$('h1').click()

SCRIPTING WITH JQUERY

Nick Doty & Ryan Greenberg

CSS meets JavaScript

REVIEW: JQUERY

Using jQuery involves two steps:

- Selects objects from the DOM using CSS selectors.
- Do something with the selected elements.

Recap from last class.

REVIEW: JQUERY

Using jQuery involves two steps:

$("p").hide()

Here we see what this looks like using jQuery's dollar sign syntax. In the first part of this statement, we select an object. **Question: what does "p" select?** (all the paragraphs on the page). In the second part we do something with it. **Question: what would you guess hide() does?**

CREATING ELEMENTS

- Create elements using fragments of HTML.
- Specify where to insert them into the DOM.
- Functions to insert elements into the DOM are in the "Manipulation" part of jQuery's documentation.
- Examples:

.append()	.prepend()
.after()	.before()

For example, $('ul').append('
A new item in my list
') will add an element to your list.

These operations happen for all of the elements you have selected. If there are 10 h2s on the page and you write $('h2').after("My new content"), it will appear 10 times, once after each h2.

JQUERY DEMONSTRATION

http://j.mp/mobileappdom

EVENTS IN JQUERY

- Old way of doing events in JavaScript
- Handling events in jQuery
- Common problems with events
- Ready vs. Load
- Default actions

THE OLD WAY

```
<button onclick= “alert(‘You clicked me’);”>
Click Me
</button>
```

```
<form onsubmit= “alert(‘Submitting’);”>
</form>
```

Problems with this approach:
- It mixes Javascript into your HTML. You should keep content and behavior separate.
- It gets messy quickly. Imagine you have lines where you want the same thing to happen when you click on each one: rewrite “onlick=‘’” 10 times.
- It is hard to mix functions and scripts that use onclick because an element can only have one onclick attribute.

EVENT LISTENERS

$('h1').click(...)

- Provide a function that contains what to happen when the event is triggered.

Adding an event listener is often called "binding" an event listener. Javascript provides the functionality to add event listeners; jQuery makes it easier.

EVENT LISTENERS

```
$('h1').click(function(){
alert("You clicked me.");
});
```

- Provide a function that contains what to happen when the event is triggered.

Demo in browser.

COMMON EVENT PROBLEMS

1. You cannot add an event listener to an element that does not exist yet.

If you try to bind an event to an element that does not exist yet, it will not work.

READY VS. LOAD

- "load" happens when all the resources for a page have been downloaded.
- "ready" happens when the DOM for a page has been created.

The solution is to bind all of your events when the document's DOM is ready. jQuery provides this functionality through the ready event.

```
$(document).ready(function(){
    [your code here]
});
```

COMMON EVENT PROBLEMS

1. You cannot add an event listener to an element that does not exist yet.
2. For elements that have a default action (links, form submissions) you have to prevent the default action.

Clicking on links and submitting forms triggers a default action in the browser: going to the new document, or submitting the form. To prevent the default action, you return false, or use event.stopPropagation:

```
$('form').submit(function(){
    // Prevent default action
    return false;
});
```

JQUERY AND "this"

