**PROIECT POO**

**DOCUMENTAȚIE**

**QUIZ CU ÎNTREBĂRI ȘI RĂSPUNSURI MULTIPLE**

NUME: FAZAKAS EDINA

GRUPA: 30223

SERIA A

Un quiz este o serie de întrebări la care jucătorul trebuie să răspundă și care sunt evaluate imediat ce un răspuns este înregistrat.

În proiectul meu jucătorul trebuie să răspundă la 10 întrebări, fiecare întrebare valorând 10 puncte, deci un jucător poate să obțină 100 de puncte în cazul în care răspunde corect la toate întrebările. Întrebările sunt legate de cunoștințe generale în JAVA. Jucătorul trebuie să introducă numele, să fie atent la instrucțiuni și să răspundă corect la cât mai multe întrebări.

Proiectul este implementat într-un pachet, și lucrez cu 3 clase. O clasă se numește Quiz, pagina care apare atunci când se deschide quiz-ul, o clasă Page1, pagina care apare după introducerea numelui jucătorului, adică în momentul în care începe quiz-ul, iar a treia clasă se numește Quiestions, asta va fi pagina care va conține întrebările.

1. **Clasa QUIZ**

Text

Description automatically generated

În această clasă am un label ”WELCOME TO QUIZ”, două butoane ”GOOD LUCK”, ”Instructions” cu intrucțiunile, și un textfield cu numele jucătorului. Dacă jucătorul nu introduce un nume, atunci se afișează mesajul ”Player name required”. Dacă se introduce numele corect și se apasă butonul START, atunci va apărea pagina principală, dacă se apasă CANCEL, atunci se va închide jocul.

Codul:

package com.company;  
  
import javafx.scene.layout.Border;  
  
import java.awt.\*;  
import java.awt.event.\*;  
import javax.swing.\*;  
import java.awt.Color;  
class Quiz implements ActionListener  
{  
 JFrame f;  
 JLabel l1,l2,l4,l5,l6;  
 JTextField t1,t2;  
 JButton b1,b2,b3,b4;  
  
 Quiz()  
 {  
 Color c1 = new Color(81, 150, 84);  
 Color c2 = new Color(63, 132, 65);  
 Color c3 = new Color(39, 86, 41);  
 Color c4 = new Color(20, 68, 22);  
 f = new JFrame("QUIZ");  
 f.getContentPane().setBackground(c1);  
 f.setVisible(true);  
 f.setSize(2000,2000);  
 f.setLayout(new FlowLayout(FlowLayout.*CENTER*,2000,25));  
 JLabel ll=new JLabel(new ImageIcon("mits1.JPEG"));  
 l1 = new JLabel("WELCOME TO QUIZ");  
 b1 = new JButton("GOOD LUCK");  
 b2 = new JButton("Instructions");  
 l2 = new JLabel("NAME:");  
 t1 = new JTextField(15);  
 b3 = new JButton("START");  
 b4 = new JButton("CANCEL");  
 l4 = new JLabel("1. Every question has 10 points, you can reach 100 points in total");  
 l6 = new JLabel("3. Good luck! :)");  
 l5 = new JLabel("2. Every question has 1 answer from A to D");  
 //b1.setForeground(Color.(255,20,147);  
 b1.setBackground(c2);  
 b1.setForeground(c4);  
 b3.setBackground(c3);  
 b4.setBackground(c3);  
 l1.setFont(new Font("cooper",Font.*BOLD*,60));  
 b1.setFont(new Font("casteller",Font.*BOLD*,50));  
 b2.setFont(new Font("cooper",Font.*BOLD*,35));  
 b3.setFont(new Font("cooper",Font.*BOLD*,25));  
 b4.setFont(new Font("cooper",Font.*BOLD*,25));  
 l4.setFont(new Font("casteller",Font.*BOLD*,35));  
 l5.setFont(new Font("casteller",Font.*BOLD*,35));  
 l6.setFont(new Font("casteller",Font.*BOLD*,35));  
 l2.setFont(new Font("casteller",Font.*BOLD*,45));  
 f.add(ll);  
 f.add(l1);  
 f.add(b1);  
 f.add(b2);  
 f.add(l4);  
 f.add(l5);  
 f.add(l6);  
 f.add(l2);  
 f.add(t1);  
 f.add(b3);  
 f.add(b4);  
 b3.addActionListener(this);  
 b4.addActionListener(this);  
 }  
 public void actionPerformed(ActionEvent ae)  
 {  
 if(ae.getSource().equals(b3))  
 {  
 try  
 {  
 String s1 =t1.getText();  
 if(t1.getText().equals(""))  
 {JOptionPane.*showMessageDialog*(null,"Player name required");}  
 else  
 {f.setVisible(false);  
 new Page1(f,s1);  
 }  
 }  
 finally  
 {}  
 }  
 else  
 {  
 f.setVisible(false);  
 }  
 }  
 public static void main(String[] args)  
 {  
 new Quiz();  
 }  
}

