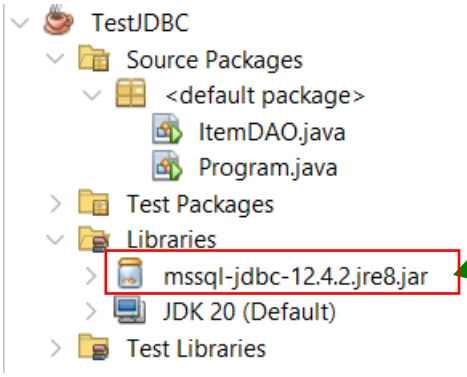
Building an Java Console Application & JDBC

Requirements

Create a java console application JDBC Driver to perform functions manage items (CRUD)

```
--------***--------
1.Print all items
2.Add new item
3.Update item
4.Remove item
Others: Exit
Enter choice:
1
001         Coffee              100
002         Milk                200
003         Cake                300
```

Step 01. Create a Java Console application named **TestJDBC** is structured as follows:



- On the project, right-click on Libraries | Add JAR/Folder then select **sqljdbc(sqljdbc42.jar)** driver that matches the Sql Server version on the machine.

-Open SQL Server Management Studio and create a database named **SampleDB** has a table named **Items** as follows:

T15PGen2.SampleD...pleDB - dbo.Items

Column Name	Data Type	Allow Nulls
ItemID	varchar(15)	☐
ItemName	varchar(50)	☐
Quantity	int	☐

T15PGen2.SampleDB - dbo.Items

ItemID	ItemName	Quantity
001	Coffee	100
002	Milk	200
003	Cake	300

Step 02. Write codes for the **ItemDAO.java** as the follows:

```
1  /* Sword Lake */
2  import java.sql.DriverManager;
3  import java.sql.Connection;
4  import java.sql.PreparedStatement;
5  import java.sql.ResultSet;
6
7  public class ItemDAO {
8      public Connection getConnection() throws Exception {
9          Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
10         String url = "jdbc:sqlserver://localhost:1433;database=SampleDB;",
11         Connection connection = DriverManager.getConnection(url, "sa", "1
12         return connection;
13     }//end Connection
14     //-------------------------------------------------------
15     public void printItems() throws Exception {
16         PreparedStatement ps = null;
17         Connection connection = null;
18         ResultSet rs = null;
19         try {
20             connection = getConnection();
21             String sql = "select ItemID,ItemName,Quantity from Items";
22             ps = connection.prepareStatement(sql);
23             rs = ps.executeQuery();
24             while (rs.next()) {
25                 //System.out.format("%-10d %-15s %5d %n",
26                 //      rs.getString(1),rs.getString(2),
27                 //      rs.getInt(3));
28                 System.out.format("%-10s %-15s %5d %n",
29                         rs.getString("ItemID"), rs.getString("ItemName"),
30                         rs.getInt("Quantity"));
31             }//end while
32         } catch (Exception ex) {
33             System.out.println(ex.getMessage());
34         } finally {
35             if (rs != null) {
36                 rs.close();
37             }
38             if (ps != null) {
39                 ps.close();
40             }
41             if (connection != null) {
42                 connection.close();
43             }
44         }
45     }//end printItems
```

```
46      //-------------------------------------------------------------
47      public void AddNewItem(String itemID, String itemName, int quantity)
48              throws Exception {
49          PreparedStatement ps = null;
50          Connection connection = null;
51          try {
52              connection = getConnection();
53              String sql = "insert Items(ItemID,ItemName,Quantity) values (?,?,?)";
54              ps = connection.prepareStatement(sql);
55              ps.setString(1, itemID);
56              ps.setString(2, itemName);
57              ps.setInt(3, quantity);
58              ps.executeUpdate();
59              System.out.println("Item has been added.");
60          } catch (Exception ex) {
61              System.out.println(ex.getMessage());
62          } finally {
63              if (ps != null) {
64                  ps.close();
65              }
66              if (connection != null) {
67                  connection.close();
68              }
69          }
70      }//end AddNewItem
71      //-------------------------------------------------------------
72      public void RemoveItem(String itemID)
73              throws Exception {
74          PreparedStatement ps = null;
75          Connection connection = null;
76          try {
77              connection = getConnection();
78              String sql = "delete Items where ItemID = ?";
79              ps = connection.prepareStatement(sql);
80              ps.setString(1, itemID);
81              ps.executeUpdate();
82              System.out.println("Item has been removed.");
83          } catch (Exception ex) {
84              System.out.println(ex.getMessage());
85          } finally {
86              if (ps != null) {
87                  ps.close();
88              }
89              if (connection != null) {
90                  connection.close();
91              }
92          }
93      }//end RemoveItem
```

```
94          //-----------------------------------------------------------
95          public void UpdateItem(String itemID, String itemName, int quantity)
96                  throws Exception {
97              PreparedStatement ps = null;
98              Connection connection = null;
99              try {
100                 connection = getConnection();
101                 String sql = "update Items set ItemName=?,Quantity =? where ItemID=?";
102                 ps = connection.prepareStatement(sql);
103                 ps.setString(1, itemName);
104                 ps.setInt(2, quantity);
105                 ps.setString(3, itemID);
106                 ps.executeUpdate();
107                 System.out.println("Item has been updated.");
108             } catch (Exception ex) {
109                 System.out.println(ex.getMessage());
110             } finally {
111                 if (ps != null) {
112                     ps.close();
113                 }
114                 if (connection != null) {
115                     connection.close();
116                 }
117             }
118         }//end UpdateItem
119     }//end class
```

