# jQuery

1. jQuery is lightweight JavaScript library .
2. jQuery decrease the syntax of JavaScript.
3. It shorts all selectors .
4. Browser independent

**jQuery Syntax** : **$(*selector*).*action*()**

1. $ = sign to define/access jQuery
2. (selector) = find HTML elements
3. action() = performed on the element(s)

**Examples:**

$(this).hide() - hides the current element.

$("p").hide() - hides all <p> elements.

$(".test").hide() - hides all elements with class="test".

$("#test").hide() - hides the element with id="test".

**The Document Ready Event :**

The code inserted between $(document).ready( ) is executed only once when page is ready for JavaScript code to execute.

**$(document).ready(function(){  
   *// jQuery methods go here...*  
});**

**Shorter form:**

**$(document).ready( ) ⬄ $( )**

**$(function(){  
   *// jQuery methods go here...*  
});**

# Selectors

1. **Element Selector :** $("element\_name")
2. **Id Selector :** $("#id\_name")
3. **Class Selector :** $(“.class\_name”)

**More Examples of jQuery Selectors:**

1. **$("\*")** - Selects all elements.
2. **$(this)** - Selects the current HTML element.
3. **$("p.intro")** : Selects all <p> elements with class="intro".
4. **$("p:first")** - Selects the first <p> element.
5. **$("ul li:first")** - Selects the first <li> element of the first <ul>
6. **$("ul li:first-child")** - Selects the first <li> element of every <ul>
7. **$("[href]")** - Selects all elements with an href attribute
8. **$("a[target='\_blank']")** - Selects all <a> elements with a target attribute value = "\_blank"
9. **$("a[target!='\_blank']")** - Selects all <a> elements with a target attribute value != "\_blank"
10. **$(":button") -** Selects all <button> elements and <input> elements of type="button"
11. **$("tr:even")** - Selects all even <tr> elements.
12. **$("tr:odd")** - Selects all odd <tr> elements

# Events

**code :**

**$(document).ready(function() {**

**$(“#box”).click(function(){**

**var a = $(‘#box’).html();**

**Console.log(a);**

**}**

**})**

**Mouse Events of jQuery:**

1. Click
2. Dblclick
3. Mouseenter
4. Mouseleave
5. contextMenu

**Keyboard Events of jQuery:**

1. Keypress
2. Keydown
3. Keyup

**Form Events of jQuery:**

1. **Submit** : when submit the form.
2. **Select**: when select the form select tag.
3. **Change**: when change the value of input text.
4. **Focus** : in form when we focus on input element it works.
5. **Blur**: when focus remove from input element it works

**Document Window Events of jQuery:**

1. **Load // not work**
2. **Unload // not work**
3. **Resize :**  when
4. **Scroll** : when we want to scroll it work.

**code :**

**$(document).ready(function() {**

**$(“#box”).scroll(function(){**

**var a = $(‘#box’).html();**

**Console.log(a);**

**}**

**})**

# jQuery method

## html for GET

1. **text()** - Returns the text content of selected elements

**code :**

**$(document).ready(function() {**

**var x =$(".amount").text();**

**console.log(x);**

**})**

1. **htm**l**()**   - Returns HTML element.

**code :**

**$(document).ready(function() {**

**var x =$(".amount").html();**

**console.log(x);**

**})**

1. **val**() - Returns the value of form fields.

**code :**

**$(document).ready(function() {**

**var x =$(".amount").val();**

**console.log(x);**

**})**

1. **attr**() – Return attribute of element .

**code :**

**$(document).ready(function() {**

**var x =$(".amount").attr();**

**console.log(x);**

**})**

## html SET :

1. text()  - Sets the text content of selected elements.

**code :**

**$(document).ready(function() {**

**$(".amount").text(“ hello world “);**

**})**

1. html()   - Sets HTML element.

