# CSY2030
# Systems Design & Development

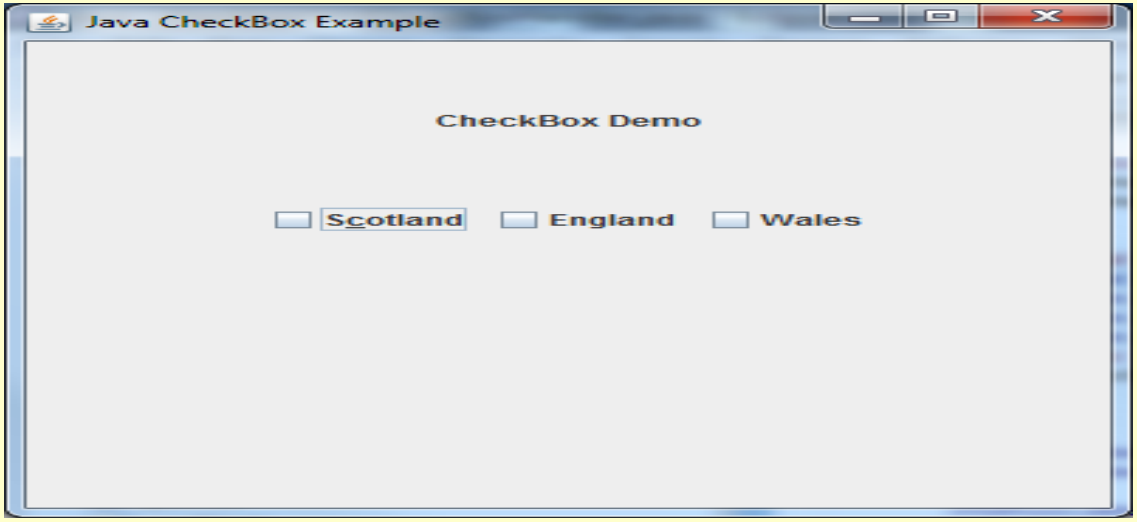
# Graphical User Interfaces 2

# Outline of Lecture

- Today we will cover the following:
  - **Check Boxes**
  - **Radio Buttons**
  - **Drop-Down Lists**

# Check Boxes

- A check box is an item that can be selected or deselected, and which displays its state to the user.
- It is achieved using the **JCheckBox** class in java
- We will now present a GUI with 3 check boxes to select or deselect i.e