1. **Clasa Page1**

Table

Description automatically generated

În această clasă se face o enumerare de întrebări, fiind ”pagina principală”. Jucătorul poate să revină oricând aici și să aleagă o întrebare la care vrea să răspundă. Aici se poate totodată salva toate răspunsurile prin apăsarea butonului ”SAVE”. În acel moment quiz-ul se va închide și se va afișa punctajul obținut împreună cu numărul de îintrebări la care a răspuns jucătorul. Deci în această pagină se permite navigarea între întrebări, totodată se urează succes prin label-ul GOOD LUCK. Întrebările sunt enumerate prin JRadioButton. După salvarea răspunsurilor jucătorul va avea opțiunea de a reîncepe quiz-ul.

Codul:

package com.company;  
  
import sun.font.TextLabel;  
  
import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import javax.swing.Timer;  
  
class Page1 implements ActionListener {  
  
 int m = 0, n = 0;  
 JFrame f1, p;  
 JRadioButton b1, b2, b3, b4, b5, b6, b7, b8, b9, b10, b11, b12, b13, b14, b15, b16, b17, b18, b19, b20;  
 JLabel l1;  
 JButton b111;  
 String ss1;  
  
  
 Page1(JFrame f, String s1) {  
  
 Color c1 = new Color(81, 150, 84);  
 Color c2 = new Color(63, 132, 65);  
 Color c3 = new Color(39, 86, 41);  
 Color c4 = new Color(20, 68, 22);  
 ButtonGroup g11 = new ButtonGroup();  
 f1 = f;  
 ss1 = s1;  
 p = new JFrame("QUIZ");  
 p.getContentPane().setBackground(c1);  
 p.setVisible(true);  
 p.setSize(2000, 2000);  
 p.setLayout(new FlowLayout(FlowLayout.*CENTER*, 2000, 5));  
  
 b1 = new JRadioButton("QUESTION 1");  
 b2 = new JRadioButton("QUESTION 2");  
 b3 = new JRadioButton("QUESTION 3");  
 b4 = new JRadioButton("QUESTION 4");  
 b5 = new JRadioButton("QUESTION 5");  
 b6 = new JRadioButton("QUESTION 6");  
 b7 = new JRadioButton("QUESTION 7");  
 b8 = new JRadioButton("QUESTION 8");  
 b9 = new JRadioButton("QUESTION 9");  
 b10 = new JRadioButton("QUESTION 10");  
 g11.add(b1);  
 g11.add(b2);  
 g11.add(b3);  
 g11.add(b4);  
 g11.add(b5);  
 g11.add(b6);  
 g11.add(b7);  
 g11.add(b8);  
 g11.add(b9);  
 g11.add(b10);  
 g11.add(b11);  
 g11.add(b12);  
 g11.add(b13);  
 g11.add(b14);  
 g11.add(b15);  
 g11.add(b16);  
 g11.add(b17);  
 g11.add(b18);  
 g11.add(b19);  
 g11.add(b20);  
 l1 = new JLabel("GOOD LUCK");  
 l1.setBackground(c4);  
 l1.setForeground(c4);  
 l1.setFont(new Font("cooper", Font.*BOLD*, 80));  
 b111 = new JButton("SAVE");  
 b1.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b2.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b3.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b4.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b5.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b6.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b7.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b8.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b9.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b10.setFont(new Font("casteller", Font.*BOLD*, 40));  
 b111.setFont(new Font("casteller", Font.*BOLD*, 45));  
  
