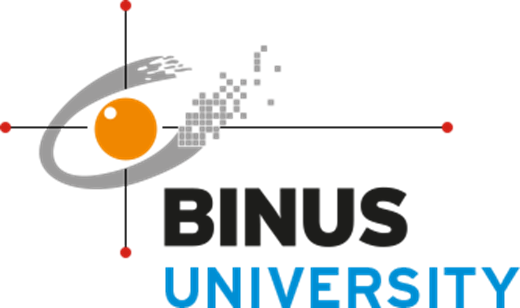
Quiz Game Application

OOP Final Project



Lecturer:

D4017\_Jude Joseph Lamug Martinez, MCS

Made by:

2702357876\_Samuel James Setiadi

Object Oriented Programming

Computer Science

Binus International University

**Table of Contents**

**1. Project Specifications……………………………………………………………………… 3**

**2. Solution Design…………………………………………………………………………….. 4**

**3. Evidence of Working Program……………………………………………………………. 9**

**4. Resources…………………………………………………………………………………… 18**

**Project Specification**

**Project Overview**

The goal of this project is to create an interactive quiz game using Java Swing components, specifically leveraging JFrame for the user interface. The quiz game will test users on various categories such as Science, Math, and History. Users will answer multiple-choice questions, and their scores will be tracked and displayed at the end of the quiz.

#### **Features**

* **User Interface:**
  + A main menu screen with options to start the quiz, view high scores, and exit the game.
  + A quiz screen displaying questions and multiple-choice answers.
  + A score screen showing the user's final score and a leaderboard of high scores.
  + A game-over screen with options to restart the quiz or return to the main menu.
* **Quiz Functionality:**
  + Three categories of questions: Science, Math, and History.
  + Randomized question order within each category.
  + Multiple-choice format with four options per question.
  + Timer for each question, adding a sense of urgency.
* **Scoring:**
  + Correct answers add to the user's score.
  + Incorrect answers do not affect the score.
  + Scores are stored and can be viewed in the view scores section.

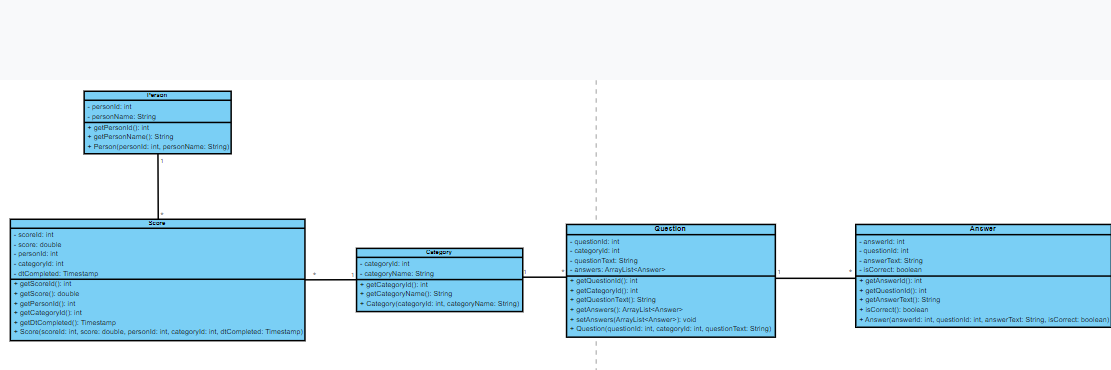
#### **Technical Requirements:**

* + **Programming Language:** Java
  + **Libraries:** Java Swing for GUI components, JDBC (Java Database Connectivity) Library
  + **Development Environment:** IntelliJ IDEA

**Solution Design**

**Class Diagrams**

**Core Domain Class Diagram**

****

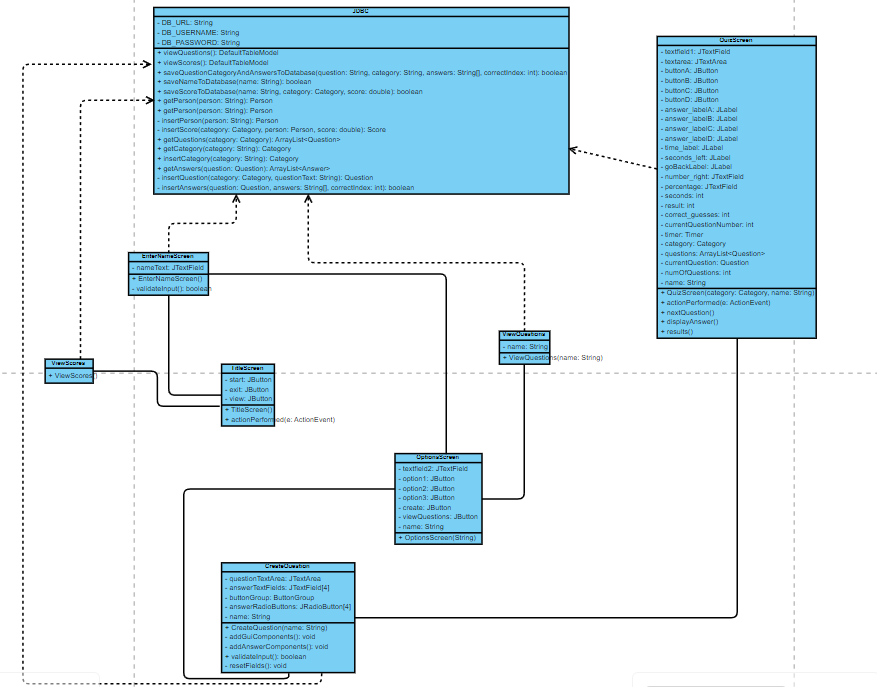
In my project, there are 5 core domain classes:

