**Question 1: What is JQuery?**  
**Answer 1:**  
JQuery is a cross-browser lightweight JavaScript library. In simple words jQuery has been designed to make navigation to any element easier, or adding/invoking event handlers on your HTML page and also simplify the way you access the elements in your web pages, provide help in working with client-side events, enable visual effects like animation, and make it easier to use Ajax in your applications. You can download the latest version of jQuery from the [official web site](http://jquery.com/).  
  
A quick look at what is available in jQuery:

* Cross-browser support and detection.
* AJAX functions
* CSS functions
* DOM manipulation
* DOM transversal
* Attribute manipulation
* Event detection and handling.
* JavaScript animation
* Hundreds of plugins for pre-built user interfaces, advanced animations, form validation, etc.
* Expandable functionality using custom plugins.

For more follow the link:

* [Introduction To jQuery](http://www.c-sharpcorner.com/UploadFile/036f9e/introduction-of-jquery/)

**Question 2. What is JQuery.noConflict?**  
  
**Answer 2:** jQuery no-conflict is an option given by jQuery to overcome the conflicts between the different js frameworks or libraries. When we use jQuery no-conflict mode, we are replacing the $ to a new variable and assigning to jQuery some other JavaScript libraries. Also use the $ (Which is the default reference of jQuery) as a function or variable name what jQuery has. And in our development life, we are not at all strict to only jQuery.  
  
To know more about **JQuery.noConflict** follow the link:

* [JQuery No-Conflict And Using Different Versions Of jQuery](http://www.c-sharpcorner.com/UploadFile/65794e/jquery-no-conflict-and-using-different-versions-of-jquery/)

**Question 3. What is a CDN?**  
  
**Answer 3:**Content Delivery Network (CDN) in simple terms is a collection of servers spread across the globe. In other words, a CDN is a network of servers in which each request will go to the closest server.  
  
**Need For a CDN**  
  
For any web application, data can be categorized into either static or dynamic content. Dynamic content is the one that generally comes from a database. Static content is like CSS, images, JavaScript, flash files, video files and so on.  
  
Now one may ask, how are requests served when a user enters an URL into the browser? Interesting, let's have a look at it. Before knowing a CDN and its usage, it is very important to understand this process:

* [What and Why of CDN](http://www.c-sharpcorner.com/UploadFile/41e70f/what-and-why-of-cdn/)

**Question 4. What are selectors in jQuery and how many types of selectors are there?**  
  
**Answer 4:** The basic operation in jQuery is selecting an element in DOM. This is done with the help of $() construct with a string parameter containing any CSS selector expression. $() will return zero or more DOM elements on which we can apply a effect or style.  
  
**$(document).ready()** indicates that code in it need to be executed once the DOM got loaded. It won't wait for the images to load for executing the jQuery script. We created an anonymous function inside ready() function to hide div1.   
  
We can rewrite $(document).ready() as jQuery (document).ready(), since $ is an alias for jQuery. Always use jQuery in place of $, if you are using more than one JavaScript library to resolve conflicts with jQuery library. The methods called on $(), will implicitly applied on all the elements returned by it without need of explicit looping. Let's say, $('.myclass').hide() will hide all elements with class as myclass with implicit looping.  
  
As we discussed earlier, $() accepts a string parameter having tag name [like div, p] or Element Id or class name as shown in the following table:

|  |  |  |
| --- | --- | --- |
| **Selector** | **jQuery Syntax** | **Description** |
| Tag Name | $('div') | All div tags in the document |
| ID | $('#TextId') | Selects element with ID as TextId. It starts with # followed Element Id. |
| Class | $('.myclass') | Selects all elements with class as myclass. It starts with '.' followed by class name |

For more details read this article:

* [Selectors in jQuery](http://www.c-sharpcorner.com/uploadfile/satisharveti/selectors-in-jquery/)

**Question 5. What is the use of jQuery .each() function?**  
  
**Answer 5:** The "j**Query.each()**" function is a general function that will loop through a collection (object type or array type). Array-like objects with a length property are iterated by their index position and value. Other objects are iterated on their key-value properties. The "jQuery.each()" function however works differently from the $(selector).each() function that works on the DOM element using the selector. But both iterate over a jQuery object.   
  
**Callback method**  
  
In the "jQuery.each()" method we're able to pass in an arbitrary array or object in which for each item will have the callback function executed.   
  
The "$.each()" function loops over any type of collection, either an array or an object collection. The "jQuery..each()"function has a callback function in which we pass the indexing value and the corresponding value of each item of the array each time. We can access the value of the current index position in the loop using the "this" keyword and append it in any DOM element.  
  
When you pass the array-like object to the .each() function, the callback can accept two arguments: index of the item, where index is the numerical zero-based index in the array of the current items and item is the value of the current array.  
  
**For example:**If we pass an array to the each function, it iterates over items in the array and accesses both the current item and its index position.  
  
**Syntax:**

1. jQuery.each(collection, callback(indexInArray, valueOfElement))
3. < script type = "text/javascript" >
4. $(document).ready(function() {
6. var arr = ["Goergie", "Johnson", "Agile", "Harrison", "Gaurav"];
8. $.each(arr, function(index, value) {
9. alert('Position is : ' + index + ' And Value is : ' + value);
10. });
12. });
13. < /script>

For more details follow the link:

* [$.each() Function in jQuery](http://www.c-sharpcorner.com/UploadFile/99bb20/each-function-in-jquery/)

**Question 6. What is difference between prop and attr?**  
  
**Answer 6:**  
*jQuery.attr()*  
  
Gets the value of an attribute for the first element in the set of matched elements.  
  
Whereas:  
  
*jQuery. prop ()*  
  
Gets the value of a property for the first element in the set of matched elements.  
  
**What Attributes actually are**  
  
Attributes carry additional information about an HTML element and come in name=”value” pairs. You can set an attribute for an HTML element and define it when writing the source code.   
  
**For example**

1. <input id="txtBox" value="Jquery" type="text" **readonly**="readonly" />

As shown above, “**id**”, "**type**” and “**value**" are attributes of the input elements.   
  
For more details follow the link:

* [Difference Between prop and attr in jQuery](http://www.c-sharpcorner.com/UPLOADFILE/97FC7A/DIFFERENCE-BETWEEN-PROP-AND-ATTR-IN-JQUERY/)

**Question 7. What is jQuery UI?**  
  
**Answer 7:**jQuery UI enable our applications to have a cool user interface and animation in a faster way. It is the set of plug-ins that include interface interactions, effects, animations, widgets and themes built on the JavaScript Library. jQuery is used to create cohesive and consistent APIs. It is a method that we can use to extend jQuery prototype objects. By that prototype object you can enable all jQuery objects to inherit any method that you add.  
  
**Interactions**  
We can use interactions for basic mouse-based behaviours to any element. Examples of Interactions are the following:

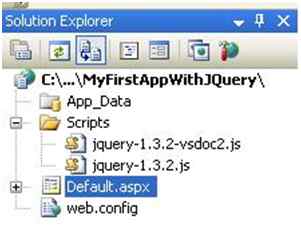
* Draggable
* Droppable
* Resizable
* Selectable
* Sortable

[Getting Started With jQuery UI Plugin](http://www.c-sharpcorner.com/UPLOADFILE/22D000/GETTING-STARTED-WITH-JQUERY-UI-PLUGIN/)  
 **Question 8. What are the methods used to provide effects?**  
  
**Answer 8:** jQuery provides many amazing effects, we can apply these effects quickly and with simple configuration. The effect may be hiding, showing, toggling, fadeout, fadein, fadeto and so on toggle(), Show() and hide() methods. Similarly we can use other methods as in the following:

* **animate( params, [duration, easing, callback] )** This function makes custom animations for your HTML elements.
* **fadeIn( speed, [callback] )**This function fades in all the matched elements by adjusting their opacity and firing an optional callback after completion.
* **fadeOut( speed, [callback] )** This function is used to fade out all the matched elements by adjusting their opacity to 0, then setting the display to "none" and firing an optional callback after completion.
* **fadeTo( speed, opacity, callback )**This function fade the opacity of all the matched elements to a specified opacity and firing an optional callback after completion.
* **stop( [clearQueue, gotoEnd ])**This function stops all the currently running animations.

For More Info follow the link:

* [jQuery Effects Methods](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-effects-methods/)

**Question 9. How we can use jQuery in ASP.NET?**  
  
**Answer 9:**As you know jQuery is a fast, lightweight JavaScript library that is CSS3 compliant and supports many browsers. The jQuery framework is extensible and handles the DOM manipulations, CSS, AJAX, Events and Animations, very nicely.   
  
**Some differences between JavaScript and jQuery**  
  
JavaScript is a language whereas jQuery is a library written using JavaScript.  
  
Let us go through an example, which will help you in understanding the use of jQuery with ASP.NET application.  
  
Make a folder with the name Scripts inside your application. Right click on **Scripts folder > Add Existing Item > Browse**to the path where you downloaded the jQuery library (**jquery-1.3.2.js**) and the intellisense documentation **(jquery-1.3.2-vsdoc2.js).** Select the files and click **Add**. The structure will look similar to the following image:  
  
  
  
In this example, I am going to display an alert on **asp:Button** click using jQuery.  
  
