

INFM 603: Information Technology and Organizational Context

Session 4: JavaScript – DOM and Events



Jimmy Lin
The iSchool
University of Maryland

Thursday, September 26, 2013



Programming... is a lot like cooking!

Arrays

- An array holds a collection of values
 - Each value is referenced with an index, starting from 0
- Creating an array:

```
var arr = new Array();  
arr[0] = 0;  
arr[1] = 3;  
arr[2] = 2;  
arr[3] = 4;
```

What happens if you don't specify
value for a particular index?

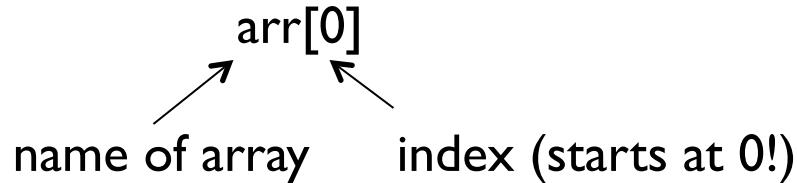
- Or, alternatively:

```
var arr = [0, 3, 2, 4];
```

- Note, arrays automatically grow in size

Using Arrays

- Referencing values in an array:



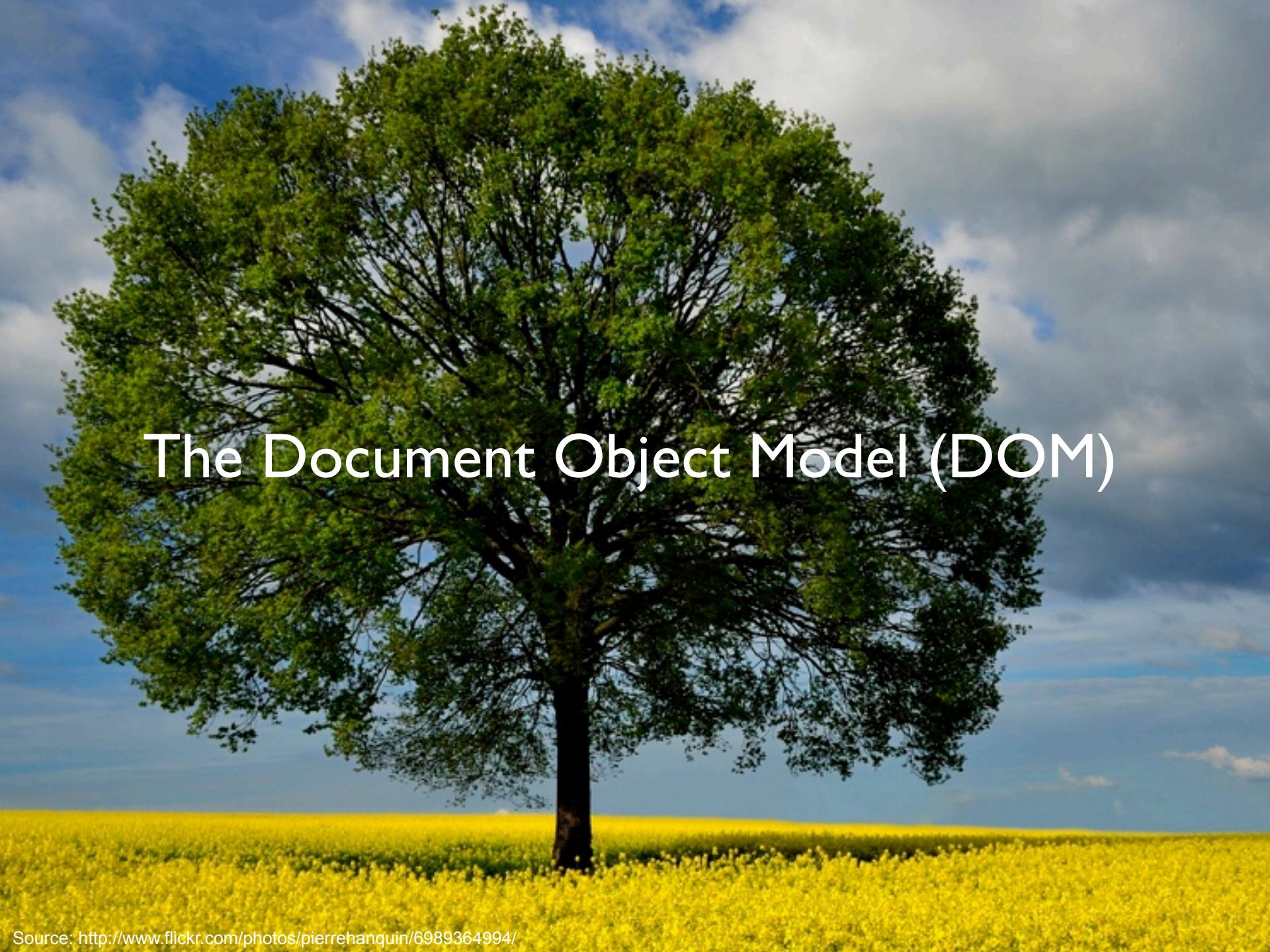
- Array values can be used in other expressions and statements:
`var f = 5 + arr[0] + arr[2];`
- Find out the length of an array: `arr.length`