1. Person: A class representing a player of the quiz game
2. Score: A class representing a score of a player
3. Category: A class representing different categories of the quiz game
4. Question: A class representing a question in the quiz game
5. Answer: A class representing a possible answer of a question

These 5 classes have association relationships between each other:

1. Question and Answer: Question has a one to many relationship with answer, this means that one question can have many possible answer choices.
2. Category and Question: Category has a one to many relationship with question, this means that one category possibly has many questions.
3. Category and Score: Category has a one to many relationship with score, this means that one category can possibly contain multiple scores.
4. Person and Score: Person has a one to many relationship with score, this means that one person can possibly have many different scores.

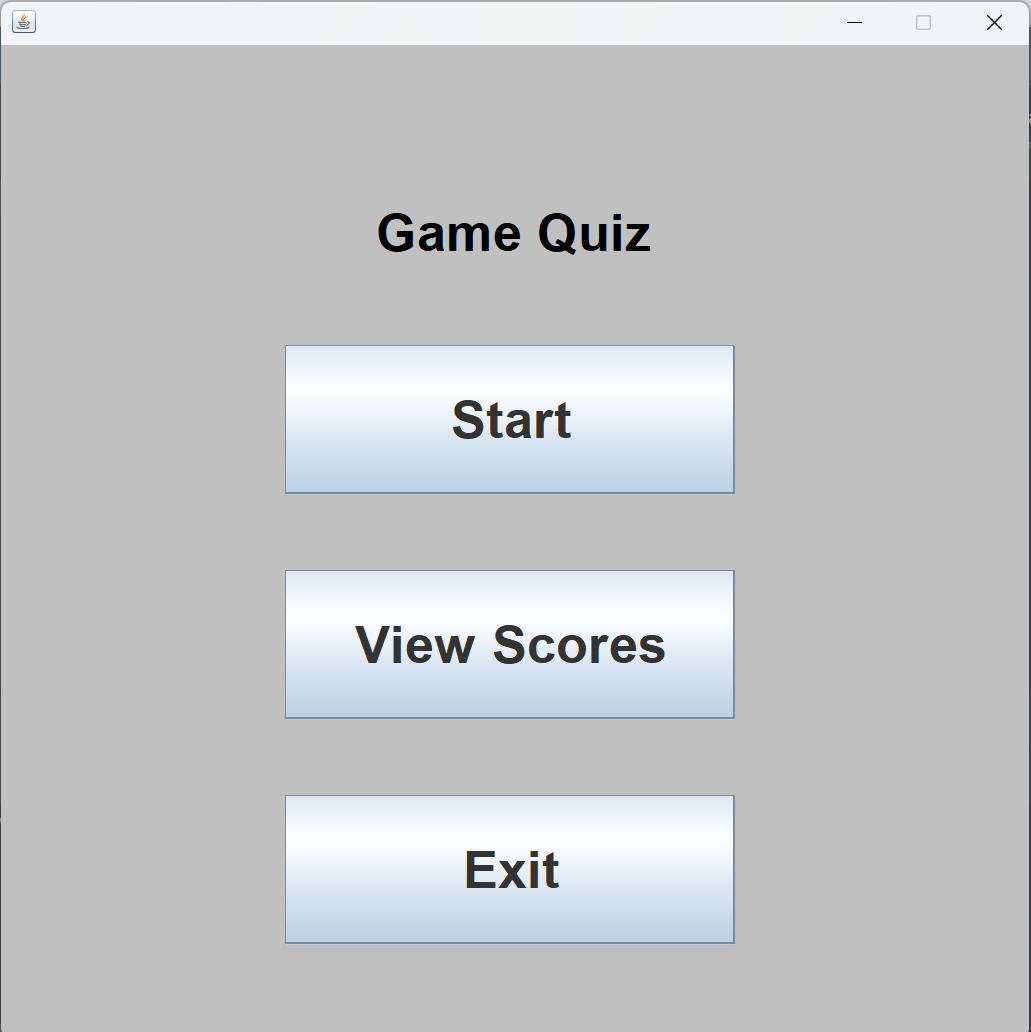
**User Interface Class Diagram**

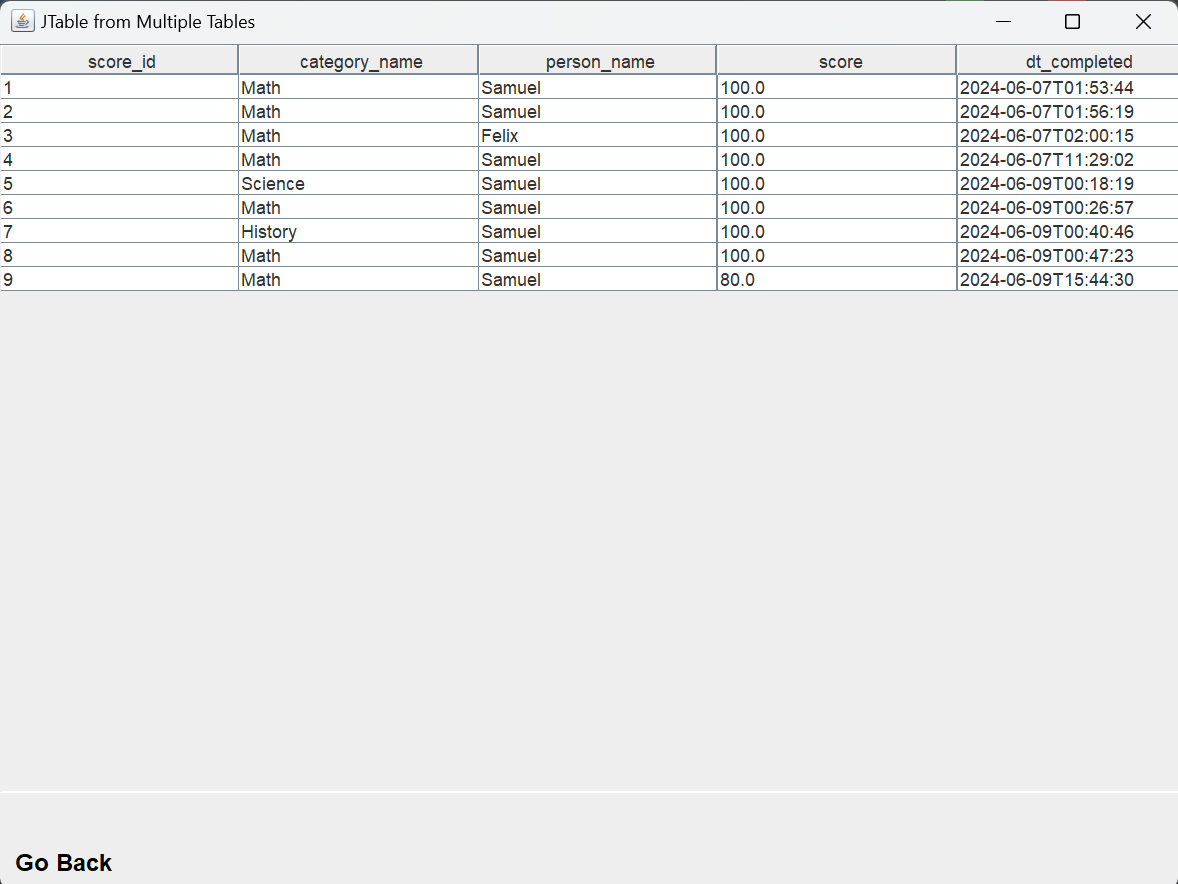
****

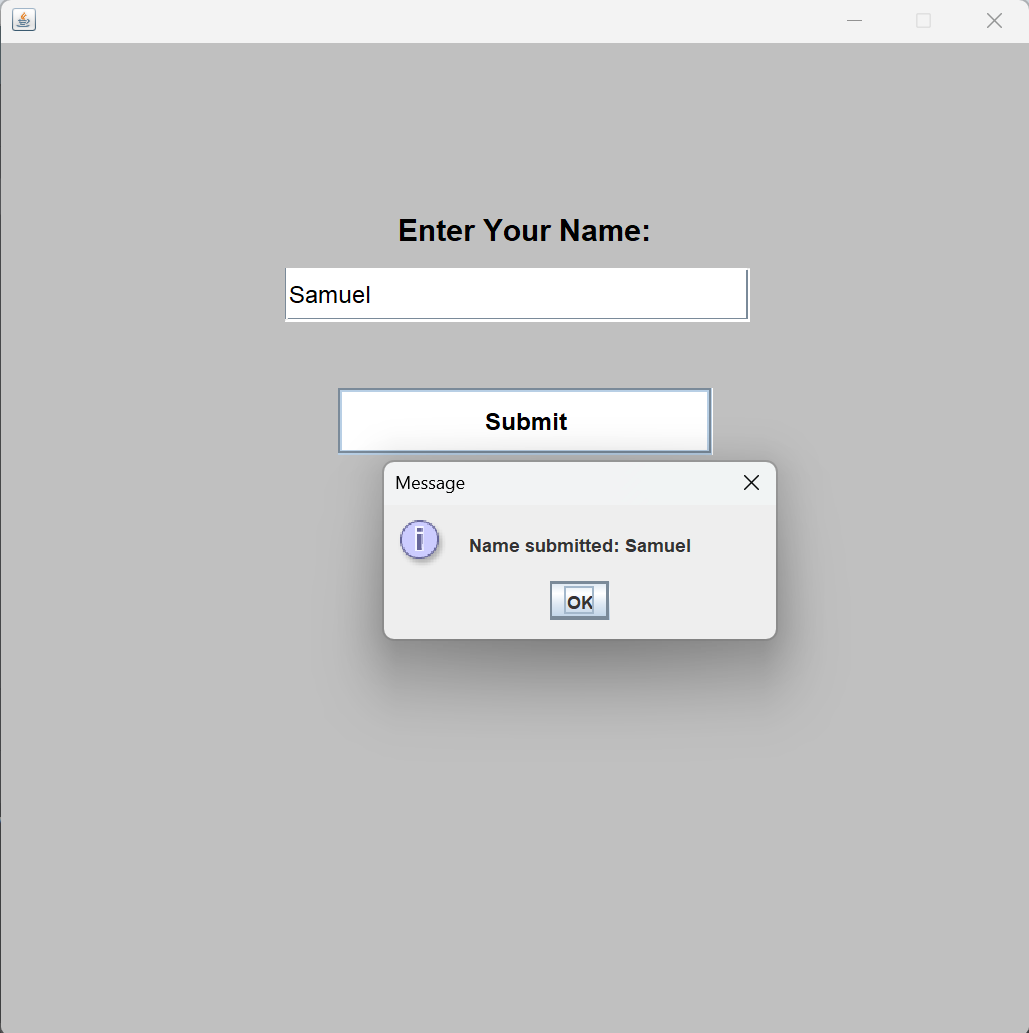
In my project, there are main 8 main classes that is related to the User Interface:

