

# WEEK 9 | PROGRAMMING

## MOAR JQUERY

Shawn Park & Jeff Zhan



Web Design DeCal  
DESIGN MEETS PROGRAMMING



# *Review jQuery*

*(<http://jsfiddle.net/suX84/3/>)*



# Today's Outline

1. Other jQuery Functions
2. DOM Traversal
3. Event Objects





***Goal Today:***  
*Practice jQuery & Learning the DOM*



# Other jQuery Functions

# Other jQuery Functions

- **.addClass('className')**
  - Ex: `<div class="title">Title</div>`
    - `$('.title').addClass('superbold');`
    - `<div class="title superbold`
- **.removeClass('className')**
- **.hasClass('className')**
  - Ex: `<div class="title superbold">Title</div>`
    - `if($('.title').hasClass('superbold')) {  
    alert('Title is super bolded!!');  
}`

# Other jQuery Functions

- **.attr('...')**
  - Ex: `<a id="link1" href="http://www.google.com">Google</a>`
  - `var sample = $('#link1').attr('href');`  
`alert(sample);`
- **.attr('...') works for any attribute**
  - Ex: `.attr('href')`
  - Useful ones:
    - `.attr('id')`
    - `.attr('class')`
    - `.attr('width')` // For `<img tags>`
    - `.attr('src')`



# Other jQuery Functions

- **Setting .attr()**
  - Syntax:
    - `.attr('src', 'images/newimg.png')`
    - `.attr('href', 'www.yahoo.com')`
  - Takes 2 arguments
    - 1<sup>st</sup> is the attribute name
    - 2<sup>nd</sup> is the attribute value





# *Demo*

Follow along:  
[jsfiddle.net/5pye4/1/](https://jsfiddle.net/5pye4/1/)