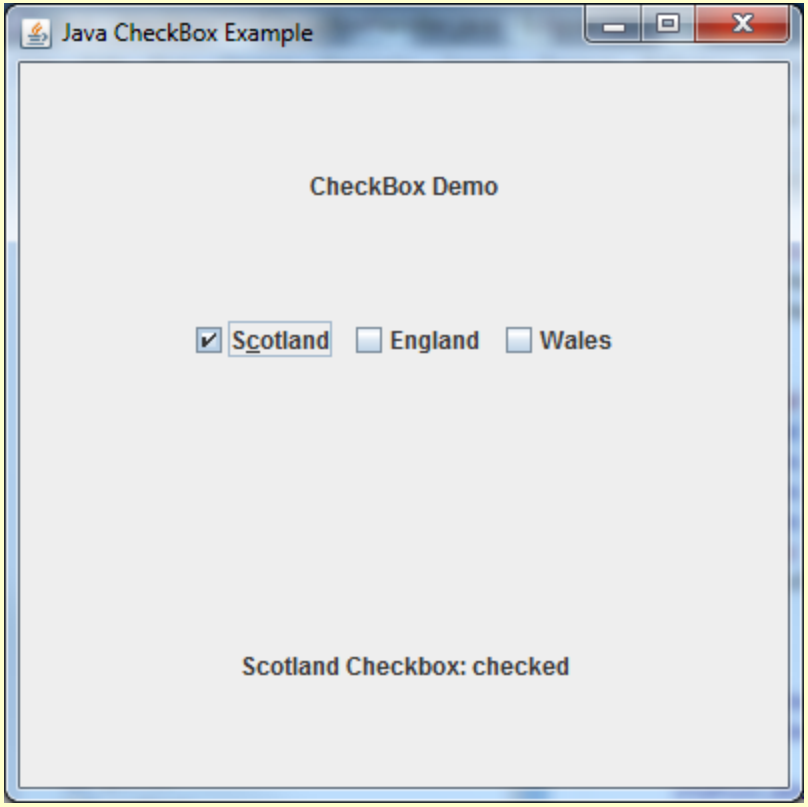

- Code for above GUI will be over next 3 slides

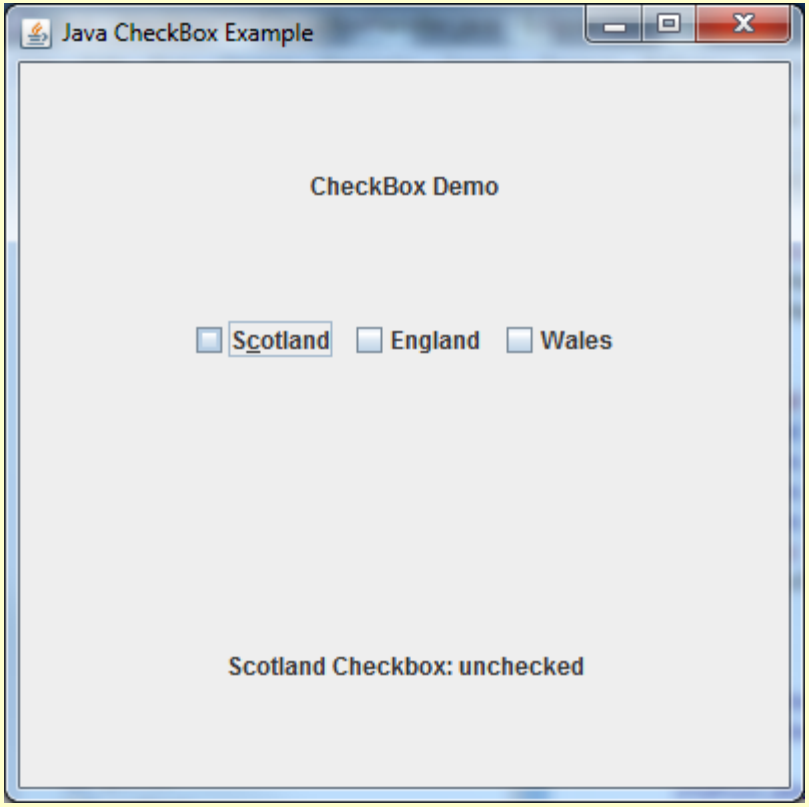
```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class CheckBoxHandler {

        private JFrame mainFrame;           // Window (application) to add components to
        private JLabel headerLabel;         // Top part (row) of window
        private JPanel controlPanel;        // Middle part (row) of window
        private JLabel statusLabel;         // Bottom part (row) of window

        public CheckBoxHandler(){
                prepareGUI();               // call prepareGUI() method
        }

        public static void main(String[] args){
                // run application
                CheckBoxHandler  CheckBoxHandler = new CheckBoxHandler();
                CheckBoxHandler.showCheckBoxDemo();
        }
```

```
private void prepareGUI(){
        mainFrame = new JFrame("Java CheckBox Example");    // give application a title
        mainFrame.setSize(400,400);                         // give application a size
        mainFrame.setLayout(new GridLayout(3, 1));          // application has 3 rows + 1 columns

       // make application receive window events
       mainFrame.addWindowListener(new WindowAdapter() {
                public void windowClosing(WindowEvent windowEvent){
                  System.exit(0);
                }
        });

        headerLabel = new JLabel("", JLabel.CENTER);        // set up top row of application

        statusLabel = new JLabel("",JLabel.CENTER);          // set up bottom row of application
        statusLabel.setSize(350,100);                       // give bottom row a size

        controlPanel = new JPanel();                    // set up middle row of application
        controlPanel.setLayout(new FlowLayout());       // make middle row flow from left to right

        mainFrame.add(headerLabel);       // add top row to application
        mainFrame.add(controlPanel);      // add middle row to application
        mainFrame.add(statusLabel);       // add bottom row to application
