Relational Databases with MySQL Week 5 Coding 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....

Rice balls

with the possibility of extras added in the rice balls

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 PreparedStatment.executeQuery() is only for Reading data and .executeUpdate() is used for Creating, Updating, and Deleting data.

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

Screenshots of Code:

```java
package application;

public class Application {
public static void main(String[] args) {
        Menu menu = new Menu();
        menu.start(); //allows us to start our application
    }

}
```

```java
 1  package application;
 2
 3
 4⊕ import java.sql.SQLException;
13
14  public class Menu {
15
16      private RiceBallDao riceBallDao = new RiceBallDao();
17      private ExtraIngredientDao extraIngredientDao = new ExtraIngredientDao();
18      private Scanner scanner = new Scanner(System.in);
19⊖     private List<String> options = Arrays.asList(
20              "Display RiceBalls",
21              "Display RiceBall's Recipe",
22              "Create RiceBall",
23              "Delete RiceBall",
24              "Create RiceBall Recipe By Ingredient",
25              "Delete RiceBall Recipe By Ingredient",
26              "Update RiceBall Recipe By Ingredient",
27              "Update RiceBall"
28              );
29
30⊖     public void start() {
31          String selection = "";
32
33          do {
34              printMenu();
35              selection = scanner.nextLine();
36              try {
37              if (selection.equals("1")) {
38                  displayRiceBalls();
39              } else if (selection.equals("2")) {
40                  displayRiceBall();
41              } else if (selection.equals("3")) {
42                  createRiceBall();
43              } else if (selection.equals("4")) {
44                  deleteRiceBall();
45              } else if (selection.equals("5")) {
46                  createExtraIngredient();
47              } else if (selection.equals("6")) {
48                  deleteExtraIngredient();
49              } else if (selection.equals("7")) {
50                  updateExtraIngredient();
51              } else if (selection.equals("8")) {
52                  updateRiceBall();
53              }
54          } catch (SQLException e) {
55              e.printStackTrace();
56          }
57              System.out.println("Please press enter to continue");
58              scanner.nextLine();
59          } while (!selection.equals("-1"));
60      }
61
```

```java
60        }
61
62⊝       private void deleteExtraIngredient() throws SQLException {
63            System.out.print("Enter extraIngredient id to delete:");
64            int id = Integer.parseInt(scanner.nextLine());
65            extraIngredientDao.deleteExtraIngredientById(id);
66
67        }
68
69⊝       private void createExtraIngredient() throws SQLException {
70            displayRiceBalls();
71            System.out.print("Enter name of new ingredient:");
72            String ingredient = scanner.nextLine();
73            System.out.print("Enter riceBall id for the new ingredient:");
74            int riceBallId = Integer.parseInt(scanner.nextLine());
75            extraIngredientDao.createNewExtraIngredient(ingredient, riceBallId);
76        }
77
78⊝       private void updateExtraIngredient() throws SQLException {
79            displayRiceBalls();
80            System.out.println("Which riceball recipe would you like to update?");
81            int id = Integer.parseInt(scanner.nextLine());
82            System.out.println("Enter the new ingredient: ");