  
 b1.setForeground(c4);  
 b2.setForeground(c4);  
 b3.setForeground(c4);  
 b4.setForeground(c4);  
 b5.setForeground(c4);  
 b6.setForeground(c4);  
 b7.setForeground(c4);  
 b8.setForeground(c4);  
 b9.setForeground(c4);  
 b10.setForeground(c4);  
  
 b1.setBackground(c1);  
 b2.setBackground(c1);  
 b3.setBackground(c1);  
 b4.setBackground(c1);  
 b5.setBackground(c1);  
 b6.setBackground(c1);  
 b7.setBackground(c1);  
 b8.setBackground(c1);  
 b9.setBackground(c1);  
 b10.setBackground(c1);  
  
 b111.setForeground(c3);  
  
  
 p.add(l1);  
 p.add(b1);  
 p.add(b2);  
 p.add(b3);  
 p.add(b4);  
 p.add(b5);  
 p.add(b6);  
 p.add(b7);  
 p.add(b8);  
 p.add(b9);  
 p.add(b10);  
  
 p.add(b111);  
 b111.addActionListener(this);  
 b1.addActionListener(this);  
 b2.addActionListener(this);  
 b3.addActionListener(this);  
 b4.addActionListener(this);  
 b5.addActionListener(this);  
 b6.addActionListener(this);  
 b7.addActionListener(this);  
 b8.addActionListener(this);  
 b9.addActionListener(this);  
 b10.addActionListener(this);  
  
  
 }  
  
  
 public void actionPerformed(ActionEvent ae) {  
 if (ae.getSource().equals(b1)) {  
 p.setVisible(false);  
 String question1 = new String("1. The range of indices for an array always start at:");  
 String answer1 = "A. Whatever programmer specifies";  
 String answer2 = "B. 1";  
 String answer3 = "C. 0";  
 String answer4 = "D. Size of array";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer3);  
 }  
 if (ae.getSource().equals(b2)) {  
 p.setVisible(false);  
 String question1 = new String("2. Which of the following is not a keyword in java?");  
 String answer1 = "A. Static";  
 String answer2 = "B. Boolean";  
 String answer3 = "C. Void";  
 String answer4 = "D. Private";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer2);  
  