- Arrays and `for` loops go hand in glove:

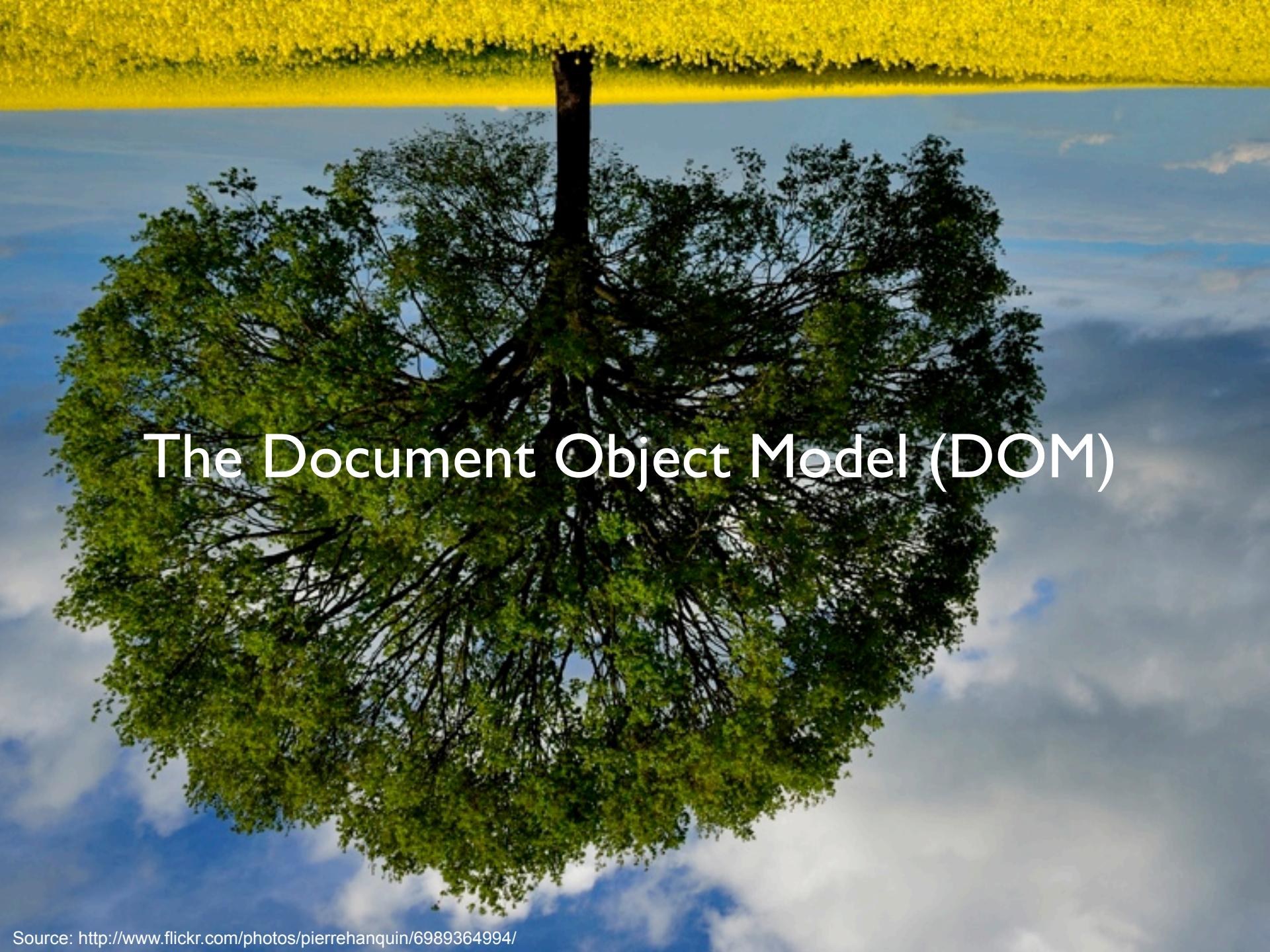
```
var arr = [0, 3, 2, 4];
var sum = 0;
for (var i=0; i<arr.length; i++) {
    sum += arr[i];
}
console.log(sum);
```



