



Chapter 12 (3.1)

Basics of JQuery



Objectives

At the end of this session, you will be able to:

- Explain the basics of jQuery and its usage with Bootstrap
- Outline selectors and functions in jQuery
- Describe JSON and its usage in jQuery
- Explain conversions to JSON
- Describe the concept of event delegation and its usage in jQuery



Introduction

- A lightweight and concise JS library
- Helps to make JS programming easy and faster
- Feature rich and open source library which simplifies complex tasks
- Easy to use API which is compatible with browsers
- Common tasks into built-in methods
- Reduce time in writing lines of code



Benefits of jQuery over JS (1-2)

Unobtrusive JS

- Web page designing through unobtrusive JS
- Separates functionality between presentation and structure of Web page

Lightweight

- Reduced library size is in kilobytes (KB)
- Browser cache the library throughout Websites instead of downloading it

Easy to use

- Simple and robust syntax
- Reduced lines of code
- Library uses CSS3 selector specification

Big and focused library

- Perform variety of functions through single library
- jQuery offers several utility functions



Benefits of jQuery over JS (2-2)

Extensibility

- Plugin framework – third party and official plugins
- Any feature can be included through plugin

Browser Compatibility

- Resolves cross browser issues
- jQuery implemented compatibility code

Strong Community

- Exclusive community where developers improve functionality
- Offers several plugins which accelerate Web development process

AJAX Support

- Creates AJAX templates
- Offers smooth interfaces and efficient loading

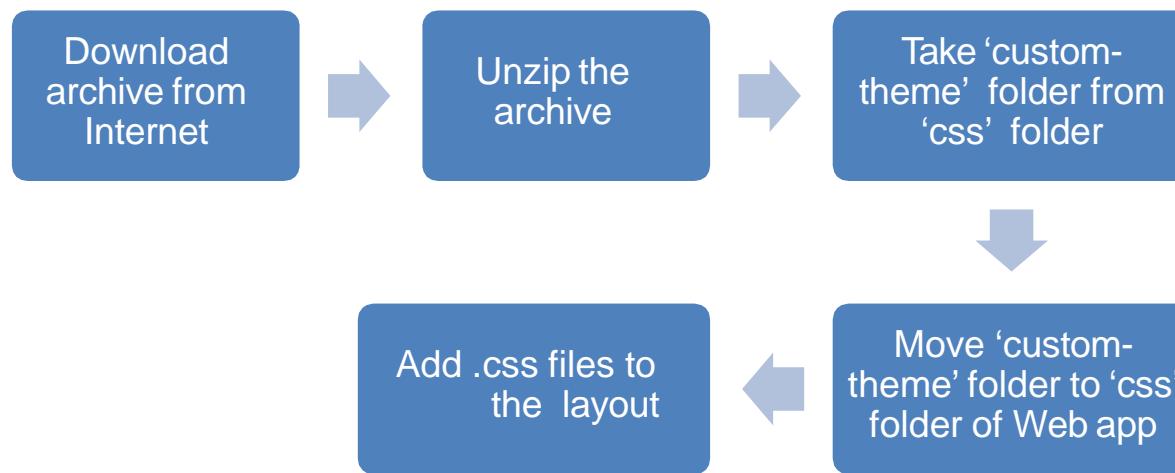
Comprehensive Documentation

- Official website has in-depth documentation and tutorials
- Helps beginners for easy use



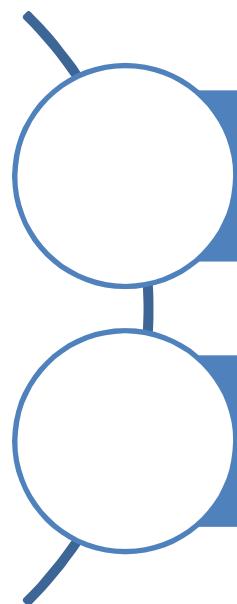
Using Bootstrap and jQuery

- jQuery UI Bootstrap theme allows using bootstrap-themed widgets
- Bootstrap and jQuery cannot exists together originally
- New theme was introduced with relevant JS and CSS using jQuery and Bootstrap
- Bootstrap theme for third party widgets
- Steps to use the theme:



Use of jQuery in Web Application

- jQuery library is a JS file
- Used for programming in jQuery to develop Web application
- Two ways to use the file:



