# 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....

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:**

**Creation sql in workbench**

Graphical user interface, text, application

Description automatically generated

**Top level application**

Graphical user interface, text, application, email

Description automatically generated

**Menu class**

Graphical user interface, text

Description automatically generated with medium confidence

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated with medium confidence

**Ship class**

Graphical user interface, text, application, email

Description automatically generated

**Ship DAO class**

Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

**DB connection class**

Graphical user interface, text, application, email

Description automatically generated

**Screenshots of Running Application:**

**Before any user input**

Text

Description automatically generated

**After user input to list ships**

Graphical user interface, text

Description automatically generated

**Creating a ship successfully**

Graphical user interface, text

Description automatically generated

**Modifying a ship successfully**

Text

Description automatically generated

**Deleting a ship successfully**

Text

Description automatically generated

**Showing results of the above successful actions**

Text

Description automatically generated

**User quitting with enter (added goodbye message at menu line 32 not screenshotted)**

Graphical user interface, text, application, email

Description automatically generated

**Bad menu choice (integer)**

Text

Description automatically generated

**On create, bad Double Speed: (could be better error message)**

Graphical user interface, text

Description automatically generated

**Delete bad due to id not existing**

Graphical user interface, text, application

Description automatically generated

**Bad update, id doesn’t exist (id check would be better before entering new fields)**

Text

Description automatically generated

**Extra paste:**

**use mysql\_week4\_coding\_assignment\_db;**

**drop table if exists spaceships;**

**create table spaceships (**

**ship\_id int auto\_increment,**

**name varchar(80) not null,**

**speed double,**

**primary key (ship\_id)**

**);**

**Github link: https://github.com/BradyBenner/mySQL\_week\_4\_coding\_assignment**