# DOM Traversal

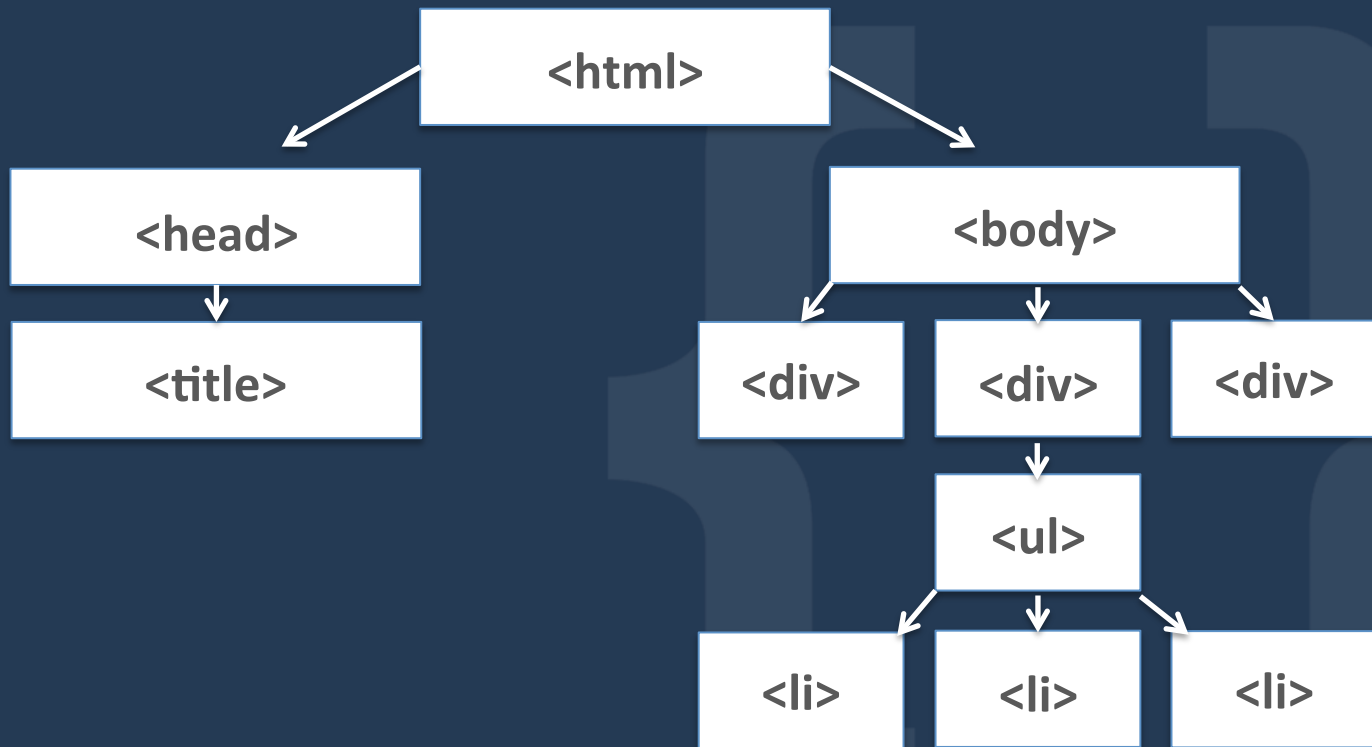
# DOM Traversal

- **What is the DOM?**
  - Document Object Model
  - A way to structure a document & represent hierarchy of elements
    - *E.x.*

```
<div id="box">  
  <div id="title">Title</div>  
</div>
```
    - In above, *#title* is a **child** of *#box* (parent-child relationship)
- **What is DOM Traversal?**
  - Accessing different elements in the DOM and utilizing their heirarchy