**Here's the Default.aspx code**

1. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2. <html xmlns="http://www.w3.org/1999/xhtml">
4. <head runat="server">
5. <title>My First Application With JQuery</title>
7. <script src="Scripts/jquery-1.3.2.js" type="text/javascript"></script>
9. <script type="text/javascript">
10. $(document).ready(function() {
11. $("#Button1").click(function() {
12. alert("Welcome jQuery !");
13. });
14. });
15. </script>
17. </head>
19. <body>
20. <form id="form1" runat="server">
21. <div>
22. <asp:Button ID="Button1" runat="server" Text="Click Me" />
23. </div>
24. </form>
25. </body>
27. </html>

For more code examples follow the link:

* [An Introduction to jQuery with ASP.NET](http://www.c-sharpcorner.com/uploadfile/rahul4_saxena/an-introduction-to-jquery-with-Asp-Net/)  
  Or
* [Consuming ASP.NET Web Service through jQuery](http://www.c-sharpcorner.com/UploadFile/sapnabeniwal/consuming-asp-net-web-service-through-jquery/)

**Question 10. How can we use hide() method on a button click using jQuery?**  
 **Answer 10:**In jQuery the hide () method is very useful. By using this method you can hide HTML elements with the hide() method. In this example we create a div element which contains text. When we click on the Button the text we use in the div will be hidden.  
  
**Example**  
We are showing you the complete code for the .aspx page below.

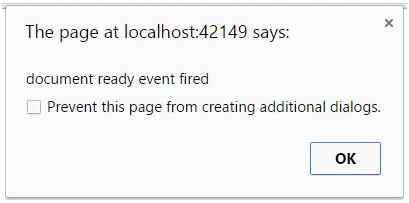
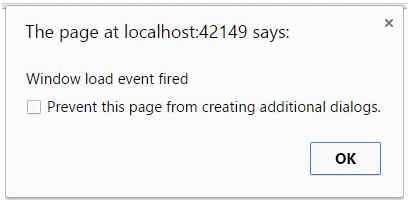
1. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2. <html xmlns="http://www.w3.org/1999/xhtml">
4. <head>
5. <title>here</title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <script type="text/javascript">
8. $(document).ready(function() {
9. $("button").click(function() {
10. $("#div1").hide();
11. });
12. });
13. </script>
14. </head>
16. <body>
17. <h2>
18. This **is** a heading</h2>
19. <div id="div1">
20. jQuery **is** great library **for** developing ajax based application.
21. <br> jQuery **is** great library **for** the JavaScript programmers, which simplifies the development of web 2.0 applications.
22. <br />
23. <br />
24. <br />
25. </div>
26. <button>
27. Hide</button>
28. </body>
29. <html>

For more follow the link:

* [html() and hide() method in jQuery](http://www.c-sharpcorner.com/UploadFile/rohatash/html-and-hide-method-in-jquery/)

**Question 11. What is the difference between $(window).load and $(document).ready function in jQuery?**  
  
**Answer 11:** $(window).load is an event that fires when the DOM and all the content (everything) on the page is fully loaded. This event is fired after the ready event.  
  
Let's look at an example.

1. <script type="text/javascript" lang="ja">
2. $(window).load(function() {
3. alert("Window load event fired");
4. });
6. $(document).ready(function() {
7. alert("document ready event fired");
8. });
9. </script>

In the preceding JavaScript, we created an anonymous function that contains an alert message. So, when the preceding two events are fired an alert window will pop-up.  
  
Run the application and let's see which event is fired first.  
  
The document ready function will be fired first.  
  
  
  
Then the window load event will be fired.  
  
  
 **When to use $(window).load instead of $(document).ready**  
  
In most cases, the script can be executed as soon as the DOM is fully loaded, so ready() is usually the best place to write your JavaScript code. But there could be some scenario where you might need to write scripts in the load() function. For example, to get the actual width and height of an image.  
  
As we know the $(window).load event is fired once the DOM and all the CSS, images and frames are fully loaded. So, it is the best place to write the jQuery code to get the actual image size or to get the details of anything that is loaded just before the load event is raised.  
  
Follow the link for more details.

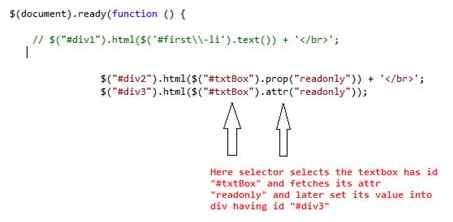
* [Basics of jQuery: Part 1](http://www.c-sharpcorner.com/UploadFile/219d4d/basics-of-jquery-part-1/)

**Question 12. How to handle Controls attribute Using jQuery?**  
  
**Answer 12:** For handle Controls attribute using jQuery we used **.addClass(),** **.removeClass(),** **.css(), .toggleClass(),**etc to manage all css and html attributes of any html control.  
  
You can follow the link:

* [Handle Controls Attribute Using jQuery](http://www.c-sharpcorner.com/UploadFile/abhikumarvatsa/handle-controls-attribute-using-jquery/)

**Question 13: What is chaining in jQuery?**  
  
**Answer 13:**Chaining is a  powerful feature of jQuery. Chaining means specifying multiple functions and/or selectors to an element.  
  
Chaining reduces the code segment and keeps it very clean and easy to understand. Generally chaining uses the jQuery built in functions that makes compilation a bit faster.  
  
By using chaining we can write the above code as follows:

1. $(document).ready(function() {
2. $("#div2").html($("#txtBox").prop("readonly")) + '</br>';
3. $("#div3").html($("#txtBox").attr("readonly"));
4. });

The code segment above is described by the following image:  
  
  
  
Follow the link for more details:

* [jQuery Interview Questions and Answers With Practices: Part 2](http://www.c-sharpcorner.com/UploadFile/97fc7a/jquery-interview-question-and-answer-with-practices-part-2/)  
    
  Or
* [jQuery - "Write Less Do More": Day 1](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-write-less-do-more-day-1/)

**Question 14: How to work with parent(), children() and siblings() methods in jQuery?**  
  
**Answer 14:**The parent() function returns the parent of the selected element by calling the jQuery parent() function. The siblings() function returns all the siblings of given HTML elements.

* [Getting Started With jQuery Traversing](http://www.c-sharpcorner.com/UploadFile/22d000/getting-started-with-jquery-traversing/)

**Question 15. What is jQuery Datepicker in jQuery ?**  
  
**Answer 15:**As per jQueryUI Documents, the jQuery UI Datepicker is a highly configurable plugin that adds datepicker functionality to your pages. You can customize the date format and language, restrict the selectable date ranges and add in buttons and other navigation options easily.  
  
By default, the datepicker calendar opens in a small overlay when the associated text field gains focus. For an inline calendar, simply attach the datepicker to a div or span.  
  
You must use the following jQuery reference in your **HTML** Code, otherwise it will not work.

1. <head>
2. <link rel="stylesheet" href="//code.jquery.com/ui/1.11.4/themes/smoothness/jquery-ui.css">
3. <script src="//code.jquery.com/jquery-1.10.2.js"></script>
4. <script src="//code.jquery.com/ui/1.11.4/jquery-ui.js"></script>
5. </head>

Display month & year menus in jQuery Datepicker.  
  
**HTML**

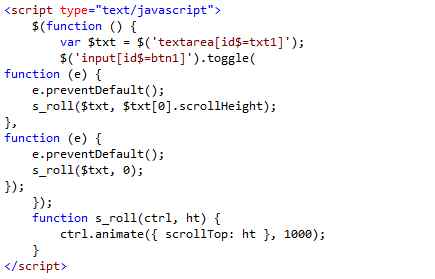
1. Date : <input id="Datepicker" type="text" />

**jQuery**  
  
The following code will show the **Month** & **Year** menus in kjquery Datepicker. The "**yearRange**" will specify the range of the year you want.

1. $(function() {
3. $('#Datepicker').datepicker({
4. dateFormat: 'dd/mm/yy',
5. changeMonth: **true**,
6. changeYear: **true**,
7. yearRange: '1950:2100'
9. });
11. })

For more details follow the link:

* j[Query Datepicker - Part 2](http://www.c-sharpcorner.com/UploadFile/2a6dc5/jquery-datepicker-part-2/)

**Question 16. How to use scrolling an ASP.NET Multiline Textbox using jQuery?**  
  
**Answer 16:** We will write the jQuery code which will be inside the <script></script> tag and always placed between head section or body section. Its your choice that where you want to place it. Let's see the jQuery code here:  
  
  
 **Explanation**  
Here we will explain the jQuery code which is given above. When the user clicks on the button (btn), we toggle the click behavior. On the first click, we cancel the postback by using e.preventDefault() and then call a function called s\_roll() passing in the textarea and the scrollHeight. The code $txt[0].scrollHeight is for scrolling downwards.

1. e.preventDefault();
2. s\_roll($txt, $txt[0].scrollHeight);

Again when the user clicks the button (btn) again, the postback is cancelled and the scrollHeight is set to 0. And by using it the multiline textbox will be scrolling upwards.

1. e.preventDefault();
2. s\_roll($txt, 0);

Here the scrollArea() function accepts the textarea that is to be scrolled as well as the scrollHeight. We then animate the scrollTop property to scroll upwards/downwards depending on the height parameter. The duration of the animation is set to 1000 milliseconds which provides a smooth scrolling effect and you can change according to your requirement. The function is given below which will animate it.

