Project Documentation

1. OOP Elements:
   1. Classes:
      1. Jeopardy.java
      2. GameViewModel.java
      3. Mjdqw5Jeopardy.java
      4. DataBaseConnection.java

* 1. Sub-Classes
     1. GameViewController.java
     2. StartViewController.java
     3. AboutViewController.java
     4. RecordViewController.java
  2. Abstract-Class
     1. Switchable.java
     2. JeopardyAbstractModel.java
  3. Interface:
     1. TimelineInterface.java

1. Code Elements Utilized:
   1. Collection Cases Utilized:
      1. GameViewModel.java
         * ArrayList<> to hold QuestionsObj objects that have Questions data
      2. DataBaseConnection.java
         * ArrayList<> to hold elements of database table for viewing
   2. Exception Handling:
      1. DataBaseConnection.java
         * Try-catch <lines 42-46>
         * Try-catch <lines 101-115>
         * Try-catch <lines 125-136>
2. Clearly Defined Model
   1. Model Classes Utilized:
      1. GameViewModel.java
      2. StartViewModel.java
3. Multiple Scenes with Dynamic Scene Display:
   1. AboutView.fxml
      1. *Back* button goes back to home screen
      2. *Exit* button will exit the game
   2. GameView.fxml
      1. *Back* button goes back to home screen
      2. *Exit* button will exit the game
   3. StartView.fxml
      1. All buttons will switch to a different page
4. About Page:
   1. *About* button in “StartView” screen will take you to About page
5. Saving and Loading Data
   1. DataBaseConnections.java
      1. Setting up AWS and MariaDB to save and load data to a SQL
   2. GameViewController.java
      1. saveScoreAction method
         * Saving a query to a SQL database
      2. viewScoreBtn method
         * Gets the size of the table and the ID’s “UserID” which is the username, and “Score” which is the previous game score.