JavaScript:
Somewhere between Functional and OO

CS 1025 Computer Science Fundamentals I

Stephen M. Watt

University of Western Ontario

“Java”Script?

- “Java”Script has nothing to do with Java, except a superficial syntactic similarity.

- It supports *both*
  functional programming with closures
  (like Scheme)
  and
  a kind of object oriented style
  (like Java, but different).

Where does it come from?

- Originally developed 1995 at Netscape and called *Mocha.*
- Name changes *Mocha* $\Rightarrow$ *LiveScript* $\Rightarrow$ *JavaScript*
- Standardized as *ECMAScript* in 1998.

Where is it Used?

- Some dialects of ECMAScript

– JavaScript (Netscape)	used in browsers, Acrobat
– JScript (Microsoft)	used in browsers
– ActionScript (Adobe)	used in Flash and Flex
– DMDScript (Digital Mars)	used in browsers

- AJAX (2005)
  - “Asynchronous JavaScript And XML”
  - XHTML, CSS, DOM, XML, XSLT, XMLHttpRequest, JavaScript
- Gadgets in iGoogle, FaceBook, MySpace, *etc.*

A First Example

```
 function fact(n) {
   return (n < 2) ? 1 : n * fact(n-1);
 }

function adderFactory(n) {
   return function (m) { return m + n; }
}

document.write("That number " + fact(6) + " is 6! ");
var add1 = adderFactory(1);
var add2 = adderFactory(2);
var add3 = adderFactory(3);
document.write("The numbers after 7 are "
       + add1(7) + " " + add2(7) + " " + add3(7) + ".");
```

Using JavaScript in a Web Page

```
<html>
  <head>
    <title>Hello</title>
    <script type="text/javascript">
      function fact(n)        { return (n<2) ? 1 : n*fact(n-1); }
      function adderFactory(n){ return function (m){ return m + n; }}
    </script>
  </head>
  <body>
    <p>There once was a gnome who liked numbers.</p>
    <script type="text/javascript">
      document.write("That number " + fact(6) + " is 6! ");
      var add1 = adderFactory(1);
      var add2 = adderFactory(2);
      var add3 = adderFactory(3);
      document.write("The numbers after 7 are "
         + add1(7) + " " + add2(7) + " " + add3(7) + ".");
    </script>
  </body>
</html>
```

Providing Your Own Library

```
<html>
  <head>
    <title>Hello</title>
    <script type="text/javascript” src=“TestLib.js”></script>
  </head>
  <body>
    <p>There once was a gnome who liked numbers.</p>
    <script type="text/javascript">
      document.write("That number " + fact(6) + " is 6! ");
      var add1 = adderFactory(1);
      var add2 = adderFactory(2);
      var add3 = adderFactory(3);
      document.write("The numbers after 7 are "
         + add1(7) + " " + add2(7) + " " + add3(7) + ".");
    </script>
  </body>
</html>
```

Modifying the Current Document

```
<html>
   <head>
      <title>This is the title</title>
      <script src="Eg03-ModifyLib.js" type="text/javascript"></script>
   </head>
   <body>
      <div id="my-stuff">
        <h1>This is a heading</h1>
        <script type="text/javascript">munge()</script>
      </div>
   </body>
</html>
```

Eg03-Modify.js

```
function munge() {
        var mypara  = document.createElement("p");
        mypara.innerHTML = "Now is the time.";
        mypara.style.backgroundColor  = "lightgrey";

        var mystuff = document.getElementById("my-stuff");
        mystuff.style.backgroundColor = "pink";
        mystuff.appendChild(mypara);
}
```