 }  
 if (ae.getSource().equals(b3)) {  
 p.setVisible(false);  
 String question1 = new String("3. What is the default value of byte variable?");  
 String answer1 = "A. 0";  
 String answer2 = "B. 0.0";  
 String answer3 = "C. null";  
 String answer4 = "D. undefined";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 if (ae.getSource().equals(b4)) {  
 p.setVisible(false);  
 String question1 = new String("4. What is class variable?");  
 String answer1 = "A. Class variables are static variables within a class but outside any method.";  
 String answer2 = "B. Class variables are variables defined inside methods, constructors or blocks.";  
 String answer3 = "C. Class variables are variables within a class but outside any method.";  
 String answer4 = "D. None of the above.";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 if (ae.getSource().equals(b5)) {  
 p.setVisible(false);  
 String question1 = new String("5. What is function overloading?");  
 String answer1 = "A. Methods with same name but different parameters.";  
 String answer2 = "B. Methods with same name but different return types.";  
 String answer3 = "C. Methods with same name, same parameter types but different parameter names.";  
 String answer4 = "D. None of the above.";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 if (ae.getSource().equals(b6)) {  
 p.setVisible(false);  
 String question1 = new String("6. What is currentThread()?");  
 String answer1 = "A. It is a Thread public static method used to obtain a reference to the current thread.";  
 String answer2 = "B. It is a thread's instance method used to get thread count.";  
 String answer3 = "C. It is a object's public static method used obtain a reference to the current thread.";  
 String answer4 = "D. It is a object's instance method used to get thread count.";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 if (ae.getSource().equals(b7)) {  
 p.setVisible(false);  
 String question1 = new String("7. Which is not a part of defining an object?");  
 String answer1 = "A. Description";  
 String answer2 = "B. Methods";  
 String answer3 = "C. Associations with other objects";  
 String answer4 = "D. Name";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 if (ae.getSource().equals(b8)) {  
 p.setVisible(false);  
 String question1 = new String("8. Which one could be used as the main container in a Java application?");  
 String answer1 = "A. JApplet";  
 String answer2 = "B. JFrame";  
 String answer3 = "C. JPanel";  
 String answer4 = "D. JButton";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer2);  
 }  
 if (ae.getSource().equals(b9)) {  
 p.setVisible(false);  
 String question1 = new String("9. What is the difference between private and public functions?");  
 String answer1 = "A. Public functions are free, you have to buy private ones";  
 String answer2 = "B. Public functions are the only ones you can download";  
 String answer3 = "C. Public functions can be used by anyone, private can only be used by other code in the class you are writing";  
 String answer4 = "D. Public functions can’t be used";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer3);  
 }  
 if (ae.getSource().equals(b10)) {  
 p.setVisible(false);  
 String question1 = new String("10. What is the keyword used in java to create an object?");  
 String answer1 = "A. This";  
 String answer2 = "B. New";  
 String answer3 = "C. Sync";  
 String answer4 = "D. New()";  
 new Question(p, m, n, f1, ss1, question1, answer1, answer2, answer3, answer4, answer2);  
 }  
  
  
 if (ae.getSource().equals(b111)) {  
 p.setVisible(false);  
 JOptionPane.*showMessageDialog*(null, "Saved successfully");  
 JOptionPane.*showMessageDialog*(null, "Player " + ss1 + " has reached " + m + " points");  
 int n = JOptionPane.*showConfirmDialog*(p, "Would you like to start another quiz?", "Quiz", JOptionPane.*YES\_NO\_OPTION*);  
 if (n == JOptionPane.*YES\_OPTION*)  
 new Quiz();  
 else p.setVisible(false);  
  
 }  
 }  
  
  
  
 public void marks(int m1, int n1) {  
 m = m + m1;  
 n = n + n1;  
 }  
  
}

1. **Clasa Question**

Diagram

Description automatically generated with medium confidence

În această clasă se afișează întrebările. Butonul SAVE ANSWER va salva răspunsurile. Putem să alegem opțiunea de a ne duce înapoi sau înainte cu întrebările ori să ne întoarcem la pagina principală. După fiecare întrebare se va afișa punctajul și numărul de întrebări la care a răspuns jucătorul. Atunci când jucătorul va ajunge la întrebarea 10, și va apăsa butonul NEXT, quiz-ul se va închide, întrebarea 10 fiind ultima întrebare din quiz. După salvarea răspunsurilor jucătorul va avea opțiunea de a reîncepe quiz-ul.

Codul:

package com.company;  
  
