**jQuery Notes**

1. Introduction to jQuery
   * jQuery is a fast, small, and feature-rich JavaScript library.
   * It provides an easy-to-use API that works across multiple browsers, making web development more efficient and accessible.
   * It makes things like HTML document traversal, event handling, and AJAX much simpler.
2. Selecting Elements (Selectors)

jQuery selectors are used to select and manipulate HTML elements based on their tag name, class, ID, or other attributes.

* **Tag Selector**: $('p')
  + Selects all <p> elements in the document.
* **Class Selector**: $('.class')
  + Selects all elements with the specified class.
* **ID Selector**: $('#id')
  + Selects a specific element with the given ID.
* **Attribute Selector**: $('[name="username"]')
  + Selects elements with a specific attribute.
* **Group Selector**: $('div, p, h1')
  + Selects multiple elements at once.
* **Descendant Selector**: $('div p')
  + Selects <p> elements inside <div> elements.
* **Child Selector**: $('div > p')
  + Selects direct child <p> elements inside <div>.

1. Events in jQuery

jQuery provides various event methods to handle user interactions easily.

* **Click Event**: $('button').click(function(){ alert('Button Clicked!'); });
  + Executes when an element is clicked.
* **Hover Event**: $('div').hover(function(){ $(this).css('background-color', 'yellow'); }, function(){ $(this).css('background-color', ''); });
  + Runs when the mouse enters and leaves an element.
* **Keyup Event**: $('input').keyup(function(){ console.log('Key Released!'); });
  + Detects when a key is released.
* **Focus Event**: $('input').focus(function(){ $(this).css('border', '2px solid blue'); });
  + Triggers when an input field gains focus.
* **Blur Event**: $('input').blur(function(){ $(this).css('border', ''); });
  + Fires when an input field loses focus.
* **Submit Event**: $('form').submit(function(event){ event.preventDefault(); alert('Form Submitted!'); });
  + Captures form submission and prevents default behavior.

1. Effects & Animations

jQuery provides various built-in methods to create effects and animations.

* **Show/Hide Elements**:
* $('p').hide(); – Hides the selected elements.
* $('p').show(); – Shows hidden elements.
* **Fade Effects**:
* $('div').fadeIn(); – Gradually displays the element.
* $('div').fadeOut(); – Gradually hides the element.
* $('div').fadeToggle(); – Toggles between fadeIn and fadeOut.
* **Slide Effects**:
* $('p').slideUp(); – Collapses the element with an upward slide.
* $('p').slideDown(); – Expands the element with a downward slide.
* $('p').slideToggle(); – Toggles between slideUp and slideDown.
* **Custom Animations**:
* $('div').animate({opacity: 0.5, left: '50px'}, 1000);

Allows custom animations by modifying CSS properties over a specified duration.

**jQuery Methods**

The **.add()** method in jQuery is used to **add** additional elements to an existing jQuery selection.

**Syntax:**

$(selector).add(selectorToAdd);

* selector → The existing elements you want to extend.
* selectorToAdd → The elements you want to add to the selection.

The **.addClass(className)** method in jQuery is used to add one or more classes to the selected elements. It modifies the element(s) by adding the specified class to them.

The **.after()** method in jQuery is used to insert content **immediately after** the selected element(s). This can be useful when you want to add new elements, text, or HTML after an existing element in the DOM.

The **.animate()** method in jQuery is used to create custom animations by modifying CSS properties of selected elements over a given duration. This method allows you to animate properties like height, width, opacity, left, top, etc.

The **.append()** method in jQuery is used to insert content **at the end** of the selected elements. This could be HTML, text, or other DOM elements.

The **.appendTo()** method in jQuery is similar to .append(), but with one key difference: **the order of the arguments**. While .append() adds content to the selected element, .appendTo() adds content to the element(s) specified in the argument.

The **.attr()** method in jQuery is used to **get** or **set** the value of an attribute of an element. You can use it to retrieve the value of an attribute or to modify it dynamically.

The **.before()** method in jQuery is used to **insert content before** the selected elements. This method allows you to add HTML elements, text, or other content immediately before an existing element in the DOM.

The **.each()** method in jQuery is used to **iterate over a set of elements** and execute a function for each element. It is commonly used to loop through elements like lists, table rows, or divs and apply changes individually.

The **.fadeTo()** method in jQuery is used to gradually change the **opacity** of an element to a specified value over a given duration. Unlike .fadeIn() and .fadeOut(), which toggle visibility, .fadeTo() allows you to set a custom opacity level (between 0 and 1).

The **.filter()** method in jQuery is used to **select a subset of elements** from a matched set **based on a condition**. It allows you to refine your selection by filtering elements that meet specific criteria.

The **.find()** method in jQuery is used to **search for descendant elements** inside a selected element. It looks for **children, grandchildren, and deeper nested elements** based on a given selector.

The **.first()** method in jQuery is used to **select the first element** from a matched set of elements. It helps when you need to manipulate only the first occurrence of a particular element.

The **.focus()** method in jQuery is used to set **focus** on a selected element, typically an **input field, textarea, or button**. This method is useful for improving **user experience** by automatically focusing on a specific form field when a page loads or when a certain event occurs.

The **.insertAfter()** method in jQuery is used to **insert an element immediately after** a specified target element in the DOM.

The **.insertBefore()** method in jQuery is used to **insert an element immediately before** a specified target element in the DOM.

The .is() method in jQuery is used to **check if an element matches a given selector, element, or jQuery object**. It returns true or false, making it useful for **conditional logic**.

The **.not()** method in jQuery is used to **filter out** elements that match a specific selector. It returns a new jQuery object containing all elements **except** the ones that match the filter.

The **.parent()** method in jQuery is used to **select the direct parent element** of the selected element(s). It only goes **one level up** in the DOM hierarchy.

The **.parents()** method in jQuery is used to **select all ancestor elements** (parent, grandparent, etc.) of the selected element, up to the <html> tag.

The **.remove()** method in jQuery is used to **remove selected elements from the DOM**, including all their child elements and associated data.

The **.removeAttr()** method in jQuery is used to **remove an attribute** from the selected element(s).

The **.removeClass()** method in jQuery is used to **remove one or multiple classes** from the selected element(s).

The **.replaceAll()** method in jQuery **replaces** all selected elements with new content.

The **.width()** method in jQuery is used to **get or set** the width of an element (excluding padding, border, and margin).

The **.has()** method in jQuery is used to **filter elements** that contain a specific child or descendant element.