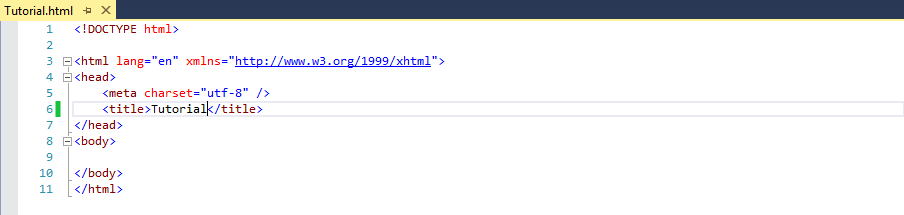
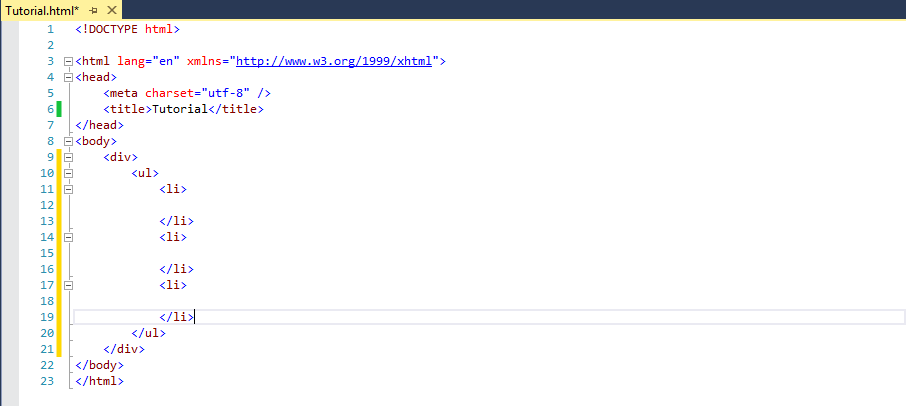
In this tutorial, we are going to learn how to build a simple working accordion menu example from scratch, using HTML and jQuery. We first are going to create the HTML page in which we will have a simple menu and then we will provide the accordion functionality with the use of certain jQuery selectors and functions.

**Building the HTML**

1. Create a new HTML document with your favorite text editor. In this tutorial I will be using Visual Studio 2013, but you are welcome to use any and should have no problem following the steps.



1. Inside the body of the HTML, let’s create a div element that will work as a container for our menu. In it, we will also create an unordered list with some list items that will represent the sections of our menu.



1. In each list item, we will create an h3 element with the name of the section and another unordered list with list items for each link in our menu. For the purposes of this tutorial, links will not point anywhere; we will use the # placeholder.

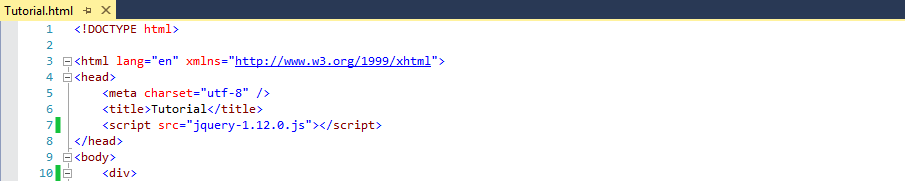


1. Save your work and run in a web browser. Compare my result with yours. If there are discrepancies, go over the steps again and fix any issues before you continue.



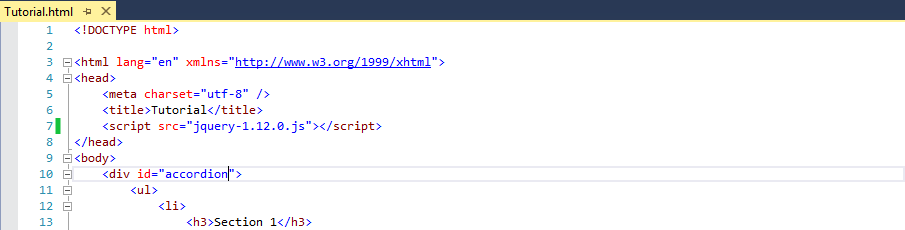
**Working with jQuery**

1. We first need to include a script tag inside the head element of our HTML file, so that it refers to an external jQuery .js file. In this tutorial, I am using version 1.12.0, but version 2.2.1 should work just fine if you choose to use it.

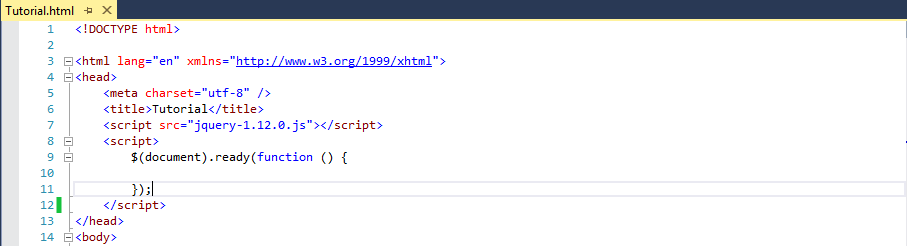


Remember to set the src attribute of the script tag to an absolute or relative path accordingly. In this tutorial, I am using an absolute path, as the jQuery .js file is setting in the same folder as our HTML document.

