

Webbteknik för ingenjörer

Laboration 4

John-Patrik Nilsson
e-mail: jpatrik.nilsson@gmail.com
Skype: j-p.nilsson

March 10, 2010

1 Abstract

ECMAScript (also known as JavaScript) is a powerful load-and-go loosely typed lambda programming language devised to function within websites and Internet applications. It is one of the fundamental parts of web development today.

Keywords: JavaScript, ECMAScript, Internet, web application, web development.

2 Laboration

2.1 Introduction

The objective of the laboration were to gain basic knowledge about ECMAScript, and consists of three parts. The first two parts are directed towards using ECMAScript to change the contents of a website, whereas the last part is focused on learning how to use an external code library (framework) to enhance a websites functionality.

2.2 Part 1 and 2: Dynamic content

The objective of these parts of the laboration were to use ECMAScript to dynamically choose content to be displayed on a website depending on the current type of season: summer, spring/fall, winter. The current type of season was determined from the current month.

Three different types of content were chosen to be displayed on the website dynamically; a randomized number, a picture, and a comment. These types of content had three versions each, one for each type of season.

Randomized number:

Summer Between 10 and 25.

Spring and fall Between -5 and 10.

Winter Between -20 and -5.

Picture:

Summer summer.png
Spring and fall spring_and_fall.png
Winter winter.png

Comment:

Summer "Nice weather! Perhaps going for a swim would be a good idea today?"
Spring and fall "Brr, cold. Don't forget your ear-muffs when going outside."
Winter "Rain. Ever thought of moving to California?"

The ECMAScript was written to exist in its own namespace, called MAIWIDGET, and it was created at the top of the ECMAScript code by using an object literal (`{}`). Containing code in namespaces are essential when using mashups, so it was considered a good idea.

The different types of content that would be displayed on the website were each handled by their own ECMAScript function. These functions were called `getTempElem`, `getPicElem`, and `getCommentElem`. Each of those functions took month (an integer 1–12) as an argument and returned a HTML element.

The assembly and the calling of these functions were handled by another function called `weatherElem`. This function took a season (string; "summer", "winter" or "spring_and_fall") as input and returned a HTML element.

The user interface were designed to be handled by a function called `weather`. The `weather` function were called whenever a user clicked on a button on the website, and this function in turn would call the other functions of the ECMAScript.

In essence, the flow of data could be explained as a list:

1. **onClick** The user presses a button after selecting a season. The click initiates the `weather` function of the ECMAScript.
2. **weather** The `weather` function calls the `weatherElem` function, and passes it the season which the user had selected.
3. **weatherElem** The `weatherElem` function considers which season to display, and after that calls the appropriate functions to create a temperature, a picture, and a comment element and puts it all together into a single `weatherElem` element, which it then returns to the `weather` function.
4. **weather** The `weather` function now have a complete `weatherElem` HTML element (div), which it displays on the website via a call to the `scriptaculous effect` function `appear`.

2.3 Part 3: Effects

Two functions belonging to the external `scriptaculous` ECMAScript library were loaded to be called with a fairly straightforward syntax from within the MAIWIDGET object. Both functions enhances the visual effects of the website and its content. The functions were called `appear` and `fade`. `Appear` were to be called by the `weather` function, and were responsible for displaying the dynamic

content of the website. Fade, the effect-wise opposite of the appear function, were called by the clear function and were allocated the responsibility of clearing (removing) all the dynamic content of the website.

The effects used by MAIWIDGET from the scriptaculous library were called like so:

```
weatherElement.appear();

MAIWIDGET.clear = function () {
  var container = document.getElementById("containerDiv");
  var kids = container.childNodes;
  for (var i = 0; i < kids.length; i++) {
    kids[i].fade();
  }
}
```

3 Final notes

The CSS code was not part of the laboration but were nevertheless added to produce a better resulting website.

The "get" functions in MAIWIDGET should have more appropriate names, for example "create", since that is what they actually do. Well, if the author had to be honest, most functions in the MAIWIDGET object should have their names substituted for something more appropriate, not to mention the object MAIWIDGET itself.

