**OOPS MINI PROJECT**

**Online Quiz Application with Countdown Timer using Java**

|  |  |
| --- | --- |
| **Name:** | **JAHNAVI .A.B** |
| **Register number:** | **2117240070121** |
| **Department:** | **AI & DS** |
| **Section:** | **B** |
| **Subject code:** | **CS23312** |
| **Subject name:** | **Object Oriented Programming** |

1. **Introduction:**

The Online Quiz Application with Countdown Timer is a Java-based mini project designed to conduct quizzes in an interactive and time-controlled manner. The application enables users to answer multiple-choice questions within a fixed time limit, automatically submitting the quiz once the timer expires. This project demonstrates core Object-Oriented Programming concepts such as classes, objects, encapsulation, inheritance, and event handling. The GUI is created using Java Swing, making it simple and user-friendly for students and instructors.

1. **Objectives:**

**The main objectives of this project are:**

1. To design a simple and interactive online quiz system using Java.
2. To implement a countdown timer for each question.
3. To evaluate user answers and display the final score automatically.
4. To demonstrate OOP principles in practical implementation.
5. To provide a foundation for advanced quiz or e-learning applications.
6. **System Requirements and Setup:**

**Hardware requirements:**

|  |  |
| --- | --- |
| Processor | Intel Core i3 or above |
| RAM | Minimum 4 GB |
| Storage | At least 1 GB free space |
| Network | Local network or internet connection |

**Software requirements:**

|  |  |
| --- | --- |
| Operating System | Windows / Linux / macOS |
| IDE | Eclipse IDE (preferred) |
| JDK | Java Development Kit (JDK 17 or above) |
| Database | MySQL Server 8.x or above |
| JDBC Driver | MySQL Connector/J |
| Libraries | java.sql, java.net, javax.swing |

1. **Setup and Installation Procedure**

Step 1: Install JDK and IDE on your system.

Step 2: Create a new Java project in your IDE.

Step 3: Copy the provided source code into the main class.

Step 4: Compile and run the application to start the quiz.

1. **Procedure / Working:**

1. The user launches the Java application.

2. The main screen displays quiz instructions and a “Start Quiz” button.

3. When the quiz begins, the timer starts counting down for each question.

4. The user selects the correct answer from multiple choices.

5. After answering all questions or when time runs out, the quiz automatically ends.

6. The final score and correct answers are displayed to the user.

**6.Flow Diagram:**

**7.CODE:**

**OnlineQuizTimer.java**

package onlinequiz;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.util.Timer;

import java.util.TimerTask;

import java.sql.\*;

public class OnlineQuizTimer extends JFrame implements ActionListener {

private static final long ***serialVersionUID*** = 1L;

JLabel questionLabel, timerLabel, scoreLabel;

JRadioButton opt1, opt2, opt3, opt4;

JButton nextButton;

ButtonGroup bg;

int index = 0, score = 0, timeLeft = 15;

Timer timer;

String username;

// Database details

final String DB\_URL = "jdbc:mysql://localhost:3306/quizdb";

final String DB\_USER = "root";

final String DB\_PASS = "12345"; // change this to your MySQL password

String[][] questions = {

{"Java is a \_\_\_ language.", "Compiled", "Interpreted", "Both", "None", "3"},

{"Which keyword is used to inherit a class?", "super", "this", "extends", "implements", "3"},

{"Which package contains Swing?", "java.awt", "javax.swing", "java.io", "java.util", "2"},

{"OOP stands for?", "Object Oriented Programming", "Order Of Process", "Output Operation Program", "None", "1"}

};

OnlineQuizTimer() {

username = JOptionPane.*showInputDialog*(this, "Enter your name:");

setTitle("🧠 Online Quiz with Timer and JDBC");

setLayout(new BorderLayout(10, 10));

setDefaultCloseOperation(***EXIT\_ON\_CLOSE***);

getContentPane().setBackground(new Color(245, 245, 255));

setSize(500, 400);