        mainFrame.setVisible(true);       // display the 3 rows

}
```

```
  private void showCheckBoxDemo(){

        headerLabel.setText("CheckBox Demo");          // print 'CheckBox Demo' in top row of application

        final JCheckBox chkScotland = new JCheckBox("Scotland");   // Create a Check Box labelled 'Scotland'
        final JCheckBox chkEngland = new JCheckBox("England");     // Create a Check Box labelled 'England'
        final JCheckBox chkWales = new JCheckBox("Wales");         // Create a Check Box labelled 'Wales'

       // if Scotland check box ticked then print 'Scotland Checkbox checked'; if it is then unticked then print 'Scotland Checkbox unchecked'
       chkScotland.addItemListener(new ItemListener() {
                              public void itemStateChanged(ItemEvent e) {
                                     statusLabel.setText("Scotland Checkbox: "  + (e.getStateChange()==1?"checked":"unchecked"));
          }
        });

       // if England check box ticked then print 'England Checkbox checked'; if it is then unticked then print 'England Checkbox unchecked'
       chkEngland.addItemListener(new ItemListener() {
             public void itemStateChanged(ItemEvent e) {
                    statusLabel.setText("England Checkbox: " + (e.getStateChange()==1?"checked":"unchecked"));
          }
        });

       // if Wales check box ticked then print 'Wales Checkbox checked'; if it is then unticked then print 'Wales Checkbox unchecked'
       chkWales.addItemListener(new ItemListener() {
             public void itemStateChanged(ItemEvent e) {
                    statusLabel.setText("Wales Checkbox: "  + (e.getStateChange()==1?"checked":"unchecked"));
          }
        });

        controlPanel.add(chkScotland);   // add 'Scotland' check box to middle row
        controlPanel.add(chkEngland);    // add 'England' check box to middle row
        controlPanel.add(chkWales);      // add 'Wales' check box to middle row

        mainFrame.setVisible(true);       // display the 3 rows

      }
}
```

