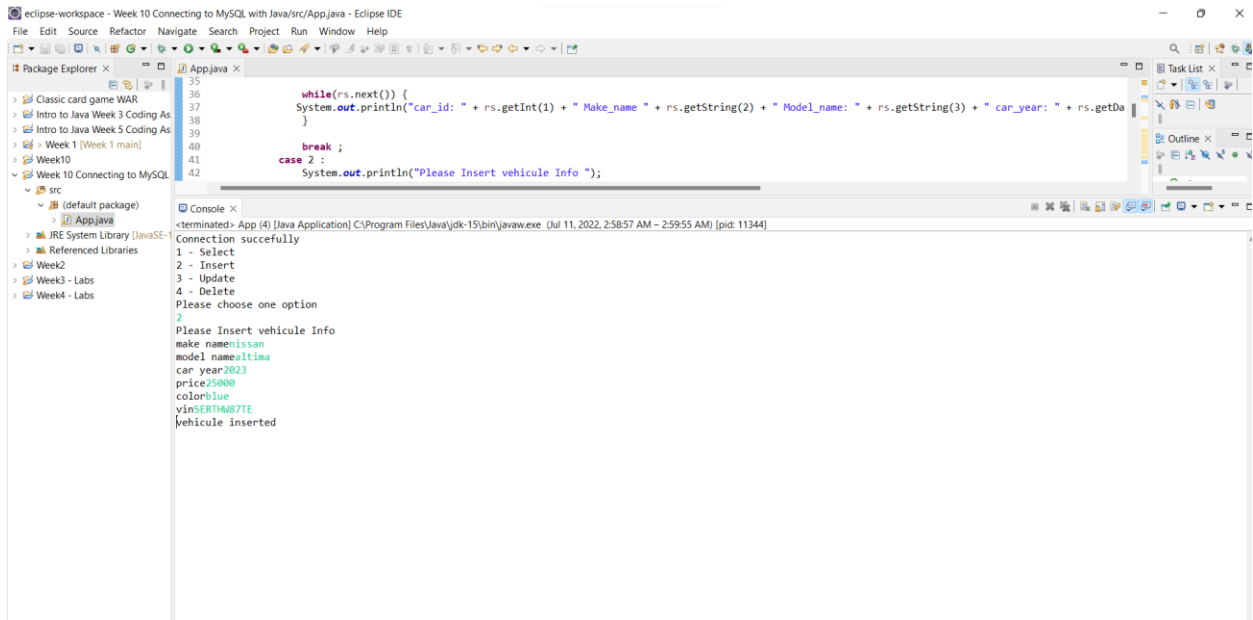
```

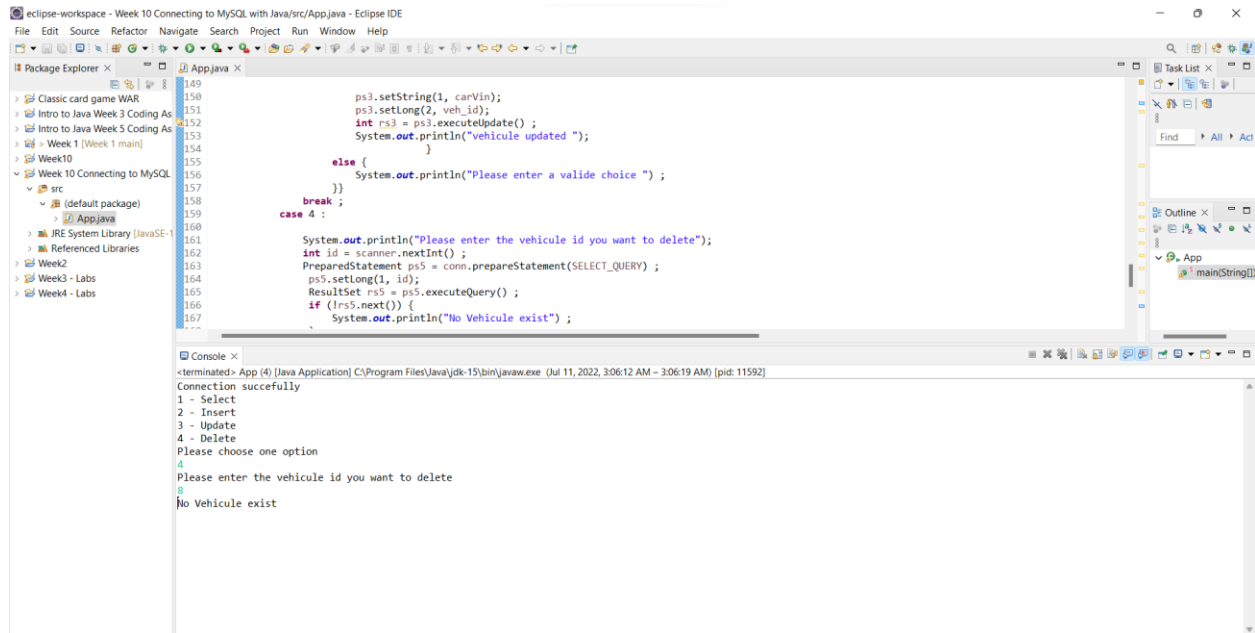
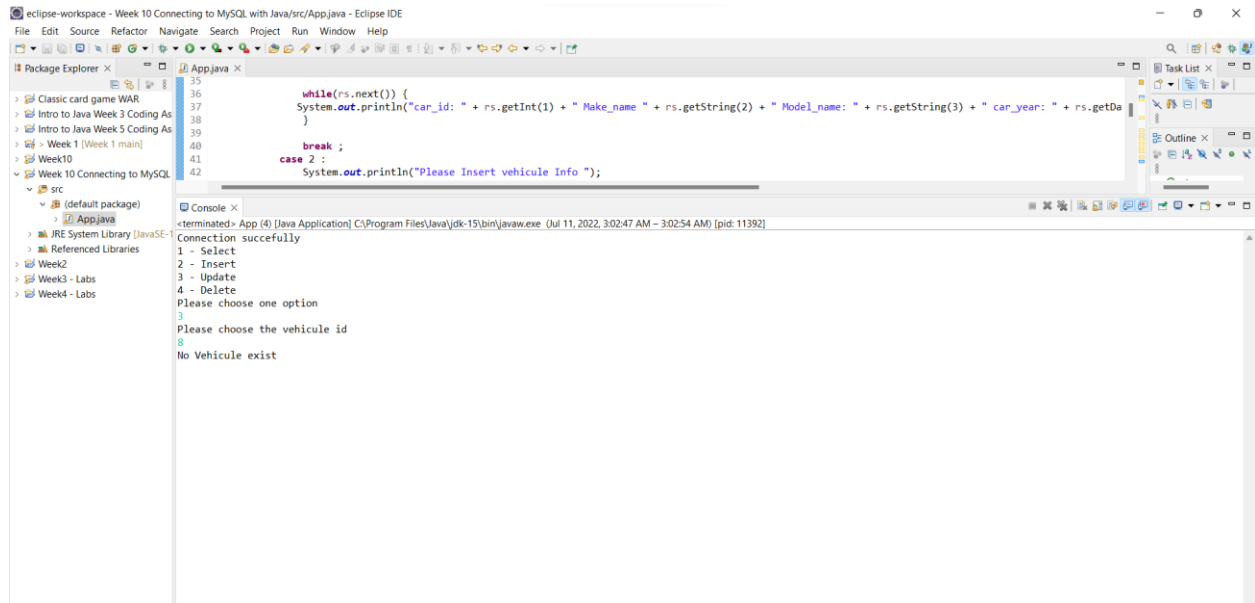
Console

