

JQUERY

Manipulating The DOM

Ashok Dudhat
30th July 2018

Agenda

- Manipulating the DOM
- Manipulating the element contents
- Add and Remove Attribute to element
- Add DOM elements relative to selected elements
- Create or Clone elements
- The remove(), detach() and empty() functions
- The wrap() and wrapAll() functions
- Explicit iterations using each()

Manipulating The DOM

jQuery can be used to
manipulate the DOM

add and remove elements

change attributes

update the content of elements

EXAMPLE

DomManipulation.html

DomManipulation.html

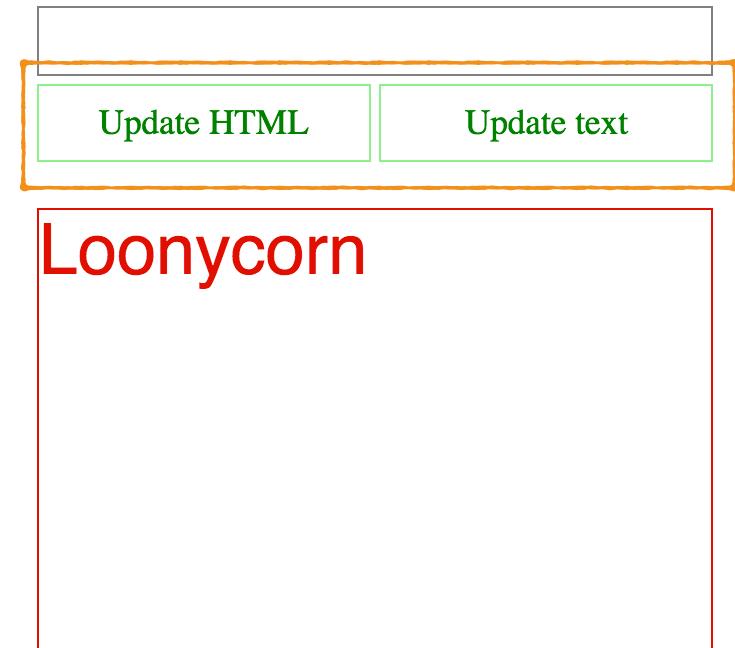
```
<body>
<input id="htmlmarkup">
<br>
<div class="updatehtml link">
    Update HTML
</div>
<div class="updatetext link">
    Update text
</div>
<div class="text">
    Loonycorn
</div>
<br>
</body>
```

The screenshot shows a web page with the following structure:

- A large orange-outlined box contains a white input field.
- Below the input field is a green-outlined button labeled "Update HTML".
- Next to it is another green-outlined button labeled "Update text".
- To the right of these buttons is a red-outlined box containing the text "Loonycorn".

DomManipulation.html

```
<body>
<input id="htmlmarkup">
<br>
<div class="updatehtml link">
    Update HTML
</div>
<div class="updatetext link">
    Update text
</div>
<div class="text">
    Loonycorn
</div>
<br>
</body>
```



DomManipulation.html

```
<body>
<input id="htmlmarkup">
<br>
<div class="updatehtml link">
    Update HTML
</div>
<div class="updatetext link">
    Update text
</div>
<div class="text">
    Loonycorn
</div>
<br>
</body>
```



DomManipulation.html

```
$(document).ready(function () {
    $('.updatehtml').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').html(htmlMarkup);
    });
    $('.updatetext').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').text(htmlMarkup);
    });
});
```

This button sets the HTML content on an element

DomManipulation.html

```
$(document).ready(function () {
    $('.updatehtml').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').html(htmlMarkup);
    });
    $('.updatetext').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').text(htmlMarkup);
    });
});
```

Extract the user input from
the input box

DomManipulation.html

```
$(document).ready(function () {
    $('.updatehtml').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').html(htmlMarkup);
    });
    $('.updatetext').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').text(htmlMarkup);
    });
});
```

Using the `.val()` function

DomManipulation.html

```
$(document).ready(function () {
    $('.updatehtml').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').html(htmlMarkup);
    });
    $('.updatetext').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').text(htmlMarkup);
    });
});
```

Set the content of the element via the **html()** function

DomManipulation.html

```
$(document).ready(function () {
    $('.updatehtml').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').html(htmlMarkup);
    });
    $('.updatetext').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').text(htmlMarkup);
    });
});
```

This parses the HTML markup
and sets it correctly as the
content

DomManipulation.html

```
$(document).ready(function () {
    $('.updatehtml').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').html(htmlMarkup);
    });
    $('.updatetext').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').text(htmlMarkup);
    });
});
```

The `.text()` function does not understand HTML

DomManipulation.html

```
$(document).ready(function () {
    $('.updatehtml').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').html(htmlMarkup);
    });
    $('.updatetext').click(function() {
        var htmlMarkup = $('input').val();
        $('.text').text(htmlMarkup);
    });
});
```

It escapes the HTML markup
in the input as text

Task : The To Do List



Html Example

```
<div id="myDIV" class="header">
  <h2>My To Do List</h2>
  <input type="text" id="myInput" placeholder="Title...">
  <span class="addBtn">Add</span>
</div>

<ul id="myUL">
  <li>Hit the gym</li>
  <li class="checked">Pay bills</li>
  <li>Meet George</li>
  <li>Buy eggs</li>
  <li>Read a book</li>
  <li>Organize office</li>
</ul>
```

EXAMPLE

AttrAndRemoveAttr.html

AttrAndRemoveAttr.html

```
<body>
<div class="add link">
    Add
</div>
<div class="remove link">
    Remove
</div>
<div class="text" id="loonyid" style="color:green; font-size:22px">
    Loonycorn
</div>
<br>
</body>
```

Add

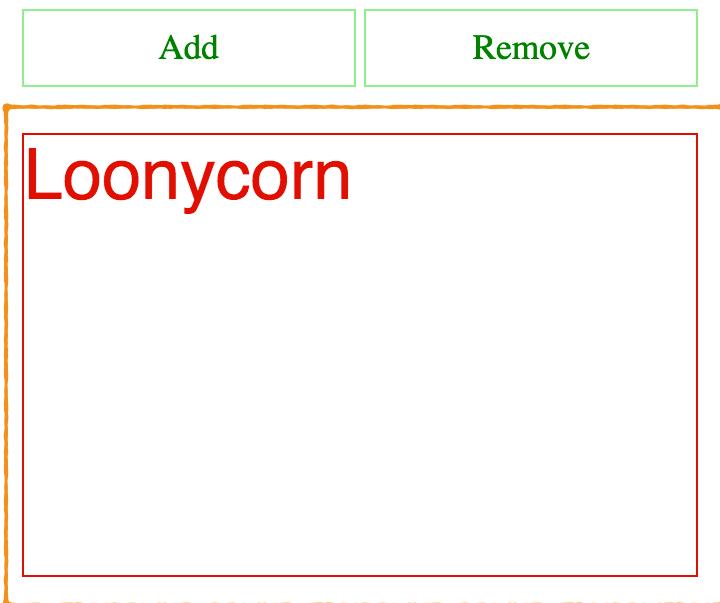
Remove

Loonycorn

AttrAndRemoveAttr.html

```
<body>
<div class="add link">
    Add
</div>
<div class="remove link">
    Remove
</div>
<div class="text" id="loonyid" style="color:green; font-size:22px">
    Loonycorn
</div>
<br>
</body>
```