83            String ingredient = scanner.nextLine();
84            extraIngredientDao.updateExtraIngredientById(id, ingredient);
85            System.out.println("Recipe has been updated");
86        }
87
88⊝       private void deleteRiceBall() throws SQLException {
89            displayRiceBalls();
90            System.out.print("Enter riceBall id to delete");
91            int id = Integer.parseInt(scanner.nextLine()); // add id verification :)
92            riceBallDao.deleteRiceBallById(id);
93
94
95        }
96
97⊝       private void createRiceBall() throws SQLException {
98            System.out.print("Enter new riceBall name:");
99            String riceBallName = scanner.nextLine();
100           riceBallDao.createNewRiceBall(riceBallName);
101
102       }
103
104⊝      private void displayRiceBall() throws SQLException {
105           System.out.print("Enter riceBall id: ");
106           int id = Integer.parseInt(scanner.nextLine());
107           RiceBall riceBall = riceBallDao.getRiceBallbyId(id);
108           System.out.println(riceBall.getRiceBallId() + ": " + riceBall.getName());
109           for (ExtraIngredient extraIngredient : riceBall.getExtraIngredients()) {
110               System.out.println("\tExtraIngredientId: " + extraIngredient.getExtraIngredientId() + " Ingredient: "
111           + extraIngredient.getIngredient());
112           }
```

```java
 89         displayRiceBalls();
 90         System.out.print("Enter riceBall id to delete");
 91         int id = Integer.parseInt(scanner.nextLine()); // add id verification :)
 92         riceBallDao.deleteRiceBallById(id);
 93
 94
 95     }
 96
 97⊖    private void createRiceBall() throws SQLException {
 98         System.out.print("Enter new riceBall name:");
 99         String riceBallName = scanner.nextLine();
100         riceBallDao.createNewRiceBall(riceBallName);
101
102     }
103
104⊖    private void displayRiceBall() throws SQLException {
105         System.out.print("Enter riceBall id: ");
106         int id = Integer.parseInt(scanner.nextLine());
107         RiceBall riceBall = riceBallDao.getRiceBallbyId(id);
108         System.out.println(riceBall.getRiceBallId() + ": " + riceBall.getName());
109         for (ExtraIngredient extraIngredient : riceBall.getExtraIngredients()) {
110             System.out.println("\tExtraIngredientId: " + extraIngredient.getExtraIngredientId() + " Ingredient: "
111         + extraIngredient.getIngredient());
112         }
113
114     }
115
116⊖    private void displayRiceBalls() throws SQLException {
117         List<RiceBall> riceBalls = riceBallDao.getRiceBalls();
118         for (RiceBall riceBall :  riceBalls) {
119             System.out.println(riceBall.getRiceBallId() + ": " + riceBall.getName());
120         }
121
122     }
123
124⊖    private void updateRiceBall() throws SQLException {
125         displayRiceBalls();
126         System.out.println("Which riceball would you like to update?");
127         int id = Integer.parseInt(scanner.nextLine());
128         System.out.println("Enter the new riceball: ");
129         String name = scanner.nextLine();
130         riceBallDao.updateRiceBallById(id, name);
131         System.out.println("Recipe has been updated");
132     }
133
134⊖    private void printMenu() {
135         System.out.println("Select an Option: \n -----------------------------------");
136         for (int i = 0; i < options.size(); i++) {
137             System.out.println(i + 1 + ") " + options.get(i));
138         }
139
140     }
141
142 }
```

```java
1  package dao;
2
3  import java.sql.Connection;
6
7  public class DBConnection {
8