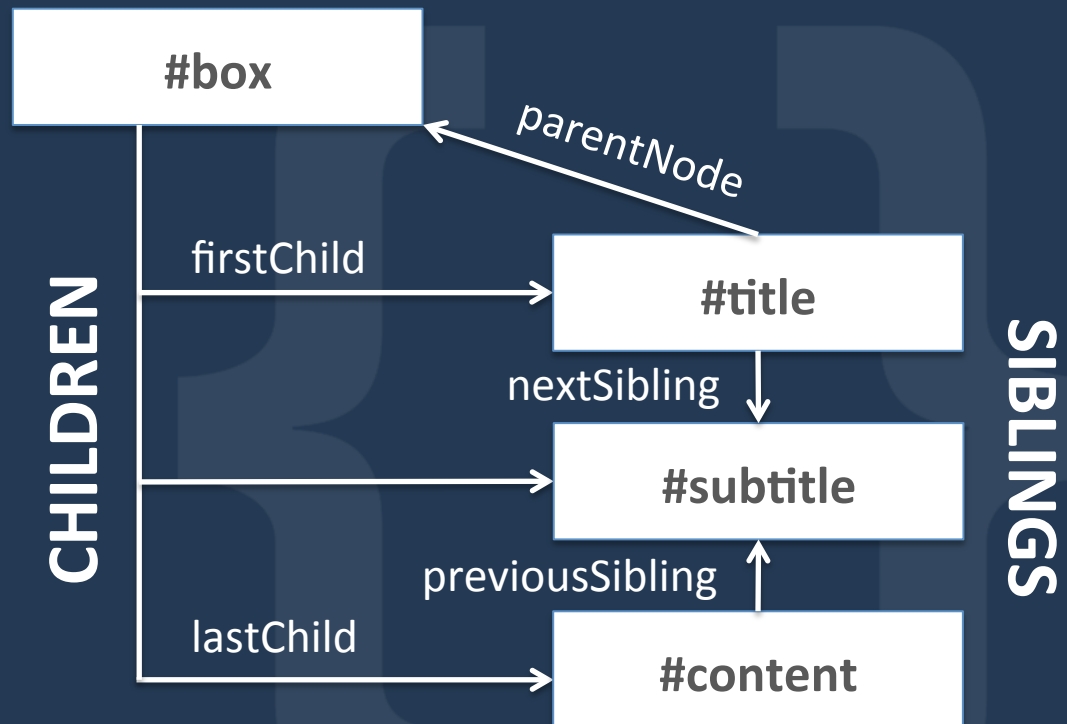


# DOM Traversal



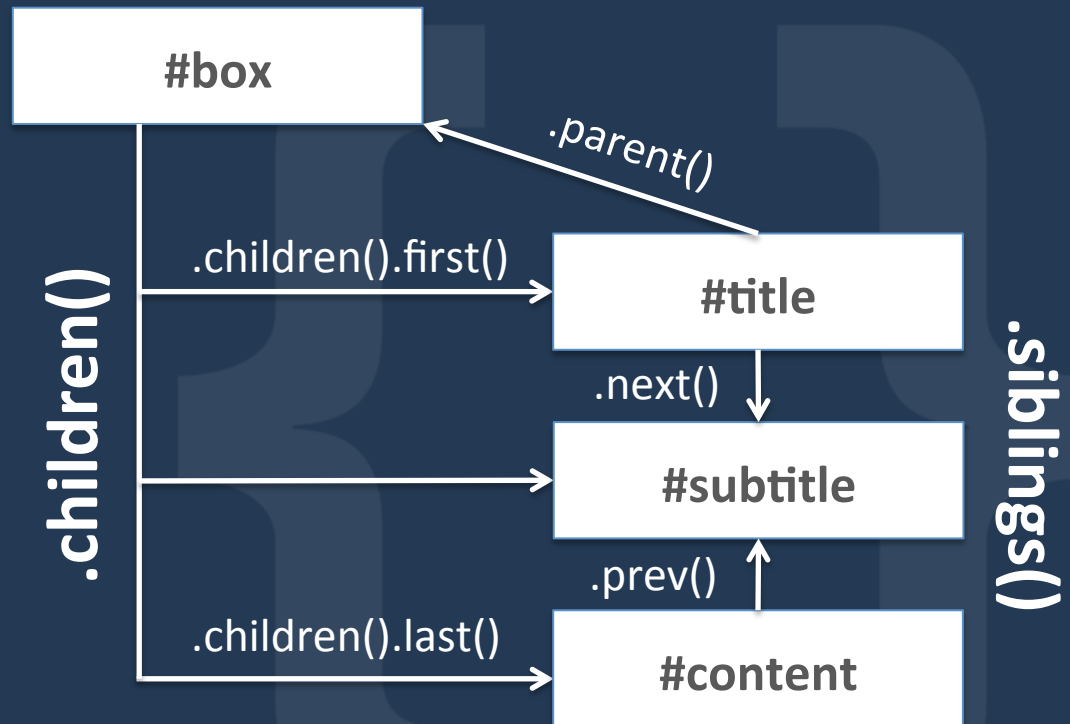
# DOM Traversal

```
<div id="box">  
  <div id="title">  
    Title  
  </div>  
  <div id="subtitle">  
    Subtitle  
  </div>  
  <div id="content">  
    Content  
  </div>  
</div>
```



# DOM Traversal - jQuery

```
<div id="box">  
  <div id="title">  
    Title  
  </div>  
  <div id="subtitle">  
    Subtitle  
  </div>  
  <div id="content">  
    Content  
  </div>  
</div>
```



# DOM Traversal - Example

## Reference

<code>.children()</code>	<code>.siblings()</code>	<code>.parent()</code>
<code>.children().first()</code>	<code>.prev()</code>	
<code>.children().last()</code>	<code>.next()</code>	

You are given `$('#menu')`. How to select Pizza element?

**Answer:** `$('#menu').children().last()`

You are given `$('#menu')`. How to select Pad Thai?

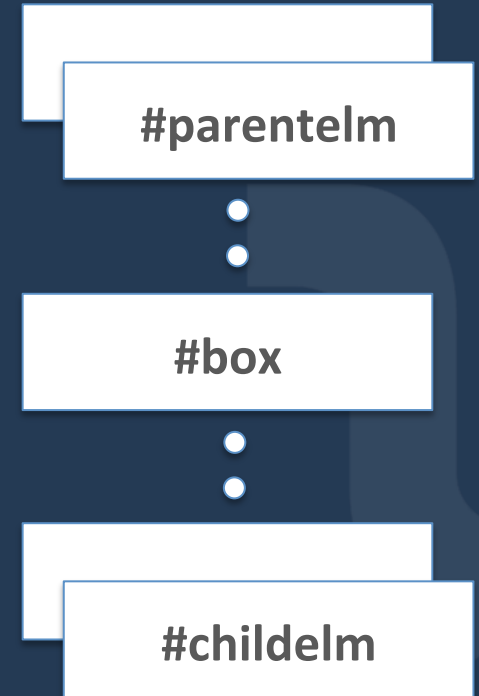
**Answer:** `$('#menu').children().first().next()`

```
<ul id="menu">  
  <li>Tacos</li>  
  <li>Pad Thai</li>  
  <li>Subway</li>  
  <li>Pizza</li>  
</ul>
```