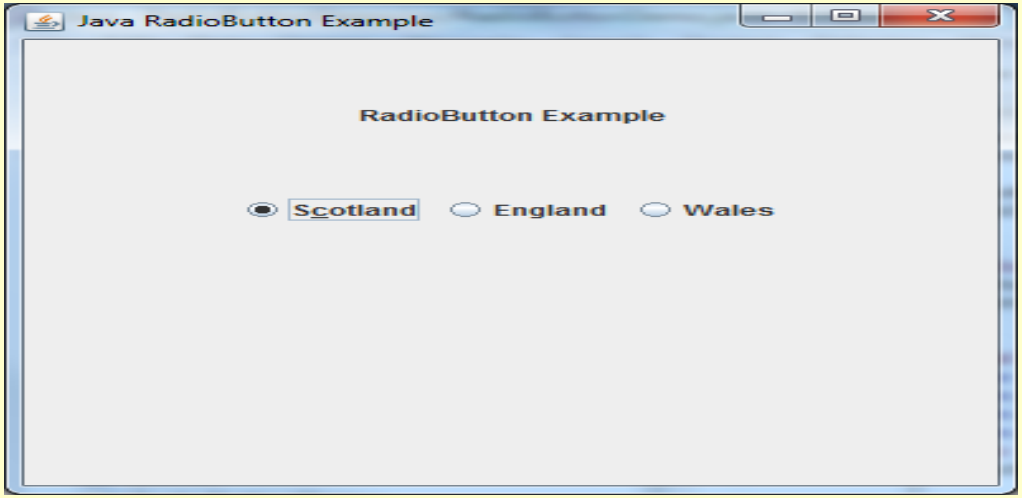
# CheckBox GUI Output



# Radio Buttons

- Like checkboxes, a radio box is an item that can be selected or deselected, and which displays its state to the user.
- It is achieved using the **JRadioButton** class in java
- We will now present a GUI with 3 radio buttons to select or deselect i.e



