

Forum: [Hack Forums](#)

Thread: [Hide and Unhide table data with JS \(or CSS, or whatever\)](#)

bevc - May 5, 2020, 9:30 pm

I have a table that's getting too big for its boots. It has tons of columns, 15 of which are very repetitious data, kinda like this.....

```
data | more data | col 1 | col 2 | col 3 | col 4 | .....  
stuff | more stuff | data 1| data 2| data 3| data 4 | .....
```

so what I'd like to do is have the 'col 1, col 2' etc data available as a kind of like a menu dropdown, where you just click or hover over an icon on each row and it unhides/shows the data.

So I'd go from having 15 columns and n rows of this stuff, to a user hovering over an icon in a particular row, and seeing 2 columns with 15 (or fewer) rows. Like an accordion, but on top of the elements below, not shifting them down.

I'm not sure of the best way to accomplish this. I want it to look like a menu dropdown in a nav bar, but don't know if it's better to create a 'hidden' table for each row somehow. Is it better trying to do this in pure css, or javascript? I don't want to spend days on it, I was hoping somebody who knows better than me can give me some tips.

Thanks.

MavenGradle - May 6, 2020, 7:50 am

Styling should always be done using CSS. Simply apply a property to the columns, like for example the following two properties:

```
Code display: none;  
visibility: hidden;
```

Be sure to read their documentation in order to fully understand what you're doing:

<https://developer.mozilla.org/en-US/docs/Web/CSS/display>

<https://developer.mozilla.org/en-US/docs/Web/CSS/visibility>

You can change style attributes or even class attributes of an element in JavaScript.

Here's an example that changes the classList property of an element in order to toggle a certain class:

```
Code const divExample = document.getElementById("divExample");  
const btnChangeStyle = document.getElementById("btnChangeStyle");
```

```
btnChangeStyle.addEventListener("click", () => {  
  divExample.classList.toggle("hidden");  
});https://jsfiddle.net/vc765w3q/  
https://developer.mozilla.org/en-US/docs/Web/API/Element/classList
```

Or by using an inline style:

```
Code const divExample = document.getElementById("divExample");  
const btnChangeStyle = document.getElementById("btnChangeStyle");
```

```
btnChangeStyle.addEventListener("click", () => {  
  divExample.style.display = divExample.style.display === 'none' ? " : 'none';  
});
```

<https://jsfiddle.net/vc765w3g/1/>
<https://developer.mozilla.org/en-US/docs/Web/API/ElementCSSInlineStyle/style>

In regards to the event you're probably looking for in JavaScript: [mouseover](#)
[mouseout](#)

Hope this helps. Happy learning!

Joeykh - May 7, 2020, 2:10 am

Hey there!

As you have stated that you could use CSS "or whatever". I suggest you use jQuery's implementation for this. It's pretty straight forward.

https://www.w3schools.com/jquery/jquery_hide_show.asp

```
Code $("#hide").click(function(){  
  $("p").hide();  
});
```

```
$("#show").click(function(){  
  $("p").show();  
});
```

MavenGradle - May 7, 2020, 4:40 am

(May 7, 2020 02:10 AM)Joeykh Wrote: [\[CITE\]](#) Hey there!

As you have stated that you could use CSS "or whatever". I suggest you use jQuery's implementation for this. It's pretty straight forward.

https://www.w3schools.com/jquery/jquery_hide_show.asp

```
Code $("#hide").click(function(){  
  $("p").hide();  
});
```

```
$("#show").click(function(){  
  $("p").show();  
});
```

I suggest that you don't. JQuery is obsolete and should no longer be considered for modern website development. Vanilla JavaScript has, over time, evolved and has taken care of the features that JQuery "added" and might I say that it has even surpassed it in terms of features. Not to mention that Vanilla JS performs better as well, with the added bonus of not having to specifically load it into your website, which reduces the time it takes to load your website in general (even if it's just a couple of milliseconds). Bloat, that's what JQuery is in this day and age. The only reason to learn JQuery today would be to maintain legacy websites and if that's your cup of tea, by all means go ahead. Any modern JavaScript developer that has even the slightest clue of what they're doing and what JQuery is would tell you the exact same thing.

10 years ago, when JavaScript was missing plenty of features, it would have been a viable library to utilize. Welcome to 2020 though, the year in which you can confidently say that JQuery should be erased the internet. Actually, it could have been removed way sooner than 2020 but people are so slow to pick up on their learning game and tend to stick around with their old and trusted tools, even if they're completely unnecessary. Not to mention the fact that a lot of companies don't tend to update their websites, because of financial reasons. Again, something that is out of the control of the developers, but like I said, JQuery is fine for legacy maintenance only.

Additionally, w3schools really isn't that great of a resource to learn from, seeing how they barely update their data and essentially teach you old tricks that no longer apply to the modern landscape.

Just look at this example:

https://www.w3schools.com/js/js_variables.asp

Not a single mention of the different ways you can declare a variable such as through the usage of the let and const keywords and which one you should use depending on the scenario you're working on. Only var..

I quote from that very same page:

Quote:Creating a variable in JavaScript is called "declaring" a variable.

You declare a JavaScript variable with the var keyword:

This is completely outdated bullshit and way too little information to actually learn something. As such, my advice would be to use better sources and to keep track of what Vanilla JS can do. Even their [page on best practices](#) in regards to JavaScript is filled with things you actually shouldn't do anymore, like for example not using let or const. It's really just a sad website now that I think of it.

bevc - May 7, 2020, 5:18 am

(May 6, 2020 07:50 AM)MavenGradle Wrote: [\[CITE\]](#) Styling should always be done using CSS. Simply apply a property to the columns, like for example the following two properties:

Code display: none;
visibility: hidden;

Thanks, I'm kinda struggling with this because I'm not sure which way to go - the information currently exists in my table, but the table is too large. But the solution I'm thinking of implementing involves creating a 'new table' for each row in my table, that will be 'unhidden' when a user hovers over a row, and there can be 'n' number of rows.

I don't know how to create a sort of 'floating hidden table' that will appear on mouse hover, and contain the right data for the nth row that you're hovering over. (Am I making any sense??? Not sure that I am). So it's not just un hiding the element, but how do I even create the element.

This is one of those problems that I run into that seems to me like it should be easy - I just want to create something that apes the look of a 'menu dropdown', and it's information that I already have displayed in the table for each row (but which is taking up too much room currently), but instead of a nav bar, it's from any given row in a table.

Aussie - May 7, 2020, 7:58 am

(May 7, 2020 04:40 AM)MavenGradle Wrote: [\[CITE\]](#) -snipity snip-

Cheers again Mvn, very detailed post. I do agree, the shit they taught at my Comp Sci degree first year, was JQuery not modern frameworks like React or Angular or even ES6 JavaScript.

In addition, W3schools does have the EC6 way of creating variables with let (though it's in a separate page called EC6/ECMAScript 2015).