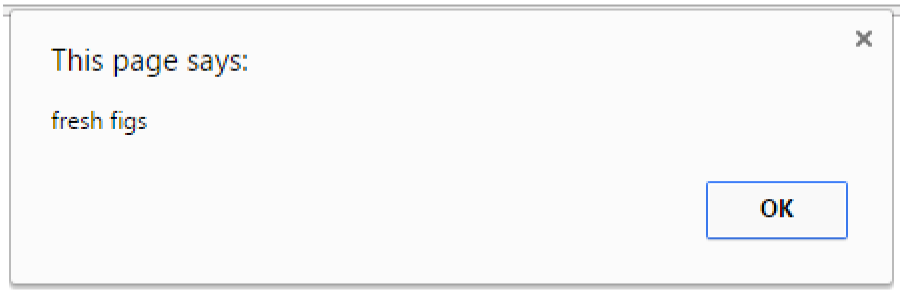
JavaScript Events Lab 2

In this exercise event handlers will be created on nodes to handle event flow for bubbling and capturing

Exercise 1 – Event Bubbling

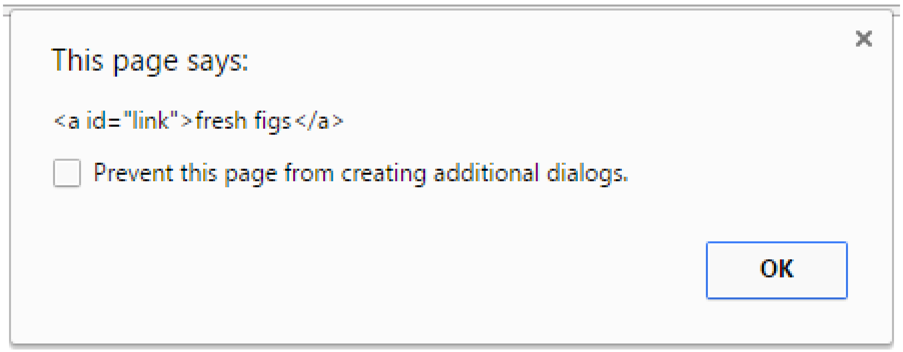
The steps outlined below will help you complete the exercise:

1. On moodle download the zip file, event-flow-bubble.zip, and extract the file to a location of your choice on your device.
2. Open the file event-flow-bubble.html, and js/event-flow-bubble.js in a text editor of your choice.
3. Review the code in js/event-flow-bubble.js. Note that event listeners exist for the <a> and <li> elements.
4. Open event-flow-bubble.html in a browser of your choice.
5. Click on the link for “great grand child link”.
6. Note the popup messages show the inner html for <a> element and then for <li> element. This is the bubble effect. The events flow outwards.
7. Now modify the code in js/event-flow-bubble.js to ensure event listeners exist for all the parent nodes all the way up to the <div> element.
8. Reuse the function showElement, for any new event listeners added to js/event-flow-bubble.js.
9. Save all changes to js/event-flow-bubble.js and reload the web page event-flow-bubble.html.
10. Click on the link for “fresh figs”.
11. Please ensure 4 alert messages appear, the first displaying the inner html for <a> and the last showing the inner html for <div>. See below:



This page says:

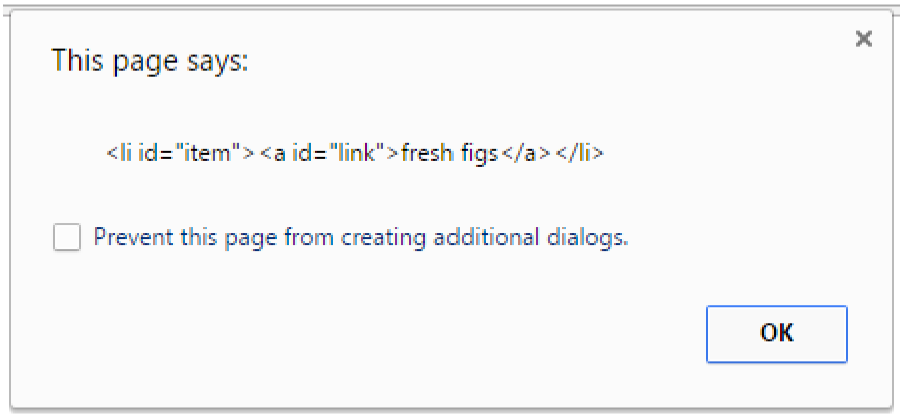
fresh figs



This page says:

<a id="link">fresh figs</a>

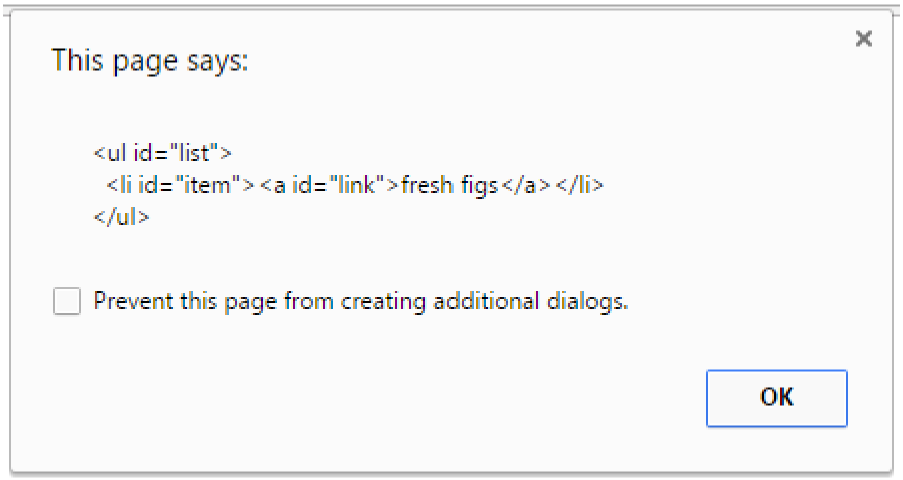
□ Prevent this page from creating additional dialogs.



This page says:

```
<li id="item"><a id="link">fresh figs</a></li>
```

☐ Prevent this page from creating additional dialogs.



This page says:

```
<ul id="list">
  <li id="item"><a id="link">fresh figs</a></li>
</ul>
```

☐ Prevent this page from creating additional dialogs.

Exercise 2 – Event Capturing

The steps outlined below will help you complete the exercise:

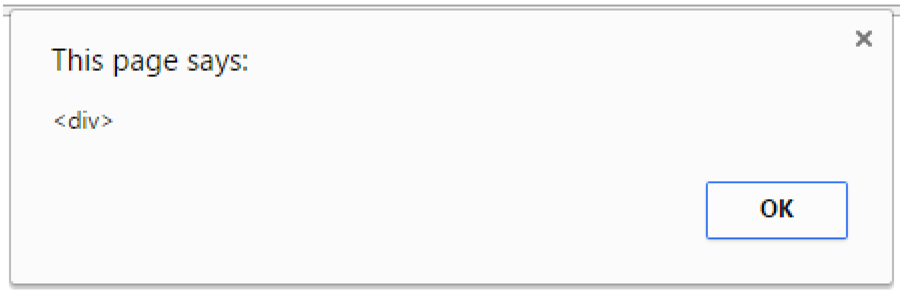
1. On moodle download the zip file, event-flow-capture.zip, and extract the details to a location of your choice on your device.
2. Open the file event-flow-capture.html, and js/event-flow-capture.js in a text editor of your choice.
3. Note that no code exists in js/event-flow-capture.js.
4. Write code which attaches an event listener to each of the elements from <div>, all the way down to <a>, so that event flows inwards (event capturing).
5. An example for such an event listener:

   ```
   el.addEventListener('click', showElement, false);
   ```

6. Instructions / Requirements for completing this exercise:
   a. The property for flow, must be updated to true.
   b. Need to create a function which accepts a string parameter, and it must be called anonymously. E.g. function(){showElement("<a>");}.
   c. The parameter passed to the function must be the element name for which the event listener is attached to.

