

ECS 189
WEB PROGRAMMING

4/17

Announcements

- Midterm Fri 4/28 – is this a problem?
- Design for weather app Friday
- Due Wednesday
- Rough design today focusing on layout, features

Using Chrome Javascript console

- View -> Developer -> Javascript console

2+3 = 5 // tries to set value of a number
> nasty error message

2+3 == 5 // the Boolean test equality operator
> true

2+3 == "5.0" // ?? try it!

===

- Can do equality testing with type checking

2+3 === 5 // true
2+3 === "5.0" // false!

Javascript tricky questions

```
a = (3.0 === 3);  
b = ("a" === 'a');
```

- What is in a? b?

Javascript tricky questions

```
a = (3.0 === 3);  
b = ("a" === 'a');
```

- What is in a? b?
- Both true.

Function assignment

```
function f () { var r = 1; return r; }  
a = f();  
b = f;
```

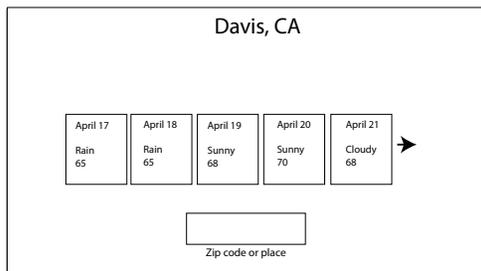
- What is in a? b?

Function assignment

```
function f () { var r = 1; return r; }  
a = f();  
b = f;
```

- What is in a? b?
- a contains the number 1 because f() evaluates the function
- b contains the function f
- Functions are objects in Javascript!

Weather App, 2017



Basic layout using flexbox

- How are elements arranged in body?

Basic layout using flexbox

- How are elements arranged in body?
Laid out vertically in order, centered horizontally
- What should CSS for body be using flexbox?

Basic layout using flexbox

- How are elements arranged in body?
Laid out vertically in order, centered horizontally
- What should CSS for body be using flexbox?
display: flex; - makes it a flexbox
flex-direction: column; - vertical layout
align-items: center; - center horizontally

The usual trick with main box

```
main {  
  display: flex;  
  flex-direction: column;  
  justify-content: center;  
  align-items: center;  
  flex-grow: 1;  
  align-self: stretch;  
}
```

The usual tricks with main box

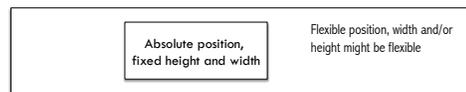
```
main {  
  display: flex;  
  flex-direction: column;  
  justify-content: center; - vertical centering  
  align-items: center; - horizontal centering  
  flex-grow: 1; - fill vertical space  
  align-self: stretch; - fill horizontal space  
}
```

Carousel

- Elements slide horizontally in and out of the box
- Eg. push arrow, weather forecast slides one day to the left.
- Implement with flexbox and Javascript

Layout using absolute positioning

- Place a box (eg. a day's forecast) explicitly in its container, not relying on flexbox.

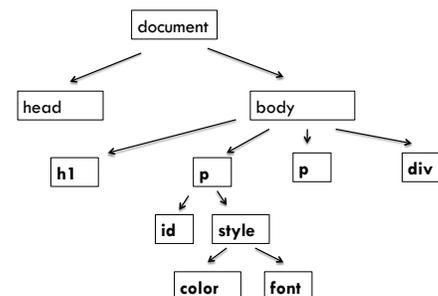


- Placement of container in grand-parent container still uses flexbox, height and/or width might be flexible

Using absolute positioning

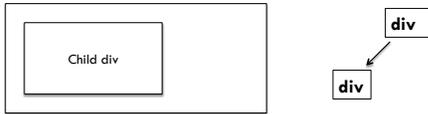
- Use flexbox to do as much as you can (eg. align-left rather than absolutely positioning something at the left of its box)
- Save absolute positioning for doing fancy things in Javascript, when you take over layout using your own code.
- Javascript code usually need to know width of parent box! Which is determined by flexbox (or other responsive layout...)

The DOM



Structure of the DOM

- Elements in the DOM have properties.
- Important ones: it's parents and children.
- For instance, say we have one div inside another; the inner is a child of the outer.



Changing style in the DOM

- We can change style as well as structure using Javascript.

```
sty= idiv.style;  
sty.backgroundColor ="pink";
```

- The style is an object as well. Its properties are the style properties we've been setting in CSS.
- Sadly, they usually don't have the same names (eg. in CSS it's background-color).

Getting children

```
<div id="outerDiv">  
<div id="innerDiv">      HTML is Green  
  <p>Child div</p>  
</div>  
</div>  
var odiv = document.getElementById("outerDiv");  
var idiv = odiv.children[0];      Javascript is pink
```

- odiv is an object in the Javascript code
- odiv has a list of children; in this case a list of length 1.

Getting parents

```
<div id="outerDiv">  
<div id="innerDiv">  
  <p>Child div</p>  
</div>  
</div>  
var idiv = document.getElementById("innerDiv");  
var odiv = idiv.parentElement;
```

- idiv is an object in the Javascript code
- It's parent is the outer div.

Getting HTML element properties

```
var odiv = document.getElementById("outerDiv");  
var width = odiv.clientWidth;
```

- clientWidth and clientHeight contain actual inner dimensions of the box in the current layout for the current viewport.