

Jquery

jQuery is a library which simplifies DOM operations, event handling, AJAX, and animations. It also takes care of many browser compatibility issues in underlying DOM and engines.

Each version of jQuery can be downloaded from <https://code.jquery.com/jquery/> in both compressed (minified) and uncompressed formats.

Examples

jQuery Namespace ("jQuery" and "\$")

jQuery is the starting point for writing any jQuery code. It can be used as a function `jQuery(...)` or a variable `jQuery.foo`.

\$ is an alias for jQuery and the two can usually be interchanged for each other (except where `jQuery.noConflict()`; has been used - see Avoiding namespace collisions).

What is jQuery?

jQuery is a fast and concise Library created by John Resig in 2006 with a nice

motto: Write less, do more.

jQuery simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development.

jQuery is a toolkit

designed to simplify various tasks by writing less code. Here is the list of important core

features supported by jQuery:

- **DOM manipulation:** The jQuery made it easy to select DOM elements, negotiate them and modifying their content by using cross-browser open source selector engine called Sizzle.
- **Event handling:** The jQuery offers an elegant way to capture a wide variety of events, such as a user clicking on a link, without the need to clutter the HTML code itself with event handlers.

- **AJAX Support:** The jQuery helps you a lot to develop a responsive and feature-rich site using AJAX technology.
- **Animations:** The jQuery comes with plenty of built-in animation effects which you can use in your websites.
- **Lightweight:** The jQuery is very lightweight library - about 19KB in size (Minified and gzipped).
- **Cross Browser Support:** The jQuery has cross-browser support, and works well in IE 6.0+, FF 2.0+, Safari 3.0+, Chrome and Opera 9.0+
- **Latest Technology:** The jQuery supports CSS3 selectors and basic XPath syntax.

How to use jQuery?

There are two ways to use jQuery.

Local Installation – You can download jQuery library on your local machine and include it in your HTML code.

CDN Based Version – You can include jQuery library into your HTML code directly from Content Delivery Network (CDN).

jQuery

Introduction

jQuery is a popular library that simplifies HTML document traversal and manipulation, event handling, animation, and Ajax interactions. It streamlines the process of working with and offers a wide range of features and plugins for web development.

Difference between jQuery and

- jQuery is a library: jQuery is a library, which means it is built on top of the core language. It provides a set of functions and methods to simplify common tasks.
- is a programming language: is a full-fledged programming language used for web development. jQuery is a tool within , and it enhances by providing a more concise and convenient way to perform tasks.

HTML/CSS Manipulation with jQuery

You can manipulate HTML and CSS elements using jQuery methods like ``.html()``, ``.text()``, ``.css()``, and more. For example:

```
// Change the text inside a paragraph
$("p").text("New text");
```

```
// Change the background color of an element
$("#myDiv").css("background-color", "blue");
```

jQuery Selectors

Selectors are used to target and select HTML elements. jQuery selectors are similar to CSS selectors. Some common examples:

```
// Select all paragraphs
$("p")
```

```
// Select elements with a specific class
$(".myClass")
```

```
// Select elements with a specific ID
$("#myID")
```

jQuery Events

jQuery simplifies event handling. You can use methods like `.on()`, `.click()`, `.hover()`, etc., to bind events to elements. For example:

```
// Handle a click event
$("button").click(function() {
    alert("Button clicked!");
});
```

Basic Events

Basic events include common interactions like `click`, `hover`, `keydown`, `submit`, and more. You can attach these events to elements as shown in the previous example.

How to Fire an Event Programmatically

You can trigger events using jQuery's `.trigger()` method. For instance:

```
// Trigger a click event on a button
```

```
$("#myButton").trigger("click");
```

Custom Logic on Event Fire

You can execute custom logic when an event is triggered. For example, you can handle form submission and validate data before submitting:

```
$("#myForm").submit(function(event) {  
    // Custom validation logic  
    if (!validateData()) {  
        event.preventDefault(); // Prevent the form from submitting  
    }  
});
```

jQuery Validation

jQuery can be used for both basic and more complex form validation.

Basic Validation

Basic validation checks can be performed using conditional statements within event handlers.

Validation with jQuery Validator

The jQuery Validation Plugin is a popular tool for advanced form validation. It provides a set of rules, messages, and methods for validating form fields.

```
$("#myForm").validate({  
    rules: {  
        email: {  
            required: true,  
            email: true  
        },  
        password: {  
            required: true,  
            minlength: 6  
        }  
    },  
    messages: {  
        email: "Please enter a valid email address",  
        password: {  
            required: "Password is required",  
            minlength: "Password must be at least 6 characters"  
        }  
    }  
});
```

```
}  
});
```

jQuery Functions

jQuery offers numerous utility functions like ``map()``, ``grep()``, ``extend()``, ``each()``, ``merge()``, etc., for various operations on arrays and objects.

Example - ``map()``

The ``map()`` function applies a function to each element of an array and returns a new array.

```
var numbers = [1, 2, 3, 4];  
var squaredNumbers = $.map(numbers, function(num) {  
    return num * num;  
});
```

Regular Expressions in jQuery

jQuery supports regular expressions for text manipulation and pattern matching. You can use regular expressions with methods like ``match()``, ``replace()``, etc.

Callback Functions

Callbacks are functions that are passed as arguments to other functions and are executed after the completion of the primary function.

Deferred & Promise Objects

jQuery provides Deferred and Promise objects for handling asynchronous operations and managing their success and failure.

Ajax

What is Ajax?

AJAX (Asynchronous and XML) is a set of technologies that allow you to make asynchronous requests to the server without refreshing the entire web page. It is often used to fetch data, update content, and interact with the server in the background.

Use of Ajax

AJAX is commonly used for tasks like loading data from a server, submitting form data without a full page reload, and creating responsive web applications.

How to Send Data with Ajax Request

You can use the `$.ajax()` method to send data to a server and receive a response. Here's a basic example:

```
$.ajax({
  url: "server.php",
  method: "POST",
  data: { key1: "value1", key2: "value2" },
  success: function(response) {
    console.log("Server response: " + response);
  },
  error: function(xhr, status, error) {
    console.error("Request failed: " + status);
  }
});
```

- **GET**: Used to retrieve data from the server.
- **POST**: Used to send data to the server, often used for form submissions.
- **PUT**: Typically used to update existing data on the server.
- **DELETE**: Used to request the removal of data from the server.

JSON Data

JSON (Object Notation) is a lightweight data interchange format. In Ajax, it's commonly used to send and receive data. You can parse JSON data using `$.parseJSON()`.

Serialization & Deserialization

Serialization is the process of converting data into a format suitable for transmission, such as converting an object into a JSON string. De-serialization is the reverse process, converting the received data (e.g., JSON) back into an object.

This document covers the basics of jQuery, including HTML/CSS manipulation, event handling, validation, common functions, regular expressions, callbacks, Deferred & Promise objects, and Ajax. It should serve as a useful reference for your web development projects.