This is the element
on which we add and
remove attributes



AttrAndRemoveAttr.html

```
$(document).ready(function () {
  $('.add').click(function() {
    var attributes = $('.text').attr('id');
    console.log(attributes);

    $('.text').attr('id', 'differentid');
    $('.text').attr('title', 'Loonycorn rocks!');
  });
  $('.remove').click(function() {
    $('#differentid').removeAttr('style');
  });
});
```

Add

Remove

Loonycorn

The add button changes the id
and the title attributes of the
element

AttrAndRemoveAttr.html

```
$(document).ready(function () {
  $('.add').click(function() {
    var attributes = $('.text').attr('id');
    console.log(attributes);

    $('.text').attr('id', 'differentid');
    $('.text').attr('title', 'Loonycorn rocks!');
  });
  $('.remove').click(function() {
    $('#differentid').removeAttr('style');
  });
});
```

Add

Remove

Loonycorn

The id is now differentid

AttrAndRemoveAttr.html

```
$(document).ready(function () {
  $('.add').click(function() {
    var attributes = $('.text').attr('id');
    console.log(attributes);

    $('.text').attr('id', 'differentid');
    $('.text').attr('title', 'Loonycorn rocks!');
  });
  $('.remove').click(function() {
    $('#differentid').removeAttr('style');
  });
});
```

Add

Remove

Loonycorn

This is what we use in the remove handler to access the element

AttrAndRemoveAttr.html

```
$(document).ready(function () {
  $('.add').click(function() {
    var attributes = $('.text').attr('id');
    console.log(attributes);

    $('.text').attr('id', 'differentid');
    $('.text').attr('title', 'Loonycorn rocks!');
  });
  $('.remove').click(function() {
    $('#differentid').removeAttr('style');
  });
});
```

Add

Remove

Loonycorn

This removes the **style** attribute on the element

AttrAndRemoveAttr.html

```
$(document).ready(function () {  
    $('.add').click(function() {  
        var attributes = $('.text').attr('id');  
        console.log(attributes);  
  
        $('.text').attr('id', 'differentid');  
        $('.text').attr('title', 'Loonycorn rocks!');  
    });  
    $('.remove').click(function() {  
        $('#differentid').removeAttr('style');  
    });  
});
```

Add

Remove

Loonycorn

This removes the explicit
green color on the text

EXAMPLE

AddingDomElements.html

AddingDomElements.html

```
<body>
<input id="userinput">
<br>
<div class="insertAfter link">
    Insert at end
</div>
<div class="after link">
    Insert at end
</div>
<br>
<div class="insertBefore link">
    Insert at start
</div>
<div class="before link">
    Insert at start
</div>
<br>
<div class="appendTo link">
    First becomes last
</div>
<div class="append link">
    First becomes last
</div>
<br>
<div class="text">
    <ul id="contendersList">
        <li>
            Hillary Clinton
        </li>
        <li>
            Donald Trump
        </li>
    </ul>
</div>
<br>
</body>
</html>
```

Insert at end	Insert at end
Insert at start	Insert at start
First becomes last	First becomes last

- Hillary Clinton
- Donald Trump

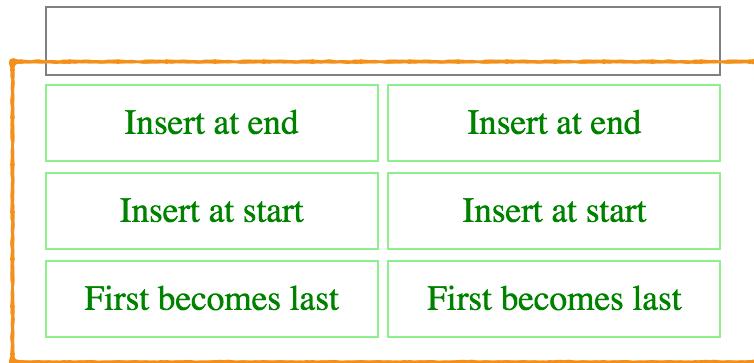
AddingDomElements.html

Insert at end	Insert at end
Insert at start	Insert at start
First becomes last	First becomes last

- Hillary Clinton
- Donald Trump

There are specific functions which allow for each of these actions

AddingDomElements.html



- Hillary Clinton
- Donald Trump

2 buttons with the same message exist because there are 2 ways to perform each action

AddingDomElements.html

```
$(document).ready(function () {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertBefore('#contendersList li:first');
    });
    $('.before').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });
    $('.appendTo').click(function() {
        $('#contendersList li:first').appendTo('#contendersList');
    });
    $('.append').click(function() {
        $('#contendersList').append($('#contendersList li:first'));
    });
});
```

AddingDomElements.html

```
$(document).ready(function () {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertAfter('#contendersList li:last');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.appendTo').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').append(value);
    });
    $('.prependTo').click(function() {
        var value = $('input').val();
        $('#contendersList').prepend($('#contendersList li:first'));
    });
});
```

We insert a new name in specified
in the input box after the last
element in the list

AddingDomElements.html

```
$(document).ready(function () {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        ('<li>' + value + '</li>').insertBefore('#contendersList li:first');
    });
    $('.prepend').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });
    $('.appendTo').click(function() {
        $('#contendersList li:first').appendTo('#contendersList');
    });
    $('.append').click(function() {
        $('#contendersList').append($('#contendersList li:first'));
    });
});
```

Extract the value from the input
box

AddingDomElements.html

```
$(document).ready(function () {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertBefore('#contendersList li:first');
    });
    $('.prepend').click(function() {
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });
    $('.append').click(function() {
        $('#contendersList').append($('#contendersList li:first'));
    });
});
```

Create a list item with the value
we just got from the input box

AddingDomElements.html

```
$(document).ready(function () {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertBefore('#contendersList li:first');
    });
    $('.before').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });
    $('.append').click(function() {
        $('#contendersList').append($('#contendersList li:first'));
    });
});
```

jQuery allows us to create items on the fly using the \$ selector!