Step 03. Write codes for the **Program.java** as the follows:

```
1   import java.util.Scanner;
2   /**
3    * @author SwordLake
4    */
5   public class Program {
6       static void printMenu() {
7           System.out.println("--------***--------");
8           System.out.println("1.Print all items");
9           System.out.println("2.Add new item");
10          System.out.println("3.Update item");
11          System.out.println("4.Remove item");
12          System.out.println("Others: Exit");
13          System.out.println("Enter choice:");
14      }//end printMenu
15      public static void main(String[] args) {
16          try {
17              String itemID, itemName;
18              int quantity, choice;
19              ItemDAO itemDAO = new ItemDAO();
20              Scanner sc = new Scanner(System.in);
21              printMenu();
22              choice = Integer.parseInt(sc.nextLine());
```

```
23              while (choice >= 1 && choice <= 4) {
                    if (choice == 1) {
25                      itemDAO.printItems();
26                  } else if (choice == 2) {
27                      System.out.println("Enter ItemID:");
28                      itemID = sc.nextLine();
29                      System.out.println("Enter ItemName:");
30                      itemName = sc.nextLine();
31                      System.out.println("Enter Quantity:");
32                      quantity = Integer.parseInt(sc.nextLine());
33                      itemDAO.AddNewItem(itemID, itemName, quantity);
34                  } else if (choice == 3) {
35                      System.out.println("Enter ItemID:");
36                      itemID = sc.nextLine();
37                      System.out.println("Enter ItemName:");
38                      itemName = sc.nextLine();
39                      System.out.println("Enter Quantity:");
40                      quantity = Integer.parseInt(sc.nextLine());
41                      itemDAO.UpdateItem(itemID, itemName, quantity);
42                  } else if (choice == 4) {
43                      System.out.println("Enter ItemID:");
44                      itemID = sc.nextLine();
45                      itemDAO.RemoveItem(itemID);
46                  } else {
47                      System.exit(0);
48                  }
49                  printMenu();
50                  choice = Integer.parseInt(sc.nextLine());
51              }//end while
52          } catch (Exception ex) {
53              System.out.println(ex.getMessage());
54          }
55      }//end main
56  }//end Program
```

Step 04. Select **Program.java** , then right-click | select **Run File** to run the app and test all functions.