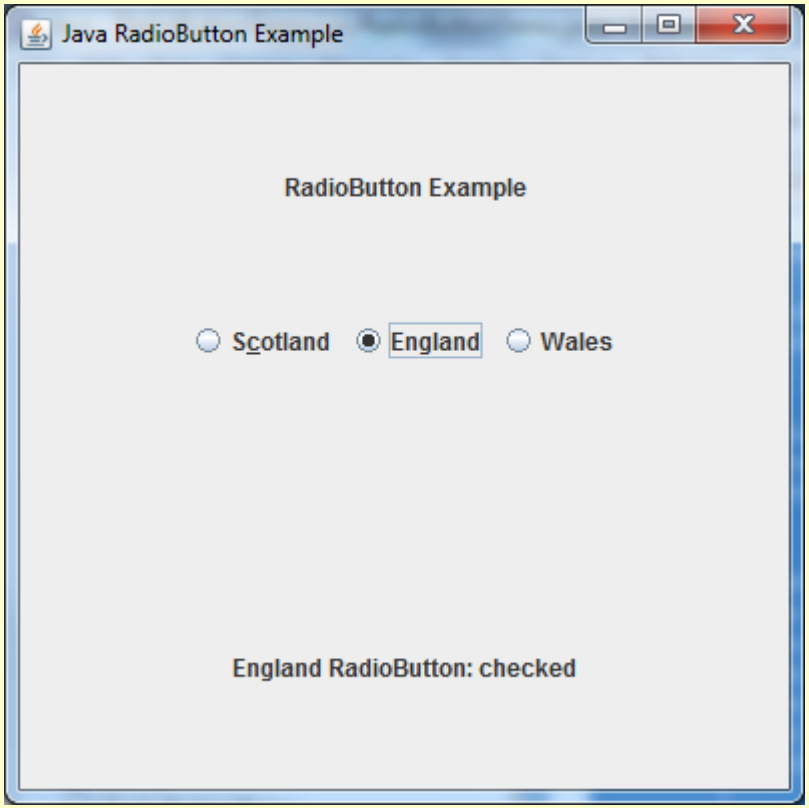

- Code for above GUI will be over next 3 slides

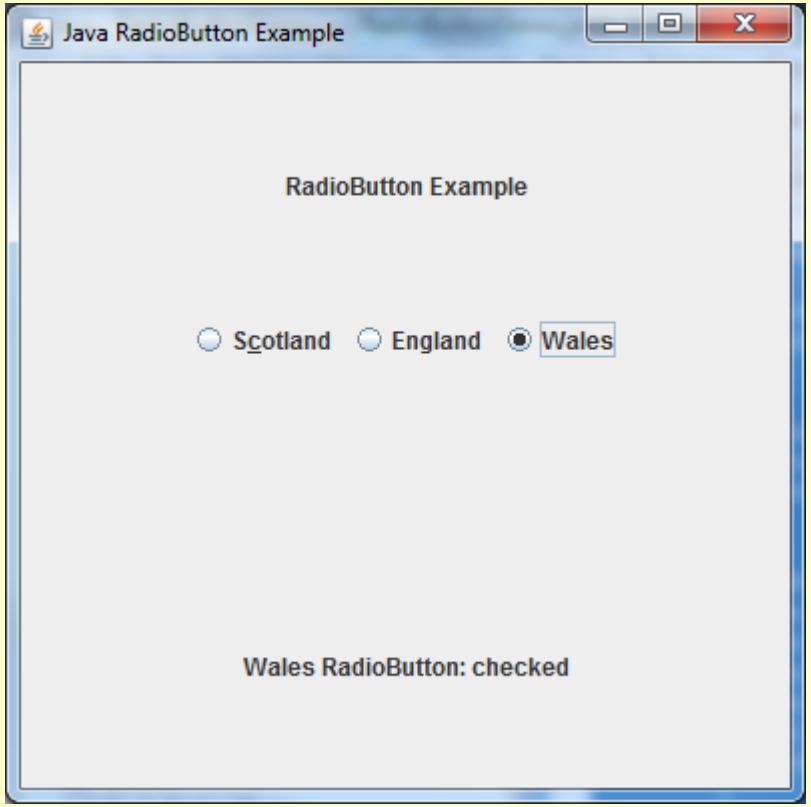
```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class RadioButtonDemo {

        private JFrame mainFrame;           // Window (application) to add components to
        private JLabel headerLabel;         // Top part (row) of window
        private JPanel controlPanel;        // Middle part of window
        private JLabel statusLabel;         // Bottom part (row) of window

        public RadioButtonDemo(){
                prepareGUI();      // call prepareGUI() method
        }

        public static void main(String[] args){
                // run application
                RadioButtonDemo  jradiobuttondemo = new RadioButtonDemo();
                jradiobuttondemo.showRadioButtonDemo();
        }
```

```
private void prepareGUI(){
        mainFrame = new JFrame("Java RadioButton Example"); // give application a title
        mainFrame.setSize(400,400);         // give application a size
        mainFrame.setLayout(new GridLayout(3, 1)); // application has 3 rows + 1 columns

        // make application receive window events
        mainFrame.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent windowEvent){
                                System.exit(0);
             }
        });

        headerLabel = new JLabel("", JLabel.CENTER); // set up top row of application

        statusLabel = new JLabel("",JLabel.CENTER); // set up bottom row of application
        statusLabel.setSize(350,100);         // give bottom row a size

        controlPanel = new JPanel();          // set up middle row of application
        controlPanel.setLayout(new FlowLayout()); // make middle row flow from left to right

        mainFrame.add(headerLabel); // add top row to application
        mainFrame.add(controlPanel); // add middle row to application
        mainFrame.add(statusLabel); // add bottom row to application
        mainFrame.setVisible(true); // display the 3 rows
}
```

```
private void showRadioButtonDemo(){
      headerLabel.setText("RadioButton Example");            // print 'RadioButton Demo' in top row of application

      final JRadioButton radScotland = new JRadioButton("Scotland", true);     // Create a Radio Button  labelled 'Scotland' and check it
      final JRadioButton radEngland = new JRadioButton("England");             // Create a Radio Button  labelled 'England'
      final JRadioButton radWales = new JRadioButton("Wales");                 // Create a Radio Button  labelled 'Wales'

     // if Scotland radio button checked print 'Scotland RadioButton checked'; if it is then unchecked print 'Scotland RadioButton  unchecked'
     radScotland.addItemListener(new ItemListener() {
          public void itemStateChanged(ItemEvent e) {
                    statusLabel.setText("Scotland RadioButton: " + (e.getStateChange()==1?"checked":"unchecked"));
       }
      });

    // if England radio button checked print 'England RadioButton checked'; if it is then unchecked print 'England RadioButton  unchecked'
     radEngland.addItemListener(new ItemListener() {
          public void itemStateChanged(ItemEvent e) {
                    statusLabel.setText("England RadioButton: " + (e.getStateChange()==1?"checked":"unchecked"));
       }
      });

    // if Wales radio button checked print 'Wales RadioButton checked'; if it is then unchecked print 'Wales RadioButton  unchecked'
      radWales.addItemListener(new ItemListener() {
           public void itemStateChanged(ItemEvent e) {
                    statusLabel.setText("Wales RadioButton: "  + (e.getStateChange()==1?"checked":"unchecked"));
       }
      });

     controlPanel.add(radScotland);  // add 'Scotland' radio button to middle row
     controlPanel.add(radEngland); // add 'England' radio button to middle row
     controlPanel.add(radWales); // add 'Wales' radio button to middle row

     mainFrame.setVisible(true); // display the 3 rows
   }
}
```