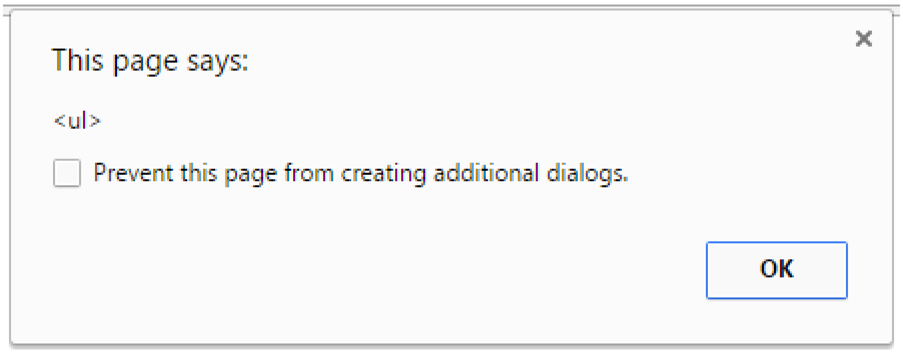
   d. The function then must contain an alert, which displays the parameter that was passed to it.

7. Save all changes to js/event-flow-capture.js and open the web page event-flow-capture.html in a browser of your choice.
8. Click on the link for “fresh figs”.
9. Please ensure 4 alert messages appear, the first displaying the element <div> and the last showing the element <a>. See below:



This page says:

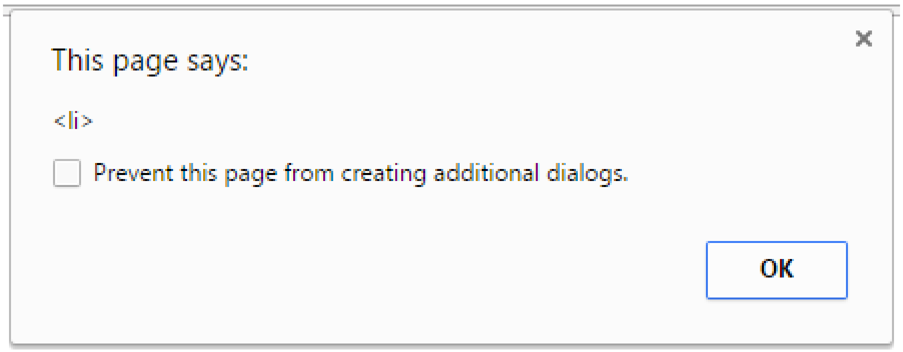
<div>



This page says:

<ul>

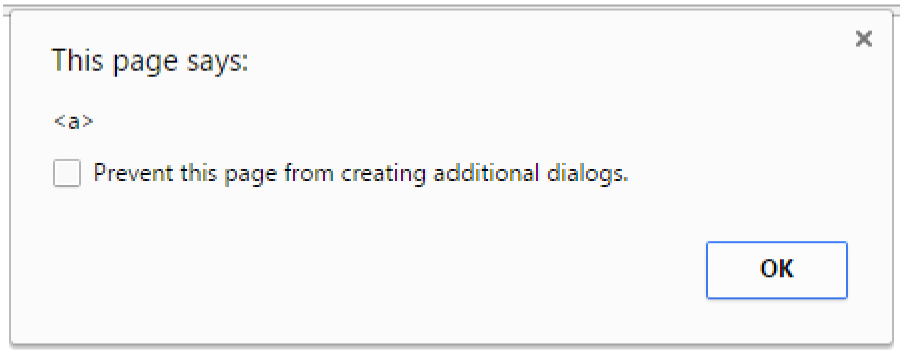
□ Prevent this page from creating additional dialogs.



This page says:

<li>

□ Prevent this page from creating additional dialogs.



This page says:

<a>

□ Prevent this page from creating additional dialogs.