

# Online Examination

## CODE :

```
package oasis_tasks;

import javax.swing.*.*;
import java.awt.*.*;
import java.awt.event.*;
import java.lang.Exception;
import java.util.Timer;
import java.util.TimerTask;

class login extends JFrame implements ActionListener {
    JButton b1;
    JPanel newPanel;
    JLabel userLabel, passLabel;
    final JTextField textField1, textField2;

    login() {
        userLabel = new JLabel();
        userLabel.setText(" Username :");
        textField1 = new JTextField(15);
        passLabel = new JLabel();
        passLabel.setText(" Password :");
        textField2 = new JPasswordField(8);
        b1 = new JButton(" SUBMIT ");
        newPanel = new JPanel(new GridLayout(3, 1));
        newPanel.add(userLabel);
        newPanel.add(textField1);
        newPanel.add(passLabel);
        newPanel.add(textField2);
        newPanel.add(b1);
        add(newPanel, BorderLayout.CENTER);
        b1.addActionListener(this);
        setTitle("Login Form ");
    }

    public void actionPerformed(ActionEvent ae) {
        String userValue = textField1.getText();
        String passValue = textField2.getText();
        if (!passValue.equals(""))
            new OnlineTestBegin(userValue);
        else {
            textField2.setText("Enter Password");
            actionPerformed(ae);
        }
    }
}
```

```

class OnlineTestBegin extends JFrame implements ActionListener {
    JLabel l;
    JLabel l1;
    JRadioButton jb[] = new JRadioButton[6];
    JButton b1, b2, log;
    ButtonGroup bg;
    int count = 0, current = 0, x = 1, y = 1, now = 0;
    int m[] = new int[10];
    Timer timer = new Timer();

    OnlineTestBegin(String s) {
        super(s);
        l = new JLabel();
        l1 = new JLabel();
        add(l);
        add(l1);
        bg = new ButtonGroup();
        for (int i = 0; i < 5; i++) {
            jb[i] = new JRadioButton();
            add(jb[i]);
            bg.add(jb[i]);
        }
        b1 = new JButton("Save and Next");
        b2 = new JButton("Save for later");
        b1.addActionListener(this);
        b2.addActionListener(this);
        add(b1);
        add(b2);
        set();
        l.setBounds(30, 40, 450, 20);
        l1.setBounds(20, 20, 450, 20);
        jb[0].setBounds(50, 80, 100, 20);
        jb[1].setBounds(50, 110, 100, 20);
        jb[2].setBounds(50, 140, 100, 20);
        jb[3].setBounds(50, 170, 100, 20);
        b1.setBounds(95, 240, 140, 30);
        b2.setBounds(270, 240, 150, 30);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(null);
        setLocation(250, 100);
        setVisible(true);
        setSize(600, 350);
        timer.scheduleAtFixedRate(new TimerTask() {
            int i = 600;

            public void run() {
                l1.setText("Time left: " + i);
                i--;
                if (i < 0) {
                    timer.cancel();
                    l1.setText("Time Out");
                }
            }
        }, 0, 600);
    }
}

```

```

        }
    }
    }, 0, 1000);
}

public void actionPerformed(ActionEvent e) {
    if (e.getSource() == b1) {
        if (check())
            count = count + 1;
        current++;
        set();
        if (current == 9) {
            b1.setEnabled(false);
            b2.setText("Result");
        }
    }
    if (e.getActionCommand().equals("Save for later")) {
        JButton bk = new JButton("Review" + x);
        bk.setBounds(480, 20 + 30 * x, 100, 30);
        add(bk);
        bk.addActionListener(this);
        m[x] = current;
        x++;
        current++;
        set();
        if (current == 9)
            b2.setText("Result");
        setVisible(false);
        setVisible(true);
    }
    for (int i = 0, y = 1; i < x; i++, y++) {
        if (e.getActionCommand().equals("Review" + y)) {
            if (check())
                count = count + 1;
            now = current;
            current = m[y];
            set();
            ((JButton) e.getSource()).setEnabled(false);
            current = now;
        }
    }
    if (e.getActionCommand().equals("Result")) {
        if (check())
            count = count + 1;
        current++;
        JOptionPane.showMessageDialog(this, "Score =" + count);
        System.exit(0);
    }
}

void set() {
    jb[4].setSelected(true);
}

