**Module 7) JQuery Basic, Effects & Advanced**

**Q1-What is jQuery?**

**Ans-** In simple words jQuery means "write less do more". jQuery is not programming language it is a tool that makes Javascript web development easier. We can use jQuery to perform multiple simple tasks that might need multiple lines of JavaScript code by using methods.

**EXAMPLE: -**

$(“div”). hide (),

$(“div”). mousedown (function () {}).

**Q2-How to Apply CSS Using JQuery, How to Add Class and Remove Class in Jquery, JQuery Animation?**

**Ans-** <!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Adding and Removing Classes in jquery </title>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.7.0/jquery.min.js"></script>

<style>

.one{

font-size: 50px;

color: red;

background-color: bisque;

}

div {

height: 100px;

width: 100px;

background-color: aqua;

position: relative;

}

</style>

</head>

<body>

<!-- Applying css using jQuery -->

<!-- <p>Applying css using jQuery</p>

<script>

$(document).ready(function() {

$("p").css({"background-color":"aquamarine", "color" : "darkcyan", "font-size" : "30px" })

})

</script> -->

<!-- Add class and remove class in jQuery -->

<!-- <p class="one">Add class and remove class in jQuery</p>

<button>click me</button>

<script>

$(document).ready(function(){

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

$("p").toggleClass("one")

})

})

</script> -->

<!-- jQuery Animation -->

<div></div>

<button>Click Me</button>

<script>

$(document).ready(function(){

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

// $("div").animate({"width":"toggle"})

$("div").animate({"height":"toggle"})

})

})

</script>

</body>

</html>

**Q3-How to create slider with animation?**

**Ans-** <!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Slider with Animation</title>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.7.0/jquery.min.js"></script>

<style>

#box-1{

width: 600px;

background-color: cadetblue;

text-align: center;

padding: 5px;

cursor: pointer;

}

#box-2{

height: 200px;

width: 610px;

background-color: aquamarine;

text-align: center;

}

</style>

</head>

<body>

<div id="box-1">Click here</div>

<div id="box-2">Slider with Animation</div>

<script>

$(document).ready(function(){

$("#box-1").click(function(){

$("#box-2").slideToggle(2000)

})

})

</script>

</body>

</html>

**Q4-Event bubbling tickling example**

**Ans-** Event bubbling example: Event bubbling is a phenomenon where an event triggered on a nested element will "bubble up" through its ancestor elements, potentially triggering the same event on multiple elements. Here's an example:

<!DOCTYPE html>

<html>

<head>

<title>Event Bubbling Example</title>

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

<style>

div {

border: 1px solid black;

padding: 10px;

}

</style>

</head>

<body>

<div id="outer">

Outer Div

<div id="inner">

Inner Div

</div>

</div>

<script>

// Event Bubbling Example

$(document).ready(function() {

$('#outer').on('click', function() {

alert('Clicked on Outer Div');

});

$('#inner').on('click', function(event) {

// Prevent the event from bubbling to the outer div

event.stopPropagation();

alert('Clicked on Inner Div');

});

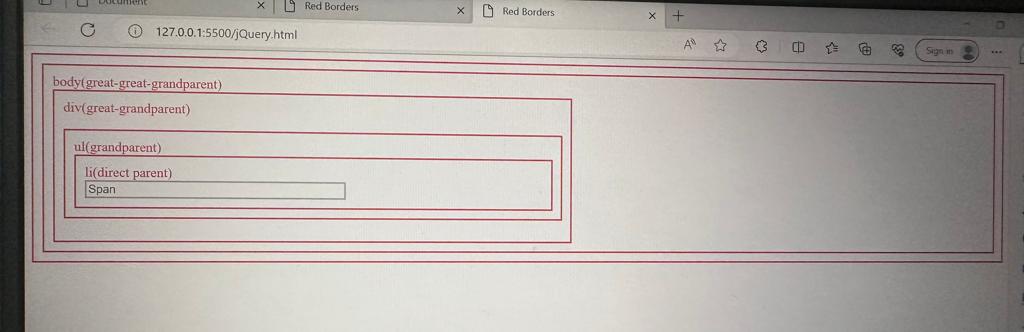
});

</script>

</body>

</html>

In this example, clicking on the inner div will trigger the click event for both the inner and outer divs, but we use event.stopPropagation() to prevent the event from reaching the outer div.



<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Red Borders</title>

    <style>

        .red-border {

            border: 2px solid red;

            padding: 10px;

        }

        ul {

            list-style: none;

        }

        input {

            border: 2px solid rgb(186, 186, 186);

        }

        .fcolour {

            color: red;

        }

        .main-box {

            width: 98%;

        }

    </style>

</head>

<body>

    <div class="main-box red-border">

        <div class="body-box red-border">

            body(great-great-grandparent)

            <div class="my-div red-border">

                div(great-grandparent)

                <ul class="my-list red-border">

                    ul(grandparent)

                    <li class="my-item red-border">

                        li(direct parent)

                        <br>

                        <input class="my-span" type="text" placeholder="Span">

                    </li>

                </ul>

            </div>

        </div>

    </div>

    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

    <script>

        $(document).ready(function () {

            $(".main-box, .body-box, .my-div, .my-list, .my-item").addClass("red-border");

            $(".my-span").removeClass("red-border");

            $(".main-box, .body-box, .my-div, .my-list, .my-item").addClass("fcolour");

        });

        $(document).ready(function () {

            // Get the width of the main and body boxes

            var mainBoxWidth = $(".main-box").width();

            var bodyBoxWidth = $(".body-box").width();

            // Set the width of div elements to half of the main and body boxes

            $(".my-div").css("width", mainBoxWidth / 2);

            // Set the span width to half of the div, ul, and li elements

            $(".my-span").css("width", mainBoxWidth / 4); // Half of the div width

            // Add red borders to all elements except span

            $(".main-box, .body-box, .my-div, .my-list, .my-item").addClass("red-border");

            $(".my-span").removeClass("red-border");

            // Add grey border to span

            $(".main-box, .body-box, .my-div, .my-list, .my-item").addClass("fcolour");

        });

    </script>

</body>

</html>