asterisk" should be included here. Instructive text that is specific to any field in the form should be located immediately before the field in the reading order.

```
//ActionScript
txtInstructions.text = "Please complete all fields and activate
    confirm to review your entries.";
txtInstructions.tabIndex = 10;
// form fields following with higher tabIndex values
txtSSNError.text = "Do not enter dashes";
txtSSNError.tabIndex = 15;
txtSSN.tabIndex = 16;

//MXML
<s:Text id="txtInstructions" text="Please complete all fields and
    activate confirm to review your entries." tabIndex="10" />
<!-- form fields following with higher tabIndex values -->
<s:Text id="txtSSNError" text="Do not enter dashes" tabIndex="15" />
<s:TextInput id="txtSSN" tabIndex="16" />
```

Error messages

After the form is submitted, any error messages should be provided in keyboard-focusable text at the beginning of the form's tab order. Error text provided elsewhere may not be immediately apparent to users of assistive technology, leading them to resubmit the form with the same errors present. Error messages should include both an indication that the overall form is in error as well as a list of the specific fields in error and reasons for the error state of each field.

Error messages may be displayed in red (or any color that stands out), but they should also start with the word "Error." Color cannot be the only means of indicating an error. See "Color" on page 44 for more details.

Note: If possible, when form validation errors are detected after submission, move the keyboard focus automatically to the overview error message positioned at the top of the form. This is an exception to the general best practice of not moving keyboard focus automatically, described in "Focus" on page 39. This approach immediately informs users of screen readers and screen magnifiers that form errors exist, so they will not assume the form has been successfully submitted. Use `setFocus()` to set the focus to the error text; for example:

```
btnSubmit.addEventListener(MouseEvent.CLICK, validateForm);

function validateForm(e: MouseEvent): void
{
 var errorText = new Label();
 // positioning of label not shown
 addChild(errorText);
 errorText.tabIndex = 10;
 errorText.text = "Error: Please provide your last name.";
 errorText.setFocus();  // set focus to the error text
}
```

Extra steps should be taken to prevent form entry errors when the data being entered is part of a legal commitment or financial arrangement. In general for these transactions, an accessible application must provide at least one of three methods for avoiding and mitigating these errors: allow the submission to be reversible, allow the user to confirm the data entered, or check the data for errors before completing the submission.

Field specific errors

When field-specific error information is provided, the text should include any known suggestions on how to correct the form if providing a suggestion would not invalidate the purpose of the field. The suggestion text may include specific examples or the format of information that is required. These directions should be