AddingDomElements.html

```
$(document).ready(function () {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').insertBefore('<li>' + value + '</li>');
    });
    $('.before').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });
    $('.appendTo').click(function() {
        $('#contendersList li:first').appendTo($('#contendersList'));
    });
    $('.prepend').click(function() {
        $('#contendersList').prepend($('#contendersList li:first'));
    });
});
```

Insert the newly created element
after the last item in the list (so it
becomes to the new last item)

AddingDomElements.html

```
$(document).ready(function () {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').before('<li>' + value + '</li>');
    });
    $('.appendTo').click(function() {
        $('#contendersList').append($('#contendersList'));
    });
    $('.append').click(function() {
        $('#contendersList').append($('#contendersList li:first'));
    });
});
```

li:last is a quick way to select the last item within the #contendersList

AddingDomElements.html

```
$(document).ready(function () {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertBefore('#contendersList li:first');
    });
    $('.append').click(function() {
        $('#contendersList').append($('#contendersList li:first'));
    });
});
```

insertAfter inserts an element relative to other elements

```
$(document).ready(function() {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertAfter('#contendersList li:last');
    });

    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });

    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertBefore('#contendersList li:first');
    });
    $('.before').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });

    $('.appendTo').click(function() {
        $('#contendersList').append('<li>' + value + '</li>');
    });

    $('.append').click(function() {
        $('#contendersList').append($('- ' + value + '
'));
    });
});
```

This is another way to accomplish exactly the same thing

```
$(document).ready(function() {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('- ' + value + '
').insertBefore('#contendersList li:first');
    });
    $('.before').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });
});
```

Select the element relative to which we place the newly created element

AddingDomElements.html

```
$(document).ready(function() {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertBefore('#contendersList li:first');
    });
    $('.before').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });
    $('.appendTo').click(function() {
        $('#contendersList').append($('input').val());
    });
    $('.append').click(function() {
        $('#contendersList').append($('input').val());
    });
});
```

Use the **after()** function rather
than the **insertAfter()**

AddingDomElements.html

```
$(document).ready(function() {
    $('.insertAfter').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertAfter('#contendersList li:last');
    });
    $('.after').click(function() {
        var value = $('input').val();
        $('#contendersList li:last').after('<li>' + value + '</li>');
    });
    $('.insertBefore').click(function() {
        var value = $('input').val();
        $('<li>' + value + '</li>').insertBefore('#contendersList li:first');
    });
    $('.before').click(function() {
        var value = $('input').val();
        $('#contendersList li:first').before('<li>' + value + '</li>');
    });
    $('.appendTo').click(function() {
        $('#contendersList li:first').appendTo('#contendersList');
    });
    $('.prepend').click(function() {
        $('#contendersList').prepend($('#contendersList li:first'));
    });
});
```

Create the new element

```
});  
$('.after').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:last').after('<li>' + value + '</li>');  
});  
$('.insertBefore').click(function() {  
    var value = $('input').val();  
    ('<li>' + value + '</li>').insertBefore('#contendersList li:first');  
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
});  
$('.appendTo').click(function() {  
    $('#contendersList li:first').appendTo('#contendersList');  
});  
$('.append').click(function() {  
    $('#contendersList').append($('#contendersList li:first'));  
});  
});
```

The `insertBefore()` and `before()` functions work in similar ways

```
});  
$('.after').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:last').after('<li>' + value + '</li>');  
});  
$('.insertBefore').click(function() {  
    var value = $('input').val();  
    ('<li>' + value + '</li>').insertBefore('#contendersList li:first');  
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
});  
$('.appendTo').click(function() {  
    $('#contendersList li:first').appendTo('#contendersList');  
});  
$('.append').click(function() {  
    $('#contendersList').append($('#contendersList li:first'));  
});  
});
```

Create the element and insert it before the first element in the list

```
});  
$('.after').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:last').after('<li>' + value + '</li>');  
});  
$('.insertBefore').click(function() {  
    var value = $('input').val();  
    $('- ' + value + '
').insertBefore('#contendersList li:first');  
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
});  
$('.appendTo').click(function() {  
    $('#contendersList li:first').appendTo('#contendersList');  
});  
$('.append').click(function() {  
    $('#contendersList').append($('#contendersList li:first'));  
});  
});
```

Create the element and insert it before the first element in the list

```
});  
$('.after').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:last').after('<li>' + value + '</li>');  
});  
$('.insertBefore').click(function() {  
    var value = $('input').val();  
    ('<li>' + value + '</li>').insertBefore('#contendersList li:first');  
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
});  
$('.appendTo').click(function() {  
    $('#contendersList li:first').appendTo('#contendersList');  
});  
$('.append').click(function() {  
    $('#contendersList').append($('#contendersList li:first'));  
});  
});
```

Select the first element and have the new element before it

```
});  
$('.after').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:last').after('<li>' + value + '</li>');  
});  
$('.insertBefore').click(function() {  
    var value = $('input').val();  
    '<li>' + value + '</li>').insertBefore('#contendersList li:first');  
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
});  
$('.appendTo').click(function() {  
    $('#contendersList li:first').appendTo('#contendersList');  
});  
$('.append').click(function() {  
    $('#contendersList').append($('#contendersList li:first'));  
});  
});
```

Select the first element and have the new element before it

```
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
  
    $('.appendTo').click(function() {  
        $('#contendersList li:first').appendTo('#contendersList');  
    });  
    $('.append').click(function() {  
        $('#contendersList').append($('#contendersList li:first'));  
    });  
});
```

There are exact equivalent
methods for **append()** and **prepend()**

```
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
  
    $('.appendTo').click(function() {  
        $('#contendersList li:first').appendTo('#contendersList');  
    });  
    $('.append').click(function() {  
        $('#contendersList').append($('#contendersList li:first'));  
    });  
});
```

AddingDomElements.html

append() and appendTo()

prepend() and prependTo()

```
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
  
    $('.appendTo').click(function() {  
        $('#contendersList li:first').appendTo('#contendersList');  
    });  
    $('.append').click(function() {  
        $('#contendersList').append($('#contendersList li:first'));  
    });  
});
```

AddingDomElements.html

In this code we do not create any new elements

```
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
});  
$('.appendTo').click(function() {  
    $('#contendersList li:first').appendTo('#contendersList');  
});  
$('.append').click(function() {  
    $('#contendersList').append($('#contendersList li:first'));  
});  
});
```

This takes the first element and appends it to the end of the list

```
}); //  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
});  
$('.appendTo').click(function() {  
    $('#contendersList li:first').appendTo('#contendersList');  
});  
$('.append').click(function() {  
    $('#contendersList').append($('#contendersList li:first'));  
});  
});
```

AddingDomElements.html

Select the first element

```
}); //
```

```
$('.before').click(function() {
    var value = $('input').val();
    $('#contendersList li:first').before('<li>' + value + '</li>');
});
$('.appendTo').click(function() {
    $('#contendersList li:first').appendTo('#contendersList');
});
$('.append').click(function() {
    $('#contendersList').append($('#contendersList li:first'));
});
});
```

