13.Dec.2021

JQuery Syllabus

1. JQuery Introduction
2. JQuery Selectors
3. JQuery Events
4. JQuery Effects
5. JQuery Traversing and Manipulation
6. JQuery Ajax
7. Data and Utility Methods
8. JQuery noConflict() Method
9. LocalStorage

1. JQuery Introduction

The main purpose is to make it much easier to use JavaScript.

A. About jQuery

i. A lightweight JavaScript Library
ii. Write less, do more!
iii. Simplify code in working with HTML DOM
iv. Author – Jon Resig
v. Initial released date – August 26, 2006
vi. Current stable released version – 3.6.0

NOTE: before getting started with jQuery, you need working knowledge of JavaScript.

B. Unobtrusive JavaScript

i. Write code separately Document Content and Script Content.
ii. Separate HTML structure and JavaScript behaviour to keep your code clean.

Example: 1

Mixed

```
<input type="button" id="btn" onclick="alert('Test')" />
```

```
<input type="button" id="btn" />
```

JavaScript:

Sperate HTML and JavaScript Code

```
var el = document.getElementById('btn');
el.onclick = function(){
  alert('Test');
};
```

C. Why jQuery

i. Write less, do more
ii. Lightweight (only 30 kB)
iii. Support CSS3 Selectors
iv. Cross Browser

Example: 2

JavaScript Code

```
var element = document.getElementById( "myid" );
```

jQuery Code

```
var element = $( "#myid" );
```

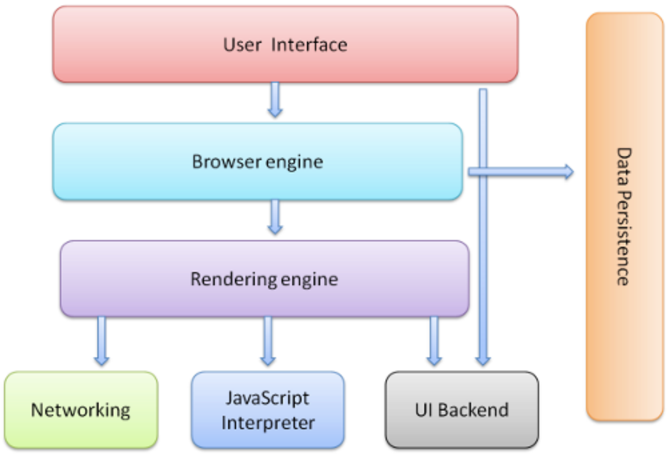

Figure : Browser components

#Different JavaScript Engines

Engines	V8	SpiderMonkey	Chakra
Browser	Chrome / Opera / Edge	Firefox	IE

D. Install and use jQuery Library

i. Go JQuery.com (or) https://code.jquery.com/jquery-3.6.0.min.js

ii. Save into your working directory

iii. Use jQuery

Example: 3

```
<!doctype html>
        <html>
        <head>
          <meta charset="utf-8">
          <title>Demo</title>
        </head>
        <body>
          <a href="http://jquery.com/">jQuery</a>
          <script src="jquery.js"></script>
          <script>
          // Your code goes here.
          </script>
        </body>
</html>
```

Explain above code!

1. Connect jQuery as external script file
2. Store jquery.js file into same directory of HTML file
3. Write custom code in script tag

Google CDN:

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.j
s"></script>
```

E. JQuery Syntax

Basic syntax is***: $(selector). action ()***

i.	$	- define / access jQuery
ii.	(selector)	- find HTML elements
iii.	action ()	- action performed on the element(s)

Example: 4

```
$( this ).hide();              - hide the current element
$( "p" ).hide();               - hide all <p> elements
$( ".test" ).hide();           - hide all elements with class = "test"
$( "#test" ).hide();           - hide the element with id = "test"
```

F. The Document Ready Event

i. All code in ready function
ii. Prevent running jQuery code from running before the document is finished loading.
iii. A good practice

Example: 5

```
$( document ).ready(function() {
    console.log( "ready!" );
});
```

Shorthand

```
$(function() {
    console.log( "ready!" );
});
```

10. JQuery Selectors

Selecting elements is a main part of jQuery library.

A. Element Selector

i. Select elements based on their tag names

`$( "p" )` - selecting all <p> elements

Example: 6

```
$( "p" ).hide();  - hide all <p> elements
```

B. ID Selector

i. Select element with id attribute to find a single, unique element.

`$( "#test" )` - selecting only element with id="test"

Example: 7