Dynamic Behaviour

```
<html>
    <head>
       <title>A Clock</title>
       <script type="text/javascript">
          function timeString() {
              var now   = new Date();
              var hours = now.getHours();
              var mins  = now.getMinutes();
              var secs  = now.getSeconds();
              var str = "" + (hours > 12) ? hours - 12 : hours;
              str += ((mins < 10) ? ":0" : ":") + mins;
              str += ((secs < 10) ? ":0" : ":") + secs;
              str += (hours < 12) ? " a.m." : " p.m.";
              return str;
         }
         function tickClock () {
              document.getElementById("clock").innerHTML = timeString();
              setTimeout(tickClock, 1000);
         }
       </script>
    </head>
    <body onLoad="tickClock()">
         <center>
            <h1>The time is <span id="clock">no time yet</span></h1>
         </center>
    </body>
</html>
```

Buttons and Colors

```
<html>
  <head>
     <title>Colours</title>
     <script type="text/javascript">
        function makeRed() {
           document.bgColor = "Red";
        }
        function makeBlue() {
           document.bgColor = "Blue";
        }
        function makeGreen() {
           document.bgColor = "Green";
        }
     </script>
  </head>
  <body>
     <center>
        <h1>Colors!</h1>
        <form>
             <input type="button" value="Red"    onclick="makeRed()">
             <input type="button" value="Blue"   onclick="makeBlue()">
             <input type="button" value="Green"  onclick="makeGreen()">
        </form>
     </center>
  </body>
</html>
```

Why We Like Closures....

```
<html>
  <head>
     <title>Colours</title>
     <script type="text/javascript">
        function makeRed() {
           document.bgColor = "Red";
        }
        function makeBlue() {
           document.bgColor = "Blue";
        }
        function makeGreen() {
           document.bgColor = "Green";
        }
     </script>
  </head>
  <body>
     <center>
        <h1>Colors!</h1>
        <form>
             <input type="button" value="Red"    onclick="makeRed()">
             <input type="button" value="Blue"   onclick="makeBlue()">
             <input type="button" value="Green"  onclick="makeGreen()">
        </form>
     </center>
  </body>
</html>
```

More Ideas

- Organizing larger programs
- Dynamic tables
- Closures – used in a Button factory

```
<html>
  <head>
    <title>Colours</title>
    <script type="text/javascript" src="Eg07-StringLib.js"></script>
    <script type="text/javascript" src="Eg07-TableLib.js"></script>
    <link rel="stylesheet" type="text/css" href="Eg07-Table.css"></link>
  </head>
  <body onload="makeTable(40,40,tableHolder)">
     <center>
        <h1>Colors!</h1>
        <div id="tableHolder"></div>
     </center>
  </body>
</html>
```

The First Library

```
var HEX_DIGITS = "0123456789ABCDEF";

function intToHex(n) {
     var loNybble = (n >> 0) % 16;
     var hiNybble = (n >> 4) % 16;
     return ""+HEX_DIGITS.charAt(hiNybble)+HEX_DIGITS.charAt(hiNybble);
}

function makeRGB(r, maxr, g, maxg) {
    var rval = Math.round((255*r)/maxr);
    var gval = Math.round((255*g)/maxg);
    var bval = Math.max(0, 255 - (rval+gval));
    return "#" + intToHex(rval) + intToHex(gval) + intToHex(bval);
}
```

