🧠 What is jQuery?

jQuery is a fast, small, and feature-rich **JavaScript library**. It simplifies tasks like HTML document traversal, event handling, animation, and AJAX with an easy-to-use API that works across a multitude of browsers.

🔍 Why Use jQuery?

- Cross-browser compatibility
- Simplified syntax (`$('#id')` vs. `document.getElementById`)
- Built-in utilities for AJAX, animations, and effects
- Reduces code verbosity

📥 Getting Started

CDN:

```
<script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
```

Basic Syntax:

```
$(selector).action();
```

🧩 Components:

1. `$`:

    - A shorthand for the **jQuery function**.
    - You can also use `jQuery(selector)`, but `$` is the common convention.

2. **`selector`**:

    - A string used to **select HTML elements**, just like CSS selectors.

    - Examples:

        - `"#header"` — selects the element with ID `header`
        - `".btn"` — selects all elements with class `btn`
        - `"p"` — selects all `<p>` tags
        - `"ul > li"` — selects direct child `<li>`s of `<ul>`

3. **`.action()`**:

    - A **jQuery method** (also called an "action") that performs something on the selected elements.

    - Examples:

        - `.hide()` — hides the selected elements
        - `.click()` — attaches a click event handler
        - `.css("color", "red")` — changes CSS

```
$("p").hide(); // Hide all paragraphs

$("#title").css("color", "blue"); // Change color of element with ID

$("#myButton").click(function () {
  alert("Button clicked!");
}); // When a button is clicked, show an alert
```

```
// jQuery "document ready" event.
$(document).ready(function () {
  // jQuery code here
});
```

Make sure that the HTML elements are fully loaded **before** you try to interact with them using jQuery.

- **`$(document)`**
  Selects the entire **HTML document** — everything between `<html>` and `</html>`.

- **`.ready()`**
  A jQuery method that runs your code **only after the DOM is fully loaded** (but before images, CSS, etc. are completely finished loading).

- **`function() { ... }`**
  This is a **callback function** — the code inside it will run when the DOM is ready.

🔍 Selectors

Selector	**Description**	**Example**
`*`	All elements	`$("*")`
`#id`	ID selector	`$("#header")`
`.class`	Class selector	`$(".menu")`
`element`	Element selector	`$("p")`
`parent > child`	Direct child	`$("ul > li")`
`:first`, `:last`	First or last matched	`$("li:first")`
`:even`, `:odd`	Even or odd matched	`$("tr:even")`
`[attr=value]`	Attribute selector	$("[type='text']")

🧭 Traversing

Method	Description
`.parent()`	Immediate parent
`.children()`	Direct children
`.find(selector)`	Descendants
`.siblings()`	All siblings
`.next()`, `.prev()`	Next or previous sibling
`.closest(selector)`	Closest ancestor matching selector
`.each(function)`	Iterate over matched elements

🎯 Events

```
$(selector).event(handler);
```

Event	Description
`click()`	Click event
`dblclick()`	Double click
`mouseenter()`	Mouse enters element
`mouseleave()`	Mouse leaves element
`keydown()`	Key pressed
`submit()`	Form submitted
`change()`	Value changed
`hover(in, out)`	Hover in and out

Example:

```
$("#btn").click(function () {
  alert("Button clicked");
});
```

✏️ DOM Manipulation

```
.html(), .text(), .val()
```

🏗 Insert

Method	Usage
`.append()`	Inside, after content
`.prepend()`	Inside, before content
`.after()`	After element
`.before()`	Before element

✂ Remove

```
.remove(), .empty()
```

🪄 Attributes & Classes

```
.attr("href"), .removeAttr("title")
.addClass("new"), .removeClass("old"), .toggleClass("active")
```

🎨 CSS Manipulation

```
.css("property", "value")
```

Example:

```
$("p").css("color", "blue");
```

🎞 Effects and Animations

Method	Description
`.hide()`, `.show()`	Hides or shows elements
`.toggle()`	Toggles visibility
`.fadeIn()`, `.fadeOut()`	Fading effects
`.slideUp()`, `.slideDown()`	Slide effects
`.animate()`	Custom animations

Example:

```
$("#box").fadeOut(1000);
```

🌐 AJAX

Basic Load

```
$("#result").load("file.html");
```

`$.get()` and `$.post()`

```
$.get("url", function (data) {});
$.post("url", { name: "John" }, function (data) {});
```

`$.ajax()`

AJAX stands for **Asynchronous JavaScript and XML**.

It's a technique used in web development that allows a web page to **communicate with a server without reloading the entire page**.

```
$.ajax({
  url: "server.php",
  method: "POST",
  data: { id: 10 },
  success: function (response) {
    console.log(response);
  },
  error: function () {
    alert("Error!");
  },
});
```

📦 What does AJAX use?

AJAX isn't a single technology—it's a **combination** of:

- **HTML/CSS** – for structure and style
- **JavaScript** – to control behavior
- **XMLHttpRequest** (or `fetch`) – to send/receive data
- **JSON or XML** – to format the data (JSON is more common today)

- **Server-side scripts** (e.g., PHP, Node.js, Python) – to process the request

---