# DOM Traversal - Searching

- What about pages with *zillions* of elements?
- **For Parent, Grandparent, etc**
  - `.closest('#parentelm')`
- **For Child, Grandchild, etc.**
  - `.find('#childelm')`





# DOM Traversal - Example

```
<div id="home-panel">
  <div class="container">
    <div class="row">
      <div class="span12">
        <div class="title">
          My Name
        </div>
      </div>
    </div>
  </div>
</div>
```

**This is a sample Twitter Bootstrap structure. How to access "My Name" element using \$('#home-panel')?**

`$('#home-panel').find('.title')`

# DOM Traversal – Use Cases

**Example 1:**

```
<div class="modal">
  <div class="x-button">X</div>
  <div class="content">
    This is a Modal!
  </div>
</div>
```

```
$('.x-button').click(function() {
  $(this).parent().fadeOut();
});
```

# DOM Traversal – Use Cases

## Example 2:

```
<div class="panels">
  <div id="home" class="panel"> ... </div>
  <div id="about" class="panel"> ... </div>
</div>
<a href="#home">
  <div class="button">Home</div>
</a>
<a href="#about">
  <div class="button">About</div>
</a>
```

```
$('.button').click(function() {
  var elm = $(this).parent().attr('href');
  $('.panels').find(elm).fadeIn();
});
```

## In CSS:

```
.panel {
  display: none;
}
```



# *Demo*

Follow along:  
[jsfiddle.net/6QS7J/](https://jsfiddle.net/6QS7J/)



# Event Objects



# Event Objects

## What are Event Objects?

```
$('.button').click(function(event) {  
    ....  
});
```

They give you **special functions** to use. Such as getting the X and Y value of when you clicked `‘.button’`!

```
$('.button').click(function(event) {  
    alert(event.pageX + “ and “ + event.pageY);  
});
```

# Event Objects – preventDefault()

We will go over 2 important special functions. First is `preventDefault()`

```
<a id="button" href="www.google.com">  
  Google  
</a>
```

```
$('#button').click(function(event) {  
  event.preventDefault();  
});
```

Now when you click on `#button`, the browser won't take you to Google. It **prevents** the default action of `<a>` tags, which takes you to a new page.

# Event Objects – stopPropagation()

The second special function is `stopPropagation()`

```
<div id="box1">
  <div id="box2">
    Click Here!
  </div>
</div>
```

```
$('#box1').click(function() {
  $('#box2').css('background', 'red');
});
$('#box2').click(function(event) {
  event.stopPropagation();
  $('#box2').css('background', 'red');
});
```

Instead of doing *#box2's* click event **and** *#box1's* click event, we now only do *#box2's*. **.stopPropagation()** stops an event from calling parent's events



# Event Objects – stopPropagation()

Last week's modal example using `stopPropagation()`

```
<div id="black">
  <div id="modal">
    This is a Modal!
  </div>
</div>
```

```
$('#black').click(function() {
  $(this).fadeOut();
});
$('#modal').click(function(event) {
  event.stopPropagation();
});
```

Because *#modal* is inside *#black*, by default when you click *#modal* it will trigger *#black's* click event. To prevent this, we added a `stopPropagation` to *#modal*.

# Event Objects

Lastly, note **event** could be whatever you want to take the variable. **event** or just **e** is usually the industry standard.

```
<div id="black">
  <div id="modal">
    This is a Modal!
  </div>
</div>
```

```
$('#black').click(function() {
  $(this).fadeOut();
});
$('#modal').click(function(berkeley) {
  berkeley.stopPropagation();
});
```

# Summary

- jQuery functions
  - .addClass(), .attr(...)
- DOM Traversal
  - Usage of .child(), .parent(), etc.
  - .find() and .closest()
- Event Objects
  - .preventDefault()
  - .stopPropagation()

All lecture material, handouts, and homework can be found at:  
<http://www.thewebdesignworkshop.co>