

IT452 Advanced Web and Internet Systems

Fall 2009

Set 2: DOM

Review – DHTML Basics

- Find the HTML object we want to change

```
var domLink = document.getElementById("linkToAnimal");
```

- Change the object's:

- HTML properties

```
domLink.href = "cat.html";
```

- CSS properties

```
domLink.style.backgroundColor = "blue";
```

Simple Paint (ex1.html) – Part 1

```
?xml version = "1.0" encoding="utf-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns = "http://www.w3.org/1999/xhtml">

<head>
    <title>IT 452 DOM demo</title>

    <script type="text/javascript" >
        <!--

        var currentAction = "red";

        function doupdate(curNode ) {
            curNode.style.backgroundColor = currentAction;
        }

        // --
    </script>
</head>
```

Simple Paint (ex1.html) – Part 2

```
<body>

    <h1> IT 452 DOM demo </h1>
    <p>Tools:</p>
    <ul>
        <li onclick="currentAction='red'    "> Set red  </li>
        <li onclick="currentAction='blue'   "> Set blue </li>
        <li onclick="currentAction='insert'"> Insert   </li>
        <li onclick="currentAction='delete'"> Delete   </li>
        <li onclick="currentAction='edit'   "> Edit     </li>
    </ul>

    <table border="1">
        <tr>
            <td onclick="doupdate(this)"> thing 1 </td>
            <td onclick="doupdate(this)"> thing 2 </td>
        </tr>
        <tr>
            <td onclick="doupdate(this)"> thing 3 </td>
            <td onclick="doupdate(this)"> thing 4 </td>
        </tr>
    </table>
</body></html>
```

What does the DOM tree look like?

ex2.html – add insert/delete/edit

```
function doupdate(curNode) {
    var parent = curNode.parentNode;

    if (currentAction == "insert") {
        newThing = document.createElement("td");
        parent.insertBefore(newThing, curNode);
        newThing.innerHTML = "hello";
    }

    else if (currentAction == "delete") {
        parent.removeChild(curNode);
    }

    else if (currentAction == "edit") {
        curNode.innerHTML = window.prompt("Enter new text", "");
    }

    else {
        curNode.style.backgroundColor = currentAction;
    }
}
```

ex3.html – fully functional

```
function doupdateDyn() {  
    doupdate(this);  
}  
  
function doupdate(curNode ) {  
    var parent = curNode.parentNode;  
    if (currentAction == "insert") {  
        newThing = document.createElement("td");  
        parent.insertBefore(newThing, curNode);  
        newThing.innerHTML = "hello";  
        newThing.onclick = doupdateDyn;  
    }  
  
    else if (currentAction == "delete") {  
        parent.removeChild(curNode);  
    }  
    else if (currentAction == "edit") {  
        curNode.innerHTML = window.prompt("Enter new text",  
            curNode.innerHTML);  
    }  
    else {  
        curNode.style.backgroundColor = currentAction;  
    }  
}
```

Some tempting ideas (too bad they don't work)

newThing.onclick = "doupdate(this) "

newThing.onclick = doupdateDyn()

ex4.html – more elegant / alternatives

```
function doupdate(curNode) {
    var parent = curNode.parentNode;

    if (currentAction == "insert") {
        newThing = document.createElement("td");
        parent.insertBefore(newThing, curNode);
        newThing.innerHTML = "hello";
        newThing.onclick = function() {doupdate(this)};
    }

    else if (currentAction == "delete") {
        parent.removeChild(curNode);
    }

    else if (currentAction == "edit") {
        curNode.firstChild.nodeValue =
            window.prompt("Enter new text", curNode.innerHTML);
    }
    else {
        curNode.style.backgroundColor = currentAction;
    }
}
```

Tips

- Remember JavaScript console
- How many children?
 - Use DOM Inspector to see
 - From ex5.html:

```
else if (currentAction == "parent") {
    var kids = parent.childNodes;
    window.alert("Number children: "+kids.length);
    window.alert("Type of first: "+kids[0].nodeType);
}
```
- See DOM reference links under IT452 page
 - Use the API!

Ex6.html: iterating over children

```
else if (currentAction == "rowInfo") {  
    var kids = parent.childNodes;  
    for (var i=0; i < kids.length; i++) {  
        var theKid = kids[i];  
        window.alert("Kid #" + i + " has type"  
            + theKid.nodeType);  
    }  
}
```