A References

<http://www.w3schools.com>

<http://script.aculo.us/>

<http://thepenry.net/jsrandom.php>

<http://www.ibm.com/developerworks/web/library/wa-jsdom/>

B Code listing

B.1 File: index.html

```
<html>
<head>
<style type="text/css">
body {
  background-color: grey;
}
button {
  margin-bottom: 3px;
}
.weatherElement {
```

```

float: right;
padding: 5px;
width: 300px;
height: 300px;
border-style: solid;
border-width: 2px;
margin: -1px;
}
p {
    text-align: center;
}
img {
    margin-left: 50px;
}
</style>
<script type="text/javascript">
var MAIWIDGET = {};
MAIWIDGET.weather = function () {
    var container = document.getElementById("containerDiv");
    var selection = document.getElementById("seasonSel").value;
    var weatherElement = this.weatherElem(selection);
    container.appendChild(weatherElement);
    weatherElement.appear();
};
MAIWIDGET.weatherElem = function (selection) {
    var displayMonth;
    if (selection == "summer") {
        displayMonth = 6;
    } else if (selection == "winter") {
        displayMonth = 1;
    } else if (selection == "spring_and_fall") {
        displayMonth = 4;
    } else {
        var date = new Date();
        displayMonth = date.getMonth() + 1;
    }
    var weatherE = document.createElement("div");
    weatherE.setAttribute("class", "weatherElement");
    weatherE.setAttribute("style", "display: none");
    weatherE.appendChild(this.getTempElem(displayMonth));
    weatherE.appendChild(this.getPicElem(displayMonth));
    weatherE.appendChild(this.getCommentElem(displayMonth));
    return weatherE;
};
// -----
MAIWIDGET.getTempElem = function (month) {
    var min, max;
    if (month >= 6 && month <= 8) {
        min = 10;
        max = 25;
    }

```

```

    } else if (month <= 2 || month == 12) {
        min = -20;
        max = -5;
    } else {
        min = -5;
        max = 10;
    }
    var temp = Math.floor(Math.random()*16+min);
    var txt = "Temperature is "+temp+" degrees C.";
    var tempElem = document.createElement("p");
    tempElem.setAttribute("id", "tempPg");
    tempElem.innerHTML = txt;
    return tempElem;
};
MAIWIDGET.getPicElem = function (month) {
    var src;
    if (month >= 6 && month <= 8) {
        src = "pics/summer.png";
    } else if (month <= 2 || month == 12) {
        src = "pics/winter.png";
    } else {
        src = "pics/spring_and_fall.png";
    }
    var img = document.createElement("img");
    img.setAttribute("id", "weatherImg");
    img.setAttribute("src", src);
    img.setAttribute("width", 200);
    img.setAttribute("height", 200);
    return img;
};
MAIWIDGET.getCommentElem = function (month) {
    var txt = "";
    if (month >= 6 && month <= 8) {
        txt = "Nice weather! Perhaps going for a
        swim would be a good idea today? ";
    } else if (month <= 2 || month == 12) {
        txt = "Brr, cold. Don't forget your ear-muffs
        when going outside.";
    } else {
        txt = "Rain. Ever thought of moving to California?";
    }
    var comment = document.createElement("p");
    comment.setAttribute("id", "commentPg");
    comment.innerHTML = txt;
    return comment;
};
MAIWIDGET.clear = function () {
    var container = document.getElementById("containerDiv");
    var kids = container.childNodes;
    for (var i = 0; i<kids.length; i++) {

```

```
        kids[i].fade();
    }
};
</script>
<script src="scriptaculous/prototype.js"
    type="text/javascript"></script>
<script src="scriptaculous/scriptaculous.js"
    type="text/javascript"></script>
</head>
<body>
<select name="season" id="seasonSel">
    <option value="today">today</option>
    <option value="winter">winter</option>
    <option value="spring_and_fall">spring and fall</option>
    <option value="summer">summer</option>
</select>
<button onClick="MAIWIDGET.weather();">display now</button>
<button onClick="MAIWIDGET.clear();">clear all</button>
<div id="containerDiv"></div>
</body>
</html>
```