import java.awt.\*;  
import java.awt.event.\*;  
import javax.swing.\*;  
class Question implements ActionListener  
{  
 int m1, n1;  
 String sss1;  
 JFrame f2,q;  
 JLabel l1;  
 JRadioButton b1,b2,b3,b4,correct1;  
 JButton b5, b6, b7, b8;  
 String answer11, answer12, answer13, answer14;  
 Question(JFrame f, int m, int n, JFrame f1, String ss1, String question, String answer1, String answer2, String answer3, String answer4, String correct)  
 {  
 Color c1 = new Color(81, 150, 84);  
 Color c2 = new Color(175, 250, 177);  
 Color c3 = new Color(24, 47, 25);  
 Color c4 = new Color(20, 68, 22);  
 m1 = m;  
 n1 = n;  
 f2 = f;  
 sss1 = ss1;  
 q = new JFrame("Question");  
 q.getContentPane().setBackground(c1);  
 ButtonGroup j=new ButtonGroup();  
 JLabel ll=new JLabel(new ImageIcon("ram.JPEG"));  
 l1 = new JLabel(question);  
 b1 = new JRadioButton(answer1);  
 b2 = new JRadioButton(answer2);  
 b3 = new JRadioButton(answer3);  
 b4 = new JRadioButton(answer4);  
 if(correct.equals(answer1))  
 correct1 = b1;  
 else if(correct.equals(answer2))  
 correct1 = b2;  
 else if(correct.equals(answer3))  
 correct1 = b3;  
 else if(correct.equals(answer4))  
 correct1 = b4;  
  
 answer11 = answer1;  
 answer12 = answer2;  
 answer13 = answer3;  
 answer14 = answer4;  
  
 b5 = new JButton("SAVE ANSWER");  
 b6 = new JButton("PREVIOUS");  
 b7 = new JButton("NEXT");  
 b8 = new JButton("FIRST PAGE");  
 l1.setFont(new Font("cooper",Font.*BOLD*,55));  
 b1.setFont(new Font("casteller",Font.*BOLD*,30));  
 b2.setFont(new Font("casteller",Font.*BOLD*,30));  
 b3.setFont(new Font("casteller",Font.*BOLD*,30));  
 b4.setFont(new Font("casteller",Font.*BOLD*,30));  
 b5.setFont(new Font("cooper",Font.*BOLD*,45));  
 b6.setFont(new Font("cooper",Font.*BOLD*,45));  
 b7.setFont(new Font("cooper",Font.*BOLD*,45));  
 b8.setFont(new Font("cooper",Font.*BOLD*,45));  
 q.setVisible(true);  
 q.setSize(2000,2000);  
 q.setLayout(new FlowLayout(FlowLayout.*CENTER*,2500,20));  
 q.add(ll);  
 b1.setForeground(c4);  
 l1.setForeground(c3);  
 b2.setForeground(c4);  
 b3.setForeground(c4);  
 b4.setForeground(c4);  
 //b5.setForeground(c4);  
 //b6.setForeground(c4);  
 //b7.setForeground(c4);  
 //b8.setForeground(c4);  
  
 b1.setBackground(c2);  
 b2.setBackground(c2);  
 b3.setBackground(c2);  
 b4.setBackground(c2);  
 b5.setBackground(c2);  
 b6.setBackground(c2);  
 b7.setBackground(c2);  
 b8.setBackground(c2);  
 j.add(b1);  
 j.add(b2);  
 j.add(b3);  
 j.add(b4);  
 q.add(l1);  
  
 q.add(b1);  
 q.add(b2);  
 q.add(b3);  
 q.add(b4);  
 q.add(b5);  
 q.add(b6);  
 q.add(b7);  
 q.add(b8);  
 b1.addActionListener(this);  
 b2.addActionListener(this);  
 b3.addActionListener(this);  
 b4.addActionListener(this);  
 b5.addActionListener(this);  
 b6.addActionListener(this);  
 b7.addActionListener(this);  
 b8.addActionListener(this);  
  
 }  
  
 public void actionPerformed(ActionEvent e)  
 {  
 if(e.getSource().equals(b5))  
 {  
 n1 = n1 + 1;  
 if(correct1.isSelected())  
 m1 = m1 + 10;  
 JOptionPane.*showMessageDialog*(null,"Player " + sss1 + " has currently " + m1 + " points\n" + n1 +" questions answered already");  
 }  
  
 if(e.getSource().equals(b8))  
 {  
 q.setVisible(false);  
 Page1 k=new Page1(f2,sss1);  
 k.marks(m1, n1);  
 }  
  