```
$( "#test" ).hide();    - hide single element with id="test"
```

C. Class Selector

i. Select elements with class attribute

`$( ".test" )` - selecting all elements with class="test"

Example: 8

```
$( ".test" ).hide();    - hide all elements with class="test"
```

D. More examples of Selectors

Syntax	**Description**
$(this)	Select the current element
$("p.intro")	Select all <p> elements with class="intro"
$("ul li:first")	Select the first <li> element of the first <ul>
$("input[name='first_name']")	Select the input element with name="first_name"
$("#contents ul.people li")	Select all <li> elements in <ul> element with class="people" of id="contents"
$(":button")	Select all <button> elements and input element with type="button"
$("tr:even")	Select all even <tr> elements

E. Choosing Selectors

i. #myTable thead tr th.special (overkill)
ii. #myTable th.special (good)

F. Check selected elements

Example: 9

Doesn't Work!	**Work!**

if ($("div.foo")) {
…
}
	if ($("div.foo").length) {
	…
	}

G. Saving Selection

```
var divs = $( "div" );
```

H. Refining and Filtering Selection

```
$( "div.foo" ).has( "p" );
// div.foo elements that contain <p> tags

$( "h1" ).not( ".bar" );
// h1 elements that don't have a class of bar

$( "ul li" ).filter( ".current" );
// unordered list items with class of current

$( "ul li" ).first();
// just the first unordered list item

$( "ul li" ).eq( 5 );
// the sixth
```

I. Selecting Form Elements

```
$( "form :checked" );
// checkboxes, radio buttons

$( "form :disabled" );
// input with disabled attributes

$( "form :enabled" );
// do not have disabled attributes

$( "form :input" );
// input, textarea, select, button

$( "form :selected" );
// selected items in <option> elements
```

11. JQuery Event Methods

Handle events when “something happens” in HTML.

A. Some event methods

i. `.click()`

ii. `.focus()`

iii. `.change()`

iv. `.on()`

Example: 10

```
// Event setup using a convenience method
$( "p" ).click(function() {
      console.log( "You clicked a paragraph!" );
});

// Equivalent event setup using the `.on()` method
$( "p" ).on( "click", function() {
      console.log( "click" );
});
```

B. Event and callback

i. A callback function is executed after the current effect is finished.

Example: 11

With callback!

```
$( "button" ).click(function(){
      $("p").hide( "slow" ,function(){
               alert("The paragraph is now hidden");
      });
});
```

Without callback!

```
$( "button" ).click(function(){
      $("p").hide(1000);
      alert("The paragraph is now hidden");
});
```

C. Prevent Current Event

To cancel the default functionality for a specific event.

Example: 12

```
$("a").click(function(event){
   alert("The link no longer took");
   event.preventDefault();
});
```

D. Working with form

Prevent form submission event.

Example: 13

```
// Preventing a default action from occurring and
stopping the event bubbling
$( "form" ).on( "submit", function( event ) {

      // Prevent the form's default submission.
      event.preventDefault();

      // Prevent event from bubbling up DOM tree,
      prohibiting delegation
      event.stopPropagation();

      // Make an AJAX request to submit the form data
});
```

12. JQuery Effects

Add simple effects and custom animation to the page.

A. Hide / Show

i. Hide and show HTML elements with the hide() and show() methods. And optionally add speed parameter such as ‘slow’, ‘fast’ and milliseconds.

Example: 14

```
$( "p" ).hide();
$( "p" ).show();
$( "p" ).hide("slow");
$( "p" ).hide();
```

ii. Using toggle() methods for shown elements are hidden and hidden elements are shown.

Example: 15

```
$( "p" ).toggle();
```

B. Fading

i. Fading elements in and out of visibility.

Example: 16

```
$( "p" ).fadeIn("slow");         - shown the hidden elements
$( "p" ).fadeOut(1000);          - hidden the shown elements
$( "p" ).fadeToggle();           - Using show and hide
$( "p" ).fadeTo("fast", 0.6);    - Changing the opacity value (between 0 and 1)
```

C. Sliding

i. Sliding effect to HTML elements up and down.

Example: 17

```
$( "p" ).slideDown();
$( "p" ).slideUp();
$( "p" ).slideToggle();
```

D. Animations

i. .animate() method is used to create custom animations.

Example: 18

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

Note: all HTML elements have a static position and cannot moved. So CSS position property change into relative, fixed or absolute.

Example: 19

