Faculty of Computing and Information Technology (FCIT) Department of Computing Indus University, Karachi



Course: Programming Fundamental (Lab) Semester: 2023

Class: SE 1ST



Project Report

Project title : Quiz game

# Submitted By:

Group Members:

|  |  |
| --- | --- |
| SYED AFNAN ALI | 751-2023 |
| SYED HAMZA AZIZ | 675-2023 |
| HAMMAD KHANZADA |  |

# Submitted To:

Miss Ayesha Quraishi

# Submitted On:

Date: 4th-Jan-2024

**Table of Contents**

|  |  |
| --- | --- |
| **S no.** | **Name of topics** |
| **1** | Introduction |
| **2** | Problem statement |
| **3** | Aims and objective |
| **4** | Features of the object |
| **5** | Overall Description |
| **6** | UML representation of project |
| **7** | Project Interface |
| **8** | Project source code |
| **9** | Conclusion and future work |
|  |  |

**INTROUCTION**

The project aims to create a quiz game in C++ where users can test their knowledge by answering a set of questions across various categories.

Background:

- Language: Developed using C++, a powerful programming language allowing for efficient management of data and logic.

- User Interface: Utilizes console input/output for simplicity but can be expanded to include a graphical interface using libraries like SFML or Qt.

- Functionality: The game presents a menu with different quiz categories/topics. Once selected, it navigates through a series of questions, prompting the user to select answers.

- Scoring: Tracks user scores based on correct answers and provides feedback at the end of the quiz.

**PROBLEM STATEMENT**

"Develop a console-based C++ quiz game that offers users to choose, presents questions one at a time, tracks scores, and provides immediate feedback on answers. The goal is to create an engaging and informative quiz experience within the console interface."

**AIMS AND OBJECTIVE**

Aim: Create an interactive C++ quiz game to engage users with, presenting questions sequentially, tracking scores.

Objectives:

1. Develop a console-based in C++.

3. Present questions one-by-one with multiple-choice answers.

4. Track and display user scores in real-time.

6. Create an engaging and user-friendly experience within the console interface.

**FEATURE OF THE OBJECT**

1. Sequential Question Presentation: Questions are displayed one at a time with multiple-choice answers.

2. Real-time Scoring: The system instantly tracks and updates the user's score.

3. Immediate Feedback: Users receive immediate feedback on the correctness of their answers.

4. Engaging Interface: The console-based interface ensures user-friendly interactions.

5. Performance Summary: After completing the quiz, users receive a summary of their performance, including total score and accuracy.

**Overall Description**

1. **Quiz Function (playQuiz) :**
2. The quiz function introduces itself and provides instructions.
3. The user is prompted to start the quiz by pressing 'S' or 's'.
4. If the user enters 'S' or 's', the program proceeds to ask ten questions, and the user's score is updated based on correct answers.
5. If an invalid character is entered, the program prompts the user to re-enter.
6. **Quiz Questions:**
7. The quiz questions cover a range of programming concepts, including programming languages, data types, debugging, and programming paradigms.
8. For each question, the user enters their choice (a, b, c, or d), and the program updates the score accordingly.
9. **Score Evaluation:**
10. After completing the quiz, the program displays the user's final score.
11. If the user scores 6 or higher, a congratulatory message is shown, indicating that the user has won.
12. The program then asks if the user wants to play again.
13. **Loop for Playing Again:**
14. If the user chooses to play again by entering 'y' or 'Y', the quiz function is called once more.
15. If the user enters any other character, the program displays a thank-you message and terminates.
16. **Overall Code Structure:**
17. The code is organized with proper indentation and comments.
18. The questions and options are hard-coded within the playQuiz function.
19. The program uses if-else statements to handle user input and determine the final outcome.
20. **User Interaction:**
21. The program provides clear instructions and feedback, making it user-friendly.
22. It congratulates the user on winning if the score is 6 or higher and asks if they want to play again.

