

IN THE CONTAINER/PARENT SWF (frame 1):

```
function loadSWF1(e:MouseEvent):void {
    myLoader.load(new URLRequest("one.swf"));
    addChild(myLoader);
    myLoader.addEventListener("killMe", killLoadedClip);
}

function killLoadedClip(e:Event):void {
    myLoader.removeEventListener("killMe", killLoadedClip);
    removeChild(myLoader);
}
```

CHILD SWF: TWO OPTIONS BELOW—CHOOSE ONLY ONE OF THESE!!!

Option 1:

ON LAST *_FRAME_* IN CHILD SWF:

```
stop();
parent.dispatchEvent(new Event("killMe"));
```

Use this only if there is some animation sequence after the Close or Return to Intro button is clicked.

Option 2:

FUNCTION FOR CLOSE BUTTON IN CHILD SWF:

```
function closeMe(e:MouseEvent):void {
    parent.dispatchEvent(new Event("killMe"));
}
```

Use this if the Close or Return to Intro button needs to do nothing else but unload the SWF it's in..

What It Does

This was new to me. We are setting up a listener as soon as we load any of the external SWFs. The listener listens for an event that sends the text “killMe” (brutal, I know). When such an event takes place, the container SWF will run the function named *killLoadedClip*.

The loaded SWF (the “child”) must refer to the parent (which means the one that loaded it). This is because myLoader exists only in the parent. Also, the function *killLoadedClip* and the listener are up in the parent. This makes it a little tricky.