```
$("button").click(function(){
    $("div").animate({
        left:'250px',
        height:'+=150px',
        width:'+=150px'
    });
});
```

Example: 20

```
$("button").click(function(){
    var div=$("div");
    div.animate({left:'100px'},"slow");
    div.animate({fontSize:'3em'},"slow");
});
```

ii. Use .stop() method to stop animation

Example: 21

```
$("#stop").click(function(){
        $("#panel").stop();
});
```

13. JQUERY TRAVERSING AND MANIPULATION

A. Traversing

i. When selecting the html elements, traversing is easy to use with parents, children and siblings.

Example: 22

```
<div class="grandparent">
    <div class="parent">
        <div class="child">
            <span class="subchild"></span>
        </div>
    </div>
    <div class="surrogateParent1"></div>
    <div class="surrogateParent2"></div>
</div>
```

Parent Methods

```
// returns [ div.child ]
$( "span.subchild" ).parent();
// returns [ div.parent ]
$( "span.subchild" ).parents( "div.parent" );
// returns [ div.child, div.parent, div.grandparent ]
$( "span.subchild" ).parents();
// returns [ div.child, div.parent ]
$( "span.subchild" ).parentsUntil( "div.grandparent" );
// returns [ div.child ]
$( "span.subchild" ).closest( "div" );
// returns [ div.child ] as the selector is also included in
the search:
$( "div.child" ).closest( "div" );
```

Children Methods

```
// returns [ div.parent, div.surrogateParent1,
div.surrogateParent2 ]
$( "div.grandparent" ).children( "div" );
// returns [ div.child, div.parent, div.surrogateParent1,
div.surrogateParent2 ]
$( "div.grandparent" ).find( "div" );
```

Sibling Methods

```
// returns [ div.surrogateParent1 ]
$( "div.parent" ).next();
// returns [] as No sibling exists before div.parent
$( "div.parent" ).prev();
// returns [ div.surrogateParent1, div.surrogateParent2 ]
$( "div.parent" ).nextAll();
// returns [ div.surrogateParent1 ]
$( "div.parent" ).nextAll().first();
// returns [ div.surrogateParent2 ]
$( "div.parent" ).nextAll().last();
// returns [ div.surrogateParent1, div.parent ]
$( "div.surrogateParent2" ).prevAll();
// returns [ div.surrogateParent1 ]
$( "div.surrogateParent2" ).prevAll().first();
// returns [ div.parent ]
$( "div.surrogateParent2" ).prevAll().last();
```

```
// returns [ div.surrogateParent1, div.surrogateParent2 ]
$( "div.parent" ).siblings();
// returns [ div.parent, div.surrogateParent2 ]
$( "div.surrogateParent1" ).siblings();
```

B. Chaining

Do one after the another on the same elements.

Example: 23

```
$("#p1").css("color","red") .slideUp(2000) .slideDown(2000);
```

C. Getting and setting elements

Powerful methods for manipulating elements and attributes.

Methods	Description
.html()	Get or set the HTML contents.
.text()	Get or set the text contents; HTML will be stripped.
.attr()	Get or set the value of the provided attribute.
.width()	Get or set the width in pixels of the first element in the selection as an integer.
.height()	Get or set the height in pixels of the first element in the selection as an integer.
.position()	Get an object with position information for the first element in the selection, relative to its first positioned ancestor. *This is a getter only*.
.val()	Get or set the value of form elements.

Example: 24

```
// Changing the HTML of an element.
$( "#myDiv p:first" ).html( "New <strong>first</strong>
paragraph!" );
```

```
// Manipulating a single attribute.
$( "#myDiv a:first" ).attr( "href", "newDestination.html" );
```

```
// Manipulating multiple attributes.
$( "#myDiv a:first" ).attr({
    href: "newDestination.html",
    rel: "nofollow"
});
```

```
// Getting input values.
var value = $( "#test" ).val();
console.log(value);
```

D. Add and Remove Elements

It is easy to add and remove elements / content.

i. Add new HTML elements using append(), prepend(), after(), before()

Example: 25

```
$("p").append("Some appended text.");
$("p").prepend("Some prepended text.");
$("img").after("Some text after");
$("img").before("Some text before");
```

ii. Remove existing HTML elements

Example: 26

```
$("#div1").remove();
$("#div1").empty();
```

E. CSS, Styling and Dimensions

i. .css() method set and get properties of selected elements.

- **Getting CSS properties**