* [Advanced Typing Scroller Using jQuery](http://www.c-sharpcorner.com/UploadFile/sapnabeniwal/advanced-typing-scroller-using-jquery/)

**Question 17. What is Ajax in jQuery?**  
  
**Answer 17:**AJAX stands for “Asynchronous JavaScript and XML”. AJAX is about exchanging data with a server, without reloading the whole page. It is a technique for creating fast and dynamic web pages.  
  
In .NET, we can call server side code using two ways:

1. ASP .NET AJAX
2. jQuery AJAX

In this article we will focus on jQuery Ajax.  
  
**$.ajax () Method**  
  
JQuery’s core method for creating Ajax requests. Here are some jQuery AJAX methods:

* **$.ajax()**-Performs an async AJAX request.
* **$.get()**-Loads data from a server using an AJAX HTTP GET request.
* **$.post()** - Loads data from a server using an AJAX HTTP POST request.

To know more [click](https://learn.jquery.com/ajax/jquery-ajax-methods/).  
  
**$.ajax () Method Configuration option**  
  
Options that we use:

* async
* type
* url
* data
* datatype
* success
* error

Let’s have a detailed overview:  
  
**async**  
  
Set to false if the request should be sent synchronously. Defaults to true.   
  
Follow the link for more details:

* [ASP.NET MVC Application - Using jQuery, AJAX](http://www.c-sharpcorner.com/UploadFile/302f8f/Asp-Net-mvc-using-jquery-ajax/)

**Question 18. Define slideToggle() effect ?**  
  
**Answer 18:**The slide methods do the up and down element. To implement slide up and down on element jQuery here are the three methods:

* slideDown()
* slideUp()
* lideToggle()

And how to use them:  
  
**1. slideDown() Method**  
This function is used to slide and hide an element on down side:

1. <script type="text/javascript">
2. $(document).ready(function() {
3. $("#btnSlideDown").click(function() {
4. $("#login\_wrapper").slideDown();
5. **return** **false**;
6. });
7. });
8. </script>

**2. slideUp() Method**  
This function is used to slide and show element up side:

1. <script type="text/javascript">
2. $(document).ready(function() {
3. $("#btnSlideUp").click(function() {
4. $("#login\_wrapper").slideUp();
5. **return** **false**;
6. });
7. });
8. </script>

**3. slideToggle() Method**  
  
This method is between slideUp() method and slideDown() method. It shows/hides an element in up/down side:

1. <script type="text/javascript">
2. $(document).ready(function() {
3. $("#btnSlideToggle").click(function() {
4. $("#login\_wrapper").slideToggle();
5. **return** **false**;
6. });
7. });
8. </script>

For more details follow the link:

* [jQuery Effects using Slide methods](http://www.c-sharpcorner.com/UploadFile/3d39b4/jquery-effects-using-slide-methods/)

**Question 19: What are the advantages of jQuery?**  
  
**Answer 19:**In JavaScript we write more code because it doesn't have more functions like animation effects functions and event handling. So if you use JavaScript, developers write more code and they often feel embrace when they execute the code on the browser and get a problem related to cross-browser support. To solve these types of problems, John has created a JavaScript library with a nice motto, "*write less and do more"* in 2006; that is called jQuery. So you can use all the functions and other capabilities available in JavaScript. It saves developer's time, testing efforts, lines of code and improves their productivity and efficiency of development. The following are some important points to use jQuery.

* Fully documented
* Lot of plugins
* Small size
* Everything works in IE 6.0+, FF 2.0+, Safari 3.0+, Chrome and Opera 9.0+

To know more about jQuery follow the link:

* [jQuery Overview](http://www.c-sharpcorner.com/UploadFile/rohatash/jquery-overview/)

**Question 20: Define bind() and unbind elements in jQuery?**  
  
**Answer:** The jQuery bind() method attaches an event handler to elements, whereas unbind() detaches an existing event handler from elements. Use basic HTML code to create the HTML elements.

* [jQuery Unbind() method to remove an attached event handler](http://www.c-sharpcorner.com/UploadFile/88d8c0/unbind-method-for-group-of-html-div-elements-using-jquery/)

**Question 21: How to remove a DOM Element using jQuery?**  
  
**Answer 21:** Sometimes we get a requirement to delete all child nodes and remove DOM elements using jQuery to make a dynamic web page. In this scenario jQuery provides a couple of methods to remove DOM elements. Here are the options:

* empty()
* remove()
* html()

For more details follow the below given link:

* [Remove DOM Elements Dynamically in jQuery](http://www.c-sharpcorner.com/UploadFile/manas1/remove-dom-elements-dynamically-in-jquery/)  
  Or
* [Remove a DOM Element Using jQuery](http://www.c-sharpcorner.com/UploadFile/65794e/remove-a-dom-element-using-jquery/)

**Question 22: What are the features of jQuery used in web applications?**  
  
**Answer 22:** jQuery has some important features such as event handling, Ajax support and animation effects functions. Here is the list of important features supported by jQuery.

1. **HTML/DOM Manipulation:**JavaScript do not have any features related to the DOM, but JavaScript in the browser does include some intelligence about the DOM.  
     
   **Example:** GetElementById() function  
     
   DOM is an important feature of jQuery. It defines the logical structure of documents and the way a document is accessed and manipulated. jQuery has the extra intelligence regarding  DOM.
2. **Event Handling:**jQuery introduced a feature called Event handling. Before starting event handling you need to understand event. Events are actions. This means that you can write code that runs when a user clicks on a certain part of the page, or when mouse is moved over a form element. jQuery contains many events, such as a user clicking on a button, moving a mouse over an element and so on.
3. **Ajax Support:** For example, when you select an item from a DropDownList or other control on the same page then that can cause loss of data. Ajax is used to update the part of the web page without reloading the page. For example, if you create a search functionality in your website like Google Search. When you enter text into the Search TextBox then without reloading the page you see the related text. You can do it easily using Ajax Methods.
4. **Animations in jQuery:** The jQuery comes with plenty of built-in animation effects that you can use in your websites. For example, animation, show, hide and so on. In jQuery the animate() method is very useful. By using this method we can change the size of elements.  
   * Animation method
   * Show method
   * Hide method

For more details follow link:

* [jQuery Overview](http://www.c-sharpcorner.com/UploadFile/rohatash/jquery-overview/)

**Question 23. What is the use of jQuery filter?**  
  
**Answer 23:**JQuery supports various types of filters, such as:

1. .eq()
2. .first()
3. .last()
4. .filter()
5. .has()
6. .not()

For more details follow link:

* [Filter in jQuery](http://www.c-sharpcorner.com/UploadFile/jayendra/filter-in-jquery/)

**Question 24. What is the use of jQuery.ajax method()?**  
  
**Answer 24:**The ajax() method is used to do an AJAX (asynchronous HTTP) request. It provides more control of the data sending and on response data. It allows the handling of errors that occur during a call and the data if the call to the ajax page is successful.  
  
Here is the list of some basic parameters required for jQuery.ajax Method:

* **type:**Specifies the type of request (GET or POST).
* **url:**Specifies the URL to send the request to. The default is the current page.
* **contentType:** The content type used when sending data to the server. The default is "*application/x-www-form-urlencoded*".
* **dataType:** The data type expected of the server response.
* **data:**Specifies data to be sent to the server.
* **success(result,status,xhr):**A function to run when the request succeeds.
* **error(xhr,status,error):** A function to run if the request fails.

For more details follow link:

* [Introduction to jQuery.ajax Call in ASP.NET](http://www.c-sharpcorner.com/UploadFile/0cefc4/introduction-to-jquery-ajax-call-in-Asp-Net/)

**Question 25. What is an attribute in jQuery?**  
  
**Answer 25:**There are many important properties of DOM or HTML elements such as for the <img> tag the src, class, id, title and other properties. jQuery provides ways to easily manipulate an elements attribute and gives us access to the element so that we can also change its properties.

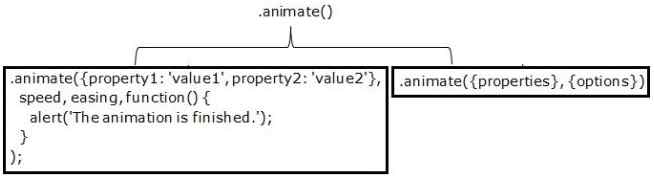
1. attr( properties ) - Set a key/value object as properties to all matched elements.
2. attr( key, fn ) - Set a single property to a computed value, on all matched elements.
3. removeAttr( name ) - Remove an attribute from each of the matched elements.
4. hasClass( class ) - Returns true if the specified class is present on at least one of the set of matched elements.
5. removeClass( class ) - Removes all or the specified class(es) from the set of matched elements.
6. toggleClass( class ) - Adds the specified class if it is not present, removes the specified class if it is present.
7. html( ) - Gets the HTML contents (innerHTML) of the first matched element.
8. html( val ) - Sets the HTML contents of every matched element.
9. text( ) - Gets the combined text contents of all matched elements.
10. text( val ) - Sets the text contents of all matched elements.
11. val( ) - Gets the input value of the first matched element.