Cooking analogy?

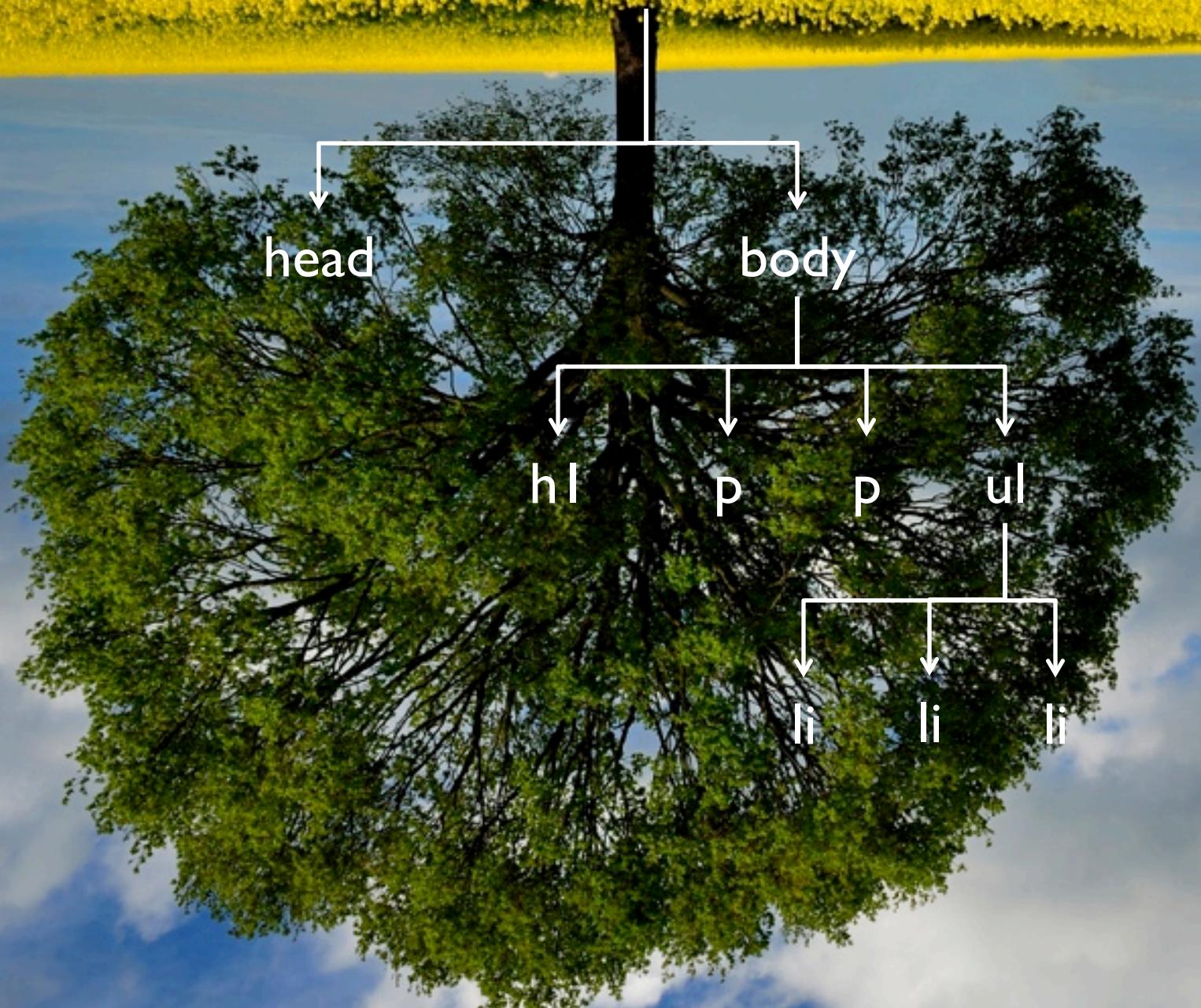
A photograph of a large, mature tree with a dense canopy of green leaves, standing alone in a field of bright yellow flowers, likely rapeseed. The tree is positioned in the center-left of the frame. The background consists of a vast, open landscape under a sky filled with scattered, white and grey clouds.