9      private final static String URL = "jdbc:mysql://localhost:3306/riceballs";
10     private final static String USERNAME = "root";
11     private final static String PASSWORD = "password";
12     private static Connection connection;
13     private static DBConnection instance;
14
15     private DBConnection(Connection connection) {
16         this.connection = connection;
17     }
18     public static Connection getConnection() {
19         if (instance == null) {
20             try {
21                 connection = DriverManager.getConnection(URL, USERNAME, PASSWORD);
22                 instance = new DBConnection(connection);
23                 System.out.println("Connection successful.");
24             } catch (SQLException e) {
25                 e.printStackTrace();
26             }
27         }
28         return DBConnection.connection;
29
30     }
31
32 }
```

```java
package dao;

import java.sql.Connection;

public class ExtraIngredientDao {
    private Connection connection;
    private final String GET_EXTRA_INGREDIENTS_BY_RICEBALL_ID_QUERY = "SELECT * FROM extraIngredients WHERE riceBall_id = ?";
    private final String DELETE_EXTRA_INGREDIENTS_BY_RICEBALL_ID_QUERY = "DELETE FROM extraIngredients WHERE riceBall_id = ?";
    private final String CREATE_NEW_EXTRA_INGREDIENT_QUERY = "INSERT INTO extraIngredients(ingredient, riceBall_id)"
            + " VALUES(?,?)";
    private final String DELETE_EXTRA_INGREDIENT_BY_ID_QUERY = "DELETE FROM extraIngredients WHERE id = ?";
    private final String UPDATE_EXTRA_INGREDIENT_BY_ID_QUERY = "UPDATE extraIngredients SET ingredient = ? WHERE id =?";

    public ExtraIngredientDao() {
        connection = DBConnection.getConnection();
    }

    public List<ExtraIngredient> getExtraIngredientsByRiceBallId(int riceBallId) throws SQLException {
        PreparedStatement ps = connection.prepareStatement(GET_EXTRA_INGREDIENTS_BY_RICEBALL_ID_QUERY);
        ps.setInt(1, riceBallId);
        ResultSet rs = ps.executeQuery();
        List<ExtraIngredient> extraIngredients = new ArrayList<ExtraIngredient>();

        while (rs.next()) {
            extraIngredients.add(new ExtraIngredient(rs.getInt(1), rs.getString(2)));
        }
        return extraIngredients;
    }

    public void createNewExtraIngredient(String ingredient, int riceBallId) throws SQLException {
        PreparedStatement ps = connection.prepareStatement(CREATE_NEW_EXTRA_INGREDIENT_QUERY);
        ps.setString(1, ingredient);
        ps.setInt(2, riceBallId);
        ps.executeUpdate();
    }

    public void deleteExtraIngredientsByRiceBallId(int riceBallId) throws SQLException {
        PreparedStatement ps = connection.prepareStatement(DELETE_EXTRA_INGREDIENTS_BY_RICEBALL_ID_QUERY);
        ps.setInt(1, riceBallId);
        ps.executeUpdate();
    }

    public void updateExtraIngredientById(int id, String ingredient) throws SQLException  {
        PreparedStatement ps = connection.prepareStatement(UPDATE_EXTRA_INGREDIENT_BY_ID_QUERY);
        ps.setInt(2, id);
        ps.setString(1, ingredient);
        ps.executeUpdate();
    }

    public void deleteExtraIngredientById(int id) throws SQLException {
        PreparedStatement ps = connection.prepareStatement(DELETE_EXTRA_INGREDIENT_BY_ID_QUERY);
        ps.setInt(1, id);
        ps.executeUpdate();
```

```java
 1  package dao;
 2
 3⊕ import java.sql.Connection;▯
11
12  public class RiceBallDao {
13      private Connection connection;
14      private ExtraIngredientDao extraIngredientDao;
15      private final String GET_RICEBALLS_QUERY = "SELECT * FROM riceBalls";
16      private final String GET_RICEBALL_BY_ID_QUERY = "SELECT * FROM riceBalls WHERE id = ?";