```
$("h1").css("fontSize");
$("h1").css("font-size");
```

- **Setting CSS properties**

```
$("h1").css("fontSize","100px");                    - individual
$("h1").css({                                       - multiple properties
        fontSize: "100px",
        color: "red"
});
```

- **CSS classes for styling**

```
var h1 = $("h1");
h1.addClass("red");
h1.removeClass("red");
h1.toggleClass("red");

if(h1.hasClass("red")){
        // your code
});
```

- **Dimensions**

```
Set => $("h1").width("50px");
Get => $("h1").width();

Set => $("h1").position();                  // the first <h1>
```

14. AJAX (ASYNCHRONOUS JAVASCRIPT AND XML)

Exchanging Data with a Server, and Updating Parts of a Web Page - Without Reloading the Whole Page. Transport data with plain HTML or JSON.

i. $.ajax()
ii. $.get() – not changing data on the server
iii. $.post() – change data of the server

Example: 27 # core $.ajax()

```
$.ajax({
    // The URL for the request
    url: "post.php",
    // The data to send (will be converted to a query string)
    data: {
        id: 123
    },
    // Whether this is a POST or GET request
    type: "GET",
    // The type of data we expect back
    dataType : "json",
})

   // Code to run if the request succeeds (is done);
  // The response is passed to the function
  .done(function( json ) {

  })
  // Code to run if the request fails; the raw request and
  // status codes are passed to the function
  .fail(function( xhr, status, errorThrown ) {
    alert( "Sorry, there was a problem!" );
    console.log( "Error: " + errorThrown );
    console.log( "Status: " + status );
    console.dir( xhr );
  })
  // Code to run regardless of success or failure;
  .always(function( xhr, status ) {
    alert( "The request is complete!" );
  });
```

Example: 28 # $.get()

```
// Get plain text or HTML

$.get( "/users.php", {
    userId: 1234
}, function( resp ) {
    console.log( resp ); // server response
});
```

Example: 29 # $.post()

```
// Get plain text or HTML
$.post( "/users.php", {
    username: “mg mg”,
   city: “Mandalay”
}, function( resp ) {
    console.log( resp ); // server response
});
```

iv. Working with Form

Example: 30

```
// Turning form data into a query string
$( "#myForm" ).serialize();
```

```
// Creating an array of objects containing form data
$( "#myForm" ).serializeArray();
```

15. Data and Utility Methods

Manipulating data using data method.

i. .data() to set and get data with selected HTML element

Example: 31

```
$( "#myDiv" ).data( "keyName", { foo: "bar" } );
$( "#myDiv" ).data( "keyName" );
```

ii. .trim(), $.each(), $.inArray() using utility methods

Example: 32

```
// Returns "lots of extra whitespace"
$.trim( "    lots of extra whitespace    " );
```

```
$.each([ "foo", "bar", "baz" ], function( idx, val ) {
    console.log( "element " + idx + " is " + val );
});
```

```
$.each({ foo: "bar", baz: "bim" }, function( k, v ) {
    console.log( k + " : " + v );
});
```

```
var myArray = [ 1, 2, 3, 5 ];

if ( $.inArray( 4, myArray ) !== -1 ) {
    console.log( "found it!" );
}
```

iii. Testing Type

Example: 33

```
$.isArray([]); // true
$.type( "test" ); // "string"
```

16. JQuery noConflict Method

Avoiding conflicts with other libraries (no-conflict mode)

Example: 34

```
<!-- Putting jQuery into no-conflict mode. -->
<script src="prototype.js"></script>
<script src="jquery.js"></script>
<script>

var $j = jQuery.noConflict();

$j(document).ready(function() {
    $j( "div" ).hide();
});
```

17. Local Storage

i. Browser storage has no expiration time
ii. Store with DOMString format
iii. Not for store server information and data

Example: 35 # useful methods

```
localStorage.setItem('myCat', 'Tom');

const cat = localStorage.getItem('myCat');

localStorage.removeItem('myCat');

localStorage.clear();
```

Exercises of day by day

1. **JavaScript revision and jQuery introduction** (JavaScript Calculator)
2. **Selectors and Events** (show welcome message when user click a button)
3. **Tab** (on/off div like as a bootstrap tab)
4. **Add to Cart** (add and remove item with no storage – ready for LS)
5. **Ajax** (working with inspiration quotes API)
6. **Local Storage** (Shopping Cart)