```

```

        if (current == 0) {
            l.setText("Que1: Who is known as father of java programming
language?");

            jb[0].setText("charles Babbage");
            jb[1].setText("James Gosling");
            jb[2].setText("M.P.Java");
            jb[3].setText("Blais Pascal");
        }
        if (current == 1) {
            l.setText("Que2: Number of primitive data types in java are?");
            jb[0].setText("6");
            jb[1].setText("7");
            jb[2].setText("8");
            jb[3].setText("9");
        }
        if (current == 2) {
            l.setText("Que3: Where is system class defined?");
            jb[0].setText("java.lang.package");
            jb[1].setText("java.util.package ");
            jb[2].setText("java.io.package");
            jb[3].setText("None");
        }
        if (current == 3) {
            l.setText("Que4: Expected created by try block is caaught in which
block.?");

            jb[0].setText("catch");
            jb[1].setText("throw");
            jb[2].setText("final");
            jb[3].setText("thrown");
        }
        if (current == 4) {
            l.setText("Que5: Which of the following is not an OOPS concept in
java?");

            jb[0].setText("Polymorphism");
            jb[1].setText("Inheritance");
            jb[2].setText("Compilation");
            jb[3].setText("Encapsulation");
        }
        if (current == 5) {
            l.setText("Que6: Identify the infinite loop?");
            jb[0].setText("for(;;)");
            jb[1].setText("for(i=0;j<1;i--)");
            jb[2].setText("for(int=0;i++)");
            jb[3].setText("if(All of the above)");
        }
        if (current == 6) {
            l.setText("Que7: When is the finalize()method called ");
            jb[0].setText("Before garbage collection");
            jb[1].setText("Before an object goes out of scope");
            jb[2].setText("Before a variable goes out of scope");
            jb[3].setText("None");
        }
    }

```

```

        if (current == 7) {
            l.setText("Que8: What is the implicit return type of constructor?");
            jb[0].setText("No return type");
            jb[1].setText("A class object in which it is defined");
            jb[2].setText("void");
            jb[3].setText("None");
        }
        if (current == 8) {
            l.setText("Que9: The class at the top of exception class is....?");
            jb[0].setText("ArithmeticException");
            jb[1].setText("Throwable");
            jb[2].setText("Object");
            jb[3].setText("Console");
        }
        if (current == 9) {
            l.setText("Que10: Which provides runtime enviroment for java byte
code to be executed?");
            jb[0].setText("JDK");
            jb[1].setText("JVM");
            jb[2].setText("JRE");
            jb[3].setText("JAVAC");
        }
        l.setBounds(30, 40, 450, 20);
        for (int i = 0, j = 0; i <= 90; i += 30, j++)
            jb[j].setBounds(50, 80 + i, 200, 20);
    }

    boolean check() {
        if (current == 0)
            return (jb[1].isSelected());
        if (current == 1)
            return (jb[1].isSelected());
        if (current == 2)
            return (jb[2].isSelected());
        if (current == 3)
            return (jb[0].isSelected());
        if (current == 4)
            return (jb[2].isSelected());
        if (current == 5)
            return (jb[3].isSelected());
        if (current == 6)
            return (jb[1].isSelected());
        if (current == 7)
            return (jb[3].isSelected());
        if (current == 8)
            return (jb[2].isSelected());
        if (current == 9)
            return (jb[2].isSelected());
        return false;
    }
}

```

```
class OnlineExam {  
    public static void main(String args[]) {  
        try {  
            login form = new login();  
            form.setSize(400, 150);  
            form.setVisible(true);  
        } catch (Exception e) {  
            JOptionPane.showMessageDialog(null, e.getMessage());  
        }  
    }  
}
```