

COM1008: Web and Internet Technology

Exercise sheet 6: Forms, user interfaces and events

Dr. Steve Maddock, s.maddock@sheffield.ac.uk

Aims

The aim of this exercise sheet is to (i) practise JavaScript programming, (ii) practise using HTML forms; (iii) practise handling events on a simple user interface.

1. The tasks

1.1 Forms

Exercise 1: Create a form that can be used to e-mail a person's name and a text comment to you. ('form2.html' in the downloads for lecture 12 can be used as a starting point for this exercise.) (**Important:** You will need to use your own sheffield e-mail address in the form.)

Exercise 2: (This exercise follows on from exercise 1 and involves changing the text field into a field containing a set of checkboxes.) Create a form that inputs a person's name, and their likes and dislikes from a list that is presented to them, e.g. the list might contain a list of foods or a list of sports. Each item in the list should have a checkbox next to it and the user can thus select more than one item. The results should be e-mailed to you.

1.2 Buttons and events

Exercise 3: Create a Web page that has a button on it. When the button is pressed the word 'Hello' should be added to the Web page. ('event1.html' in the downloads for lecture 12 can be used as a starting point. However, you will also need a div element with an id on the Web page, so you can target that element to display the word. The file 'results_button1.html' in the downloads for lecture 11 should be consulted to help in understanding how to do this.)

Exercise 4: This is an extension to the previous exercise. Create a Web page that has two buttons on it. When the first button is pressed the word 'Hello' should be added to the Web page before the two buttons. When the second button is pressed, the word

'Goodbye' should be added to the Web page after the two buttons. As the buttons are pressed more and more the page will fill up with the repeated word 'Hello' and the repeated word 'Goodbye'.

1.3 User interface elements (and no forms)

Exercise 5: Write a program to calculate the area and circumference of a circle. A set of HTML input elements should be used to get the user input and to display the results. (The file 'roomarea1_noform.html' in the downloads for lecture 12 can be used as a starting point.)

Exercise 6: Create a Web page that has two buttons and a paragraph with some random text. When button 1 is pressed the text in the paragraph should be displayed in red. When button 2 is pressed, the text in the paragraph should be displayed in blue.

Hints:

First, make sure you have created a css file with two classes in it, e.g.

```
*.red {
    color: red;
}

*.blue {
    color: blue;
}
```

Then link the css file to your html file as you have done in many previous examples.

Then, use JavaScript to add an event listener to each button in your html file. Each button should call a function when it is clicked. For example, the following function would locate the paragraph of text with id 'paragraph' and change its css class to "red"

```
function makeRed() {
  var p =
document.getElementById('paragraph
');
  //console.log(p);
  //console.log(p.className);
  p.className = "red";
}
```

Here, the css class for an element is changed using:

element.className = “*nameOfClass*”;

A similar function can be written for the button that changes the paragraph of text to blue.

This approach changes the className. There are alternative approaches to add and delete css classes. For example:

element.className += “ *nameOfClass*”;

will add the new css class to the existing list of classes (if any). The space after the first quotes is important. Other approaches, including deleting a css class from the list, at:

<https://developer.mozilla.org/en/docs/Web/API/Element/classList>