

# **Animations**

# **jQuery Review**

# What is the difference?

- `attr` vs. `val`
- `addClass` vs. `toggleClass`
- `find` vs. `children`
- `html` vs. `append`
- `hide` vs. `remove`

# **Which method should you use?**

- To get the text in a text field?
- To get the href of a link?
- To add some text to the beginning of an element?
- To add a click event listener?

# What does it select?

- `$(' .highlighted' )`
- `$(' a, img' )`
- `$(' img.large' )`
- `$(' #list li' )`

# What's wrong?

```
// Log the URL of any link clicked.  
$(a).on('click', function() {  
    var url = $('this').attr('href');  
    console.log(url);  
})
```

# What's wrong?

```
// Log a message when a form is submitted.  
$( 'button' ).submit( function() {  
    event.preventDefault();  
    console.log( 'Form submitted' );  
});
```

# **Animations**



# Prerequisites

1. CSS Positioning
2. Dynamically changing the appearance of elements

# CSS Positioning

The `position` attribute tells the browser what strategy to use for layout elements on the page.

The `top`, `left`, `right`, and `bottom` attributes mean different things depending on the strategy you're using.

# CSS Positioning Strategies

- `inherit`: use position value of container
- `static`: normal layout
- `relative`: offsets from normal layout position
- `fixed`: offsets from browser window
- `absolute`: offsets from nearest container with a position attribute other than `static` ??!

# **Dynamically changing the appearance of elements**

1. Change the style attribute of the element
2. Change the class or id so that different CSS rules apply.

# 1) **Change the style attribute of the element**

What jQuery method can we use to update the style attribute?

# css method

- More convenient than writing `attr( 'style' )`
- Be careful of units.

# **Animations Lab**

# Exercises

Using the `css` method:

1. Get the border-width of the first list item on the page.
2. Change the border color of all the list items to red.
3. Write a script so that when you click a list item, its border-width is set to 10px.
4. [Stretch] Write a script so that when you click the rocket ship, it moves 10px to the right.



# Exercises

1. Make sure the rocket ship is using absolute positioning.
2. Figure out which container the rocket ships positioning will be relative to.
3. Change the page's CSS so that the rocket ship is positioned 500px from the top and the left of the page.
4. [Stretch] Write a script so the rocket ship appears wherever you click on the page.

# Change the style attribute

- **Advantage:** Ultimate power and flexibility
- **Disadvantage:** Mixes styles and code which can be ugly

## **2) Adding / Removing classes**

1. Create a CSS class before-hand with the style changes you will want to make.
2. Add that class to an element dynamically with jQuery.

*This is how "highlight" worked in our jQuery lab!*

# Adding / Removing classes

- **Advantage:** Keeps styling information out of JS. Can easily change the look without changing behaviour.
- **Disadvantage:** Less control. Can't make incremental changes.

# Exercises

1. Create a "big" CSS class that sets the rocket ship's width to 100px.
2. Add and remove the "big" class to the rocket ship to see it grow and shrink.
3. Make "item-1" have a red border by adding a class to it in the console. Now remove the class in the console.

# Animations

- Introduction
- Manual update loops
- Basic jQuery Animations
- Advanced jQuery Animations
- CSS animations with Transitions
- CSS animations with Keyframes

# **Introduction to Animations**

It's a wide world. We'll just get a taste of  
what's possible.

# **What is "Animation"?**

**"Updating the appearance of elements over time"**



# Animation Showcase

- Ball moves to cursor <http://jsfiddle.net/RwtHn/5/>
- Canvas snow effect <http://thecodeplayer.com/walkthrough/html5-canvas-snow-effect>

<http://www.snownews.org/demo>

- Keyframes <http://www.impressivewebs.com/demo-files/css3-animated-scene/>
- `animate.css`

# Manual Update Loops

1. Define some code to update the scene
2. Use `setInterval` to run update code 30 times per second

# Exercises

1. Make the rocket ship grow twice as quickly.
2. Animate the rocket ship launching into the air using an animation loop.