**UML representation of project**

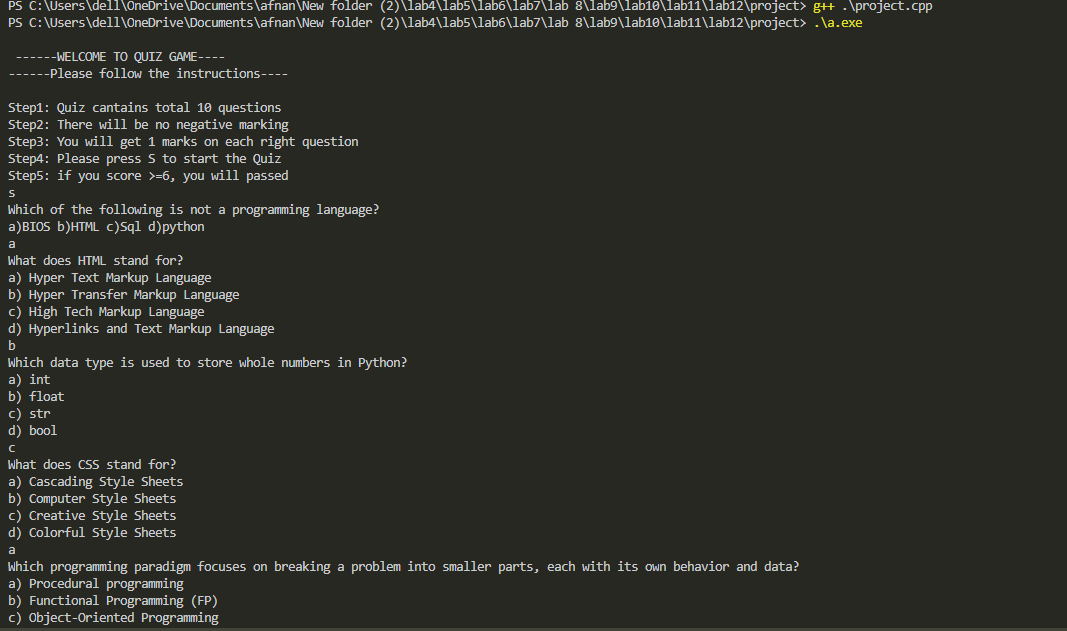
|  |
| --- |
| QuizGame |
| - Finalresult: int -Playagain: char |
| + main() + Playquiz(): int |

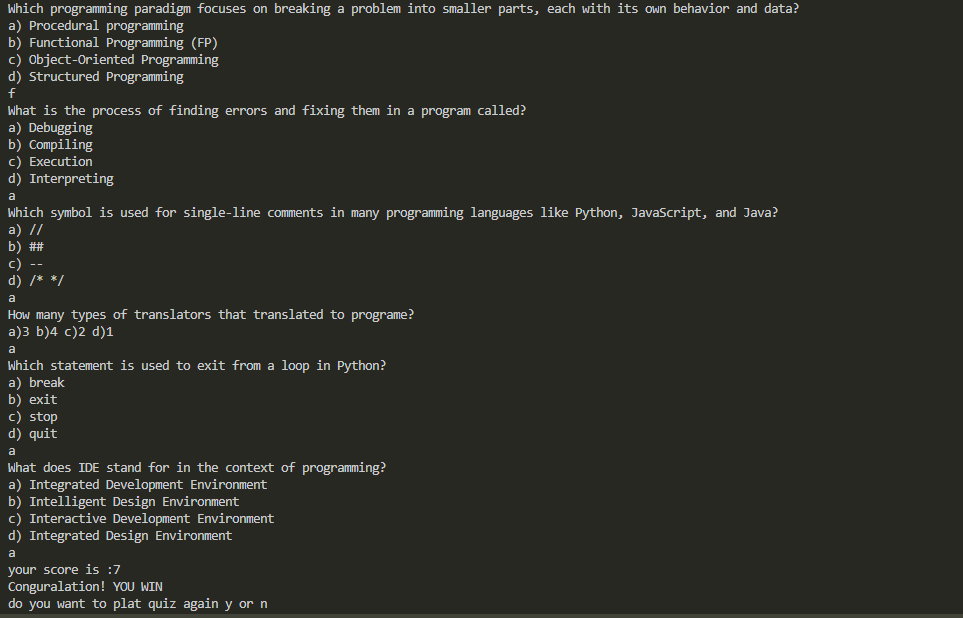
|  |
| --- |
| Quizround |
| - score: int |
| + playquiz() |

|  |
| --- |
| Std::cout |
|  |

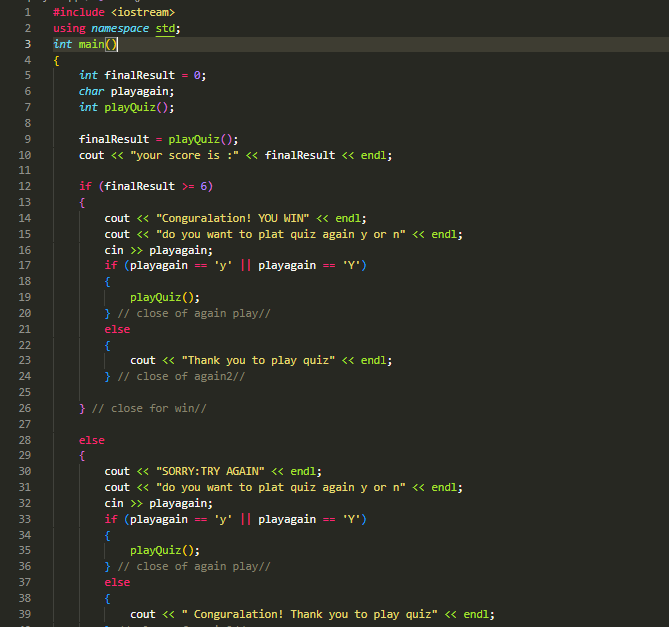
|  |
| --- |
| Std::cin |
| + operator>>(char&) |