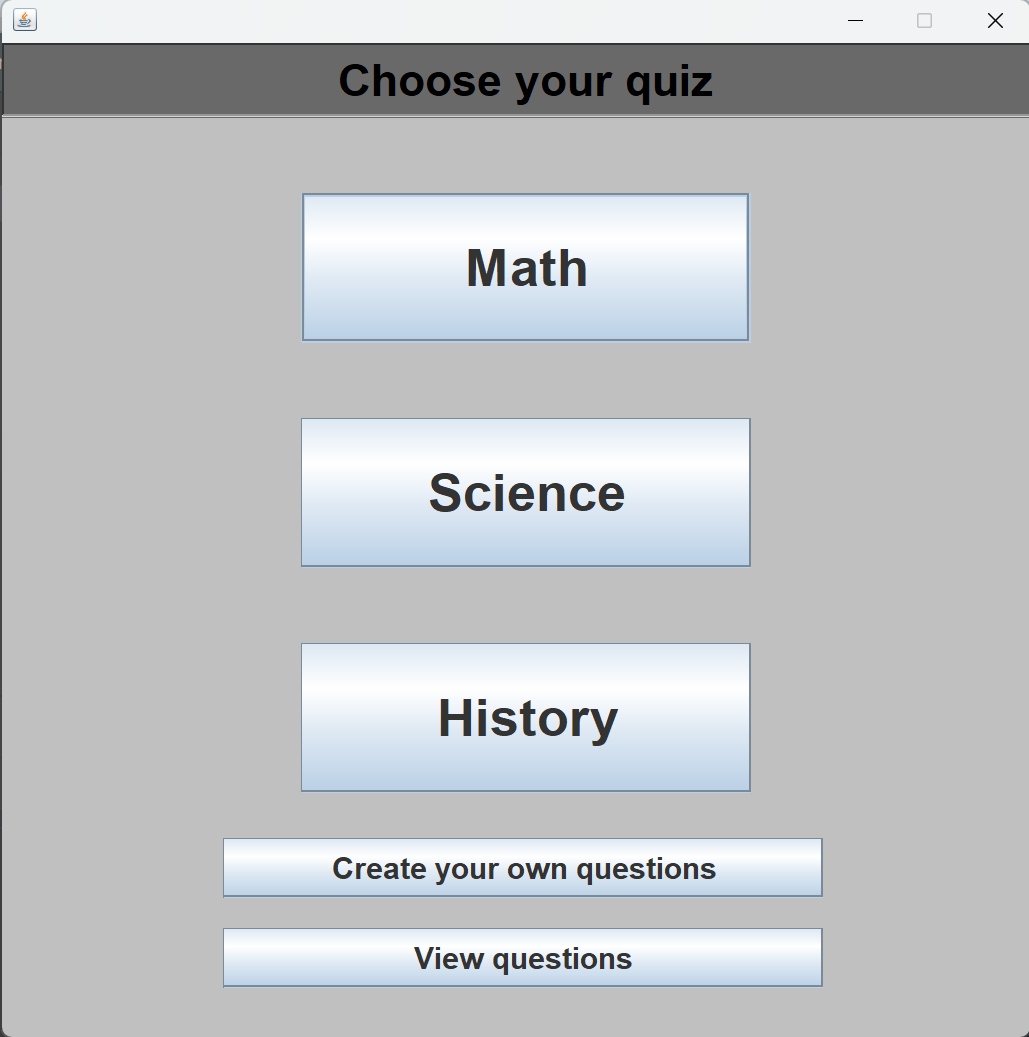
1. TitleScreen: This will be the first screen that will pop up when the application is first run, this page contains the title and 3 buttons, one for viewing the previous scores, one for exiting the application and one for moving on to the next page which is the EnterNameScreen. It has an association relationship with EnterNameScreen and ViewScore because an instance of EnterNameScreen when the start button is pressed and creates an instance of the ViewScore screen when the view scores button is pressed.
2. ViewScore: This screen is shown when the user pressed the view score button in the title screen. This class uses the static method viewScores from the JDBC class to get the data from the MySQL database related to the score of the player and uses JTable to showcase the data. A go back button is also created so that the user can go back to the TitleScreen. It has a dependency relationship with JDBC and association relationship with TitleScreen because it depends on the data retrieved from the database to showcase the data and when the go back button is pressed it creates an instance of the TitleScreen.
3. EnterNameScreen: This screen is where the player enters their name. The name will be stored into the database by using the static method called saveNameToDatabase. When the user presses the submit button a pop up will be shown to tell the users the name they submitted and then it will move on to the next page which is the OptionScreen class. This class has an association relationship with TitleScreen and OptionsScreen and dependency on JDBC.
4. OptionsScreen: This screen is where the player can choose the category of quiz they want to play. They can also create their own question and view all the questions. There are 5 buttons on this screen. One for choosing the science quiz, one for the math quiz, one for the history quiz, one to pop up the creating questions screen, and one to pop up the view questions screen. This class has an association relationship with EnterNameScreen, CreateQuestion, and ViewQuestions.
5. CreateQuestion: This screen is where the player can create their own questions and it will be saved into the mySQL database. There are text fields, radio buttons, and dropdowns that the player has to input. This is done by using the static method from the JDBC class saveQuestionCategoryAndAnswersToDatabase. Input is first validated before saving to the database. A pop up message is shown up when the submit button is pressed and then all the fields will be reset. A go back button is also created so that the user can go back to the OptionsScreen. This class has association relationships with QuizScreen and OptionsScreen and dependency with JDBC.
6. ViewQuestions: This screen will pop up when the view questions button is pressed in the OptionsScreen. This class uses the static method viewQuestions from the JDBC class to get the data from the MySQL database related to the score of the player and uses JTable to showcase the data. A go back button is also created so that the user can go back to the OptionsScreen. ViewQuestions has dependency on JDBC and association relationship with OptionsScreen.
7. QuizScreen: This screen is where the quiz will start. The player chooses the category first in the OptionsScreen to move to the QuizScreen. The number of questions will vary for each category depending on the number of questions that are stored in the database. The questions are taken from the database using methods from the JDBC class. There are only 4 answer options for each question and 4 buttons A,B,C,D are there to represent each answer option. There is a timer of 30 seconds for the player to answer each question and then the answer will be displayed. A pause of 2 seconds will be activated when the answer is displayed. Their answer label will change red to represent incorrect answer or green to represent correct answer to display the answer. After all the questions have been asked then the final score will be calculated and displayed. The player has the option to save the score using the save score button and it will be stored in the database. This class has dependency with JDBC because the data for the questions and answers need to be retrieved for it to work.
8. JDBC: This class is where all the database operations are handled. This class uses static methods so that other classes can easily access the methods. The methods created in the class are for either inserting data to the database or retrieving data from the database.