[jQuery Attribute Basics](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-attributes-basics/)  
  
**Question 26. What are jQuery Events?**  
  
**Answer 26:**When we design dynamic web pages, we need to apply some events such as Mouse Click, for forms submit the form after a button click, change a color after a click, etc. So in layman language, events are actions that are used for dynamic web pages. When we perform these actions on an HTML page, we can do whatever we want.  
  
We use some event handlers to perform the action. Some important handlers are bind(), unbind(), blur(), off(), hover(), on(), one(), ready(), trigger() etc.

* [Overview of jQuery Events](http://www.c-sharpcorner.com/UploadFile/79037b/overview-of-jquery-events/)

**Question 27. What is the jQuery Unbind() method?**  
  
**Answer 1:**The jQuery bind() method attaches an event handler to elements, whereas unbind() detaches an existing event handler from elements. Use basic HTML code to create the HTML elements.  
  
For more details follow link:

* [jQuery Unbind() method to remove a attached event handler](http://www.c-sharpcorner.com/UploadFile/88d8c0/unbind-method-for-group-of-html-div-elements-using-jquery/)

**Question 28. What is the jQuery Animation?**  
  
**Answer 28:** In short, the *.animate* method is used to perform a custom animation of a set of CSS properties. The .*animate()* method comes in two flavours. The first takes four arguments and the second takes two arguments.  
  
  
  
For more details follow link:

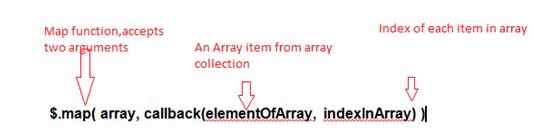
* [jQuery .animate() Method Part 1](http://www.c-sharpcorner.com/UploadFile/abhikumarvatsa/jquery-animate-method-part-1/)

**Question 29. How can you find browser and browser version in jQuery?**  
  
**Answer 29:** using $.browser property of jQuery  returns the browser information.  
  
*Using $.browser is not recommended by jQuery itself, so this feature has been moved to the jQuery.migrate plugin which is available for downloading if the user want. It is a vulnerable practice to use the same. Use it only if needed. It is always better to not use browser specific codes.*  
  
For more details follow link:

* [Find Browser And Browser Version Using jQuery](http://www.c-sharpcorner.com/UploadFile/65794e/find-browser-and-browser-version-using-jquery/)

**Question 30. What is $.each() function in jQuery?**  
  
**Answer 30:** The "jQuery.each()" function is a general function that will loop through a collection (object type or array type). Array-like objects with a length property are iterated by their index position and value. Other objects are iterated on their key-value properties. The "jQuery.each()" function however works differently from the $(selector).each() function that works on the DOM element using the selector. But both iterate over a jQuery object.  
 **For example:** If we pass an array to the each() function, it iterates over items in the array and accesses both the current item and its index position.  
  
For more details follow link:

* [$.each() function in jQuery](http://www.c-sharpcorner.com/UploadFile/99bb20/each-function-in-jquery/)

**Question 31: What is the difference between Map and Grep function in jQuery?**  
  
**Answer 31:** In $.map() you need to loop over each element in an array and modify its value whilst the $. Grep() method returns the filtered array using some filter condition from an existing array.  
  
The basic structure of Map() is: **$.map ( array, callback(elementOfArray, indexInArray) )**

* [Diifference Between Map and Grep Function in jQuery](http://www.c-sharpcorner.com/UploadFile/97fc7a/diiffernece-between-map-and-grep-function-in-jquery/)

**Question 32: What are jQuery plugins?**  
  
**Answer 32:**Plugins are a piece of code. In jQuery plugins it is a code written in a standard JavaScript file. These JavaScript files provide useful jQuery methods that can be used along with jQuery library methods.   
  
Any method you use in plugins must have a semicolon (;) at the end. The method must return an object (jQuery), unless explicitly noted otherwise. Use each to iterate over the current set of matched elements. It produces clean and compatible code that way. Prefix the filename with jQuery, follow that with the name of the plugin and conclude with .js. (For example, jquery.plug-in.js). Always attach the plugin to jQuery directly instead of $, so users can use a custom alias via the noConflict() method (via the jQuery Team).

* [jQuery Plugins](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-plugins/)

**Question 33: Define jQuery .animate() method?**  
  
**Answer 33:** In jQuery the animate() method is very useful. By using this method we can change the size of elements. In this example we will create a div element which contains an Image; when we move the mouse over the image, the image size will change. First of all you add an image to the application, add a new form to the application and add the following HTML code to the aspx page.

1. <div style="height: 100px; width: 100px; position: relative">
2. <img src="animate.gif" id="img" />
3. </div>

Now add the following code in the head section.

1. <script type="text/javascript">
2. $(document).ready(function() {
4. $("div").mouseover(function() //mouseover function will execute when mouse pointer will reach on <div>element
5. {
6. $("img").animate({
7. height: 300
8. }, "slow"); //image height will change by using animate method
9. $("img").animate({
10. width: 300
11. }, "slow");
12. $("img").animate({
13. height: 100
14. }, "slow");
15. $("img").animate({
16. width: 100
17. }, "slow");
18. });
19. });
20. </script>

In the above code we create a mouseover function.

1. $("img").animate({ height: 300 }, "slow"); //image height will change by using animate method
2. $("img").animate({ width: 300 }, "slow");
3. $("img").animate({ height: 100 }, "slow");
4. $("img").animate({ width: 100 }, "slow");

For more follow the link:

* [CSS() and Animate() Method in jQuery](http://www.c-sharpcorner.com/UploadFile/rohatash/css-and-animate-method-in-jquery/)

**Question 34:**What is the difference between bind() and live() method in jQuery ?  
  
**Answer 34:**The binding of event handlers are the most confusing part of jQuery for many developers working on jQuery projects. Many of them unsure of which is better to use. In this article we will see the main differences between Bind and Live methods in jQuery.  
  
**Bind() Method**  
The bind method will only bind event handlers for currently existing items. That means this works for the current element.  
  
**Example**

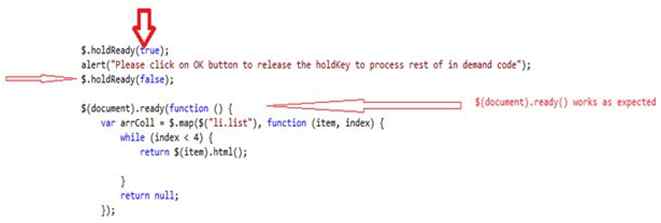
1. $(document).ready(function () {
2. $('P').bind('click', function () {
3. alert("Example of Bind Method");
4. e.preventDefault();
5. });
6. });

**Live() Method**  
The Live method can bind event handlers for currently existing items or future items.  
  
**Example**

1. $(document).ready(function() {
2. $('P').live('click', function() {
3. alert("Example of live method");
4. e.preventDefault();
5. });
6. $('body').append('<p>Adding Future items</p>');
8. });

Follow the link:

* [Difference Between Bind and Live Methods in jQuery](http://www.c-sharpcorner.com/UploadFile/rohatash/difference-between-bind-and-live-methods-in-jquery/)

**Question 35: What is jQuery.holdReady() function?**  
  
**Answer 35:** jQuery.holdReady() function is what we can hold or release the execution of jQuery's ready event. This method should be called before we run the ready event. To delay the ready event, we need to call jQuery.holdReady(true);  
  
When we want to release the ready event then we need to call jQuery.holdReady(false);  
  
This function is helpful when we want to load any jQuery plugin before the execution of the ready event or want to perform certain events/functions before document.ready() loads. For example, some information.  
  
**For example**  
  


* [jQuery Interview Questions and Answers With Practices: Part 2](http://www.c-sharpcorner.com/UploadFile/97fc7a/jquery-interview-question-and-answer-with-practices-part-2/)  
    
  or
* [HoldReady Functions in jQuery 1.9.1](http://www.c-sharpcorner.com/UploadFile/97fc7a/holdready-functions-in-jquery-1-9-1/)

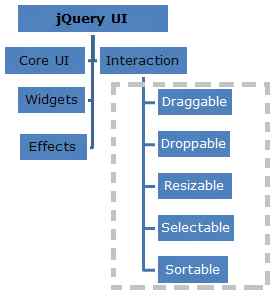
**Question 36: What is resize() function in jQuery?**  
 **Answer 36:** This method in jQuery is used for changing of the size of the element. You can use by .resize() function. For more visit the following link:

* [Changing Size of The Text Using jQuery](http://www.c-sharpcorner.com/UploadFile/sapnabeniwal/changing-size-of-the-text-using-jquery/)

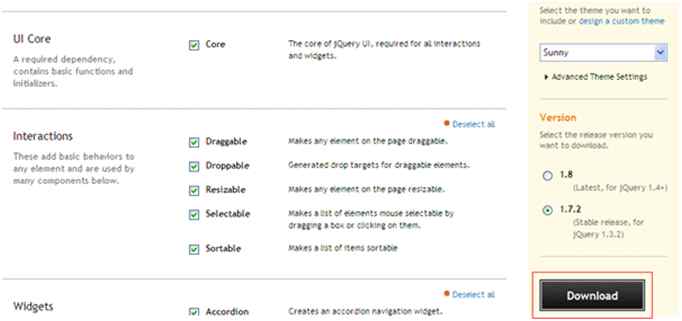
**Question 37: Define Add or Remove class in jQuery?**  
  
**Answer 37:***addclass* will be used for adding a new CSS class after replacing the old class and *removeClass* will work for removing the selected class.

