question, assistive technology will render the text of the question when each button is selected. This ensures that the action of the form control is clear and that a unique text representation is provided for each form field.

Form element groupings must be implemented *programmatically*; merely using visual indications, such as lines or rectangles, on the form is not sufficient. In Flex, this is done using a MX FormHeading, FormItem, or Panel container. Developers should choose the container that is best suited to the type of components in the group. The FormHeading component displays a title above the grouped controls, which is then automatically associated with the children elements via the container's accessible name. The FormHeading component should be used to group components other than radio buttons and check boxes.

The FormItem component should be used to group radio button and check boxes, because it provides a visual group label to the left of radio buttons and check boxes, which have their own labels positioned to the right. One FormItem component should enclose the entire set of radio buttons or check boxes. When radio buttons are used, developers must also define the `groupName` property and a corresponding RadioButtonGroup control to ensure that the radio buttons work together as a group. The following example illustrates the use of FormHeading and FormItem components to group form elements:

```
// FormHeading Example, the text "Registration" will be prepended to the
     accessible name of all fields within the form
<mx:Form>
  <mx:FormHeading label="Registration" />
  <mx:FormItem id="f1" color="#ff0000" required="true" label="Enter Name :">
    <mx:TextInput id="txt1" />
  </mx:FormItem>
  <mx:FormItem label="Enter Password :">
    <mx:TextInput displayAsPassword="true"/>
  </mx:FormItem>
  <mx:FormItem label="Select Gender:">
    <s:ComboBox id="c1" width="150">
      <s:dataProvider>
        <mx:ArrayList>
          <fx:String>Male</fx:String>
          <fx:String>Female</fx:String>
        </mx:ArrayList>
      </s:dataProvider>
    </s:ComboBox>
  </mx:FormItem>
</mx:Form>
<s:Button label="Submit" />

// FormItem example showing proper radio button grouping using
     FormItem and RadioButtonGroup with the groupName attribute
<mx:FormItem label="Do you want to receive emails?:">
  <s:RadioButtonGroup id="RadioGroup1" />
  <s:RadioButton label="Yes" groupName="RadioGroup1" />
  <s:RadioButton label="No" groupName="RadioGroup1" />
</mx:FormItem>
```

In some cases, it may be impractical to provide a visual grouping for multiple form elements. When this is the case, developers should include the name of the group in each element's `accessibleName` property. Such grouping significantly increases the usability of forms for screen reader users. It may be preferable to state the answer first followed by the question in the radio button's accessible name. This enables the user to quickly answer without having to fully listen to the query for each element. This is particularly helpful when there are many radio buttons in the group and the group name (question) is constant. Users can listen to the full accessible name for the first radio button, and proceed through the remaining buttons quickly to select the appropriate answer.