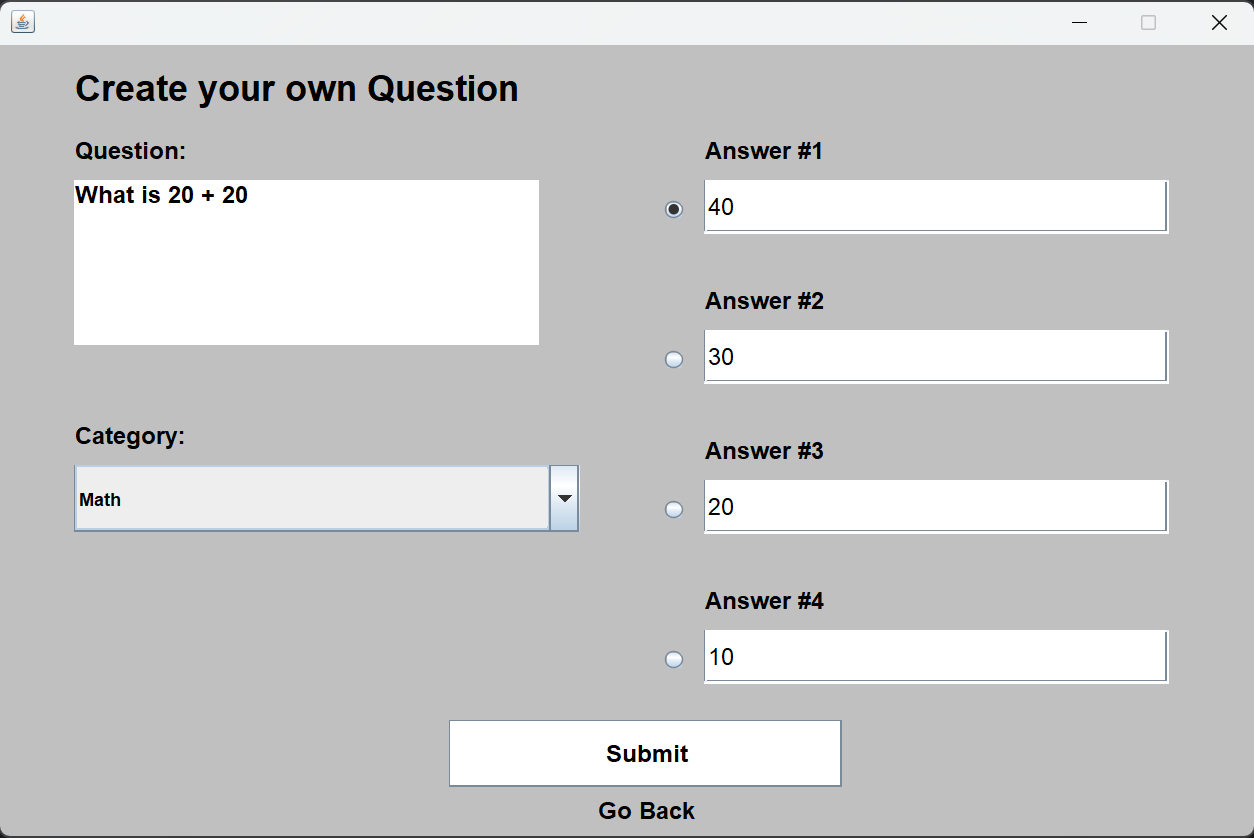
**Evidence of Working Program**

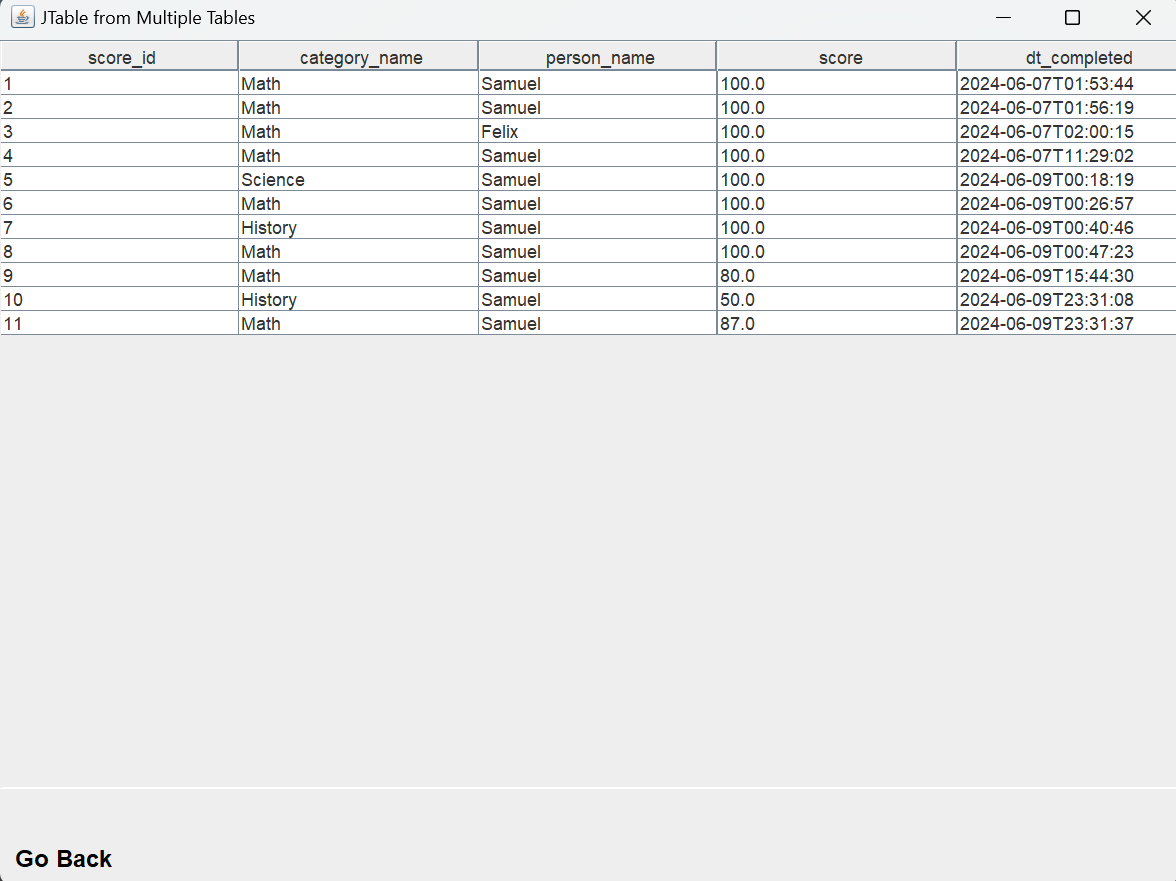
****

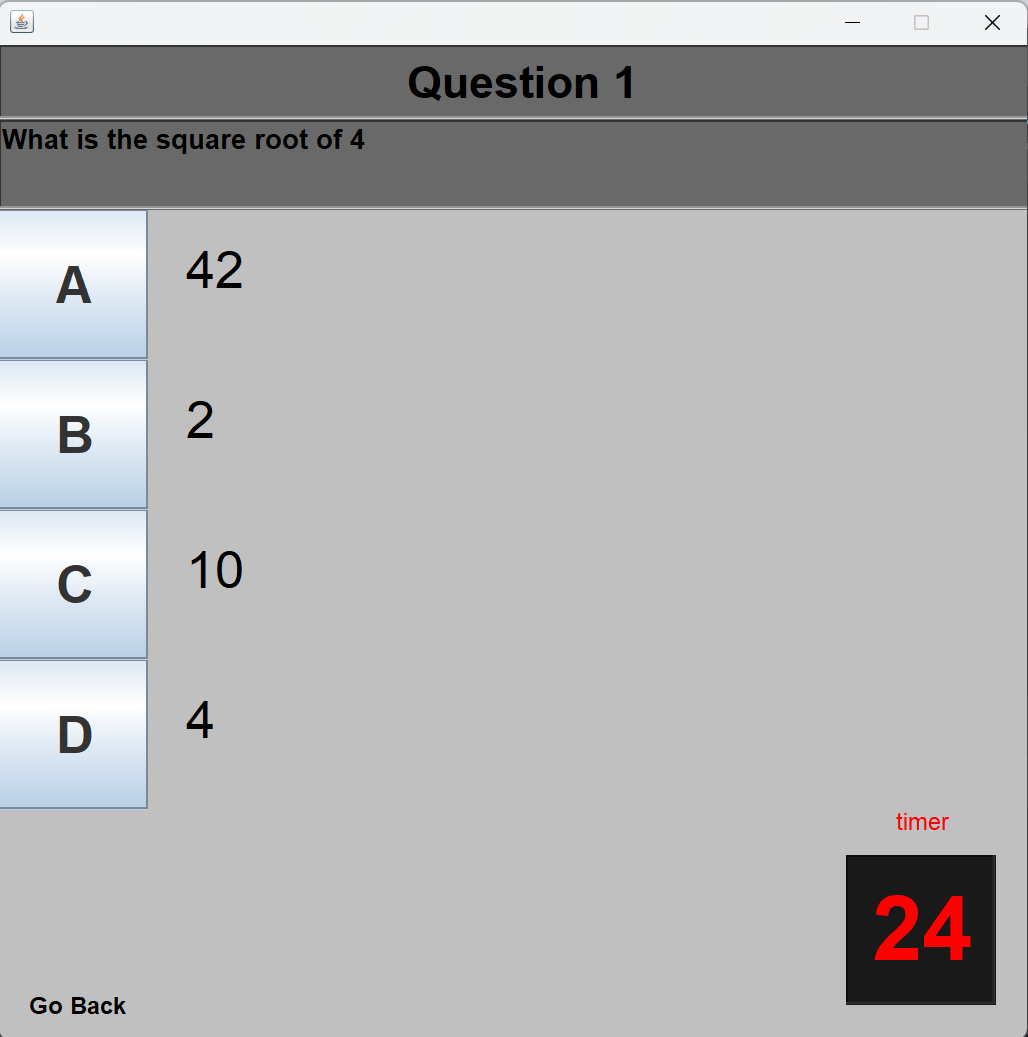
****

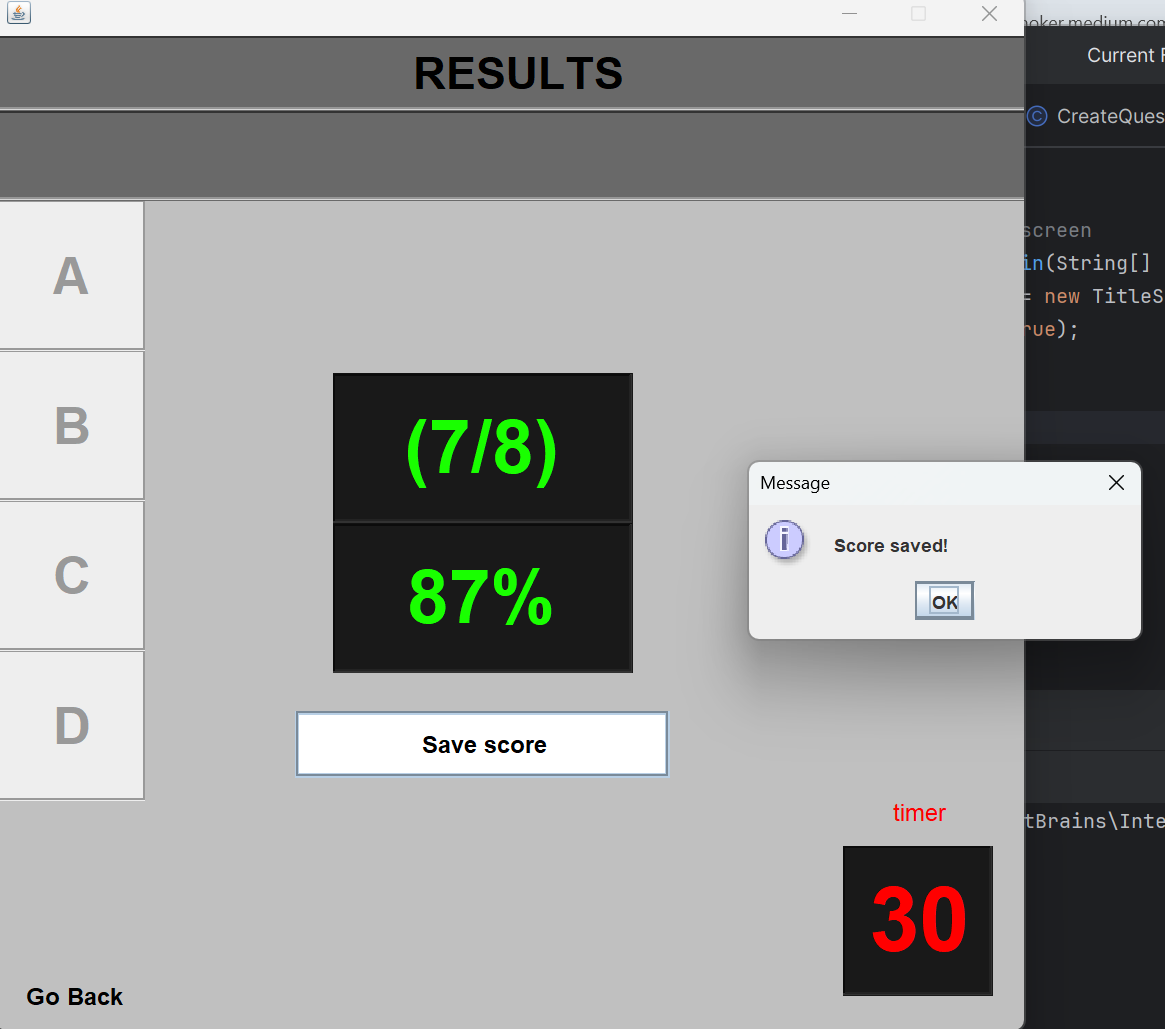