1. $(document).ready(function() {
2. $('.button').click(function() {
3. **if** (**this**.id == "add") {
4. $('#animTarget').addClass("myClass", "fast")
5. } **else** **if** (**this**.id == "toggle") {
6. $('#animTarget').toggleClass("myClass", 1000, "easeOutSine")
7. } **else** **if** (**this**.id == "switch") {
8. $("#animTarget").switchClass("myClass", "switchclass", "fast")
9. } **else** {
10. $('#animTarget').removeClass("myClass", "fast")
11. }
12. })
13. });

* [How To Work With jQuery UI Effects](http://www.c-sharpcorner.com/UploadFile/22d000/how-to-work-with-jquery-ui-effects/)

**Question 38: What is the usage of Draggable, Droppable, Resizable, Selectable in jQuery UI?**  
 **Answer 38:**There are only 5 plugins available in the interaction section; that is Draggable, Droppable, Resizable, Selectable and Sortable. Interaction Plugins handles complex behaviors such as drag and drop, resizing, selection and sorting.  
  
Graphical representation of jQuery UI subordinates:  
  
  
  
**Draggable:** It enables draggable functionality on any DOM element. Move the draggable object by clicking on it with the mouse and dragging it anywhere within the viewport.  
  
**Droppable:** It enables any DOM element to be droppable, a target for draggable elements.  
  
**Resizable:** It enables any DOM element to be resizable. With the cursor, grab the right or bottom border and drag to the desired width or height.  
  
**Selectable:**It enables a DOM element (or group of elements) to be selectable. Draw a box with your cursor to select items. Hold down the Ctrl key to make multiple non-adjacent selections.  
  
**Sortable:** It enables a group of DOM elements to be sortable. Click on and drag an element to a new spot within the list, and the other items will adjust to fit. By default, sortable items share draggable properties.

* [jQuery UI: Interaction Plugins](http://www.c-sharpcorner.com/UploadFile/abhikumarvatsa/jquery-ui-interaction-plugins/)

**Question 39: What is the history of jQuery UI and how to use it?**  
  
**Answer 39:**jQuery UI is really very easy to learn and it provides abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets, built on top of the jQuery JavaScript Library which you can use to build highly interactive web applications. The whole jQuery UI is categorized into four groups; they are core, interactions, widgets and effects.  
  
  
  
The components of jQuery UI are:

* **Core:** It's a perquisite for other widgets and effects to work properly.
* **Interactions:** It allows us to add behavior like Draggable, Droppable, Sortable, etc on the UI elements.
* **Widgets:**It provides UI controls like tabs, dialog, slider, etc.
* **Effects:** It provides ready to use effects like clip, bounce, explode, etc.  
    
  [Introduction to jQuery UI](http://www.c-sharpcorner.com/uploadfile/satisharveti/introduction-to-jquery-ui/)

**Question 40: What $(document).ready(function()) is and when to use it?**  
  
**Answer 40:** $(document).ready(function()) is a jQuery event that fires as soon as the DOM is fully loaded and ready to be manipulated by script. This is the earliest point in the page load process where the script can safely access elements in the page's HTML DOM. This event is fired before all the images and CSS are fully loaded.

* [Basics of jQuery: Part 1](http://www.c-sharpcorner.com/UploadFile/219d4d/basics-of-jquery-part-1/)

**Question 41: Define jQuery UI Autocomplete ?**  
  
**Answer 41:** Autocomplete is one of the best widgets ON websites and is used in nearly all websites. jQuery has a powerful widget, autocomplete, and in this article I will try to explain how to use jQuery Autocomplete in websites. All the way and all other features of autocomplete. We can make autocomplete, using AJAX, to call to build a list (server-side) and then bind that list into a text box using JavaScript. However there are other alternatives to make autocomplete rather then this in an easy way. The most robust and efficient tool of autocomplete is **jQuery-ui autocomplete** and this tool is free and there is no need to license it.  
  
**Points to remember**

* The Autocomplete widget requires some functional CSS, otherwise it won't work. If you build a custom theme, use the widget's specific CSS as a starting point.
* This widget manipulates its element's value programmatically, therefore a native change event may not be fired when the element's value changes.

[jQuery UI Autocomplete](http://www.c-sharpcorner.com/UploadFile/736ca4/autocomplete-in-jquery/)  
  
**Question 42: What is jQuery UI Sortable and how to use it?**  
  
**Answer 42:** The jQuery UI is a library provided by jQuery for a better user interface. Using sortable we can reorder the DOM elements in the defined area. Users have to click on the item and drag that item to a new place. The other items will be automatically arranged. Accordingly, use the following procedure to enable sortable elements:

1. Include the jQuery js file.
   1. <script src="http://ajax.aspnetcdn.com/ajax/jquery/jquery-1.10.0.min.js"></script>
2. Include the jQuery UI js file.
   1. <script src="http://ajax.aspnetcdn.com/ajax/jquery.ui/1.10.3/jquery-ui.min.js" type="text/javascript"></script>

[jQuery UI Sortable](http://www.c-sharpcorner.com/UploadFile/f18cf0/jquery-ui-sortable/)  
  
**Question 43: Why to use jQuery $ sign ?**  
  
**Answer 43:**The basic operation in jQuery is selecting an element in DOM. This is done with the help of $() construct with a string parameter containing any CSS selector expression. $() will return zero or more DOM elements on which we can apply an effect or a style.

* [Selectors in jQuery](http://www.c-sharpcorner.com/uploadfile/satisharveti/selectors-in-jquery/)

**Question 44: What is slice() method in jQuery ?**  
  
**Answer 44:**This method selects a subset of the matched elements by giving a range of indices. In other words, it gives the set of DOM elements on the basis of it's parameter (start, end).  
  
**Syntax:**.slice( start, end[Optional] )   
  
**Start:**This is the first and mandatory parameter of the slice method. This specifies from where to start to select the elements.   
  
**End:**This is an optional parameter. It specifies the range of the selection. This indicates where to stop the selection of elements, excluding end element.   
  
**Note:**The Negative Indices started from -1. This last element is denoted by index -1 and so on.

* [.slice() Method in jQuery](http://www.c-sharpcorner.com/UploadFile/99bb20/slice-method-in-jquery/)

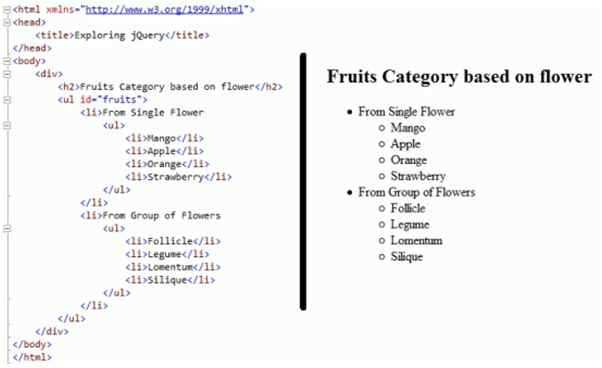
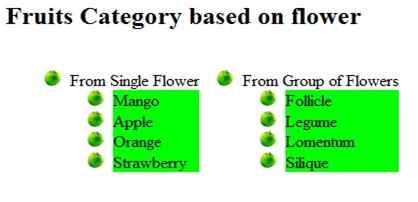
**Question 45: What is jQuery Effects – Fading ?**  
  
**Answer 45:**The fade methods define visibility of content in UI, in other words how the web page is hidden/shown. To use the fade methods of jQuery I need a jQuery library in my project, so I directly used the [Google AJAX Libraries](http://code.google.com/apis/ajaxlibs/) content delivery network to serve jQuery from Google. Doing so has several advantages over hosting jQuery on our server, decreased latency, increased parallelism, and better caching. We add the script to our project.

1. <script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.js"></script>

Our UI design is ready so now implement the fade methods. Here are the four types of jQuery fade methods:

1. fadeIn()
2. fadeOut()
3. fadeToggle()
4. fadeTo()

[jQuery Effects Using Fade Methods](http://www.c-sharpcorner.com/UploadFile/3d39b4/jquery-effects-using-fade-methods/)

**Question 46: How to work with jQuery css() method ?**  
  
**Answer 46:**The following code is very simple, no styles are applied. Now, we want to achieve the following style by using jQuery and some CSS.  
  
   
  


* [JQuery and CSS Selectors: Part 2](http://www.c-sharpcorner.com/UploadFile/abhikumarvatsa/jquery-and-css-selectors-part-2/)

**Question 47: What is**[**queue()**](http://www.w3schools.com/jquery/eff_queue.asp)**in Jquery ? Use of**[**queue()**](http://www.w3schools.com/jquery/eff_queue.asp)**in jquery ?**  
  
**Answer 47:**Delay comes under the custom effect category in jQuery. Its sole use is to delay the execution of subsequent items in the execution queue.   
  
*delay( duration [, queueName ] )*  
  
*queueName* is a name of the queue in which the delay time is to be inserted. By default it is a "fx" queue. A "fx" queue is also known as an effects queue.