Append i.e. add it to the end of
the list

```
});  
$('.before').click(function() {  
    var value = $('input').val();  
    $('#contendersList li:first').before('<li>' + value + '</li>');  
});  
$('.appendTo').click(function() {  
    $('#contendersList li:first').appendTo('#contendersList');  
});  
$('.append').click(function() {  
    $('#contendersList').append($('#contendersList li:first'));  
});  
});
```

AddingDomElements.html

The same action expressed in a slightly different way

After adding element

My To Do List	
Title...	Add
DCI	X
Students	X
Thamaria	X
HTML Course	X
Termin with doctor	X
German Class	X

After selecting element

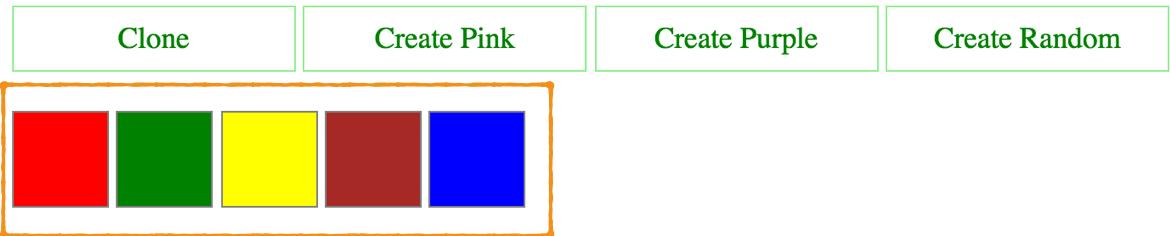
My To Do List		
Title...		Add
✓ BEI		X
Students		X
Thamata		X
HTML Course		X
Termn with doctor		X
German Class		X

EXAMPLE

CloningAndCreatingElements.html

CloningAndCreatingElements.html

```
<body>
<div class="clone link">
  Clone
</div>
<div class="createpink link">
  Create Pink
</div>
<div class="createpurple link">
  Create Purple
</div>
<div class="createrandom link">
  Create Random
</div>
<br>
<br>
<div id="colorboxes">
  <div class="red box">
  </div>
  <div class="green box">
  </div>
  <div class="yellow box">
  </div>
  <div class="brown box">
  </div>
  <div class="blue box">
  </div>
</div>
<br>
</body>
```



A whole bunch of colored boxes

CloningAndCreatingElements.html

```
<body>
<div class="clone link">
    Clone
</div>
<div class="createpink link">
    Create Pink
</div>
<div class="createpurple link">
    Create Purple
</div>
<div class="createrandom link">
    Create Random
</div>
<br>
<br>
<div id="colorboxes">
    <div class="red box">
    </div>
    <div class="green box">
    </div>
    <div class="yellow box">
    </div>
    <div class="brown box">
    </div>
    <div class="blue box">
    </div>
</div>
<br>
</body>
```



Clone and create new ones,
with specific colors or at
random

CloningAndCreatingElements.html

```
$(document).ready(function () {
    var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];

    $('.clone').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        $(className).first().clone().appendTo($('#colorboxes'));
    });

    $('.createpink').click(function() {
        var jqEl = $('

</div>');
        $('#colorboxes').append(jqEl);
    });

    $('.createpurple').click(function() {
        $('#colorboxes').append($('

</div>'));
    });

    $('.createrandom').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        className = className.substr(1);
        $('#colorboxes').append($('

</div>', {
            'class': className + ' box'
        }));
    });
});


```

CloningAndCreatingElements.html

```
$('document').ready(function() {
    var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];

    $('.clone').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        $(className).first().clone().appendTo($('#colorboxes'));
    });

    $('.createpink').click(function() {
        var jqEl = $('

</div>');
        $('#colorboxes').append(jqEl);
    });

    $('.createpurple').click(function() {
        $('#colorboxes').append($('

</div>'));
    });
});


```

Clone an existent box element to
create a new one

CloningAndCreatingElements.html

```
$();
var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];

$('.clone').click(function() {
    var className = classNames[Math.floor(Math.random() * classNames.length)];
    $(className).first().clone().appendTo($('#colorboxes'));
});

$('.createpink').click(function() {
    var jqEl = $('

</div>');
    $('#colorboxes').append(jqEl);
});

$('.createpurple').click(function() {
    $('#colorboxes').append($('

</div>'));
});


```

An array of class names for the various colored boxes

An array of class names for the various colored boxes

CloningAndCreatingElements.html

```
$(document).ready(function () {
    var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];

    $('.clone').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        $(className).first().clone().appendTo($('#colorboxes'));
    });

    $('.createpink').click(function() {
        var jqEl = $('

</div>');
        $('#colorboxes').append(jqEl);
    });

    $('.createpurple').click(function() {
        $('#colorboxes').append($('

</div>'));
    });

    $('.createorange').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        $('#colorboxes').append($('

</div>', {
            'class': className
        }));
    });
});


```

Choose one of the class names at random

CloningAndCreatingElements.html

```
$(document).ready(function () {  
    var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];  
  
    $('.clone').click(function() {  
        var className = classNames[Math.floor(Math.random() * classNames.length)];  
        $(className).first().clone().appendTo($('#colorboxes'));  
    });  
  
    $('.createpink').click(function() {  
        var jqEl = $('<div class="pink box"></div>');  
        $('#colorboxes').append(jqEl);  
    });  
  
    $('.createpurple').click(function() {  
        $('#colorboxes').append($('<div class="purple box"></div>'));  
    });  
  
    $('.createrandom').click(function() {  
        var className = classNames[Math.floor(Math.random() * classNames.length)];  
        var jqEl = $('<div class="random box"></div>');  
        jqEl.addClass(className + ' box');  
        $('#colorboxes').append(jqEl);  
    });  
});
```

Select an existing element which has that class applied and clone it

CloningAndCreatingElements.html

```
$(document).ready(function () {
    var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];

    $('.clone').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        $(className).first().clone().appendTo($('#colorboxes'));
    });

    $('.createpink').click(function() {
        var jqEl = $('<div class="pink box"></div>');
        $('#colorboxes').append(jqEl);
    });

    $('.createpurple').click(function() {
        $('#colorboxes').append($('<div class="purple box"></div>'));
    });

    $('.createrandom').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        var newElement = $(`<div class="${className} box"></div>`);
        $('#colorboxes').append(newElement);
    });
});
```