**code :**

**$(document).ready(function() {**

**$(".amount").html(“<h1> helo wolo </h1>”);**

**console.log(x);**

**})**

1. val() - Sets or returns the value of form fields.

**code :**

**$(document).ready(function() {**

**$(".amount").val(“ shuvo ”);**

**})**

1. attr() – return attribute of element .

**code :**

**$(document).ready(function() {**

**$(".amount").attr(“class” , “red”);**

**console.log(x);**

**})**

## Add Elements**:**

1. **append**() - Inserts content at the end of the selected elements.

**code :**

**$(document).ready(function() {**

**$(".amount").append(“<h1> hello world </h1>”);**

**})**

1. **appendTo()** -

Syntax: **$(content).appendTo(selector);**

**code:**

**$(document).ready(function() {**

**$ (“<h1> hello world </h1> <p> new tag </p>”).appendTo("#amount");**

**})**

1. **prepend**() - Inserts content at the beginning of the selected elements

**code :**

**$(document).ready(function() {**

**$(".amount").prepend(“<h1> hello world </h1>”);**

**})**

1. **prependTo()** -

Syntax: **$(content).prependTo(selector);**

**code:**

**$(document).ready(function() {**

**$ (“<h1> hello world </h1>”).prependTo("#amount");**

**})**

1. **after**() - Inserts content after the selected elements

**code :**

**$(document).ready(function() {**

**$(".amount").after(“<h1> hello world </h1>”);**

**})**

1. **before**() - Inserts content before the selected elements

**code :**

**$(document).ready(function() {**

**$(".amount").before(“<h1> hello world </h1>”);**

**})**

## Remove Elements:

1. **remove()** - Removes the selected element (and its child elements).

**code :**

**$(document).ready(function() {**

**$(".amount").remove(“<h1> hello world </h1>”);**

**})**

1. **empty() -** Removes the child elements from the selected element.

**code :**

**$(document).ready(function() {**

**$(".amount").empty(“<h1> hello world </h1>”);**

**})**

## CSS Classes:

1. **addClass() -** Adds one or more classes to the selected elements.

**code :**

**$(document).ready(function() {**

**$(".amount").addClass(“first main”);**

**})**

1. **removeClass()** - Removes one or more classes from the selected elements

**code :**

**$(document).ready(function() {**

**$(".amount").removeClass(“first”);**

**})**

1. **toggleClass()** - Toggles between adding/removing classes from the selected elements.

**code :**

**$(document).ready(function() {**

**$(".amount").toggleClass(“first”);**

**})**

1. **css() -** we can set CSS style by this method.

**Syntax : css( "propertyname" , "value");**

**code :**

**$(document).ready(function() {**

**$(".amount").css(“border”, “2px solid red”);**

**})**

## On / off method :

**ON method is use for multiple events on same element .**

**Single event :**

**code :**

**$(document).ready(function()**

**$(".amount").on(“click”, function(){**

**Console.log(“hello”);**

**})**

**})**

**Multiple event :**

**Code:**

**$(document).ready(function() {**

**$(".amount").on({**

**“click : function(){**

**Console.log(“click”);**

**},**

**“mouseover” : function(){**

**Console.log(“mouseover”);**

**},**

**“mouseout” : function(){**

**Console.log(“mouseout”);**

**}**

**});**

**});**

**Off method remove multiple events.**

## Clone method:

We can clone one element to another element.

**Code:**

**$(document).ready(function() {**

**$(".amount").click(function(){**

**$(‘#box h1’).clone().appendTo(‘#box2’);**

**});**

**});**

## Replace html

1**. replaceWith():** The replaceWith() method replaces selected elements with new content.