Download the library file from [jQuery.com](https://jquery.com) and refer it in HTML code and finally, save the downloaded file in the directory

Include the library into the HTML code while referring it to CDN



jQuery Syntax

- Customized syntax:
`$(selector).action()`
- Web developer finds and selects HTML element and triggers desired actions
- jQuery code exists in **document ready** event. Syntax is:

```
$(document).ready(function()
{
    //Code jQuery functions here
});
```

- Wait for DOM to load fully before requests or responses are handled by HTML pages



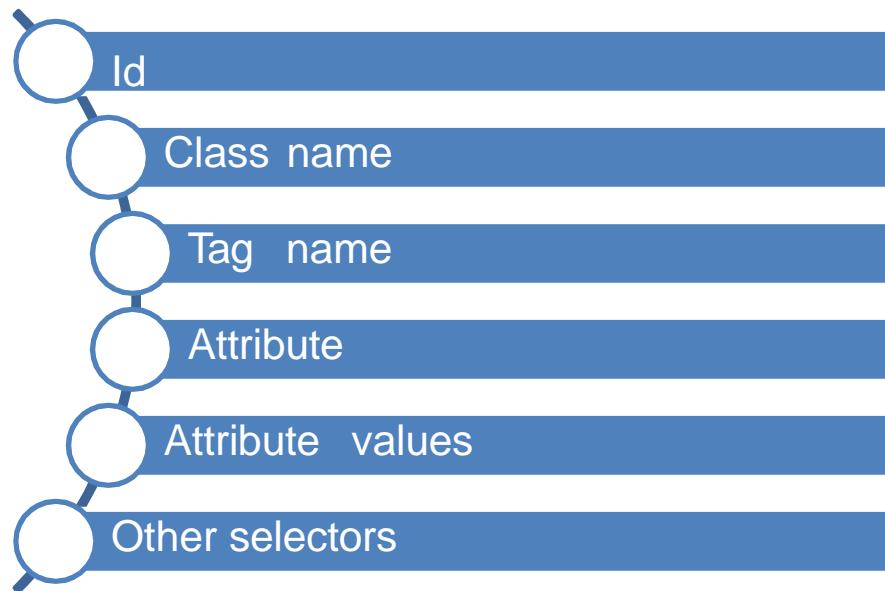
jQuery Functions

- Function definition can be done in four ways:

Basic Declaration	Declaration as a Variable name	Self-invoking Function Declaration	User-defined Function Declaration
<ul style="list-style-type: none">• Easy way of declaration• Used for quick testing• Not recommended to use	<ul style="list-style-type: none">• Defining a function through an expression assigned to a variable• Variable is used as a function	<ul style="list-style-type: none">• Expression where script invokes automatically without being called• No name and no explicit calls	<ul style="list-style-type: none">• Developers creating own functions• jQuery.fn or \$.fn creates a user-defined function

jQuery Selectors (1-3)

- A function in the form of expression to search matching element in DOM
- Offers selecting single or multiple HTML elements
- Selectors begin with jQuery identifier (\$) followed by parentheses
- Library allows to access elements by:





jQuery Selectors (2-3)

Id Selector

- Refers to the id attribute of a tag to search relevant element
- Id is unique and used to find single tag at a time
- Document.getElementById() function
- Syntax: `$('#<id>')`

Class Selector

- Search HTML elements having specified class
- getElementsByClassName() function
- Syntax: `('.<class>')`



jQuery Selectors (3-3)

Element Selector

- Access elements as per name
- Specify element selector within parentheses followed with \$ character
- Syntax is: \$(<element>)

Checked Selector

- Access and select all selected radio buttons and checkboxes
- Specify :checked in double quotes within parentheses followed by \$ character
- Syntax is: \$(":checked")



JSON and Conversions to JSON

- It is a lightweight format used for exchanging and storing data
- Alternate to eXtensible Markup Language (XML)
- Data exchanged between a server and a client
- Exchange in simple text format
- Independent of programming language
- Convert an object into JSON and send it to the server or vice versa
- Free from complex parsing



Creating JSON Objects

- Similar to JS objects
- Written in the form of key/value or name/value pairs
- Name is in double quotes and is separated by value with a colon
- Syntax: {"name":value}



Data Access from JSON Object

- Use dot (.) notation and corresponding property name
- Syntax: <Object>.<propertyName>;
- Example: var firstName = employeeJSON.firstName;
- JSON object is created when document is loaded