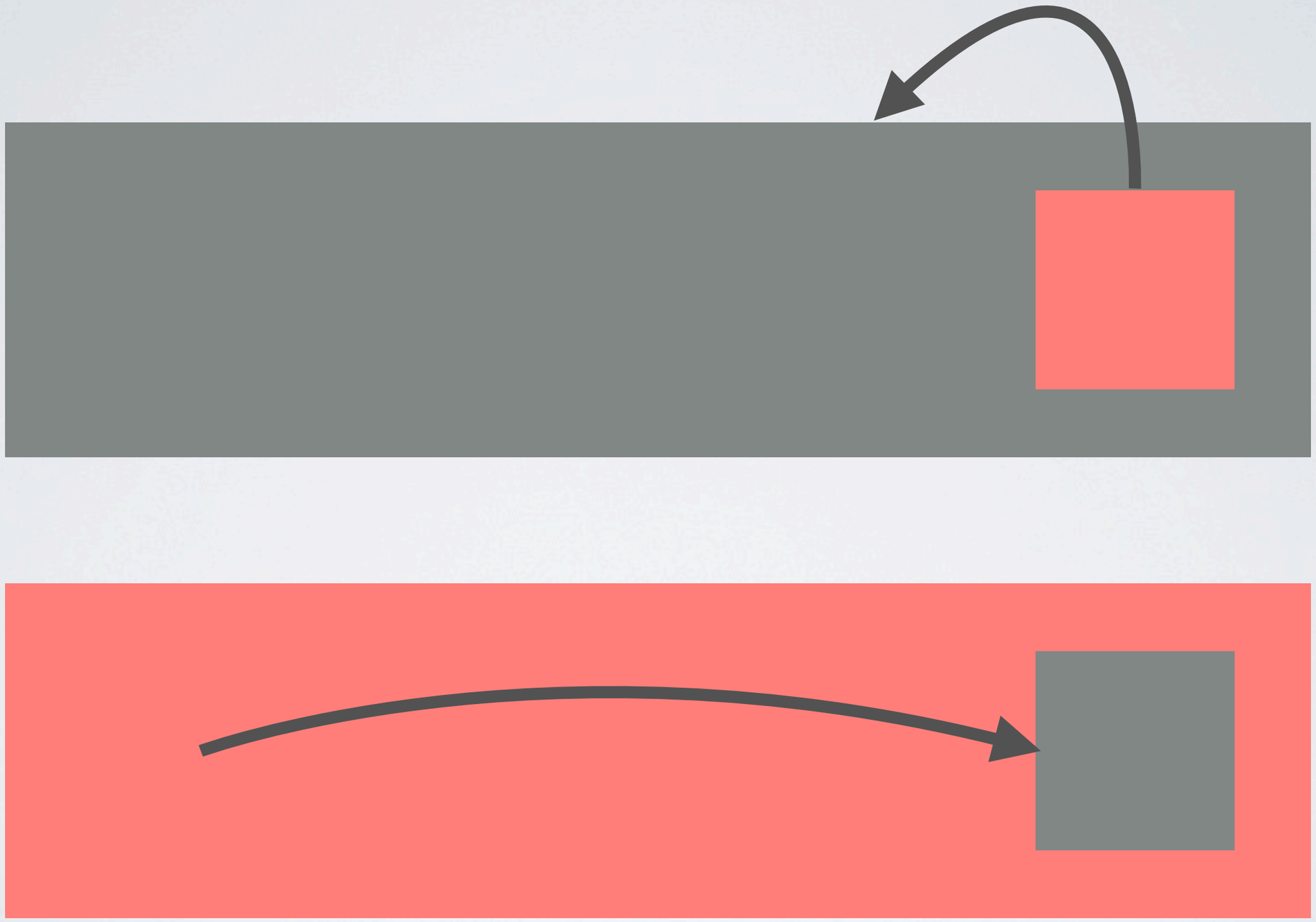
- When jQuery triggers an event handler "this" represents the element that was acted upon.
- "this" is not a jQuery object, so you must wrap it with $(…) before you can do something to it.

TRAVERSING THE DOM

- Moving from the currently selected element to a related element.
- You can move in any direction: up to parents, down to children, and back and forth between siblings.

When handling events, you often want to operate on a related element, not just the element receiving the event. When an item's delete button is clicked, you want to delete the item, not the delete button.

TRAVERSING THE DOM



You can move from an element to its parent(s), or vice-versa from a element to its children. To get the closest parent of a given type, use .closest();

```
$(this).closest('li');
```

See http://api.jquery.com/category/traversing/ for more operations. In particular, look at .find(), .children(), and .parents(). To move between sibling elements, see .next() and .prev().

AJAX WITH JQUERY

- $.get()
- $.post()
- $.getJSON()

AJAX will be covered in greater detail later in this course. jQuery eliminates the differences between browsers' varying implementations of AJAX.

AJAX WITH JQUERY

- $.get(*url*, *parameters*, *callback*)
- $.get('/greeting.php', {'name': 'Erik'}, function(rsp){alert(rsp);});

QUESTIONS?