17      private final String CREATE_NEW_RICEBALL_QUERY = "INSERT INTO riceBalls(name) VALUES (?)";
18      private final String DELETE_RICEBALL_BY_ID_QUERY = "DELETE FROM riceBalls WHERE id =?";
19      private final String UPDATE_RICEBALL_BY_ID_QUERY = "UPDATE riceBalls SET name = ? WHERE id=?";
20⊖     public RiceBallDao() {
21          connection = DBConnection.getConnection();
22          extraIngredientDao = new ExtraIngredientDao();
23      }
24
25⊖     public List<RiceBall> getRiceBalls() throws SQLException {
26          ResultSet rs= connection.prepareStatement(GET_RICEBALLS_QUERY).executeQuery();
27          List<RiceBall> riceBalls = new ArrayList<RiceBall>();
28
29          while (rs.next()) {
30              riceBalls.add(populateRiceBall(rs.getInt(1), rs.getString(2)));
31          }
32          return riceBalls;
33
34      }
35
36⊖     public RiceBall getRiceBallbyId(int id) throws SQLException {
37          PreparedStatement ps = connection.prepareStatement(GET_RICEBALL_BY_ID_QUERY);
38          ps.setInt(1, id);
39          ResultSet rs = ps.executeQuery();
40          rs.next();
41          return populateRiceBall(rs.getInt(1), rs.getString(2));
42      }
43
44⊖     public void createNewRiceBall(String riceBallName) throws SQLException {
45          PreparedStatement ps = connection.prepareStatement(CREATE_NEW_RICEBALL_QUERY);
46          ps.setString(1, riceBallName);
47          ps.executeUpdate();
48      }
49
50⊖     public void deleteRiceBallById(int id) throws SQLException {
51          extraIngredientDao.deleteExtraIngredientsByRiceBallId(id);
52          PreparedStatement ps = connection.prepareStatement(DELETE_RICEBALL_BY_ID_QUERY);
53          ps.setInt(1, id);
54          ps.executeUpdate();
55      }
56
57⊖     public void updateRiceBallById(int id, String name) throws SQLException {
58          PreparedStatement ps = connection.prepareStatement(UPDATE_RICEBALL_BY_ID_QUERY);
59          ps.setInt(2, id);
60          ps.setString(1, name);
```
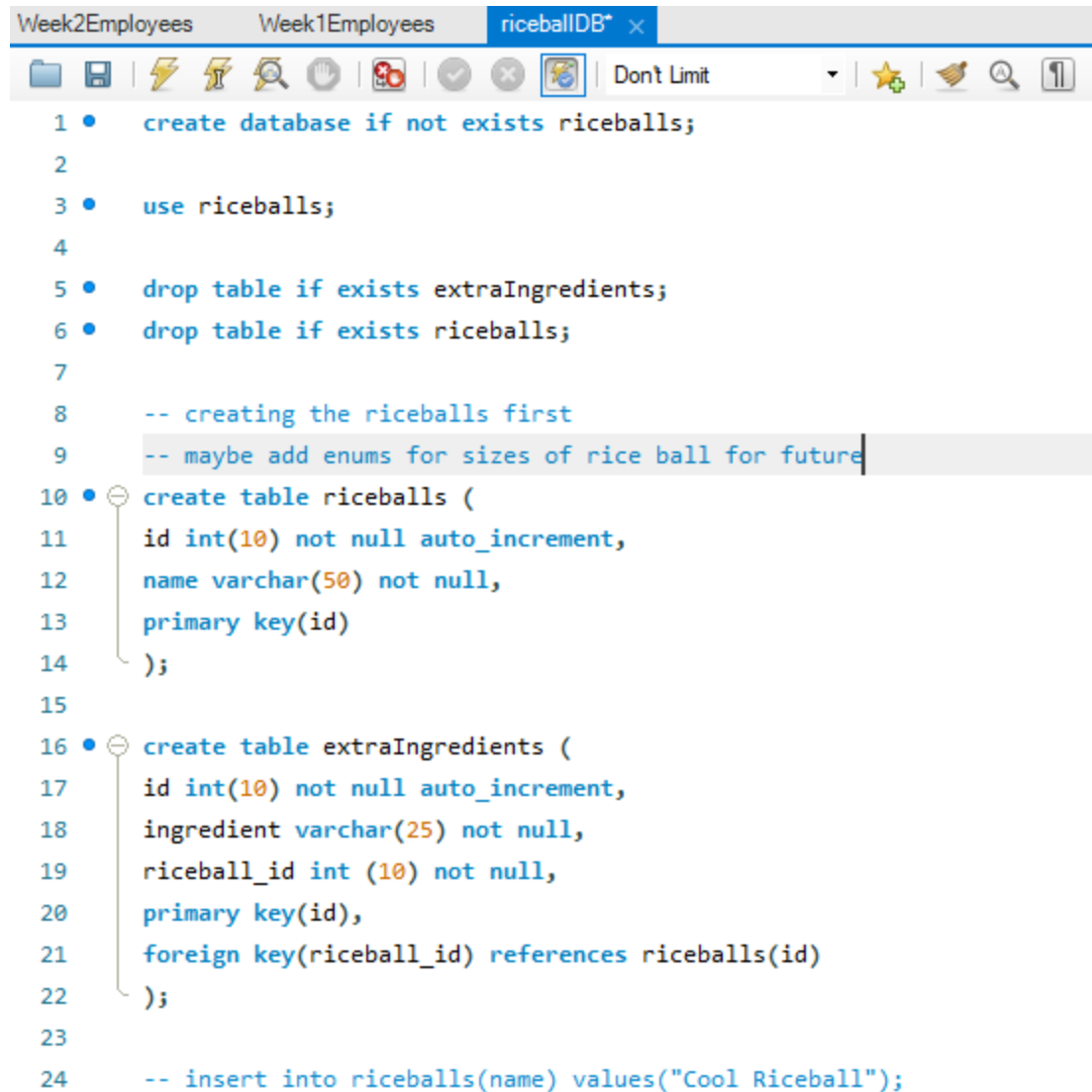
```java
    private final String GET_RICEBALL_BY_ID_QUERY = "SELECT * FROM riceBalls WHERE id = ?";
    private final String CREATE_NEW_RICEBALL_QUERY = "INSERT INTO riceBalls(name) VALUES (?)";
    private final String DELETE_RICEBALL_BY_ID_QUERY = "DELETE FROM riceBalls WHERE id =?";
    private final String UPDATE_RICEBALL_BY_ID_QUERY = "UPDATE riceBalls SET name = ? WHERE id=?";
    public RiceBallDao() {
        connection = DBConnection.getConnection();
        extraIngredientDao = new ExtraIngredientDao();
    }

    public List<RiceBall> getRiceBalls() throws SQLException {
        ResultSet rs= connection.prepareStatement(GET_RICEBALLS_QUERY).executeQuery();
        List<RiceBall> riceBalls = new ArrayList<RiceBall>();

        while (rs.next()) {
            riceBalls.add(populateRiceBall(rs.getInt(1), rs.getString(2)));
        }
        return riceBalls;

    }

    public RiceBall getRiceBallbyId(int id) throws SQLException {
        PreparedStatement ps = connection.prepareStatement(GET_RICEBALL_BY_ID_QUERY);
        ps.setInt(1, id);
        ResultSet rs = ps.executeQuery();
        rs.next();
        return populateRiceBall(rs.getInt(1), rs.getString(2));
    }

    public void createNewRiceBall(String riceBallName) throws SQLException {
        PreparedStatement ps = connection.prepareStatement(CREATE_NEW_RICEBALL_QUERY);
        ps.setString(1, riceBallName);
        ps.executeUpdate();
    }

    public void deleteRiceBallById(int id) throws SQLException {
        extraIngredientDao.deleteExtraIngredientsByRiceBallId(id);
        PreparedStatement ps = connection.prepareStatement(DELETE_RICEBALL_BY_ID_QUERY);
