



## jQuery Animations

- The jQuery animate() method is used to create custom animations.

### Syntax:

***\$(selector).animate({params},speed,callback);***

- *The required params parameter defines the CSS properties to be animated.*
- *The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.*

### Eg:

```
$("#button").click(function(){  
    $("#div").animate({left: '250px'});  
});
```



multiple properties can be animated at the same time

Example

```
$("#button").click(function(){  
    $("#div").animate({  
        left: '250px',  
        opacity: '0.5',  
        height: '150px',  
        width: '150px'  
    });  
});
```



## Queue Functionality

By default, jQuery comes with queue functionality for animations.

This means that if you write multiple **animate()** calls after each other, jQuery creates an "internal" queue with these method calls.

Then it runs the animate calls ONE by ONE.

So, if you want to perform different animations after each other, we take advantage of the queue functionality:

### Example

```
$("#button").click(function(){  
    var div = $("#div");  
    div.animate({height: '300px', opacity: '0.4'}, "slow");  
    div.animate({width: '300px', opacity: '0.8'}, "slow");  
    div.animate({height: '100px', opacity: '0.4'}, "slow");  
    div.animate({width: '100px', opacity: '0.8'}, "slow");  
});
```



## jQuery **stop()** Method

- The jQuery **stop()** method is used to stop an animation or effect before it is finished.
- The **stop()** method works for all jQuery effect functions, including sliding, fading and custom animations.

Eg:     `$("#flip").click(function(){`  
          `$("#panel").slideDown(5000);`  
          `});`  
          `$("#stop").click(function(){`  
          `$("#panel").stop();`  
          `});`



## jQuery **stop()** Method

- The jQuery **stop()** method is used to stop an animation or effect before it is finished.
- The **stop()** method works for all jQuery effect functions, including sliding, fading and custom animations.

Eg:     `$("#flip").click(function(){`  
          `$("#panel").slideDown(5000);`  
          `});`  
          `$("#stop").click(function(){`  
          `$("#panel").stop();`  
          `});`



## jQuery Callback Functions

A callback function is executed after the current effect is 100% finished. JavaScript statements are executed line by line. However, with effects, the next line of code can be run even though the effect is not finished. This can create errors.

To prevent this, you can create a callback function.

A callback function is executed after the current effect is finished.

### Syntax:


***\$(selector).hide(speed,callback);***

## Example without Callback

```
$("#button").click(function(){  
    $("#p").hide(1000);  
    alert("The paragraph is now hidden");  
});
```

## Example with Callback

```
$("#button").click(function(){  
    $("#p").hide("slow",function(){  
        alert("The paragraph is now hidden");  
    });  
});
```



## jQuery – Chaining

Chaining allows us to run multiple jQuery methods (on the same element) within a single statement.

This way, browsers do not have to find the same element(s) more than once.

To chain an action, you simply append the action to the previous action.

**Eg:**     ***`$(document).ready(function(){`***

***`$("button").click(function(){`***

***`$("#p1").css("color", "red").slideUp(2000).slideDown(2000);`***

***`});`***

***`});`***





## Get Content and Attributes

jQuery contains powerful methods for changing and manipulating HTML elements and attributes.


One very important part of jQuery is the possibility to manipulate the **DOM**.

The **DOM** defines a standard for accessing HTML and XML documents.

### Get Content - `text()`, `html()`, and `val()`

Three simple, but useful, jQuery methods for DOM manipulation are:

1. **`text()`** - Sets or returns the text content of selected elements
2. **`html()`** - Sets or returns the content of selected elements (including HTML markup)
3. **`val()`** - Sets or returns the value of form fields



Eg: `<p id="test">This is some <b>bold</b> text in a paragraph.</p>`  
`<script>`  
    `$(document).ready(function(){`  
        `$("#btn1").click(function(){`  
            `alert("Text: " + $("#test").text());`  
        `});`  
        `$("#btn2").click(function(){`  
            `alert("HTML: " + $("#test").html());`  
        `});`  
    `});`  
`</script>`



**Eg:**

```
<script>  
    $(document).ready(function(){  
        $("button").click(function(){  
            alert("Value: " + $("#test").val());  
        });  
    });  
</script>
```

```
<p>Name: <input type="text" id="test" value="Mickey  
Mouse"></p>
```



## Get Attributes

The jQuery ***attr()*** method is used to get attribute values.

Eg: ***<script>***

```
$(document).ready(function(){  
    $("button").click(function(){  
        alert($("#w3s").attr("href"));  
    });  
});
```

***</script>***

***<p><a href="http://www.w3schools.com" id="w3s">W3Schools.com</a></p>***



## Set Content and Attributes

the same three methods from the previous is used to set content.

**Eg:**

```
$(document).ready(function(){  
  
    $("#btn1").click(function(){  
  
        $("#test1").text("Hello world!");  
  
    });  
  
    $("#btn2").click(function(){  
  
        $("#test2").html("<b>Hello world!</b>");  
  
    });  
  
    $("#btn3").click(function(){  
  
        $("#test3").val("Dolly Duck");  
  
    });  
});
```



## Set Attributes - attr()

The jQuery attr() method is also used to set/change attribute values.

**Eg:** `<script>  
$(document).ready(function(){  
 $("button").click(function(){  
 $("#w3s").attr("href",  
"http://www.w3schools.com/jquery");  
 });  
});  
</script>`

`<p><a href="http://www.w3schools.com"  
id="w3s">W3Schools.com</a></p>`



## jQuery - Add Elements

With jQuery, it is easy to add new elements/content.

jQuery methods that are used to add new content:

1. ***append()*** - Inserts content at the end of the selected elements
2. ***prepend()*** - Inserts content at the beginning of the selected elements
3. ***after()*** - Inserts content after the selected elements
4. ***before()*** - Inserts content before the selected elements

## jQuery **append()** and **prepend()** Method

The jQuery **append()** method inserts content *AT THE END* of the selected HTML elements.

The jQuery **prepend()** method inserts content *AT THE BEGINNING* of the selected HTML elements.

Eg:

```
<script>
    $(document).ready(function(){
        $("#btn1").click(function(){
            $("p").append(" <b>Appended text</b>.");
        });

        $("#btn2").click(function(){
            $("ol").append("<li>Appended item</li>");
        });
    });
</script>
```



## jQuery after() and before() Methods

The jQuery **after()** method inserts content **AFTER** the selected HTML elements.

The jQuery **before()** method inserts content **BEFORE** the selected HTML elements.

Eg:

```
$(document).ready(function(){  
    $("#btn1").click(function(){  
        $("img").before("<b>Before</b>");  
    });  
  
    $("#btn2").click(function(){  
        $("img").after("<i>After</i>");  
    });  
});
```