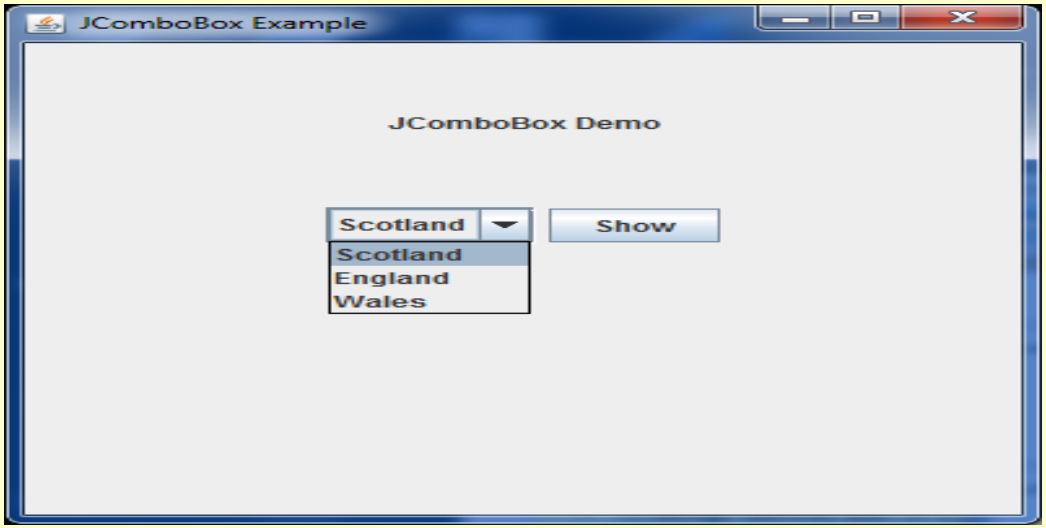
# Radio Button GUI Output



# Drop-Down Lists

- A **drop-down list** is a graphical control element, that allows the user to choose one value from a list
- It is achieved using the **JComboBox** class in java
- We will now present a GUI with a drop down list of 3 countries  i.e




- Code for above GUI will be over next 3 slides

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class ComboBoxDemo {

        private JFrame mainFrame;       // Window (application) to add components to
        private JLabel headerLabel;     // Top part (row) of window
        private JLabel statusLabel;     // Middle part of window
        private JPanel controlPanel;    // Bottom part (row) of window

        public ComboBoxDemo(){
                prepareGUI();           // call prepareGUI() metho
        }

         public static void main(String[] args){
                // run application
                ComboBoxDemo  swingControlDemo = new ComboBoxDemo();
                swingControlDemo.showComboboxDemo();
        }
```

```
private void prepareGUI(){
        mainFrame = new JFrame("JComboBox Example");      // give application a title
        mainFrame.setSize(400,400);                       // give application a size
        mainFrame.setLayout(new GridLayout(3, 1));        // application has 3 rows + 1 column

        // make application receive window events
        mainFrame.addWindowListener(new WindowAdapter() {
                public void windowClosing(WindowEvent windowEvent){
                        System.exit(0);
                }
        });

        headerLabel = new JLabel("", JLabel.CENTER);      // set up top row of application

        statusLabel = new JLabel("", JLabel.CENTER);      // set up bottom row of application
        statusLabel.setSize(350,100);                     //give bottom row a size


        controlPanel = new JPanel();                  // set up middle row of application
        controlPanel.setLayout(new FlowLayout());  // make middle row flow from left to right

        mainFrame.add(headerLabel);      // add top row to application
        mainFrame.add(controlPanel);     // add middle row to application
        mainFrame.add(statusLabel);      // add bottom row to application
        mainFrame.setVisible(true);      // display the 3 rows
  }
```

```
private void showComboboxDemo(){
        headerLabel.setText("JComboBox Demo");          // print 'JCombList Demo' in top row of application

        final DefaultComboBoxModel countryList = new DefaultComboBoxModel();          // create list object

        countryList.addElement("Scotland");  // add "Scotland" to drop-down list
        countryList.addElement("England");   // add "England" to drop-down list
        countryList.addElement("Wales");     // add "Wales" to drop-down list

        final JComboBox countryCombo = new JComboBox(countryList);          // add list to JComboBox object
        countryCombo.setSelectedIndex(0);                                   // show 1st element of list

        JScrollPane countryListScrollPane = new JScrollPane(countryCombo);      // create scrollable view of list

        JButton showButton = new JButton("Show");       // create button and label it "Show"

        showButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String data = "";
                if (countryCombo.getSelectedIndex() != -1) {
                    data = "Country Selected: " + countryCombo.getItemAt(countryCombo.getSelectedIndex());
            }
            statusLabel.setText(data);
          }
        });

        controlPanel.add(countryListScrollPane);          // add drop-down list to middle row
        controlPanel.add(showButton);                     // add button to middle row
        mainFrame.setVisible(true);                       // display the 3 rows
    }
}
```

# Drop-Down Lists GUI Output