An existing element which has the class applied

CloningAndCreatingElements.html

```
$(document).ready(function () {
    var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];

    $('.clone').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        $(className).first().clone().appendTo($('#colorboxes'));
    });

    $('.createpink').click(function() {
        var jqEl = $('

</div>');
        $('#colorboxes').append(jqEl);
    });

    $('.createpurple').click(function() {
        $('#colorboxes').append($('

</div>'));
    });

    $('.createrandom').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        className = className.substr(1);
        $('#colorboxes').append($('

<div></div>').attr('class', className));
    });
});


```

The first of the selected elements

CloningAndCreatingElements.html

```
$(document).ready(function () {
    var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];

    $('.clone').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        $(className).first().clone().appendTo($('#colorboxes'));
    });

    $('.createpink').click(function() {
        var jqEl = $('

</div>');
        $('#colorboxes').append(jqEl);
    });

    $('.createpurple').click(function() {
        $('#colorboxes').append($('

</div>'));
    });

    $('.createrandom').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        className = className.substr(1);
        $('#colorboxes').append($('

</div>', [
            { 'class': className + ' box' }
        ]));
    });
});


```

Clone it

CloningAndCreatingElements.html

```
$(document).ready(function () {
    var classNames = ['.red', '.green', '.yellow', '.brown', '.blue'];

    $('.clone').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        $(className).first().clone().appendTo($('#colorboxes'));
    });

    $('.createpink').click(function() {
        var jqEl = $('

</div>');
        $('#colorboxes').append(jqEl);
    });

    $('.createpurple').click(function() {
        $('#colorboxes').append($('

</div>'));
    });

    $('.createrandom').click(function() {
        var className = classNames[Math.floor(Math.random() * classNames.length)];
        var jqEl = $('

</div>');
        jqEl.attr('class', className + ' box');
        $('#colorboxes').append(jqEl);
    });
});


```

Append the clone to the outer <div>

CloningAndCreatingElements.html

```
var className = classNames[Math.floor(Math.random() * classNames.length)];  
$(className).first().clone().appendTo($('#colorboxes'));  
});  
  
$('.createpink').click(function() {  
    var jqEl = $('

</div>');  
    $('#colorboxes').append(jqEl);  
});  
  
$('.createpurple').click(function() {  
    $('#colorboxes').append($('

</div>'));  
});  
  
$('.createrandom').click(function() {  
    var className = classNames[Math.floor(Math.random() * classNames.length)];  
    className = className.substr(1);  
    $('#colorboxes').append($('

</div>', {  
        'class': className + ' box'  
    }));  
});  
});


```

Create a new pink box to add to the DOM

CloningAndCreatingElements.html

```
var className = classNames[Math.floor(Math.random() * classNames.length)];  
$(className).first().clone().appendTo($('#colorboxes'));  
});  
  
$('.createpink').click(function() {  
    var jqEl = $('

</div>');  
    $('#colorboxes').append(jqEl);  
});  
  
$('.createpurple').click(function() {  
    $('#colorboxes').append($('

</div>'));  
});  
  
$('.createrandom').click(function() {  
    var className = classNames[Math.floor(Math.random() * classNames.length)];  
    className = className.substr(1);  
    $('#colorboxes').append($('

</div>', {  
        'class': className + ' box'  
}));  
});


```

Create an element on the fly - just specify the HTML as you would a selector

CloningAndCreatingElements.html

```
var className = classNames[Math.floor(Math.random() * classNames.length)];  
$(className).first().clone().appendTo($('#colorboxes'));  
});  
  
$('.createpink').click(function() {  
    var jqEl = $('

</div>');  
    $('#colorboxes').append(jqEl);  
});  
  
$('.createpurple').click(function() {  
    $('#colorboxes').append($('

</div>'));  
});  
  
$('.createrandom').click(function() {  
    var className = classNames[Math.floor(Math.random() * classNames.length)];  
    className = className.substr(1);  
    $('#colorboxes').append($('

</div>', {  
        'class': className + ' box'  
    }));  
});  
});


```

Assign it to a variable - the element has been created but not added to the DOM yet

CloningAndCreatingElements.html

```
var className = classNames[Math.floor(Math.random() * classNames.length)];
$(className).first().clone().appendTo($('#colorboxes'));

$('.createpink').click(function() {
    var jqEl = $('<div class="pink_box"></div>');
    $('#colorboxes').append(jqEl);
});

$('.createpurple').click(function() {
    $('#colorboxes').append($('<div class="purple_box"></div>'));
});

$('.createrandom').click(function() {
    var className = classNames[Math.floor(Math.random() * classNames.length)];
    className = className.substr(1);
    $('#colorboxes').append($('<div></div>', {
        'class': className + ' box'
    }));
});
});
```

Append the element to DOM

CloningAndCreatingElements.html

```
$('.createpink').click(function() {
    $('#colorboxes').append(jqEl);
});

$('.createpurple').click(function() {
    $('#colorboxes').append($('<div class="purple_box"></div>'));
});

$('.createrandom').click(function() {
    var className = classNames[Math.floor(Math.random() * classNames.length)];
    className = className.substr(1);
    $('#colorboxes').append($('<div></div>', {
        'class': className + ' box'
    }));
});
});
```

A more concise way to create and append an element to DOM

CloningAndCreatingElements.html

```
$(' .createtwo').click(function() {
    $('#colorboxes').append($('.classestwo').clone(true));
});

$('.createrandom').click(function() {
    var className = classNames[Math.floor(Math.random() * classNames.length)];
    className = className.substr(1);
    $('#colorboxes').append($('', {
        'class': className + ' box'
    }));
});
```

Create an element and specify its attributes dynamically

CloningAndCreatingElements.html

```
$(' .createtwo').click(function() {
    $('#colorboxes').append('<div></div>');
});

$('.createrandom').click(function() {
    var className = classNames[Math.floor(Math.random() * classNames.length)];
    className = className.substr(1);
    $('#colorboxes').append('<div></div>', {
        'class': className + ' box'
    });
});
});
```

Choose a class name at random (a random colored box)

CloningAndCreatingElements.html

```
$('button').click(function() {
    $('#colorboxes').append($('div').addClass('purple_box'));
});

$('.createrandom').click(function() {
    var className = classNames[Math.floor(Math.random() * classNames.length)];
    className = className.substr(1);
    $('#colorboxes').append($('', {
        'class': className + ' box'
    }));
});
});
```

Append a `<div>` with this class name

CloningAndCreatingElements.html

```
$(' .createtwo').click(function() {
    $('#colorboxes').append($(' <div> class=purple_box </div> '));
});

$('.createrandom').click(function() {
    var className = classNames[Math.floor(Math.random() * classNames.length)];
    className = className.substr(1);
    $('#colorboxes').append($(' <div></div> ', {
        'class': className + ' box'
    }));
});
});
```

Specify the attributes applied to
the `<div>` in the form of a
properties object

EXAMPLE

RemoveDetachEmptyElements.html

RemoveDetachEmptyElements.html

Events can be removed from the
DOM in 2 ways

remove

detach

RemoveDetachEmptyElements.html

remove

This removes the element from
the DOM and also removes all
event handlers!