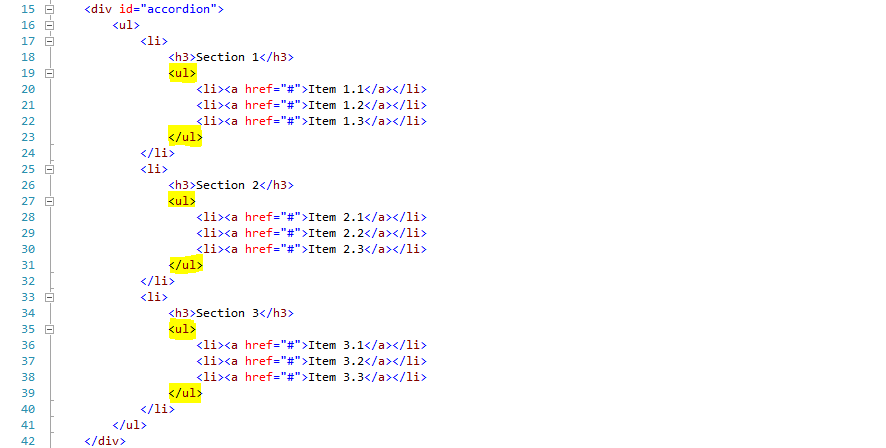
1. Also in the head element of our HTML file, we will add another script element immediately below the one we added in point 1. In this second script element, we will use jQuery functions and selectors to give our menu the behavior of an accordion.
2. Now let’s give the div element in our HTML document an id attribute, so that we can easily access to it using jQuery selectors. In this example, I will use “accordion”. You are free to use whichever id you see fit.



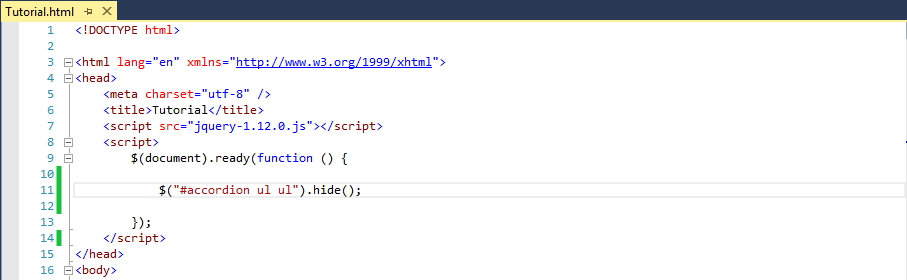
1. The first function we will use is $(document).ready(). It takes another function as an argument and ensures this nested function executes only when the DOM hierarchy is fully constructed. We will then write the rest of the code inside this second function.



1. Inside our function, we will first use a jQuery selector to work with only those unordered lists inside our main unordered list in the menu:



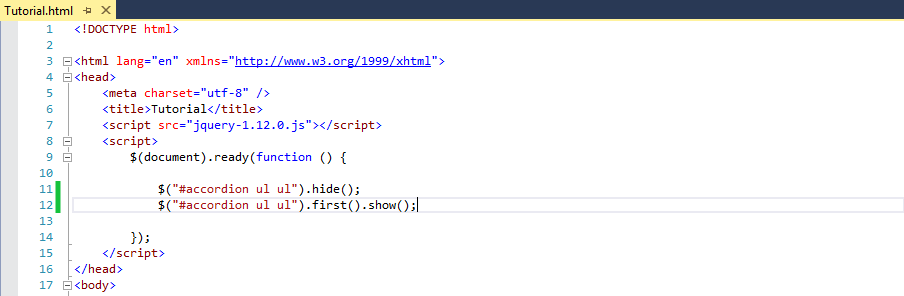
The $("#accordion ul ul") selector will allow us to do that. Then we will use the .hide() function to make the selected elements hidden as soon as the DOM is ready.



1. If you run the HTML file now, you will notice all list items are hidden and you can only see main unordered list’s h3 elements.



Yet this is no the functionality we want. When the documents loads, we want to shot the first unordered list’s list items. We will then use another function on the same selector that will allow us to choose the first element targeted by the selector and the .show() function to make it visible. Keeping it simple.