// Panel for question and options

JPanel centerPanel = new JPanel(new GridLayout(6, 1, 10, 10));

centerPanel.setBackground(new Color(245, 245, 255));

questionLabel = new JLabel();

questionLabel.setFont(new Font("Segoe UI", Font.***BOLD***, 18));

questionLabel.setForeground(new Color(40, 40, 90));

timerLabel = new JLabel("Time Left: " + timeLeft + " sec", SwingConstants.***CENTER***);

timerLabel.setFont(new Font("Segoe UI", Font.***PLAIN***, 14));

timerLabel.setForeground(Color.***RED***);

scoreLabel = new JLabel("", SwingConstants.***CENTER***);

bg = new ButtonGroup();

opt1 = createOptionButton();

opt2 = createOptionButton();

opt3 = createOptionButton();

opt4 = createOptionButton();

bg.add(opt1);

bg.add(opt2);

bg.add(opt3);

bg.add(opt4);

centerPanel.add(questionLabel);

centerPanel.add(opt1);

centerPanel.add(opt2);

centerPanel.add(opt3);

centerPanel.add(opt4);

centerPanel.add(timerLabel);

nextButton = new JButton("Next ➡️");

nextButton.setFont(new Font("Segoe UI", Font.***BOLD***, 14));

nextButton.setBackground(new Color(90, 130, 230));

nextButton.setForeground(Color.***WHITE***);

nextButton.setFocusPainted(false);

nextButton.setBorder(BorderFactory.*createEmptyBorder*(10, 20, 10, 20));

nextButton.addActionListener(this);

JPanel bottomPanel = new JPanel();

bottomPanel.setBackground(new Color(245, 245, 255));

bottomPanel.add(nextButton);

add(centerPanel, BorderLayout.***CENTER***);

add(bottomPanel, BorderLayout.***SOUTH***);

loadQuestion();

setVisible(true);

}

JRadioButton createOptionButton() {

JRadioButton btn = new JRadioButton();

btn.setFont(new Font("Segoe UI", Font.***PLAIN***, 16));

btn.setBackground(new Color(245, 245, 255));

return btn;

}

void loadQuestion() {

if (index < questions.length) {

questionLabel.setText("Q" + (index + 1) + ": " + questions[index][0]);

opt1.setText(questions[index][1]);

opt2.setText(questions[index][2]);

opt3.setText(questions[index][3]);

opt4.setText(questions[index][4]);

bg.clearSelection();

startTimer();

} else {

showResult();

}

}

void startTimer() {

timeLeft = 15;

if (timer != null)

timer.cancel();

timer = new Timer();

timer.scheduleAtFixedRate(new TimerTask() {

public void run() {

timerLabel.setText(" Time Left: " + timeLeft + " sec");

if (timeLeft == 0) {

timer.cancel();

index++;

loadQuestion();

}

timeLeft--;

}

}, 0, 1000);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == nextButton) {

checkAnswer();

index++;

loadQuestion();

}

}

void checkAnswer() {

String correct = questions[index][5];

if ((opt1.isSelected() && correct.equals("1")) ||

(opt2.isSelected() && correct.equals("2")) ||

(opt3.isSelected() && correct.equals("3")) ||

(opt4.isSelected() && correct.equals("4"))) {

score++;

}

}

void showResult() {

getContentPane().removeAll();

scoreLabel.setText(" Your Score: " + score + "/" + questions.length);

scoreLabel.setFont(new Font("Segoe UI", Font.***BOLD***, 20));

scoreLabel.setForeground(new Color(30, 70, 150));

add(scoreLabel, BorderLayout.***CENTER***);

revalidate();

repaint();

saveResultToDatabase();

}

void saveResultToDatabase() {

try {

Class.*forName*("com.mysql.cj.jdbc.Driver");