RemoveDetachEmptyElements.html

remove

This removes the element from
the DOM and also removes all
event handlers!

If you re-attach the element
then the original event handlers
will no longer exist

RemoveDetachEmptyElements.html

Events can be removed from the
DOM in 2 ways

remove

detach

RemoveDetachEmptyElements.html

detach

This removes the element from the
DOM but preserves the event
handlers which exist on the element

RemoveDetachEmptyElements.html

detach

This removes the element from the
DOM but preserves the event
handlers which exist on the element

If you re-add the element then we
get back its original state
complete with event handlers

RemoveDetachEmptyElements.html

Events can be removed from the
DOM in 2 ways

remove

detach

RemoveDetachEmptyElements.html

remove

detach

It's often useful to remove/
detach elements from the DOM
before manipulation changes on
the element

RemoveDetachEmptyElements.html

remove

detach

This avoids the very expensive
re-rendering of the page when
the changes are made

RemoveDetachEmptyElements.html

remove

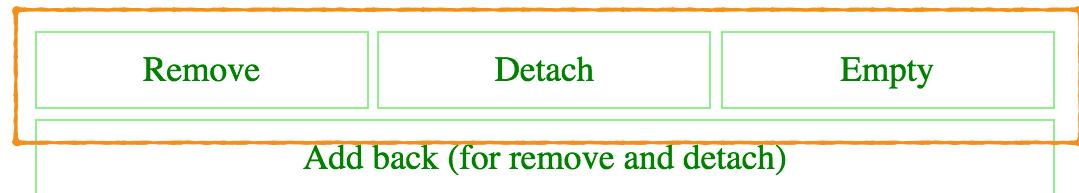
detach

This avoids the very expensive
re-rendering of the page when
the changes are made

Once changes are complete you can
add it back leading to a one time
render - this is much more performant

RemoveDetachEmptyElements.html

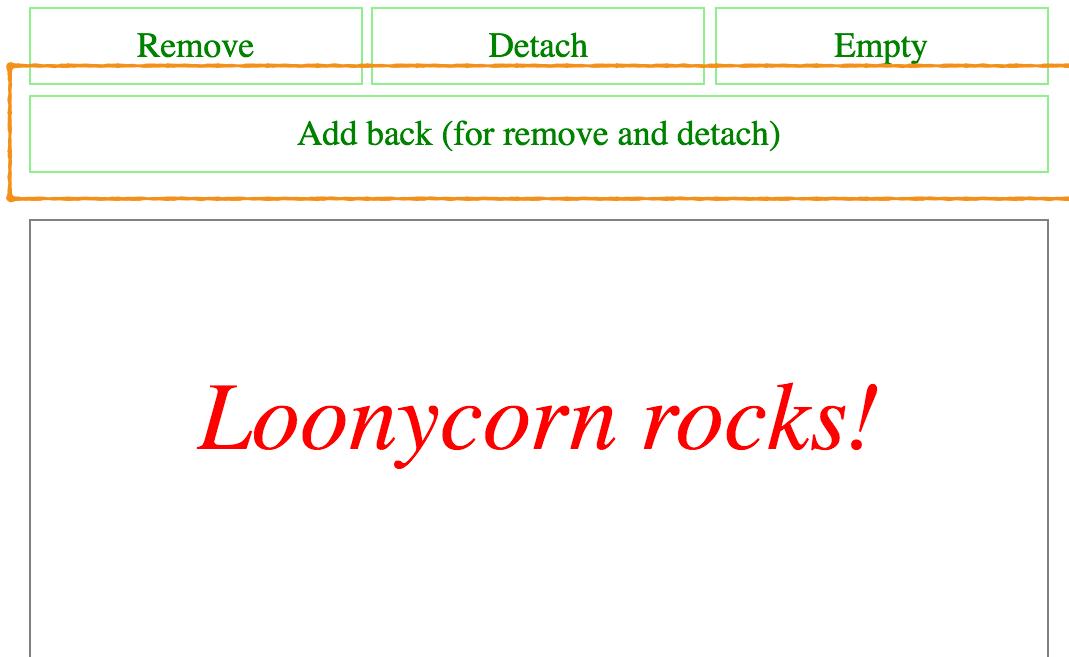
```
<body>
<div class="remove link">
  Remove
</div>
<div class="detach link">
  Detach
</div>
<div class="empty link">
  Empty
</div>
<br>
<div class="link add">
  Add back (for remove and detach)
</div>
<br>
<br>
<div id="container">
  <div class="box">
    <i>Loonycorn rocks!</i>
  </div>
</div>
<br>
</body>
```



Loonycorn rocks!

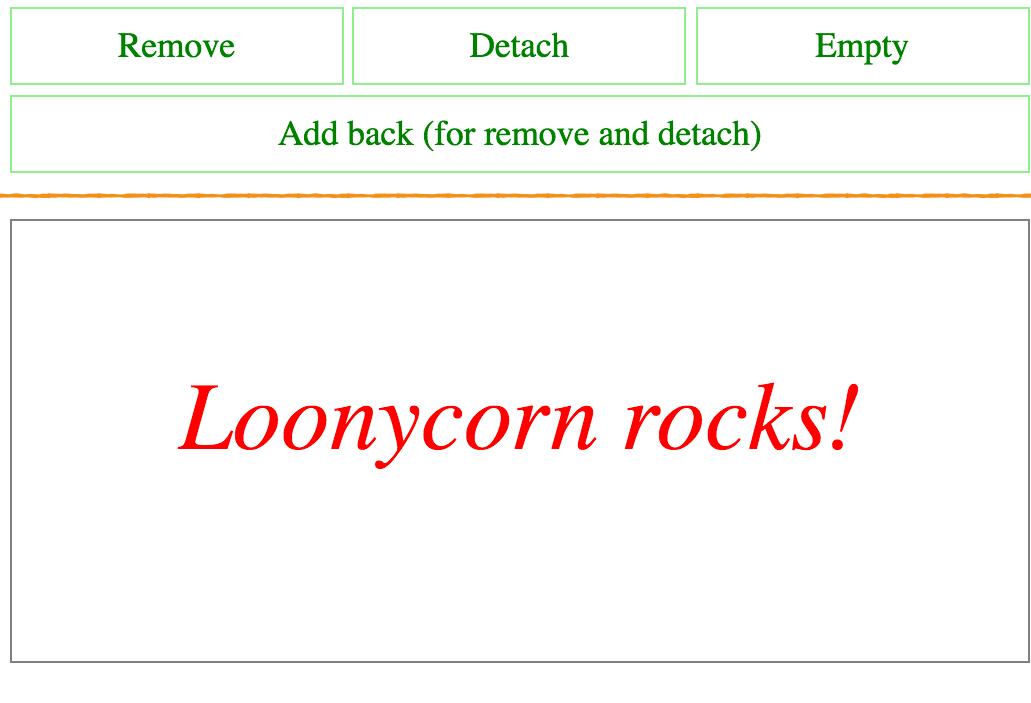
RemoveDetachEmptyElements.html

```
<body>
<div class="remove link">
    Remove
</div>
<div class="detach link">
    Detach
</div>
<div class="empty link">
    Empty
</div>
<br>
<div class="link add">
    Add back (for remove and detach)
</div>
<br>
<br>
<div id="container">
    <div class="box">
        <i>Loonycorn rocks!</i>
    </div>
</div>
<br>
</body>
```