****

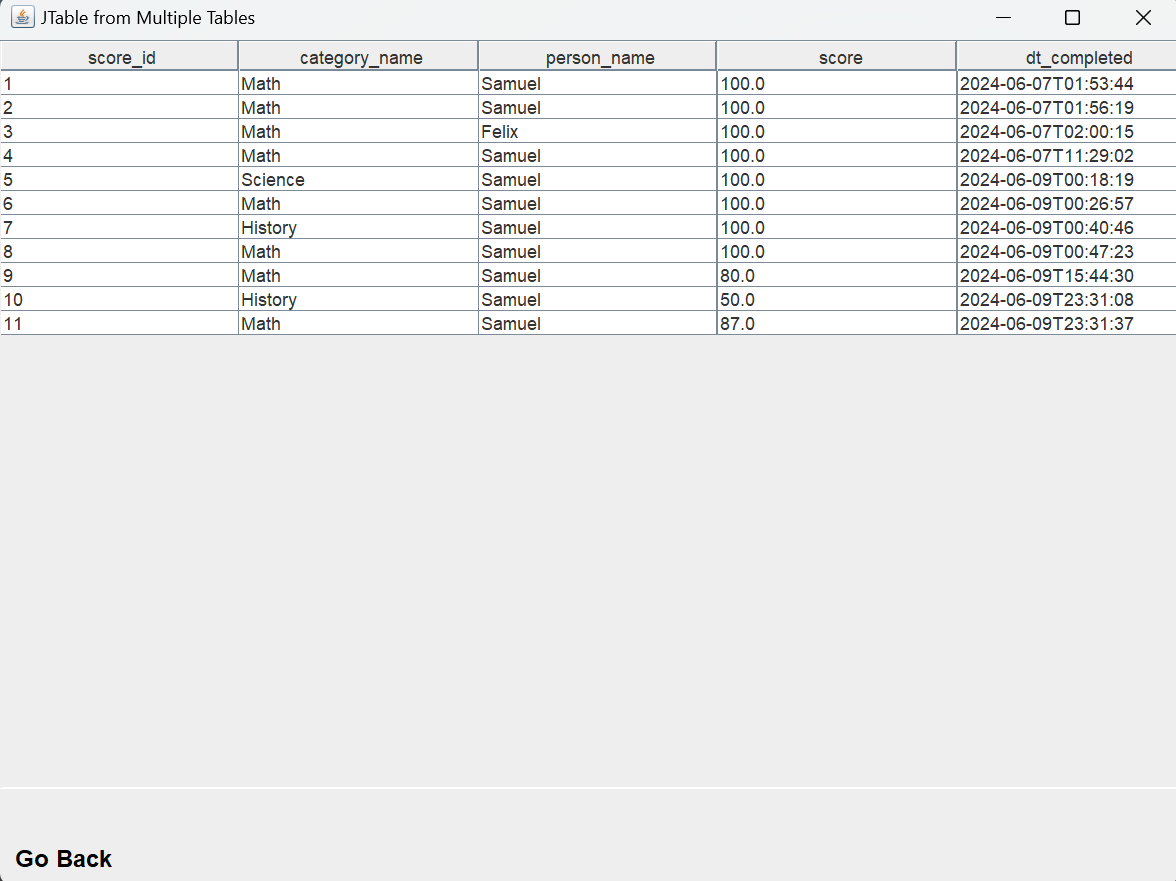
****

****

****

****

****

****

**Resources**

GeeksforGeeks. (2023, November 17). *Java JFrame*. GeeksforGeeks. <https://www.geeksforgeeks.org/java-jframe/>

TapTap. (2023, August 17). *Connect MySQL to Java in IntelliJ 2024 | Full JDBC Tutorial for DB Connectivity* [Video]. YouTube. <https://www.youtube.com/watch?v=9ntKSLLDeSs>

TapTap. (2024, March 31). *ASMR Programming - Quiz Game with MySQL Database - Java Swing Project* [Video]. YouTube. <https://www.youtube.com/watch?v=npjPPR0I2p8>