        ps.setInt(1, id);
        ps.executeUpdate();
    }

    public void updateRiceBallById(int id, String name) throws SQLException {
        PreparedStatement ps = connection.prepareStatement(UPDATE_RICEBALL_BY_ID_QUERY);
        ps.setInt(2, id);
        ps.setString(1, name);
        ps.executeUpdate();
    }

    private RiceBall populateRiceBall(int id, String name) throws SQLException {
        return new RiceBall(id, name, extraIngredientDao.getExtraIngredientsByRiceBallId(id));
    }
}
```

```java
1   package entity;
2
3   public class ExtraIngredient {
4       private int extraIngredientId;
5       private String ingredient;
6
7       public ExtraIngredient(int extraIngredientId, String ingredient) {
8           this.setExtraIngredientId(extraIngredientId);
9           this.setIngredient(ingredient);
10      }
11
12      public int getExtraIngredientId() {
13          return extraIngredientId;
14      }
15
16      public void setExtraIngredientId(int extraIngredientId) {
17          this.extraIngredientId = extraIngredientId;
18      }
19
20      public String getIngredient() {
21          return ingredient;
22      }
23
24      public void setIngredient(String ingredient) {
25          this.ingredient = ingredient;
26      }
27  }
28
```

```java
1  package entity;
2
3  import java.util.List;
4
5  public class RiceBall {
6      private int riceBallId;
7      private String name;
8      private List<ExtraIngredient> extraIngredients;
9
10     public RiceBall(int riceBallId, String name, List<ExtraIngredient> extraIngredients) {
11         this.setRiceBallId(riceBallId);
12         this.setName(name);
13         this.setExtraIngredients(extraIngredients);
14     }
15
16     public int getRiceBallId() {
17         return riceBallId;
18     }
19
20     public void setRiceBallId(int riceBallId) {
21         this.riceBallId = riceBallId;
22     }
23
24     public String getName() {
25         return name;
26     }
27
28     public void setName(String name) {
29         this.name = name;
30     }
31
32     public List<ExtraIngredient> getExtraIngredients() {
33         return extraIngredients;
34     }
35
36     public void setExtraIngredients(List<ExtraIngredient> extraIngredients) {
37         this.extraIngredients = extraIngredients;
38     }
39
40 }
41
42
```

```
Week2Employees        Week1Employees        riceballDB*  ×

                                        Don't Limit          ▾  ⭐  🧹  🔍  📑

  1 ●      create database if not exists riceballs;

  2

  3 ●      use riceballs;

  4

  5 ●      drop table if exists extraIngredients;
  6 ●      drop table if exists riceballs;

  7

  8        -- creating the riceballs first
  9        -- maybe add enums for sizes of rice ball for future
 10 ● ⊖  create table riceballs (
 11        id int(10) not null auto_increment,
 12        name varchar(50) not null,
 13        primary key(id)
 14      );

 15

 16 ● ⊖  create table extraIngredients (
 17        id int(10) not null auto_increment,
 18        ingredient varchar(25) not null,
 19        riceball_id int (10) not null,
 20        primary key(id),
 21        foreign key(riceball_id) references riceballs(id)
 22      );

 23

 24        -- insert into riceballs(name) values("Cool Riceball");
```

Screenshots of Running Application:

```
Connection successful.
Select an Option:
 -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
1
1: plain
3: seaweed warpped riceball
Please press enter to continue

Select an Option:
 -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
8
1: plain
3: seaweed warpped riceball
Which riceball would you like to update?