  
 if(e.getSource().equals(b6))  
 {  
 if(answer11.equals("A. Whatever programmer specifies")) {  
 //JOptionPane.showMessageDialog(null,"Player " + sss1 + " has currently " + m1 + " points\n" + n1 +" questions answered already");  
 }  
  
 else if(answer11.equals("A. Static")) {  
 q.setVisible(false);  
 String question1 = new String("1. The range of indices for an array always start at:");  
 String answer1 = "A. Whatever programmer specifies";  
 String answer2 = "B. 1";  
 String answer3 = "C. 0";  
 String answer4 = "D. Size of array";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer3);  
 }  
 else if(answer11.equals("A. 0")) {  
 q.setVisible(false);  
 String question1 = new String("2. Which of the following is not a keyword in java?");  
 String answer1 = "A. Static";  
 String answer2 = "B. Boolean";  
 String answer3 = "C. Void";  
 String answer4 = "D. Private";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer2);  
 }  
 else if(answer11.equals("A. Class variables are static variables within a class but outside any method.")) {  
 q.setVisible(false);  
 String question1 = new String("3. What is the default value of byte variable?");  
 String answer1 = "A. 0";  
 String answer2 = "B. 0.0";  
 String answer3 = "C. null";  
 String answer4 = "D. undefined";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 else if(answer11.equals("A. Methods with same name but different parameters.")) {  
 q.setVisible(false);  
 String question1 = new String("4. What is class variable?");  
 String answer1 = "A. Class variables are static variables within a class but outside any method.";  
 String answer2 = "B. Class variables are variables defined inside methods, constructors or blocks.";  
 String answer3 = "C. Class variables are variables within a class but outside any method.";  
 String answer4 = "D. None of the above.";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
  
 else if(answer11.equals("A. It is a Thread public static method used to obtain a reference to the current thread.")) {  
 q.setVisible(false);  
 String question1 = new String("5. What is function overloading?");  
 String answer1 = "A. Methods with same name but different parameters.";  
 String answer2 = "B. Methods with same name but different return types.";  
 String answer3 = "C. Methods with same name, same parameter types but different parameter names.";  
 String answer4 = "D. None of the above.";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
  
 else if(answer11.equals("A. Description")) {  
 q.setVisible(false);  
 String question1 = new String("6. What is currentThread()?");  
 String answer1 = "A. It is a Thread public static method used to obtain a reference to the current thread.";  
 String answer2 = "B. It is a thread's instance method used to get thread count.";  
 String answer3 = "C. It is a object's public static method used obtain a reference to the current thread.";  
 String answer4 = "D. It is a object's instance method used to get thread count.";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
  
 else if(answer11.equals("A. JApplet")) {  
 q.setVisible(false);  
 String question1 = new String("7. Which is not a part of defining an object?");  
 String answer1 = "A. Description";  
 String answer2 = "B. Methods";  
 String answer3 = "C. Associations with other objects";  
 String answer4 = "D. Name";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
  
 else if(answer11.equals("A. Public functions are free, you have to buy private ones")) {  
 q.setVisible(false);  
 String question1 = new String("8. Which one could be used as the main container in a Java application?");  
 String answer1 = "A. JApplet";  
 String answer2 = "B. JFrame";  
 String answer3 = "C. JPanel";  
 String answer4 = "D. JButton";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer2);  
 }  
  
 else if(answer11.equals("A. This")) {  
 q.setVisible(false);  
 String question1 = new String("9. What is the difference between private and public functions?");  
 String answer1 = "A. Public functions are free, you have to buy private ones";  
 String answer2 = "B. Public functions are the only ones you can download";  
 String answer3 = "C. Public functions can be used by anyone, private can only be used by other code in the class you are writing";  
 String answer4 = "D. Public functions can’t be used";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer3);  
 }  
 }  
  