RemoveDetachEmptyElements.html

```
<body>
<div class="remove link">
  Remove
</div>
<div class="detach link">
  Detach
</div>
<div class="empty link">
  Empty
</div>
<br>
<div class="link add">
  Add back (for remove and detach)
</div>
<br>
<br>
<div id="container">
  <div class="box">
    <i>Loonycorn rocks!</i>
  </div>
</div>
<br>
</body>
```



RemoveDetachEmptyElements.html

```
$(document).ready(function () {
  var el = $('.box');

  el.click(function() {
    alert('Click event! And Loonycorn rocks!');
  });

  $('.remove').click(function() {
    el = $('.box').remove();
  });

  $('.detach').click(function() {
    el = $('.box').detach();
  });

  $('.empty').click(function() {
    $('.box').empty();
  });

  $('.add').click(function() {
    $('#container').append(el);
  });
});
```

All actions which apply to the element with the box class applied

RemoveDetachEmptyElements.html

```
$(document).ready(function () {
    var el = $('.box');

    el.click(function() {
        alert('Click event! And Loonycorn rocks!');
    });

    $('.remove').click(function() {
        el = $('.box').remove();
    });

    $('.detach').click(function() {
        el = $('.box').detach();
    });

    $('.empty').click(function() {
        $('.box').empty();
    });

    $('.add').click(function() {
        $('#container').append(el);
    });
});
```

Store a reference
to the element
which is to be
acted on

RemoveDetachEmptyElements.html

```
$(document).ready(function () {
  var el = $('.box');

  el.click(function() {
    alert('Click event! And Loonycorn rocks!');
  });

  $('.remove').click(function() {
    el = $('.box').remove();
  });

  $('.detach').click(function() {
    el = $('.box').detach();
  });

  $('.empty').click(function() {
    $('.box').empty();
  });

  $('.add').click(function() {
    $('#container').append(el);
  });
});
```

Add an alert on click, any event handler works as well for this

RemoveDetachEmptyElements.html

```
$(document).ready(function () {
    var el = $('.box');

    el.click(function() {
        alert('Click event! And Loonycorn rocks!');
    });

    $('.remove').click(function() {
        el = $('.box').remove();
    });

    $('.detach').click(function() {
        el = $('.box').detach();
    });

    $('.empty').click(function() {
        $('.box').empty();
    });

    $('.add').click(function() {
        $('#container').append(el);
    });
});
```

remove() and
detach() are
functions on the
jQuery object

RemoveDetachEmptyElements.html

```
$(document).ready(function () {
    var el = $('.box');

    el.click(function() {
        alert('Click event! And Loonycorn rocks!');
    });

    $('.remove').click(function() {
        el = $('.box').remove();
    });

    $('.detach').click(function() {
        el = $('.box').detach();
    });

    $('.empty').click(function() {
        $('.box').empty();
    });

    $('.add').click(function() {
        $('#container').append(el);
    });
});
```

Store a reference
to the jQuery object
which was removed
from the DOM

We'll need it to add
it back

RemoveDetachEmptyElements.html

```
$(document).ready(function () {
    var el = $('.box');

    el.click(function() {
        alert('Click event! And Loonycorn rocks!');
    });

    $('.remove').click(function() {
        el = $('.box').remove();
    });

    $('.detach').click(function() {
        el = $('.box').detach();
    });

    $('.empty').click(function() {
        $('.box').empty();
    });

    $('.add').click(function() {
        $('#container').append(el);
    });
});
```

The empty()
function just
removes all the
content within the
selected element

RemoveDetachEmptyElements.html

```
$(document).ready(function () {
    var el = $('.box');

    el.click(function() {
        alert('Click event! And Loonycorn rocks!');
    });

    $('.remove').click(function() {
        el = $('.box').remove();
    });

    $('.detach').click(function() {
        el = $('.box').detach();
    });

    $('.empty').click(function() {
        $('.box').empty();
    });

    $('.add').click(function() {
        $('#container').append(el);
    });
});
```

It does not remove
the element itself

RemoveDetachEmptyElements.html

```
$(document).ready(function () {
    var el = $('.box');

    el.click(function() {
        alert('Click event! And Loonycorn rocks!');
    });

    $('.remove').click(function() {
        el = $('.box').remove();
    });

    $('.detach').click(function() {
        el = $('.box').detach();
    });

    $('.empty').click(function() {
        $('.box').empty();
    });

    $('.add').click(function() {
        $('#container').append(el);
    });
});
```

The add button
simply adds the
element back to the
DOM using append()

