Module 7) jQuery Basic, Effects & Advanced

• *What is jQuery?*

jQuery is a lightweight, "write less, do more", JavaScript library.

The purpose of jQuery is to make it much easier to use JavaScript on your website.

jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

The jQuery library contains the following features:

- HTML/DOM manipulation
- CSS manipulation
- HTML event methods
- Effects and animations
- AJAX
- Utilities

In addition, jQuery has plugins for almost any task out there.

• *How to Apply CSS Using jQuery, How to Add Class and Remove Class in jQuery, jQuery Animation?*

jQuery Manipulating CSS

jQuery has several methods for CSS manipulation.

- addClass() - Adds one or more classes to the selected elements
- removeClass() - Removes one or more classes from the selected elements
- toggleClass() - Toggles between adding/removing classes from the selected elements
- css() - Sets or returns the style attribute

jQuery Animations - The animate() Method

The jQuery animate() method is used to create custom animations.

Syntax:

$(*selector*).animate({*params*},*speed,callback*);

The required params parameter defines the CSS properties to be animated.

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds.

The optional callback parameter is a function to be executed after the animation completes.

The following example demonstrates a simple use of the animate() method; it moves a <div> element to the right, until it has reached a left property of 250px:

Example
```
$("button").click(function(){
  $("div").animate({left: '250px'});
});
```

• *How to create slider with animation?*

jQuery Sliding Methods
With jQuery you can create a sliding effect on elements.

jQuery has the following slide methods:

- slideDown()
- slideUp()
- slideToggle()

• *Event bubbling tickling example*

Event bubbling is a method of event propagation in the HTML DOM API when an event is in an element inside another element, and both elements have registered a handle to that event. It is a process that starts with the element that triggered the event and then bubbles up to the containing elements in the hierarchy. *In event bubbling, the event is first captured and handled by the innermost element and then propagated to outer elements.*

Syntax:

```
addEventListener(type, listener, useCapture)
```

- **type**: Use to refer to the type of event.
- **listener**: Function we want to call when the event of the specified type occurs.
- **userCapture**: Boolean value. The boolean value indicates the event phase. By Default useCapture is false. It means it is in the bubbling phase.

Example: This example shows the working of event bubbling in JavaScript.

```
<body>
    <h2>Bubbling Event in Javascript</h2>

    <div id="parent">
        <button>
            <h2>Parent</h2>
        </button>
        <button id="child">
```

```html
        <p>Child</p>

    </button>
</div><br>


<script>
    document.getElementById(
    "child").addEventListener("click", function () {
        alert("You clicked the Child element!");
    }, false);

    document.getElementById(
    "parent").addEventListener("click", function () {
        alert("You clicked the parent element!");
    }, false);
</script>

</body>
```