  
  
 if(e.getSource().equals(b7))  
 {  
 if(answer11.equals("A. Whatever programmer specifies")) {  
 q.setVisible(false);  
 String question1 = new String("2. Which of the following is not a keyword in java?");  
 String answer1 = "A. Static";  
 String answer2 = "B. Boolean";  
 String answer3 = "C. Void";  
 String answer4 = "D. Private";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer2);  
 }  
  
 else if(answer11.equals("A. Static")) {  
 q.setVisible(false);  
 String question1 = new String("3. What is the default value of byte variable?");  
 String answer1 = "A. 0";  
 String answer2 = "B. 0.0";  
 String answer3 = "C. null";  
 String answer4 = "D. undefined";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 else if(answer11.equals("A. 0")) {  
 q.setVisible(false);  
 String question1 = new String("4. What is class variable?");  
 String answer1 = "A. Class variables are static variables within a class but outside any method.";  
 String answer2 = "B. Class variables are variables defined inside methods, constructors or blocks.";  
 String answer3 = "C. Class variables are variables within a class but outside any method.";  
 String answer4 = "D. None of the above.";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 else if(answer11.equals("A. Class variables are static variables within a class but outside any method.")) {  
 q.setVisible(false);  
 String question1 = new String("5. What is function overloading?");  
 String answer1 = "A. Methods with same name but different parameters.";  
 String answer2 = "B. Methods with same name but different return types.";  
 String answer3 = "C. Methods with same name, same parameter types but different parameter names.";  
 String answer4 = "D. None of the above.";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
 else if(answer11.equals("A. Methods with same name but different parameters.")) {  
 q.setVisible(false);  
 String question1 = new String("6. What is currentThread()?");  
 String answer1 = "A. It is a Thread public static method used to obtain a reference to the current thread.";  
 String answer2 = "B. It is a thread's instance method used to get thread count.";  
 String answer3 = "C. It is a object's public static method used obtain a reference to the current thread.";  
 String answer4 = "D. It is a object's instance method used to get thread count.";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
  
 else if(answer11.equals("A. It is a Thread public static method used to obtain a reference to the current thread.")) {  
 q.setVisible(false);  
 String question1 = new String("7. Which is not a part of defining an object?");  
 String answer1 = "A. Description";  
 String answer2 = "B. Methods";  
 String answer3 = "C. Associations with other objects";  
 String answer4 = "D. Name";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer1);  
 }  
  
 else if(answer11.equals("A. Description")) {  
 q.setVisible(false);  
 String question1 = new String("8. Which one could be used as the main container in a Java application?");  
 String answer1 = "A. JApplet";  
 String answer2 = "B. JFrame";  
 String answer3 = "C. JPanel";  
 String answer4 = "D. JButton";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer2);  
 }  
  
 else if(answer11.equals("A. JApplet")) {  
 q.setVisible(false);  
 String question1 = new String("9. What is the difference between private and public functions?");  
 String answer1 = "A. Public functions are free, you have to buy private ones";  
 String answer2 = "B. Public functions are the only ones you can download";  
 String answer3 = "C. Public functions can be used by anyone, private can only be used by other code in the class you are writing";  
 String answer4 = "D. Public functions can’t be used";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer3);  
 }  
  
 else if(answer11.equals("A. Public functions are free, you have to buy private ones")) {  
 q.setVisible(false);  
 String question1 = new String("10. What is the keyword used in java to create an object?");  
 String answer1 = "A. This";  
 String answer2 = "B. New";  
 String answer3 = "C. Sync";  
 String answer4 = "D. New()";  
 new Question(q, m1, n1, f2, sss1, question1, answer1, answer2, answer3, answer4, answer2);  
 }  
  
 else if(answer11.equals("A. This")) {  
 q.setVisible(false);  
 JOptionPane.*showMessageDialog*(null,"Saved successfully");  
 JOptionPane.*showMessageDialog*(null,"Player " + sss1 + " has reached " + m1 + " points");  
 int n = JOptionPane.*showConfirmDialog*(q, "Would you like to start another quiz?", "Quiz", JOptionPane.*YES\_NO\_OPTION*);  
 if(n == JOptionPane.*YES\_OPTION*)  
 new Quiz();  
 else q.setVisible(false);  
 }  
 }  
 }  
}