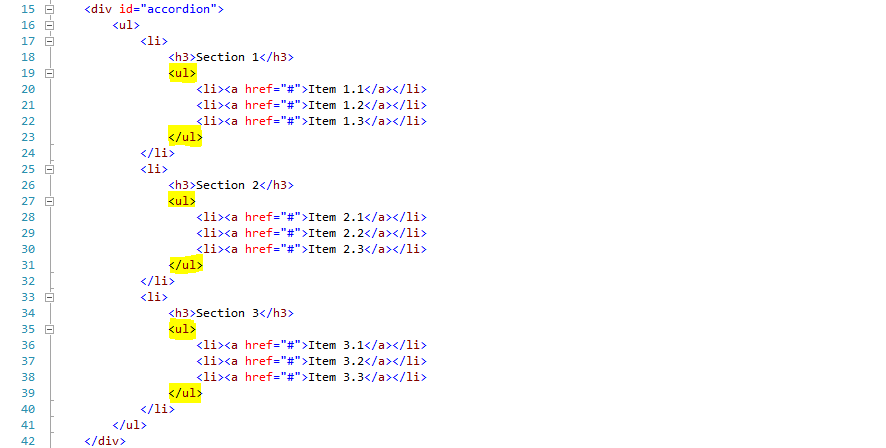
If you run our HTML file again, you will see the result as intended:



1. Our accordion menu is looking good, but we still want to be able to click the headings of our menu and show the adequate item list and hide the others accordingly. For this, we will use the $("#accordion h3") selector to work only with the h3 elements in our menu and with .click() function we will append to it any functionality we want when they are clicked inside its nested function.



1. Now, to achieve the previously mentioned behavior. Inside the nested click function, we will use $(“this”) selector to specifically refer to the h3 element being clicked and we will use the .next() function to work with the HTML element that comes immediately after it. This will be the unordered list immediately below the clicked h3 element.

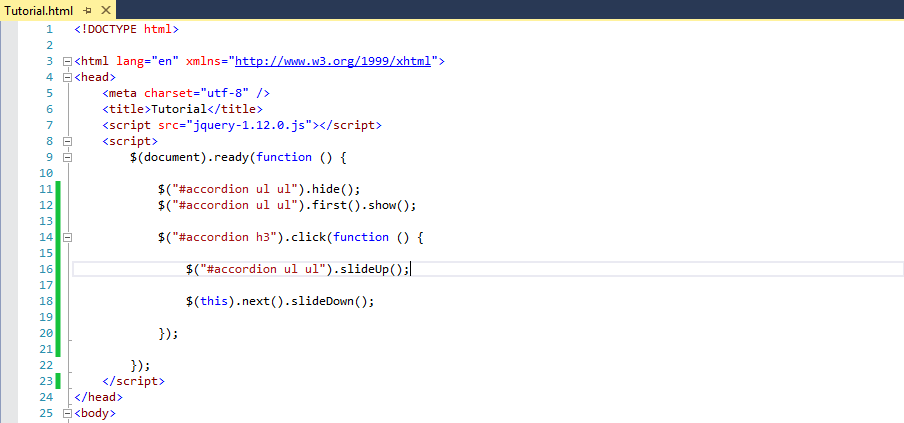


Using method chaining, we will also use the .slideDown() function to show the unordered list.



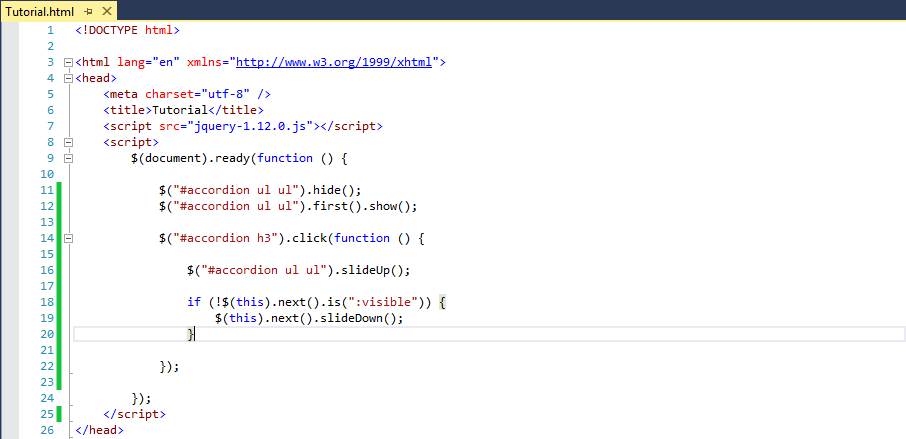
Run the application and click any h3 element, you will notice that the second and third h3 elements trigger the event, but not the first as it is already slid down.

1. If we click an h3 element, we also want the others to slide up, so that only the unordered list we want is being shown. To achieve this, we will simple slide up all unordered list elements in our menu as soon as one h3 element is clicked.



Run the application and click any h3 element, our menu now behaves like an accordion. But if you click the h3 element of a section that is already slid down, this trigger a strange animation, as the event slides up all unordered lists and immediately after it slides down the one we want to show.

1. To fix this problem, we will only slide down the unordered list if is not already visible using the .is() function and :visible selector. We then have to surround line number 18 with an if statement with the following condition: !$(this).next().is(":visible").



1. Finally, run our HTML document and test our accordion. If you followed the steps carefully, it should be working. If it is not, go over the steps again and fix any issues.

You can also go to <https://jqueryui.com/accordion/#collapsible> and get a simplified code for an accordion menu from the jQuery UI library.

