POWERING
POTENTIAL

Web Development

JQuery

Chapter 5
Events

OBJECTIVES

In this chapter you'll learn

- What events are
- Different types of events
- How to create event handlers
- How to capture information from events
- How events bubble up through the DOM hierarchy

Events

- **There are various ways of adding events**
  - During design-time
  - Dynamically at run-time
  - In this part we will look at the second option

- **The classic addEventListener() method**

```
var myControl = document.getElementById('footer');
myControl.addEventListener('click', function(){    });
```

- **In IE 8 and below do:**

```
myControl.attachEvent('onclick', function(){    });
```

How to setup events using Jquery

- **Simple click event example**

```
<script>
     $(document).ready(function () {
          $("#div1").click(function (e) {
                alert("Hello!");
          });
      });
</script>
<body>
    <div id="div1">Welcome</div>
```

- **See also: http://api.jquery.com/category/events/**

How to get information from an event

```
</head>
    <script src="https://code.jquery.com/jquery-3.3.1.min.js"></script>
    <script>
        $(document).ready(function () {

            $("#div1").click(function (e) {
                alert($(e.target).text());
                // or simply:
                alert($(this).text());
            });
        });
    </script>
</head>
<body>
    <div id="div1">Click on this text!</div>
</body>
</html>
```

How to get information from an event

```
<!DOCTYPE html>
<html>
<head>
<script src="https://code.jquery.com/jquery-3.3.1.min.js"></script>
<script>
    $(document).ready(function () {
        $('#UKimage').click(function (e) {
            var X = e.pageX - $(this).offset().left;
            var Y = e.pageY - $(this).offset().top;
            alert(X + " , " + Y);
        });
    });
</script>
</head>
<body>
    <img src="https://img.posterlounge.co.uk/images/l/1698693.jpg" alt="UK map" id="UKimage" />
</body>
</html>
```

More examples... Using a button Click event

```
 <script src="https://code.jquery.com/jquery-3.3.1.min.js"></script>
    <script>
        $(document).ready(function () {

            $('#SaveButton').click(function () {
                var info = $('#NameTextBox').val() + "<br/>" +
                    $('#Over18Checkbox').is(':checked') + "<br/>" +
                    $('#OptionsSelect').val();
                $('#infoDiv').html(info);
            });
        });
    </script>
</head>
<body>
    Name:<input id="NameTextBox" type="text" />
    Over 18<input id="Over18Checkbox" type="checkbox" checked/>
    <select id="OptionsSelect">
        <option>Option 1</option>
        <option>Option 2</option>
    </select>
    <input type="button" value="Save" id="SaveButton"/>
    <div id="infoDiv">.....</div>
```

Lab – Part 1

- In this exercise you will create event handlers using JQuery.
- Duration 30 minutes

There are lots of other events

- Hover, keydown, keyup, keypress
- focus, focusin, focusout, mousemove, scroll…

```
<script>
    $(document).ready(function () {
      $('#image1').mousemove(function (e) {
        var X = e.pageX - $(this).offset().left;
        var Y = e.pageY - $(this).offset().top;
        $("#div1").html(X + "," + Y);
      });
      $('#image1').mouseenter(function (e) {
        alert('Mouse over image detected')
      });
      $('#image1').mouseout(function (e) {
        alert('Mouse leaving the image detected')
      });
    });
  </script>
</head>
<body>
  <div id="div1">move mouse over the image</div>
  <img id="image1" src="team.jpg" alt="Team image" />
```

Events calling other event handlers

```
 <script src="https://code.jquery.com/jquery-
1.10.2.js"></script>
    <script>
        $(document).ready(function () {

            $('#button1').click(function (e) {
                button2.click();
            });

            $('#button2').click(function (e) {
                alert('Hello!');
            });
        });
    </script>
</head>
<body>
    <input id="button1" type="button" value="button1" />
    <input id="button2" type="button" value="button2" />
```

Events Bubbling

- Events bubble up through containing elements

```
 <script>
    $(document).ready(function () {

      $("#div1").click(function (e) {
        alert($(e.target).attr('id') + " clicked showing in div1");
      });
      $("#div2").click(function (e) {
        alert($(e.target).attr('id') + " clicked showing in div2");
      });
      $("p").click(function (e) {
        alert($(e.target).attr('id') + " clicked");
      });
    });
  </script>
</head>
<body>
  <div id="div1">
    Outer div1
    <div id="div2">
      inner div2
      <p id="p1">Click here first (Paragraph in div2)</p>
    </div>
  </div>
```

Events Bubbling – Handling key presses

- Following code only allows numbers and '-' character to be inserted into any control in the #PhoneDiv

```
<script>
       $(document).ready(function () {

           $('#PhoneDiv').keypress(function (e) {
               if ((e.keyCode < 48 || e.keyCode > 57)
                        && e.keyCode != 45)
               {
                   return false;
               }
           });
       });
   </script>
</head>
<body>
   <div id="PhoneDiv">
       <span>Phone:</span><input type="text" maxlength="12" />
   </div>
```

Stop event bubbling and inhibit typing the last character

Summary

In this chapter you learned

- What events are
- Different types of events
- How to create event handlers
- How to capture information from events
- How events bubble up through the DOM hierarchy

LAB – PART 2

- **The Event code in previous lab was good but it did not give immediate feedback to the user when each character was typed, neither did it erase any invalid character**

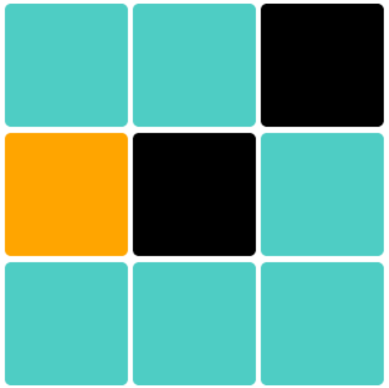
- **Duration 15 minutes**

If time permits…

- **Here is an exercise to have some fun with!**

- **Create a simple Tic-Tac-Toe game**
  - Game consists of 3x3 divs

```
<div></div><div></div><div></div><br>
<div></div><div></div><div></div><br>
<div></div><div></div><div></div><br>
```



  - Set an initial style for all divs with a suitable colour and margin
  - Put a "reset" game button to wipe the board
  - Toggle the style of a div when clicked
  - Alert user if the clicked div already has been played
  - If you have time you can detect the game-over condition