Connection conn = DriverManager.*getConnection*(DB\_URL, DB\_USER, DB\_PASS);

String query = "INSERT INTO results (username, score, total) VALUES (?, ?, ?)";

PreparedStatement pst = conn.prepareStatement(query);

pst.setString(1, username);

pst.setInt(2, score);

pst.setInt(3, questions.length);

pst.executeUpdate();

conn.close();

JOptionPane.*showMessageDialog*(this, " Your score has been saved to the database!");

} catch (Exception ex) {

JOptionPane.*showMessageDialog*(this, " Database Error: " + ex.getMessage());

}

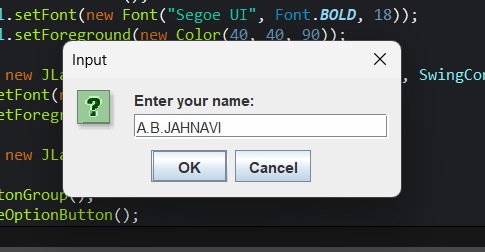
}

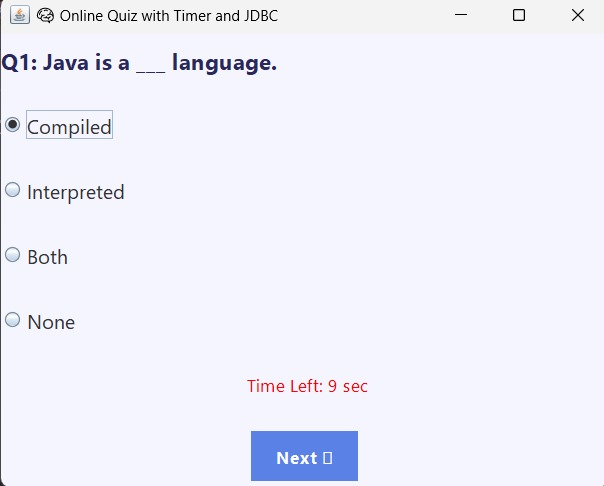
public static void main(String[] args) {

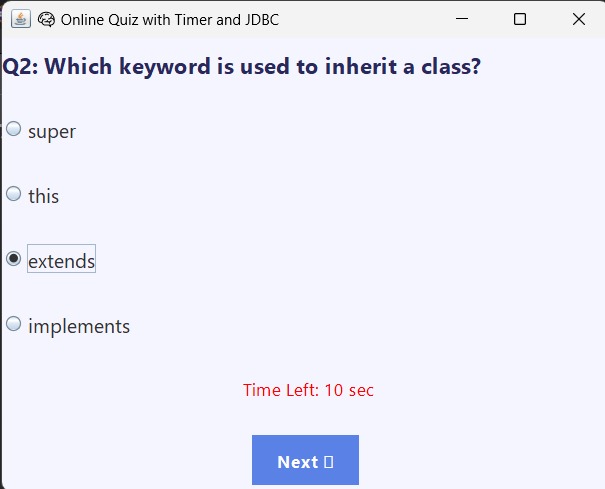
new OnlineQuizTimer();