Using JSON Arrays

- JSON arrays hold several objects
- To store the data of more than one employee in JSON object
- Include objects in square brackets separated by comma (,) notation
- Example of JSON array holding colors of rainbow:

```
[ "Red", "Blue", "Green", "Yellow", "Violet", "Indigo", "Orange" ]
```



Converting String to JSON Object

- Parse a string using `JSON.parse()`
- Convert string into JSON object or an array
- String should be used in JSON format to avoid errors

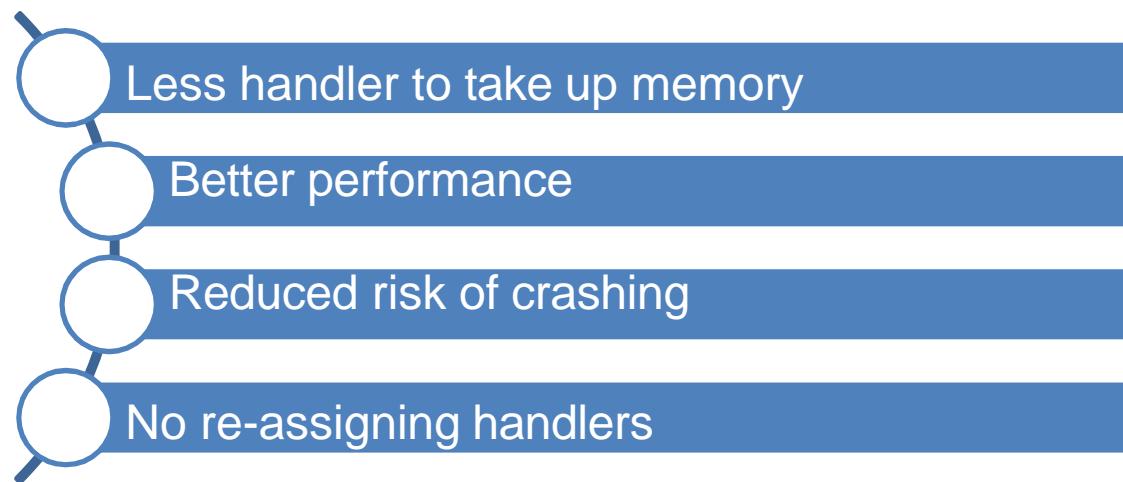


Event Delegation in jQuery (1-2)

- Event is an exact moment when an action is performed by a user
- Designing of dynamic Web pages
- Event listener listens to the raised event which is associated with source of event
- Event handler is a custom function that conveys for handling an event
- Event delegation assigns single listener to a parent element
- On() and delegate() methods enables event delegation

Event Delegation in jQuery (2-2)

- Main features are – event bubbling and target property
- Boost performance
- Target property is the element that originated the event
- Benefits of event delegation:





Summary (1-3)

- jQuery is a lightweight, open-source JS library that makes JS programming easier and more efficient.
- jQuery uses unobtrusive JS, offers a variety of reusable functions, is extensible, and is compatible with modern browsers.
- Web developers can either download the jQuery file or include it by referring to a CDN for using jQuery in their Web applications.
- The customized syntax of jQuery is `$(selector).action()`.
- There are four ways of declaring a function in jQuery, which are basic, as a variable, self-invoking, and user-defined.



Summary (2-3)

- A jQuery Selector refers to a function in the form of an expression for searching the matching element in the DOM.
- The jQuery library allows selecting the HTML elements by ID, class name, tag name, attribute, and attributed values.
- JSON is a lightweight format for exchanging and storing data as well as an easier alternative to XML.
- Unlike in JS, the key or name in JSON is always a string.
- Each name/value pair in JSON format is referred to as a property.
- A JSON object is a collection of properties, each separated by a colon.



Summary (3-3)

- The `JSON.stringify()`method converts a JSON object into a JSON string.
- The `JSON.parse()`method converts a JSON string into a JSON object or array.
- Event delegation allows assigning a single listener to a parent element, which will listen for all descendants corresponding to a selector.
- The `on()` and `delegate()` methods facilitate delegating an event in jQuery.
- Event bubbling in which an event bubbles upwards from the source element in the DOM tree.
- Event bubbling and target property are the main features of event delegation.



Q & A