1
Enter the new riceball:
jasmine short-grain rice rice ball
Recipe has been updated
Please press enter to continue

Select an Option:
 -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
1
1: jasmine short-grain rice rice ball
3: seaweed warpped riceball
Please press enter to continue
```

```
Select an Option:
-------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
2
Enter riceBall id: 1
1: jasmine short-grain rice rice ball
        ExtraIngredientId: 1 Ingredient: supermayoo
Please press enter to continue
```

```
. _____ r____ _____ __ _____
Select an Option:
  -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
3
Enter new riceBall name:Edamame RiceBall
Please press enter to continue

Select an Option:
  -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
3
Enter new riceBall name:Katsuobushi RiceBall
Please press enter to continue
3
Select an Option:
  -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
3
Enter new riceBall name:Salmon RiceBall
Please press enter to continue
3
Select an Option:
  -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
```

```
3
Enter new riceBall name:Salmon RiceBall
Please press enter to continue
3
Select an Option:
 ------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
3
Enter new riceBall name:Shiokombu RiceBall
Please press enter to continue

Select an Option:
 ------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
3
Enter new riceBall name:Furikake RiceBall
Please press enter to continue

Select an Option:
 ------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
3
Enter new riceBall name:Nori RiceBall
Please press enter to continue
```

```
1: jasmine short-grain rice rice ball
3: seaweed warpped riceball
4: Edamame RiceBall
5: Katsuobushi RiceBall
6: Salmon RiceBall
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Please press enter to continue

Select an Option:
-----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
4
1: jasmine short-grain rice rice ball
3: seaweed warpped riceball
4: Edamame RiceBall
5: Katsuobushi RiceBall
6: Salmon RiceBall
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Enter riceBall id to delete3
Please press enter to continue

Select an Option:
-----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
1
1: jasmine short-grain rice rice ball
4: Edamame RiceBall
5: Katsuobushi RiceBall
6: Salmon RiceBall
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Please press enter to continue
```

```
Select an Option:
 -------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
4

1: jasmine short-grain rice rice ball
4: Edamame RiceBall
5: Katsuobushi RiceBall
6: Salmon RiceBall
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Enter riceBall id to delete1
Please press enter to continue

Select an Option:
 ----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
1

4: Edamame RiceBall
5: Katsuobushi RiceBall
6: Salmon RiceBall
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Please press enter to continue
```

```
Please press enter to continue

Select an Option:
------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
5
4: Edamame RiceBall
5: Katsuobushi RiceBall
6: Salmon RiceBall
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Enter name of new ingredient:Seaweed flakes
Enter riceBall id for the new ingredient:9
Please press enter to continue

Select an Option:
------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
2
Enter riceBall id: 9
9: Nori RiceBall
        ExtraIngredientId: 2 Ingredient: Seaweed flakes
Please press enter to continue
```

```
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Enter name of new ingredient:Seaweed flakes
Enter riceBall id for the new ingredient:9
Please press enter to continue

Select an Option:
 -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
2
Enter riceBall id: 9
9: Nori RiceBall
        ExtraIngredientId: 2 Ingredient: Seaweed fla
Please press enter to continue

Select an Option:
 -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
6
Enter extraIngredient id to delete:2
Please press enter to continue

Select an Option:
 -----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
2
Enter riceBall id: 9
9: Nori RiceBall
Please press enter to continue
```

```
Enter riceBall id: 9
9: Nori RiceBall
Please press enter to continue

Select an Option:
 ------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
5
4: Edamame RiceBall
5: Katsuobushi RiceBall
6: Salmon RiceBall
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Enter name of new ingredient:Sesame Seeds
Enter riceBall id for the new ingredient:6
Please press enter to continue

Select an Option:
 ------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
2
Enter riceBall id: 6
6: Salmon RiceBall
        ExtraIngredientId: 3 Ingredient: Sesame Seeds
Please press enter to continue

Select an Option:
 ------------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
```

```
Select an Option:
-----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
7
4: Edamame RiceBall
5: Katsuobushi RiceBall
6: Salmon RiceBall
7: Shiokombu RiceBall
8: Furikake RiceBall
9: Nori RiceBall
Which riceball recipe would you like to update?
3
Enter the new ingredient:
Salmon flakes
Recipe has been updated
Please press enter to continue

Select an Option:
-----------------------------------
1) Display RiceBalls
2) Display RiceBall's Recipe
3) Create RiceBall
4) Delete RiceBall
5) Create RiceBall Recipe By Ingredient
6) Delete RiceBall Recipe By Ingredient
7) Update RiceBall Recipe By Ingredient
8) Update RiceBall
2
Enter riceBall id: 6
6: Salmon RiceBall
        ExtraIngredientId: 3 Ingredient: Salmon flakes
Please press enter to continue
```

URL to GitHub Repository: