

JQuery Notes and Functions

What is JQuery?

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. JQuery possesses versatility and extensibility.

How is jquery different from javascript?

JavaScript:

- **Programming Language:**

JavaScript is a full-fledged programming language used for adding interactivity and dynamic behavior to web pages. It allows you to manipulate the Document Object Model (DOM), handle events, perform calculations, and much more.

- **Foundation:**

JavaScript is the foundation for many front-end frameworks and libraries like jQuery, React, Angular, and Vue.js.

- **Complexity:**

JavaScript can be complex and verbose, requiring more lines of code for certain tasks, especially when dealing with cross-browser compatibility.

jQuery:

- **Library:**

jQuery is a JavaScript library, meaning it's a collection of pre-written JavaScript code designed to simplify common web development tasks.

- **Purpose:**
jQuery simplifies DOM manipulation, event handling, animations, AJAX calls, and other common web development tasks.
- **Syntax:**
jQuery offers a concise and easy-to-understand syntax, allowing you to achieve complex functionalities with fewer lines of code compared to plain JavaScript.

When to use jquery over javascript?

- **JavaScript:**
Use JavaScript when you need full control over your code, want to build complex web applications, or are working with a framework that doesn't rely on jQuery.
- **jQuery:**
Use jQuery when you need to simplify common web development tasks, want to write concise code, or need to ensure cross-browser compatibility easily.

Advantages of Jquery over javascript:

jQuery is a JavaScript library that has many advantages, including:

- **DOM manipulation:** jQuery makes it easy to select, modify, change, or insert elements into the DOM.
- **Event handling:** jQuery makes it easy to respond to user interactions like clicks, keypresses, and form submissions.
- **Cross-browser compatibility:** jQuery is compatible across browsers.
- **Beginner-friendly:** jQuery's straightforward syntax makes it easy for novice developers to create functional web components.

- **AJAX calls:** jQuery allows web pages to be updated asynchronously without reloading the whole page.
- **Lightweight:** jQuery is fast and lightweight.

Syntax of JQuery:

Jquery works on the css methodology

`$(selector).function(callback function)`

Functions available in the JQuery are as follow:

- selectors:
 - id : `$("#id")`
 - class : `$(".class")`
 - tag : `$("tag")`
 - combination : `$("div h1")` --- selects all h1 present in div only.
 - last : `$("p:last")` --- selects last p available in complete file
 - first : `$("p:first")` --- selects first p available in complete file
 - gt : `$("p:gt(2)")` --- selects all p greater than 2 available
 - lt : `$("p:lt(2)")` ----- selects all p lesser than 2 available
 - gt : `$("p:eq(2)")` ----- selects p equal to 2 only
 - even : `$("p:even")` ----- selects all even p
 - odd : `$("p:odd")` ----- selects all odd p
- `hide()` function hide the element
- `fadeIn(time)` fades the element to null in mentioned time
- `fadeOut(time)` fades the element from null in mentioned time
- `fadeTo(time, opacity)` fades the element to mentioned opacity value which varies from 0 to 1.
- `addClass("className");`
- `removeClass("className")`
- `toggleClass("className")`

- empty()
- remove()
- css function:
css()
- Getters and setters:
text(), html(), attr(), val()
- Adding elements in the html
append(), prepend(), before(), after()
- Mouse events:
click(), dblclick(), contextMenu(), mouseenter(), mouseleave()
- keyboard events:
keyup(), keydown(), keypress()
- form events:
focus(), blur(), submit(), select(), change()
- Window events:
resize(), scroll()

Example:-

```
$(document).ready(function() {  
    // Write here your code  
})
```