The Document Object Model (DOM)

A photograph of a large, mature tree with a dense canopy of green leaves. The tree stands against a backdrop of a clear blue sky with scattered white and grey clouds. The perspective is from below, looking up at the tree's branches.

The Document Object Model (DOM)

document



Asking the DOM to “do stuff”

the *method* is what you want the document “to do”, usually a verb phrase



document.method(“argument”);

document represents the entire page and contains the DOM



arguments are additional details that you specify

More on the dot notation later...

DOM: Selecting Nodes

- Selecting a DOM node by id:

```
document.getElementById("id");
```

- Note, returns a DOM node

- Selecting DOM nodes by tag:

```
document.getElementsByTagName("p");
```

- Note, returns a collection (treat as an array)

- Once you select a DOM node:

- Get a node's children: list.childNodes
- Get a node's number of children: list.childNodes.length
- Natural to iterate over child nodes using for loops

BTW, <div> tags are very useful for grouping elements together.

DOM: Manipulating Nodes

- Simplest way to manipulate DOM nodes: select the node and modifying its innerHTML property:

```
var p = document.getElementById("para");
```

```
p.innerHTML = "some text";
```

- innerHTML can be *any* HTML!

- Modify a child node using innerHTML:

```
document.getElementById("list").childNodes[1].innerHTML = "new item";
```

DOM: Building Nodes

- Building DOM nodes programmatically:

```
var p = document.createElement("p");
p.innerHTML = "here is some new text.";
document.getElementById("div1").appendChild(p);
```

```
var newItem = document.createElement("li");
newItem.innerHTML = "new list item";
document.getElementById("list").appendChild(newItem);
```

- Set setAttribute method to set attributes

```
document.getElementById("para").setAttribute("style", "font-family: arial");
```

DOM: Removing Nodes

- Select the node to remove, then use the `removeChild` method in its parent:

```
var list = document.getElementById("list");
var listItem = list.childNodes[1];
list.removeChild(listItem);
```

Let's build a table!



```
var t = document.createElement("table");
t.setAttribute("border", 1);
var row1 = document.createElement("tr");
var row1col1 = document.createElement("td");
row1col1.innerHTML = "A";
var row1col2 = document.createElement("td");
row1col2.innerHTML = "B";

row1.appendChild(row1col1);
row1.appendChild(row1col2);

t.appendChild(row1);

document.getElementById("div1").appendChild(t);
```

Events

- GUI are driven by events
- When an event happens, an event handler is called to “handle” the event
- Easier to show in an example...

Note, what I'm showing is slightly easier than what's in the book...

Forms

The image shows a collection of German tax forms from 2005 and 2006, illustrating the complexity of German tax reporting. The forms include:

- Anlage N (2005):** Eintragung zur Einkommensteuererklärung (Antrag auf Lohnsteuer-Ermäßigung).
- Anlage V (2005):** Einkünfte aus Vermietung und Verpachtung.
- Anlage KAP (2005):** Antrag auf Festsetzung der Arbeitnehmer-Sparzusage.
- Anlage R (2005):** Renten und andere Leistungen.
- Anlage EUR (2005):** Einnahmenüberschussrechnung – Anlage EUR.
- Finanzamt (2006):** Umsatzsteuer-Voranmeldung 2006.

These forms cover various aspects of taxation, including income from employment, self-employed individuals, children, capital gains, and business profits. They are filled with handwritten data and contain numerous checkboxes and dropdown menus.

A wide-angle photograph of a massive dam under construction, likely the Hoover Dam. The dam's concrete walls are covered in a grid of steel reinforcement bars (rebar). Scaffolding and construction equipment are visible along the top and sides of the dam. In the foreground, water flows over a partially completed section of the dam. In the background, there are mountains and a clear blue sky.

Putting everything together...