RemoveDetachEmptyElements.html

```
$(document).ready(function () {
  var el = $('.box');

  el.click(function() {
    alert('Click event! And Loonycorn rocks!');
  });

  $('.remove').click(function() {
    el = $('.box').remove();
  });

  $('.detach').click(function() {
    el = $('.box').detach();
  });

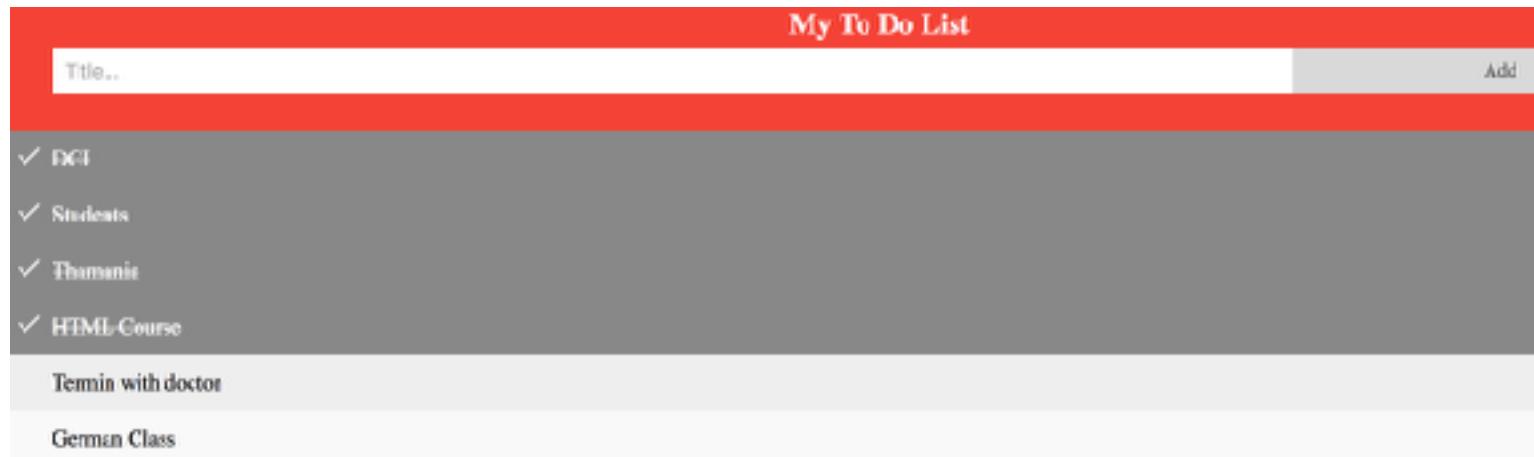
  $('.empty').click(function() {
    $('.box').empty();
  });

  $('.add').click(function() {
    $('#container').append(el);
  });
});
```

The event handlers
are preserved in the
case of `detach()` but
not in the case of
`remove()`

Task Continue...

After selecting multiple element



After deleting element

My To Do List	
Title	Add
✓ DCI	x
✓ Students	x
✓ Thomania	x
Termin with doctor	x
German Class	x

After deleting element



EXAMPLE

WrapAndWrapAll.html

WrapAndWrapAll.html

```
<body>
<div class="wrap link">
    Wrap
</div>
<div class="wrapall link">
    Wrap All
</div>
<br>
<br>
<div id="container">
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
</div>
<br>
</body>
```

Wrap

Wrap All

Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!

WrapAndWrapAll.html

```
$('document').ready(function () {
    $('.wrap').click(function() {
        $('.box').wrap('<div class="outer-box"></div>');
    });

    $('.wrapall').click(function() {
        $('.box').wrapAll('<div class="outer-box"></div>');
    });
});
```

The `wrap()` function wraps each selected element with the HTML specified

```
<body>
<div class="wrap link">
    Wrap
</div>
<div class="wrapall link">
    Wrap All
</div>
<br>
<br>
<div id="container">
    <div class="box">
        Loyncorn rocks!
    </div>
    <div class="box">
        Loyncorn rocks!
    </div>
    <div class="box">
        Loyncorn rocks!
    </div>
    <div class="box">
        Loyncorn rocks!
    </div>
</div>
<br>
</body>
```

WrapAndWrapAll.html

```
$('document').ready(function () {
  $('.wrap').click(function() {
    $('.box').wrap('<div class="outer-box"></div>');
  });
  $('.wrapall').click(function() {
    $('.box').wrapAll('<div class="outer-box"></div>');
  });
});
```

Select elements with the box class applied and wrap each with HTML.

```
<body>
<div class="wrap link">
  Wrap
</div>
<div class="wrapall link">
  Wrap All
</div>
<br>
<br>
<div id="container">
  <div class="box">
    Loyncorn rocks!
  </div>
  <div class="box">
    Loyncorn rocks!
  </div>
  <div class="box">
    Loyncorn rocks!
  </div>
  <div class="box">
    Loyncorn rocks!
  </div>
</div>
<br>
</body>
```

WrapAndWrapAll.html

```
$(document).ready(function () {
    $('.wrap').click(function() {
        $('.box').wrap('<div class="outer-box"></div>');
    });

    $('.wrapall').click(function() {
        $('.box').wrapAll('<div class="outer-box"></div>');
    });
});
```

```
<div class="outer-box">
    <div class="box">
        Loyncorn rocks!
    </div>
</div>
```

Each element gets
the outer-box as a
wrapper

WrapAndWrapAll.html

```
$(document).ready(function () {
    $('.wrap').click(function() {
        $('.box').wrap('<div class="outer-box"></div>');
    });

    $('.wrapall').click(function() {
        $('.box').wrapAll('<div class="outer-box"></div>');
    });
});
```

```
<div class="outer-box">
    <div class="box">
        Loonycorn rocks!
    </div>
</div>
```

Each element gets
the outer-box as a
wrapper

WrapAndWrapAll.html

```
$('document').ready(function () {
    $('.wrap').click(function() {
        $('.box').wrap('<div class="outer-box"></div>');
    });
    $('.wrapall').click(function() {
        $('.box').wrapAll('<div class="outer-box"></div>');
    });
});
```

The **wrapAll()** method adds the HTML as the parent of all the selected elements

```
<body>
<div class="wrap link">
    Wrap
</div>
<div class="wrapall link">
    Wrap All
</div>
<br>
<br>
<div id="container">
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
</div>
<br>
</body>
```

WrapAndWrapAll.html

```
$(document).ready(function () {
    $('.wrap').click(function() {
        $('.box').wrap('<div class="outer-box"></div>');
    });

    $('.wrapall').click(function() {
        $('.box').wrapAll('<div class="outer-box"></div>');
    });
});
```

```
<div id="container">
    <div class="outer-box">
        <div class="box">
            Loonycorn rocks!
        </div>
        <div class="box">
            Loonycorn rocks!
        </div>
        <div class="box">
            Loonycorn rocks!
        </div>
        <div class="box">
            Loonycorn rocks!
        </div>
    </div>
</div>
```

All the elements with
the **box** class applied
gets the **outer-box** as a
wrapper

EXAMPLE

Each.html

Each.html

```
<body>
<div class="each link">
    Each
</div>
<br>
<br>
<div id="container">
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
    <div class="box">
        Loonycorn rocks!
    </div>
</div>
<br>
</body>
```

Each

Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!

Each.html

```
$('document').ready(function () {  
    $('.each').click(function() {  
        $('.box').each(function() {  
            $(this).addClass('outer-box');  
        });  
    });  
});
```

Each

The `.each()` function is
an explicit iteration
over all the selected
elements

Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!

Each.html

```
$(document).ready(function () {
    $('.each').click(function() {
        $(' .box').each(function() {
            $(this).addClass('outer-box');
        });
    });
});
```

Each

Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!

Add the **outer-box** class
to each selected element

Each.html

```
$(document).ready(function () {
    $('.each').click(function() {
        $('.box').each(function() {
            $(this).addClass('outer-box');
        });
    });
});
```

Each

Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!
Loonycorn rocks!

`$(this)` refers to each selected element in turn

Thank You