

The example below illustrates how the `FocusableLabel` class defined above can be used:

```
//ActionScript
import FocusableLabel;
...
var txtTimeLeft: FocusableLabel = new FocusableLabel();
txtTimeLeft.text = String(getTimeLeft()); // getTimeLeft function not shown
// positioning of txtTimeLeft not shown
addChild(txtTimeLeft);
txtTimeLeft.tabIndex = 13; // set the tab order for this element
```

Note: As described in "Controlling dynamic and automatically updating content" on page 14, developers should provide keyboard-accessible controls that allow the user to pause, hide, or stop automatically updating text.

Rollover content

Meaningful content that appears as the mouse rolls over an element must also be accessible via the keyboard. Generally, this is achieved by ensuring that the user can tab to the item that activates the rollover, that Enter or Space can activate the rollover content, and that the now visible rollover content itself can be reached via the Tab key even if the content is not actionable.

If the rollover content is not extensive, it can be placed in an accessibility property (for example, `accessibilityDescription`), which will enable users to access it with assistive technology without having to activate the rollover. Support for triggering the rollover via the keyboard is still required, however, as this property is not readily available to users who do not have screen readers. The following example illustrates how an event listener can be used to show rollover content:

```
btnActuator1.addEventListener(MouseEvent.CLICK, showRollOver);
btnActuator1.tabIndex = 10;
btnActuator2.addEventListener(MouseEvent.CLICK, showRollOver);
btnActuator1.tabIndex = 12;
btnActuator3.addEventListener(MouseEvent.CLICK, showRollOver);
btnActuator1.tabIndex = 14;
```

```
function showRollOver(e:MouseEvent): void
{
    switch (e.target)
    {
        case btnActuator1:
        {
            lblRollOverContent.text = "Content for Rollover 1";
            lblRollOverContent.visible = true;
            lblRollOverContent.tabIndex = 11;
            lblRollOverContent.setFocus();
        }
        case btnActuator2:
        {
            lblRollOverContent.text = "Content for Rollover 2";
            lblRollOverContent.visible = true;
            lblRollOverContent.tabIndex = 13;
            lblRollOverContent.setFocus();
            // display the rollover content for the second actuator
        }
        case btnActuator3:
        {
            // display the rollover content for the third actuator
        }
    }
}
```