**Code :**

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

**$(" p:first").replaceWith("Hello world!");**

**});**

**2. replaceAll():** The replaceAll() method in jQuery replaces the selected elements with new HTML elements.

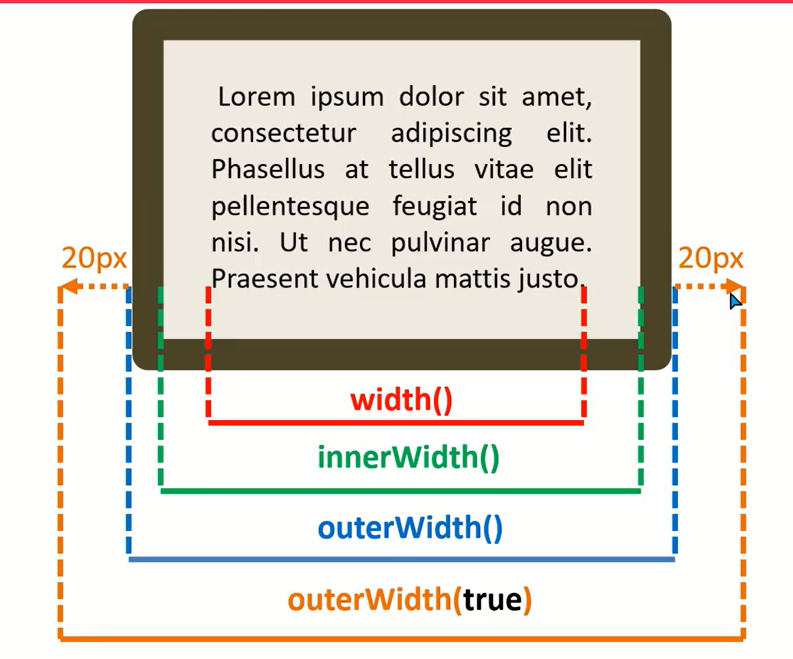
**Code :**

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

**$(" p:first").replaceAll("Hello world!");**

**});**

## Dimension Methods:



1. **width():**

**Code:**

**$(document).ready(function() {**

**$(".amount").click(function(){**

**var x = $(this).width();**

**console.log(x);**

**});**

**});**

1. **innerWidth()**

**Code:**

**$(document).ready(function() {**

**$(".amount").click(function(){**

**var x = $(this).innerwidth();**

**console.log(x);**

**});**

**});**

1. **outerWidth():**

**Code:**

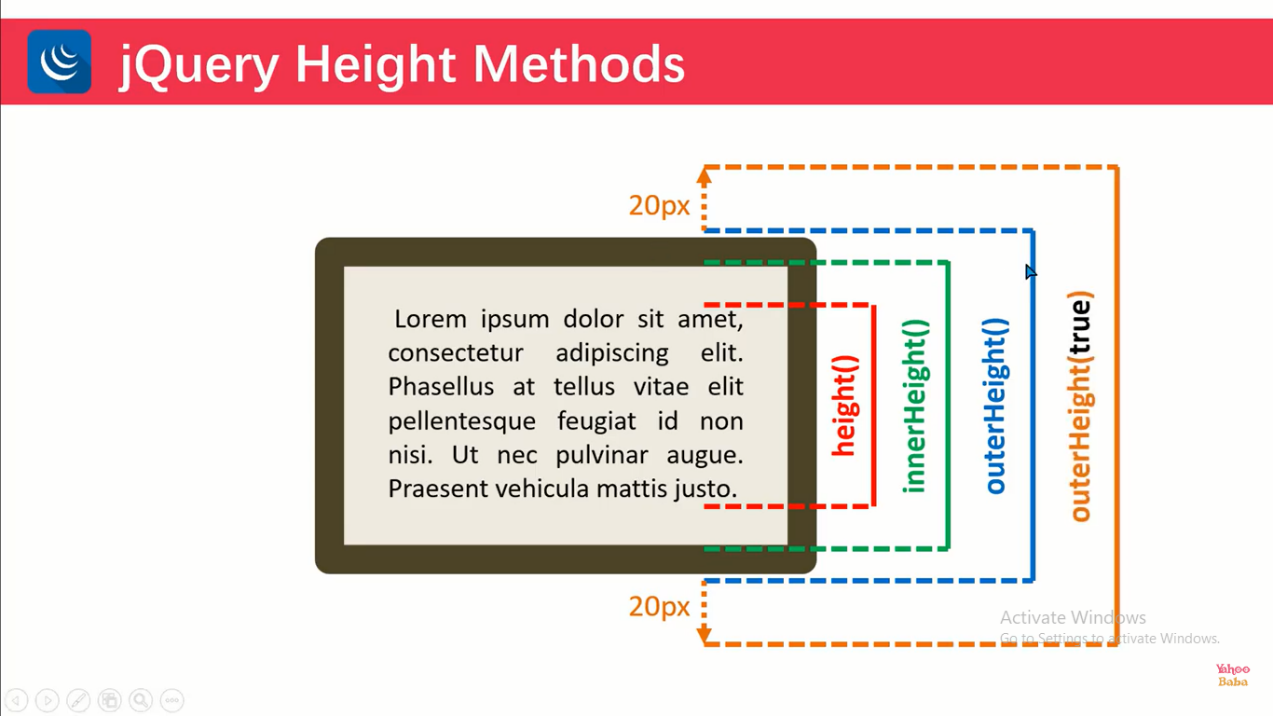
**$(document).ready(function() {**

**$(".amount").click(function(){**

**var x = $(this).**outerwidth();

**});**

**});**



1. **height():**

**Code:**

**$(document).ready(function() {**

**$(".amount").click(function(){**

**var x = $(this).height ();**

**console.log(x);**

**});**

**});**

1. **innerHeight()**

**Code:**

**$(document).ready(function() {**

**$(".amount").click(function(){**

**var x = $(this).innerHeight ();**

**console.log(x);**

**});**

**});**

1. **outerHeight():**

**Code:**

**$(document).ready(function() {**

**$(".amount").click(function(){**

**var x = $(this).outerHeight ();**

**console.log(x);**

**});**

**});**

## Position and offset

## Scroll

## hasClass

## display html

**hide():**

**show():**

**toggle():**

**fadeIn():**

**fadeout():**

**fadeToggle():**

**fadeTo():**

**slideDown():**

**slideUp():**

**slideToggle():**

## animate()

1. The animate() method performs a custom animation of a set of CSS properties.
2. This method changes an element from one state to another with CSS styles.

**Syntax:**

$(selector).animate( {**styles**}, **speed**, easing, callback)

**$(selector).animate( {styles} , {options} )**

**Styles:**

1. It is required.
2. Specifies one or more CSS properties/values to animate.
3. The property names must be camel-cased when used with the animate() method.
4. Only numeric values can be animated.
5. String values cannot be animated except for the string’s "show", "hide" and "toggle".

Properties that can be animated:

1. backgroundPositionX
2. backgroundPositionY
3. borderWidth
4. borderBottomWidth
5. borderLeftWidth
6. borderRightWidth
7. borderTopWidth
8. borderSpacing
9. margin
10. marginBottom
11. marginLeft
12. marginRight
13. marginTop
14. opacity
15. outlineWidth
16. padding
17. paddingBottom
18. paddingLeft
19. paddingRight
20. paddingTop
21. height
22. width
23. maxHeight
24. maxWidth
25. minHeight
26. minWidth
27. fontSize
28. bottom
29. left
30. right
31. top
32. letterSpacing
33. wordSpacing
34. lineHeight
35. textIndent

**speed**:

1. it is Optional.
2. Specifies the speed of the animation.
3. Default value is 400 milliseconds
4. Possible values:
   1. milliseconds (like 100, 1000, 5000)
   2. "slow"
   3. "fast"

**Easing**:

1. Optional.
2. Specifies the speed of the element in different points of the animation.
3. Default value is "swing". Possible values:
   1. "swing" - moves slower at the beginning/end, but faster in the middle
   2. "linear" - moves in a constant speed