* [Timers in jQuery: Delay Method](http://www.c-sharpcorner.com/UploadFile/4aac15/timers-in-jquery-delay-method)

**Question 48: How jQuery selectors are executed?**  
  
**Answer 48:**A selector starts with $(). In the parentheses may be an element, a class or an ID. For example:

1. *<div class=”leftBorder”> C# Corner</div>*
2. *<div ID=”leftPanel”>C# Corner</div>*

For the preceding code, jQuery syntax (for selectors ) will be:

1. $(“div”).action
2. $(“.leftBorder”).action
3. $(“#leftPanel”).action

So here we used the following three things: HTML tag name, class name and ID name. There are jQuery selectors. The factory function $() is a synonym of the jQuery() function.

* [jQuery Selectors Basics](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-selectors-basics/)

**Question 49: What are the advantages of Ajax?**  
  
**Answer 49:**Ajax stands for Asynchronous JavaScript and XML; in other words Ajax is the combination of various technologies such as JavaScript, CSS, XHTML, and DOM, etc.  
  
AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the entire page.  
  
We can also define Ajax is a combination of client side technologies that provides asynchronous communication between the user interface and the web server so that partial page rendering occurs instead of complete page post back.  
 **Advantages of AJAX based application**

* Improved application performance by reducing the amount of data downloaded from the server.
* Rich, responsive and Slick UI with no page flickers.
* Eliminates frequent page refresh which usually happens in a typical request/response model (Everything is updated on fly).
* Easy to implement as there are variety of AJAX implementations available around.
* AJAX mechanism works behind the scene nothing much required from user perspective.
* Works with all the web browsers.
* Avoids the round trips to the server.
* Rendering of webpage faster.
* Decreases the consumption of server resources.
* Response time of application is faster.
* Rendering of data is dynamic.

[Introduction to Ajax and Ajax Control Toolkit](http://www.c-sharpcorner.com/UploadFile/0c1bb2/introduction-to-ajax-and-ajax-control-toolkit-part-1/)  
  
**Question 50: How can you use array with jQuery?**  
  
**Answer 50:** Arrays are zero indexed, ordered lists of values. They are really handy for storing a set of values of the same data type.  
  
*var names = [“Name1”,”Name2”] //Recommended*  
  
Both of the preceding approaches are kind of static declarations. Now let's do some dynamic programming with Arrays.

1. var namearray = [];
2. namearray.push(“Name1”) //Index 0
3. namearray.push(“Name2”) //Index 1
4. namearray.push(“Name3”) //Index 2

Here, .push() is a jQuery function used in conjunction with Arrays that adds an element at the end of the array. Items can be inserted by specifying the index as well, as follows:

1. namearray[0] = “Name1”;
2. namearray[1] = “Name2”;
3. namearray[2] = “Name3”;

Now let’s print the values of the array:

1. Console.log(namearray);

The statement above will produce the output as [ *"Name1", "Name2",”Name3”*].  
  
We can see that we just printed the array object but not the individual values, so to extract individual values the following statement can be executed:

1. Console.log(namearray[0]) //Name1;
2. Console.log(namearray[1]) //Name2;

How to print an array of values using a for loop in jQuery:

1. var myArray = ["Name1", "Name2", "Name3"];
2. **for** (var i = 0; i < myArray.length; i = i + 1) {
3. console.log(myArray[i]);
4. }

How to print an array of values using $.each() in jQuery:

1. $.each(myArray, function (index, value) {
2. console.log(index + ": " + value);
3. });