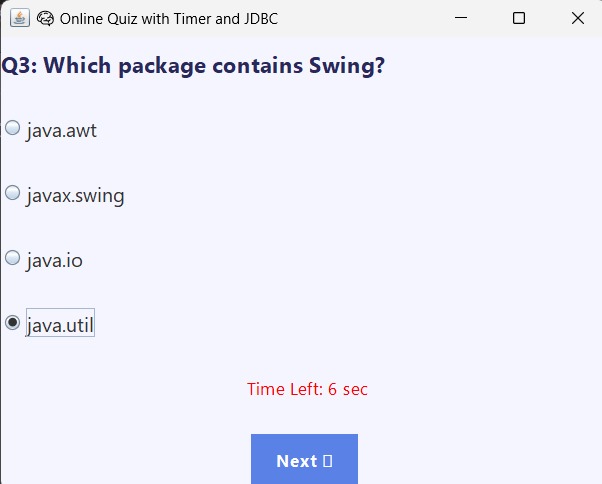
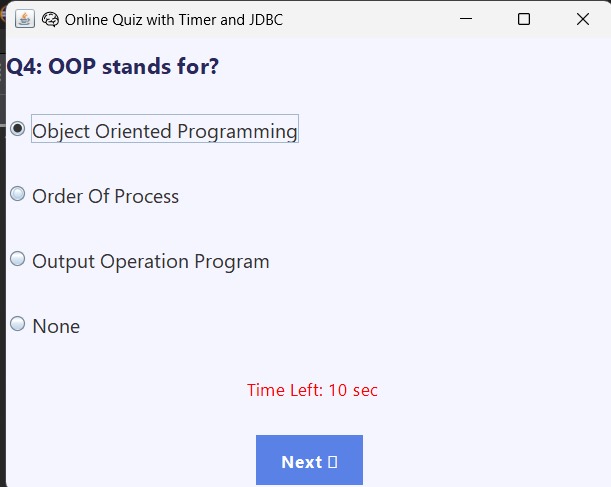
}

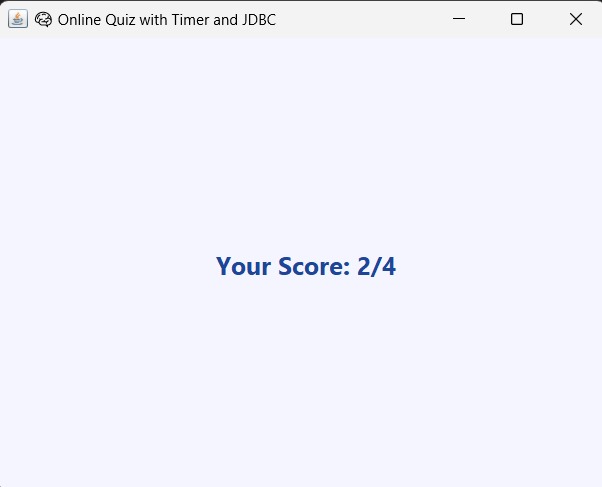
}

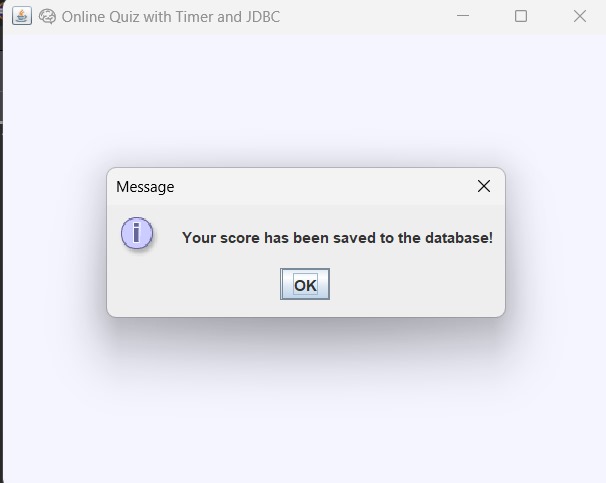
**8.Screenshots:**

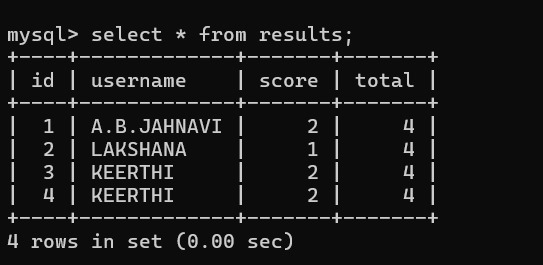












**9.Github Repository Link:**

**https://github.com/JAHNAVI-BOOPATHY/OnlineQuizTimer.git**