**Callback:**

1. It Is optional.
2. A function to be executed after the animation completes.

**Code :**

**$(document).ready(function () {**

**$("#b1").click(function () {**

**$("#box").animate({**

**height: "200px",**

**width: "200px"**

**} ,**

**{**

**duration: 1000,**

**easing: "linear",**

**complete: function () {**

**$(this).after( "<p>maximum height</p>");**

**}**

**} );**

**});**

**});**

## Method chaining

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

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

## Traversing

**Code :**

**$(document).ready(function(){**

**$("span").parent().css(‘border’,’2px solid red’);**

**});**

**$(document).ready(function(){**

**$("div").find("span");**

**});**

**parent():**

The parent() method returns the direct parent element of the selected element.

This method only traverse a single level up the DOM tree.

**parents() :** The parents() method returns all ancestor elements of the selected element, all the way up to the document's root element (<html>).

**parentsUntil() :** The parentsUntil() method returns all ancestor elements between two given arguments.

**children():**

The children() method returns all direct children of the selected element.

This method only traverses a single level down the DOM tree.

**find() :**  method returns descendant elements of the selected element, all the way down to the last descendant.

**siblings() :** The siblings() method returns all sibling elements of the selected element.

**next() :** The next() method returns the next sibling element of the selected element.

**nextAll() :** The nextAll() method returns all next sibling elements of the selected element.

**nextUntil() :** The nextUntil() method returns all next sibling elements between two given arguments.

**prev():**

**prevAll():**

**prevUntil():**

**has():** The has() method returns all elements that have one or more elements inside of them, that matches the specified selector.

**Is():**

The is() method checks if one of the selected elements matches the selectorElement.

**Syntax** : **$(selector).is( selectorElement, function(index , element))**

**selectorElement :**

1. Required.
2. Returns true if there is at least one match from the given argument, and false if not.

**function(index, element)** : Optional. Specifies a function to run for the group of selected elements.

* 1. index - the index position of the element
  2. element - the current element (the "this" selector can also be used)

**Code :**

**if ($("p").parent().is("div")) {**

**alert("Parent of p is div");**

**}**

## Filtering

**Code :**

**$(document).ready(function(){**

**$("span").first().css(‘border’,’2px solid red’);**

**});**

**$(document).ready(function(){**

**$("div").eq(0);**

**});**

**first() :** The first() method returns the first element of the specified elements.

**last() :** The last() method returns the last element of the specified elements.

**eq() :**

1. the eq() method returns an element with a specific index number of the selected elements.
2. The index numbers start at 0, so the first element will have the index number 0 and not 1.

**filter() :**

1. The filter() method lets you specify a criteria.
2. Elements that do not match the criteria are removed from the selection, and those that match will be returned.

**Code :**

**$(document).ready(function(){**

**$("ul li").filter(‘#test’).css(‘border’,’2px solid red’);**

**});**

**not():** The not() method returns all elements that do not match the criteria.

**Code :**

**$(document).ready(function(){**

**$("ul li").not(‘#test’).css(‘border’,’2px solid red’);**

**});**

**slice():** The slice() method selects a subset of elements based on its index.

**Syntax** :  **$(selector).slice( start, stop)**

**Code :**

**$(document).ready(function(){**

**$("ul li").slice(2).css(‘border’,’2px solid red’);**

**});**

## each

1. it works as loop
2. it searches all same tag**.**

Syntax : **$(selector).each(function(index , element))**

**Code :**

**$(document).ready(function(){**

**$(“li”).each(function(){**

**Console.log($(this).text());**

**});**

**});**

## Event method:

1. Event.pageX
2. Event.pageY
3. Event.type
4. Event.target
5. Event.preventDefault
6. Event.isDefaultPrevented
7. Event.stopPropagation
8. Event.data