|  |
| --- |
| What is jQuery Selectors? Give some examples.  1. jQuery Selectors are used to select one or a group of HTML elements from your web page. 2. jQuery supports all the CSS selectors as well as many additional custom selectors. 3. jQuery selectors always start with dollar sign and parentheses: $(). 4. There are three building blocks to select the elements in a web document.  1) Select elements by tag name **Example :** $(div) It will select all the div elements in the document.  2) Select elements by ID **Example :** $(“#xyzid”) It will select single element that has an ID of xyzid.  3) Select elements by class **Example :** $(“.xyzclass”) It will select all the elements having class xyzclass. |
| How can we give face effect in jQuery?  1. In jQuery we have three methods to give the fade effect to elements: fadeIn, fadeOut and fadeTo. 2. This methods change the opacity of element with animation.  **Syntax:**  $(selector).fadeIn(speed,callback) $(selector).fadeOut(speed,callback) $(selector).fadeTo(speed,opacity,callback)  1. “speed” can be one of following values : “slow”, “fast”, “normal” or milliseconds. 2. “opacity” specify the value that allows the fading to given opacity. 3. “callback” is the function which we want to run once the fading effect is complete.  **For example**  $("clickme").click(function() { $("mydiv").fadeTo("slow",0.50); });  $("clickme").click(function() { $("mydiv").fadeOut(3000); }); |
| Explain the animate function.  -The animate function is used to apply the custom animation effect to elements.  **Syntax:**  $(selector).animate({params}, [duration], [easing], [callback])  1. **“param”** defines the CSS properties on which you want to apply the animation. 2. **“duration”** specify how long the animation will run. It can be one of following values : “slow”, “fast”, “normal” or milliseconds. 3. **“easing”** is the string which specify the function for the transition. 4. **“callback”** is the function which we want to run once the animation effect is complete.  **For example**  <div id="clickToAnimate"> Click Me </div> <div id="mydiv" style=”width:200px; height:300px; position: relative; right: 20px;"> </div>  Following is the jQuery to animate opacity, left offset, and height of the mydiv element  $('# clickToAnimate’).click(function()  {      $('#book').animate({opacity: 0.30,left: '+=20',height: 'toggle'}, 3000, function()  {      // run after the animation complete. }); }); |
| What is .siblings() method in jQuery?  1. When we want to fetch siblings of every elements in the set of matched elements then we can use siblings() method. 2. We filter the elements fetched by an optional selector. 3. Syntax : .siblings([selector]) 4. “selector” is the selector expression which specify the matched elements.  **For example**  <ul> <li> item 1 </li> <li id=”second\_item”> item 2 </li> <li class=”myitem”> item 3 </li> <li class=”myitem”> item 4 </li> </ul>  Now we want to find the siblings of the element of id “second\_item” and change the text color to Blue :  $(‘li.second\_item’).siblings().css(‘color’,’blue’);  If we want specific sibling elements for example the elements having class “myitem” then we can pass a optional selector :  $(‘li.second\_item’).siblings(‘.myitem’).css(‘color’,’blue’); |
| Explain width() vs css(‘width’).  1. In jQuery, there are two way to change the width of an element. 2. One way is using .css(‘width’) and other way is using .width().  **For example**  $(‘#mydiv’).css(‘width’,’300px’); $(‘#mydiv’).width(100);  1. The difference in .css(‘width’) and .width() is the data type of value we specify or return from the both functions. 2. In .css(‘width’) we have to add “px” in the width value while in .width() we don’t have to add. 3. When you want to get the width of “mydiv” element then .css(‘width’) will return ‘300px’ while .width() will return only integer value 300. |
| What is the use of jQuery.data()?  1. **jQuery.data()** is used to set/return arbitrary data to/from an element. 2. **Syntax:** jQuery.data(element, key, value) 3. **“element”** is the DOM element to which the data is associated. 4. **“key”** is an arbitrary name of the piece of data. 5. **“value”** is value of the specified key. 6. Suppose we want to set the data for a span element:  jQuery.data(span, “item”, { val1: 10, val2: "myitem" });  If we want to retrieve the data related to div element and set it to label’s data:  $("label:val1").text(jQuery.data(div, "item").val1); $("label:val2").text(jQuery.data(div, "item").val2); |
| Explain bind() vs live() vs delegate() methods.  - The bind() method will not attach events to those elements which are added after DOM is loaded while live() and delegate() methods attach events to the future elements also. - The difference between live() and delegate() methods is live() function will not work in chaining. It will work only on an selector or an element while delegate() method can work in chaining.  **For example**  $(document).ready(function() { $("#myTable").find("tr").live("click",function() {      alert($(this).text()); }); });  Above code will not work using live() method. But using delegate() method we can accomplish this.  $(document).ready(function() { $("#dvContainer")children("table").delegate("tr","click",function() {      alert($(this).text()); }); }); |
| Explain the each() function.  - The each() function specify the function to be called for every matched element.  **Syntax :**  $(selector).each(function (index, element))  1. **“index”** is the index position of the selector. 2. **“selector”** specifies the current selector where we can use “this” selector also. 3. In the case when we need to stop the each loop early then we can use “return false;”  **For example**  $("#clickme").click(function() { $("li").each(function() {      document.write($(this).text()) }); });  This will write the text for each “li” element. |
| Explain slideToggle() effect.  - slideToggle() effect is used to give animated sliding effect to an element.  **Syntax:**  slideToggle([ duration] [, easing] [, callback])  1. **“duration”** is the number specifying how long the animation will run. 2. **“easing”** is the string which specify the function for the transition. 3. **“callback”** is the function which we want to run once the animation is complete. 4. If the element is visible then this effect will slide the element up side and make it completely hidden. If the element is hidden then slideToggle() effect will slide it down side and make it visible. 5. We can specify the toggle speed with this effect.  **For example**  $("#clickme").click(function() { $("#mydiv").slideToggle(“slow”, function() {      //run after the animation is complete. }); }); |
| What is difference between $(this) and ‘this’ in jQuery?  Refer the following example:  $(document).ready(function() { $(‘#clickme’).click(function() {      alert($(this).text());      alert(this.innerText); }); });  - this and $(this) references the same element but the difference is that “this” is used in traditional way but when “this” is used with $() then it becomes a jQuery object on which we can use the functions of jQuery. - In the example given, when only “this” keyword is used then we can use the jQuery text() function to get the text of the element, because it is not jQuery object. Once the “this” keyword is wrapped in $() then we can use the jQuery function text() to get the text of the element. |
| What is the use of param() method.  1. The **param()** method is used to represent an array or an object in serialize manner. 2. While making an ajax request we can use these serialize values in the query strings of URL. 3. **Syntax:** $.param(object | array, boolValue) 4. **“object | array”** specifies an array or an object to be serialized. 5. **“boolValue”** specifies whether to use the traditional style of param serialization or not.  **For example:**  personObj=new Object(); empObject.name="Arpit"; empObject.age="24"; empObject.dept=”IT”; $("#clickme").click(function() {      $("span").text($.param(empObject)); });  It will set the text of span to “name=Arpit&age=24&dep=IT” |
| What is jQuery.holdReady() function?  - By using jQuery.holdReady() function we can hold or release the execution of jQuery’s ready event. - This method should be call before we run ready event. - To delay the ready event, we have to call  jQuery.holdReady(true);  - When we want to release the ready event then we have to call  jQuery.holdReady(false);  - This function is helpful when we want to load any jQuery plugins before the execution of ready event.  **For example**  $.holdReady(true); $.getScript("xyzplugin.js", function()  {      $.holdReady(false); }); |
| Explain .empty() vs .remove() vs .detach().  - **.empty()** method is used to remove all the child elements from matched elements. - **.remove()** method is used to remove all the matched element. This method will remove all the jQuery data associated with the matched element. - **.detach()** method is same as .remove() method except that the .detach() method doesn’t remove jQuery data associated with the matched elements. - **.remove()** is faster than .empty() or .detach() method.  **Syntax:**  $(selector).empty(); $(selector).remove(); $(selector).detach(); |
| How to read, write and delete cookies in jQuery?  - To deal with cookies in jQuery we have to use the Dough cookie plugin. - Dough is easy to use and having powerful features.  **1. Create cookie**  $.dough("cookie\_name", "cookie\_value");  **2. Read Cookie**  $.dough("cookie\_name");  **3. Delete cookie**  $.dough("cookie\_name", "remove"); |
| Is window.onload is different from document.ready()?  - The window.onload() is Java script function and document.ready() is jQuery event which are called when page is loaded. - The difference is that document.ready() is called after the DOM is loaded without waiting for all the contents to get loaded. While window.onload() function waits until the contents of page is loaded. - Suppose there is very large image on a page, at that time window.onload() will wait until that image is loaded totally. - So while using the window.onlaod() function the execution will be slow, but the document.ready() will not wait until the image is loaded. |
| What is Chaining in jQuery?  - Chaining is very powerful feature of jQuery. - Chaining means specifying multiple function and/or selectors to an element.  **Examine the below example**  $(document).ready(function() {      $('#mydiv').css('color', 'blue');      $('#mydiv').addClass('myclass');      $('#mydiv').fadeIn('fast'); }  By using chaining we can write above code as follows  $(document).ready(function() {      $('#mydiv').css('color', 'blue').addClass('myclass').fadeIn('fast'); });  - Advantage of chaining is that it makes your code simple and simple to manage. - The execution becomes faster because the code search for the element only once. |
| What is difference between sorting string array and sorting numerical array in jQuery?  The sort method is used to sort any array elements. It sorts the string elements alphabetically.  **For example**  $(document).ready(function() {      var mylist = [ “Apple”,”Orange”,”Banana”];      mylist = mylist.sort();      $(“#mydiv”).html(list.join(“”)); });  It will give following output: Apple Banana Orange  Now we declare a numerical array and use sort() method to sort its elements.  $(document).ready(function() {      var mylist = [ “20”,”3””100”,”50”];      mylist = mylist.sort();      $(“#mydiv”).html(list.join(“”)); });  It will give following output: 100 20 3 50 |
| What is difference between prop and attr?  1. In jQuery both prop() and attr() function is used to set/get the value of specified property of an element. 2. The difference in both the function is that attr() returns the default value of the property while the prop() returns the current value of the property.  **For example**  <input value="My Value" type="text"/> $('input').prop('value', 'Changed Value');  - .attr('value') will return 'My Value' - .prop('value') will return 'Changed Value' |
| How to always reference latest version of jQuery?  When you reference the jQuery on your web page, you have to specify the version number also.  <script type=”text/javascript” src=”http://ajax.googleapis.com/ajax/libs/jquery/1.5.1/jquery.min.js”> </script>  Above code will always load the 1.5.1 version of jQuery. If you reference the latest jQuery then you don’t need to change the code every time the new version of jQuery is released.  To achieve this you have to use following code  <script type=”text/javascript” src=”http://code.jquery.com/jquery-latest.min.js”> </script>  This code will always reference the latest version of jQuery in your page. |
| What is resize() function in jQuery?  The resize() function is called whenever the browser size is changed. This event can be only used with $(window).  **Syntax:**  .resize([event\_data], handler(event\_object))  - The “event\_data” is the data to be sent to the handler. - The “handler(event\_object)” is a function to be called each time when the window is resized.  **For example**  $(window).resize(function()  { $('#message).text('window is resized to ' + $(window).width() + ‘x’ + $(window).height()); }); |
| How can jquery library be added to pages? Write a basic jquery code?  The jquery library is a collection of all the jquery methods. It is stored in the form of a single java script file. Th format of adding a jquery file to an html page is:  <head> <script type=”text/javascript” src”jquery.js”></script> </head>  An example of a javascript that will hide all the <p> elements in the page.  <html> <head> <script type="text/javascript" src="jquery.js"></script> <script type="text/javascript"> $(document).ready(function() { $("button").click(function() {      $("p").hide(); }); }); </script> </head> <body> <h2>This is a heading</h2> <p>This is a paragraph.</p> <p>This is another paragraph.</p> <button>Click me</button> </body> </html> |
| What are the types of selectors that are used in jquery? Give examples.  - Jquery enables the user to select specifically the element that is to be effected. jquery allows the user to select in the following ways: - **jquery element selectors :** With the use of css selectors the users can select the elements of an html document.  **For ex.**  $("p") will select all the <p> elements. $("p.intro") will select all <p> elements with class="intro" defined in them. $("p#demo") this will select all <p> elements with id="demo".  - jquery attribute selectors: the xpath expressions are used by jquery to select elements of an html document with defined attributes. **For ex.**  $("[href]") is used to select all elements which have an href attribute. $("[href$='.jpg']") can select all elements with an href attribute which will end with ".jpg". |