# Basic jQuery Animations

1. Add timing parameters to effect method.
2. jQuery handles the updates and looping for you.

```
$( '.shape' ).slideUp(500);
```

```
$( '.shape' ).fadeOut( 'slow' );
```

# Exercises

1. Make "item-1" slide up over 1 second.
2. Make "item-2" fade out slowly using the 'slow' parameter.
3. Toggle "item-3" quickly, using the 'fast' parameter.

# Experiment

Write a script that:

1. Fades out item 1 over 5 seconds.
2. Fades out item 2 in 5 seconds.
3. Do the animations run at the same time or one after the other?

# Sequential Animations

**Q:** How can you start one animation after another one is complete?

**A:** Provide an `onComplete` callback as last parameter to effects.

# Exercises

Write a script that:

1. Fades out item-1 in 1 second
2. THEN fade out item-2 in 1 second
3. THEN fade out item-3 in 1 second



# **Transitions (Tweening)**

# Transitions (Tweening)

1. Manual
2. jQuery Animate
3. CSS Transitions

# **1. Manual Transitions**

## 2. jQuery Animate

- <http://api.jquery.com/animate/>
- Convenient but not great.

# 3. CSS Transitions

- Smoooooooooth! (Hardware accelerated)
- Reference: <https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/UsingCSSTransitions>

# CSS Transitions

1. Tell the browser how to transition a property.
2. Change the value of the property (using a class, or manually)
3. The browser will "tween" the value for you.

# Transition Attribute

transition: [attribute] [duration]

transition: opacity 0.3s

# CSS Transitions by Adding Classes

```
li {  
  /* 1) Set the original property value */  
  opacity: 0;  
  /* 2) Describe how to transition that property */  
  transition: opacity 0.5s;  
}  
  
/* 3) Create a class that sets the final property value */  
li.active {  
  opacity: 1;  
}
```

Works great with: opacity, width, height, left, top.



# Exercises

Using a CSS transition and a helper class:

1. When "Button 1" is clicked, animate the rocket ship to double its size. When clicked again, return it to its previous size.
2. When "Button 2" is clicked, animate the rocket ship to slide to the right 100px, and back when clicked again.
3. When the rocket ship is clicked, animate it to fade out (using CSS transitions).

# Exercises

1. Use the `css` method to animate the rocket 10px to the right in the console.
2. **Rocket driver:** Make the arrow keys animate the rocket left, up, right and down.

**Demo**

Animated ball

# jQuery UI

- <http://jqueryui.com/>
- Experiment with site
- Sortables

# Exercises

1. Make the two lists sortable
2. Enable dropping items from one list into the other.

# CSS Keyframes

*For more complex, multi-step, animations.*

## **Examples:**

- animate.css
- Sunrise: <http://www.impressivewebs.com/demo-files/css3-animated-scene/>

# CSS Keyframes References

- <http://css-tricks.com/snippets/css/keyframe-animation-syntax/>
- <http://www.smashingmagazine.com/2011/05/17/an-introduction-to-css3-keyframe-animations/>
- [http://www.w3schools.com/css/css3\\_animations.asp](http://www.w3schools.com/css/css3_animations.asp)

```
/* 1) Create animation keyframes */
@keyframes drive {
    0%      { left: 0; bottom: 0; }
    50%     { left: 0; bottom: 200px; }
    100%    { left: 200px; bottom: 200px; }
}
```

```
/* 2) Create a class to add the animations dynamically */
#rocket.animate {
    animation: drive 5s;
}
```

```
/* 3) Add -webkit- prefixes */
```



# Exercise

- Make your own crazy 'drive' animation for the rocket ship, and execute it when "button-1" is clicked.

# Vendor prefixing

–webkit, –moz, –o

- <http://pleeease.io/play/>
- <http://leaverou.github.io/prefixfree/>

# Common Javascript Recipes

- Validating forms (w/ html5 and w/ jQuery)
- Star / Un-star
- Tabs
- Show more
- Tooltips & Modals w/ Bootstrap

# Demo

Sortable FAQs app

# Homework

- See homework.pdf