```

<terminated> App (4) [Java Application] C:\Program Files\Java\jdk-15\bin\javaw.exe (Jul 11, 2022, 2:57:44 AM - 2:58:03 AM) [pid: 1664]
Connection successfully
1 - Select
2 - Insert
3 - Update
4 - Delete
Please choose one option
1
Please enter the vehicle id
1
car_id: 1 Make_name BMW Model_name: Sentra car_year: 2016-01-01 Price: 16000.25Colorbk VIN: CTRES254SY

```







```
149
150         ps3.setString(1, carVin);
151         ps3.setLong(2, veh_id);
152         int rs3 = ps3.executeUpdate();
153         System.out.println("vehicle updated ");
154     }
155     else {
156         System.out.println("Please enter a valide choice ");
157     }
158     break;
159 }
160 case 4 :
161
162     System.out.println("Please enter the vehicle id you want to delete");
163     int id = scanner.nextInt();
164     PreparedStatement ps5 = conn.prepareStatement(SELECT_QUERY);
165     ps5.setLong(1, id);
166     ResultSet rs5 = ps5.executeQuery();
167     if (!rs5.next()) {
168         System.out.println("No Vehicule exist");
169     }
170 }
```

Console Output:

```
<terminated> App (4) [Java Application] C:\Program Files\Java\jdk-15\bin\javaw.exe (Jul 11, 2022, 3:07:30 AM – 3:07:36 AM) [pid: 21952]
Connection successefully
1 - Select
2 - Insert
3 - Update
4 - Delete
Please choose one option
4
Please enter the vehicle id you want to delete
4
vehicle deleted
```

```
144         else if (option2 == 6) {
145             String UPDATE_QUERY = "update vehicles set vin = ? where car_id = ? ";
146             PreparedStatement ps3 = conn.prepareStatement(UPDATE_QUERY);
147             System.out.println("Please enter the new car VIN: ");
148             String carVin = scanner.next();
149
150             ps3.setString(1, carVin);
151             ps3.setLong(2, veh_id);
152             int rs3 = ps3.executeUpdate();
153             System.out.println("vehicle updated ");
154         }
155     }
156     else {
157         System.out.println("Please enter a valide choice ");
158     }
159     break;
160 }
161 case 4 :
162
163     System.out.println("Please enter the vehicle id you want to delete");
164     int id = scanner.nextInt();
165     PreparedStatement ps5 = conn.prepareStatement(SELECT_QUERY);
166     ps5.setLong(1, id);
167     ResultSet rs5 = ps5.executeQuery();
168     if (!rs5.next()) {
169         System.out.println("No Vehicule exist");
170     }
171 }
```

Console Output:

```
<terminated> App (4) [Java Application] C:\Program Files\Java\jdk-15\bin\javaw.exe (Jul 11, 2022, 3:14:12 AM – 3:14:27 AM) [pid: 5272]
Connection successefully
1 - Select
2 - Insert
3 - Update
4 - Delete
Please choose one option
5
Please enter one option between 1 to 4
```

**URL to GitHub Repository:**

**<https://github.com/cel90/Week-10-Connecting-to-MySQL-with-Java>**