| How can images be made to appear scrolling one over another?  - Jquery provides the user with the ability to change the attributes of a property dynamically. The jquery slide method can be used to change the height of elements gradually. This can be used to give the scroll effect of an image over image. The jquery comprises of the following slide methods:  1. $(selector).slideDown(speed,callback) 2. $(selector).slideUp(speed,callback) 3. $(selector).slideToggle(speed,callback)  - The speed parameter is used to effect the speed of change of the jquery. The parameters for it can be slow, fast , normal and time in milliseconds. The parameter of callback is used to refer to the name of the function to be executed once the completion of function occurs. |
| What are the various ajax functions?  Ajax allows the user to exchange data with a server and update parts of a page without reloading the entire page. Some of the functions of ajax are as follows:  **1. $.ajax() :** This is considered to be the most low level and basic of functions. It is used to send requests . This function can be performed without a selector. **2. $.ajaxSetup() :** This function is used to define and set the options for various ajax calls. **For ex.**  $.ajaxSetup({ "type":"POST", "url":"ajax.php", "success":function(data) {      $("#bar")      .css("background","yellow")      .html(data); } });  **3. Shorthand ajax methods :** They comprise of simply the wrapper function that call $.ajax() with certain parameters already set. **4. $.getJSON() :** This is a special type of shorthand function which is used to accept the url to which the requests are sent. Also optional data and optional callback functions are possible in such functions. |
| What are the guidelines for an application to follow the principles of progressive enhancement.  Progressive enhancement is web development technique that would allow the application to be accessible to any computer with any Internet connection. For an application to work on the principles of progressive enhancement the following rules / guidelines must be met:  1. The basic content must be available to all browsers with any type of Internet connections. 2. The basic functionalities of an application must be working in all browsers. 3. The application will not override any of the user specified browser settings. 4. The css files are linked externally to control the look and feel of the page. 5. The javascripts should also be externally linked as it can enhance the user experience. |
| How can events be prevented to work after an ajax request?  **There are two ways to handle this issue:**  **1. Use of event delegation :** The event delegation technique works on principle by exploiting the event bubbling. It uses event bubbling to capture the events on elements which are present anywhere in the domain object model. In jquery the user can make use of the live and die methods for the events delegation which contains a subset of event types.  **For ex.** Handling even delegation, handling of clicks on any <a> element:  $('#mydiv').click(function(e) {      if( $(e.target).is('a') )      fn.call(e.target,e); }); $('#mydiv').load('my.html')  **2. Event rebinding usage :** When this method is used it requires the user to call the bind method and the added new elements.  **For ex.**  $('a').click(fn); $('#mydiv').load('my.html',function() {      $('#mydiv a').click(fn); }); |
| How can an element be checked if it contains a specific class?  The hasClass method defined can be used to check if an element actually contains the specified class.  **For ex :** usage of the hasClass:  $("div").click(function() { if ( $(this).hasClass("protected")) $(this) .animate({ left: -10 }) .animate({ left: 10 }) .animate({ left: -10 }) .animate({ left: 10 }) .animate({ left: 0 }); });  The is() method can also be used with a selector for a more advanced level of matching. **For ex.**  if ( $('#myDiv').is('.pretty.awesome') ) $('#myDiv').show();  This method can be used to test various other things, such as it could be used to detect if the specified element is hidden or not. |
| Why is the block display style used for animations?  In html only the block level elements can have custom heights and widths. So when a user defines an animation method for usage such as show, hide, slide up etc the display css property of the block being animated is set to display block style. On completion of the animation the display style of the block would be changed to its original value. This procedure does not work properly for inline elements and the following workarounds can be applied to it:  - If the user wants the element to remain inline and only want to animate it in and out he can use the fadein and fadeout animation instead of using the show method. - The user can also use a block level element with float to make the element appear inline with the rest of the content around it. |
| What are the approaches of extracting a query string with regular expressions?  There are two approaches of doing so:  **1. String based approach :** This is the simple way of extracting the information of a query string using the string-based replacement technique. This method makes use of the .replace() method to function. **For ex :**  var data = string.replace("http://localhost/view.php?", "");  The above code works fine for the string based approach but has some flexibility issues. It cannot deal effectively with domain name and file name changes.  **2. Regular expression approach :** They are a powerful pattern matching tool available in modern programming languages. For the extraction of a query string a pattern would have to be used which looks for a question mark in the string. Once it does it returns everything after it. The regular expression in JS are delimited using the forward slashes at the end of an expression. |
| How does jquery store data related to an element?  In plain java scripts the information about an element can be stored by adding a domain object model property to the element. This results in memory leak problems in certain browsers. In jquery the user does not has to worry about memory management issues.  **For ex :** Storing and retrieving data related to an element:  $('#myDiv').data('keyName', { foo : 'bar' }); $('#myDiv').data('keyName'); // { foo : 'bar' }  Jquery allows the user to store any kind of information on an element. The $.fn.data is used the most to store data related to an element.  **For ex :** Storing of relationship between elements with the use of $.fn.data :  $('#myList li').each(function()  {      var $li = $(this), $div = $li.find('div.content');      $li.data('contentDiv', $div); }); // later, the user is not required to look for the div again; // the data can simply be read from the list items data; var $firstLi = $('#myList li:first'); $firstLi.data('contentDiv').html('new content'); |
| Explain the common methods of sending a request to a server.  The two most common methods of sending a request to a server are :  **1. GET method :** The get method is mostly used for non destructive operations. These operations get data from the server and does not change the data on it. A good example of the application of the search query to a server. In most of the cases GET will send all of the data to be sent in the form of a query string. **2. POST method :** The POST method is primarily used for destructive operations. These operations can change the data on a server. A good example is a user saving an entry on a site will get the POST request. These requests are not cached by the browser. A query can be a part of a url but any data that is to be sent is done separately as post data. |
| Create a plugin that would add and remove a class on hover.  The plugin can be considered to be simply a new method that can be used by a user to extend the prototype object of a jquery. A plugin performs some actions on a collection of elements. Each method that comes with the jquery core can be considered to be a plugin.  The code for creating a plugin that would add and remove a class on hover would be as follows:  (function($) {    $.fn.hoverClass = function(c)  {      return this.hover(      function() { $(this).toggleClass(c); } ); }; })(jQuery);  // using the plugin $('li').hoverClass('hover'); |
| Explain the use of the .pushStack() method.  The pushStack() method works by accepting an array of DOM elements and pushes them into a stack. This is done so that call to methods like .end() and .andSelf are able to behave correctly. The jquery internally uses this method to keep track of all the previous collections of jquery while using a chain traversing method. Good examples of such methods could be .parents() and .filter().  **For ex :**  // select some divs $('div.container') // find some spans inside those divs and add a class to them .find('span').addClass('baby') // pop those spans off the "stack", // returning to the previous collection (div.container) .end() // add a class to the parent of each div.container .parent().addClass('daddy'); |
| How is the deferred method in jquery important in relation to animate method?  The .animate() method is used to create animations with other shorthands using it. The queue() method can be used to link together multiple animation methods to create an unique effect. These methods are effective when all the data is available locally and all the methods are executed on as single system only. In case the user wants to use the animation methods on a data that resides on the server and wants to handle at a single go the user can make used of the .deferred method.  **For ex :**  var my$ = $.sub(); my$.fn.animate = function( props, speed, easing, callback )  { var options = speed && typeof speed === "object" ? jQuery.extend({}, speed) :  {      complete: callback || !callback && easing ||      jQuery.isFunction( speed ) && speed,      duration: speed,      easing: callback && easing || easing &&      !jQuery.isFunction(easing) && easing }; var dfd = my$.Deferred(), complete = options.complete, count = this.length; options.complete = function()  {      complete && complete.call( this );      if ( !--count )  {      dfd.resolve(); } } }; |
| Explain some of the key concepts of good code organization patterns.  Some of the key concepts that should be followed while:  1. The codes should always be divided into specific units of functionality. They can be divided into modules , services etc. This concept is also known as encapsulation. 2. Use inheritance techniques to avoid rewriting a code again and again. 3. The jquery is dom centric in nature but not all applications are such in nature. All funtionalities should not have a dom representation. 5. All the units of functionality should be existing in a loosely coupled state. This implies that a module can exist independently. Also the communication between the units should be possible and should be handled using a messaging system for example custom events. |
| How can related code be encapsulated? Give example.  The object literal is one of the simplest ways that the user can encapsulate related code together. It helps by removing any anonymous functions from the users code. It can also be used to centralize configuration options. **For ex :** An object literal being implemented  var myFeature =  { myProperty : 'hello', myMethod : function()  {      console.log(myFeature.myProperty); }, init : function(settings)  {      m2yFeature.settings = settings; }, readSettings : function()  {      console.log(myFeature.settings); } }; myFeature.myProperty; // 'hello' myFeature.myMethod(); // logs 'hello' myFeature.init({ foo : 'bar' }); myFeature.readSettings(); // logs { foo : 'bar' } |
| Write a code for the implementation of a module pattern.  The object literal does not provide any privacy for the methods or properties. The module pattern allows the user to offer privacy to functions and variables. It can be used to set to expose limited API.  **For ex : The module pattern code:**  var feature =(function()  { var privateThing = 'secret', publicThing = 'not secret', changePrivateThing = function()  {      privateThing = 'super secret'; }, sayPrivateThing = function()  {      console.log(privateThing);      changePrivateThing(); }; return  {      publicThing : publicThing,      sayPrivateThing : sayPrivateThing } })(); feature.publicThing; // 'not secret' feature.sayPrivateThing(); |
| Write the code to define a RequireJs module with its dependencies.  The RequireJS is a dependency management tool that can be used by the user to manage script modules. It can be used to load scripts once a page has been loaded. This helps in evenly distributing the downloads.  **For ex :** RequireJS module with dependencies defined:  require.def("my/shirt", ["my/cart", "my/inventory"], function(cart, inventory)  { //return an object to define the "my/shirt" module. return  {      color: "blue",      size: "large"      addToCart: function()  { inventory.decrement(this); cart.add(this); } } } ); |
| Explain the use of the $.fn.bind and $.fn.trigger.  Both the $.fn.bind and $.fn.triggers are two important jquery methods. They are primarily used with custom events.  **1. $.fn.bind :** This method accepts an event type and an event handling function as an argument for itself. This method can also accept event-related data as a second argument. **2. $.fn.trigger :** This method can accept an event type as an argument. It can also take an array of values. **Ex :** Depicting the usage of $.fn.bind and $.fn.triggers using custom data in both the cases:  $(document).bind('myCustomEvent', { foo : 'bar' }, function(e, arg1, arg2)  {      console.log(e.data.foo); // 'bar'      console.log(arg1); // 'bim'      console.log(arg2); // 'baz' }); $(document).trigger('myCustomEvent', [ 'bim', 'baz' ]); |
| What is the Struts2 jQuery plugin and its advantages.  The struts2 jquery plugin is used to: 1. It provides the user with an easy integration of ajax and widgets. 2. It reduces the number of code to be written by a coder.  **For ex :**  <div id="result">Result Div</div> <s:url id="ajax" value="/ajax1.action"/> <sj:a id="ajaxlink" href="%{ajax}" indicator="indicator" targets="result" effect="highlight"> Run AJAX Action </sj:a> <img id="indicator" src="images/indicator.gif" alt="Loading..." style="display:none"/>  The advantages of using it are: 1. It reduces the number of code to be written dramatically. 2. It also supports ajax form validation. 3. The code becomes much easier to read and hence prevents errors from occurring and is easy to debug. |