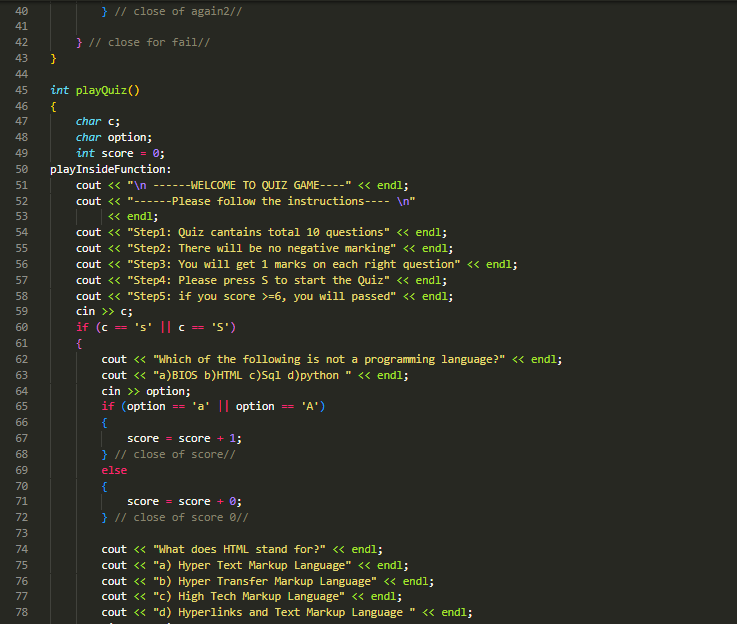
**PROJECT INTERFACE**

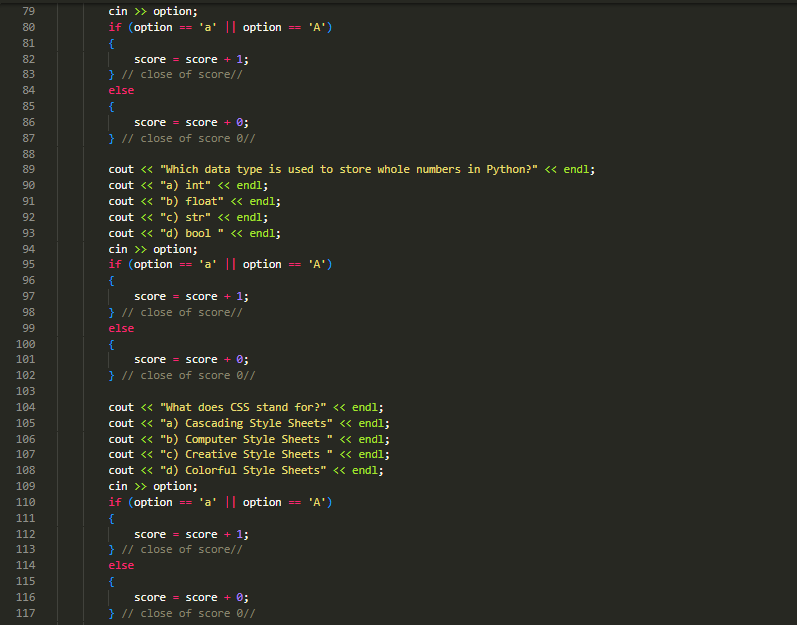
****

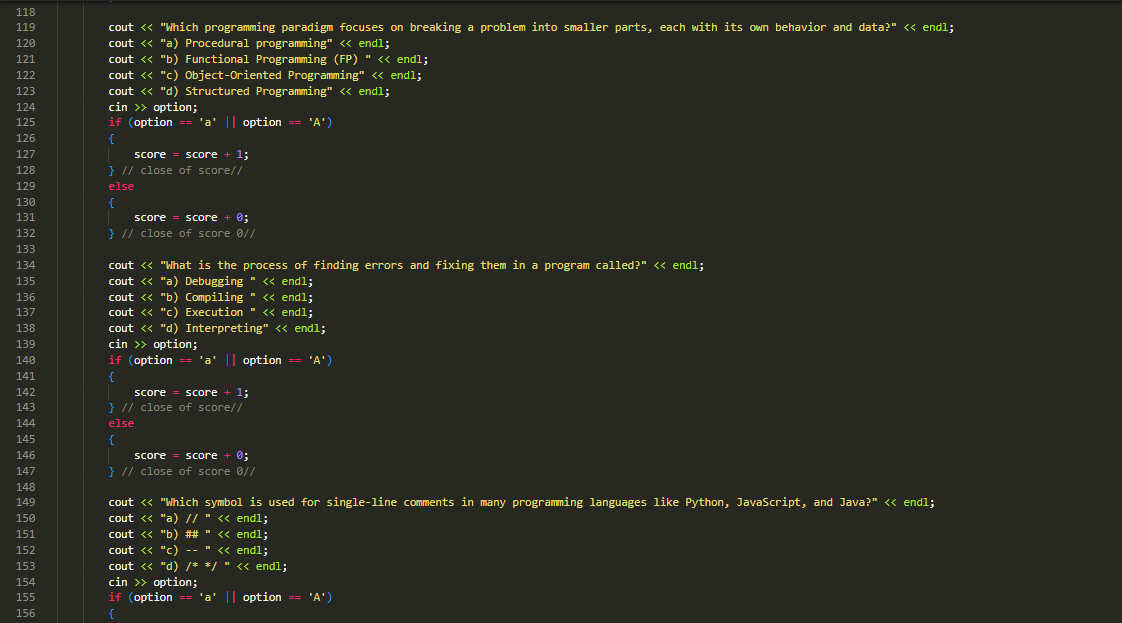
****

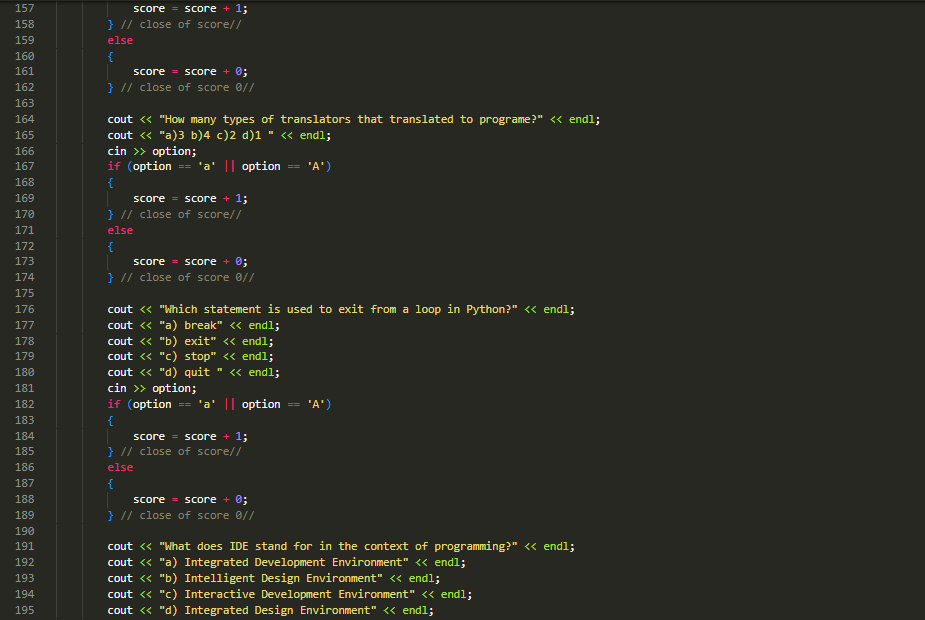
**PROJECT SOURCE CODE**

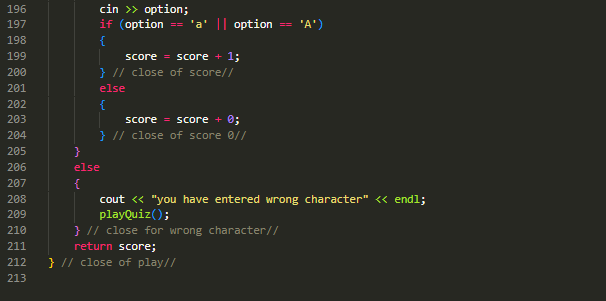
****

****

****

****

****

****

**Conclusion and future work**

Conclusion:

The developed C++ quiz game provides an engaging and interactive experience, real-time scoring, and immediate feedback. Its console-based interface ensures ease of use while testing users' knowledge.

Future Work:

1. Enhanced User Interface: Explore options to create a graphical interface for a more visually appealing experience.

2. Multiplayer Functionality: Implement multiplayer features for competitive play.

3. Additional Question Sets: Continuously expand the question database for each category to keep the game fresh and challenging.

4. Difficulty Levels: Introduce varying difficulty levels within categories to cater to different skill levels.

5. Analytic and Performance Tracking:\*\* Include user analytic and tracking mechanisms to monitor user progress and tailor quizzes accordingly.

These potential future developments aim to enhance the game's features, engagement, and scalability for a broader user base and increased enjoyment.