jQuery interview questions and answers part 4

Search

Interview | Current affairs | Civil services | Banking | Engineering | MBA | GD | Placement papers | HR | CV | Cover letter | Aptitude | English | **Online test** | Ask question! | Prep. online

jQuery interview questions and answers part 4

Part 1 Part 2 Part 3 Part 4 Part 5

38. Explain the common methods of sending a request to a server.

The two most common methods of sending a request to a server are :
-GET
-POST
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.
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.

39. 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');
```

40. 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');
```

41. 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();
```

Interview questions

```
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();
}
};
```

42. Explain some of the key concepts of good code organization patterns.

Some of the key concepts that should be followed while
- 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.
- Use inheritance techniques to avoid rewriting a code again and again.
- The jquery is dom centric in nature but not all applications are such in nature. All funtionalities should not
have a dom representation.
- 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.

Write your comment - Share Knowledge and Experience