The Second Library

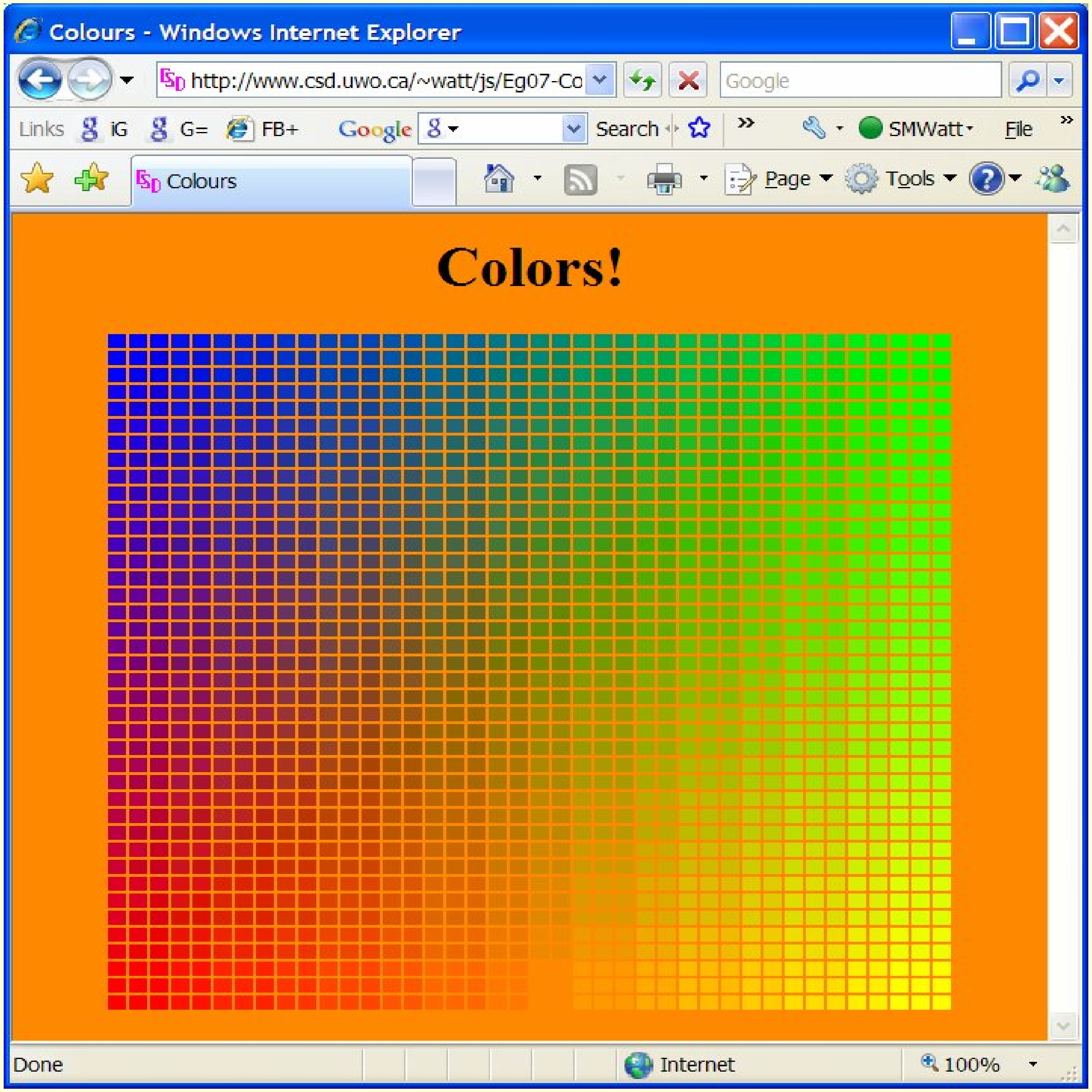
```
function setBackground(newcolor) {
   return function() { document.bgColor = newcolor; }
}

function makeTable(nrows, ncols, holder) {
    var theTable = document.createElement("table");
    var theBody  = document.createElement("tbody");
    for (r = 0; r < nrows; r++) {
        var theRow = document.createElement("tr");
        for (c = 0; c < ncols; c++) {
            var colorString     = makeRGB(r, nrows-1, c, ncols-1);
            var theCell         = document.createElement("td");
            theCell.bgColor     = colorString;
            theCell.onmouseover = setBackground(colorString);
            theRow.appendChild(theCell);
        }
        theBody.appendChild(theRow);
    }
    theTable.appendChild(theBody);
    holder.appendChild(theTable);
}
```

Some CSS Styling

```
td {
    height: 6pt;
    width: 6pt
}

table {
    border-spacing: 3pt
}
```


Colours - Windows Internet Explorer
http://www.csd.uwo.ca/~watt/js/Eg07-Co
Google
Links iG G= FB+
Search
SMWatt
File
Colours
Page
Tools
Done
Internet
100%

Colors!