JQUERY:

What is jQuery

* jQuery is a small and lightweight JavaScript library.
* jQuery is cross-platform.
* jQuery means "write less do more".
* jQuery simplifies AJAX call and DOM manipulation.

jQuery is a small, light-weight and fast JavaScript library. It is cross-platform and supports different types of browsers. It is also referred as ?write less do more? because it takes a lot of common tasks that requires many lines of JavaScript code to accomplish, and binds them into methods that can be called with a single line of code whenever needed. It is also very useful to simplify a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

Why jQuery is required

Sometimes, a question can arise that what is the need of jQuery or what difference it makes on bringing jQuery instead of AJAX/ JavaScript? If jQuery is the replacement of AJAX and JavaScript? For all these questions, you can state the following answers.

* It is very fast and extensible.
* It facilitates the users to write UI related function codes in minimum possible lines.
* It improves the performance of an application.
* Browser's compatible web applications can be developed.
* It uses mostly new features of new browsers.

So, you can say that out of the lot of JavaScript frameworks, jQuery is the most popular and the most extendable. Many of the biggest companies on the web use jQuery.

Some of these companies are:

* Microsoft
* Google
* IBM
* Netflix

# jQuery History

jQuery was first released in January 2006 by **John Resig** at BarCamp NYC. It is currently headed by Timmy Wilson and maintained by a team of developers.

Nowadays, jQuery is widely used technology. Most of the websites are using jQuery.

### jQuery Release History

Let's see the release dates of jQuery versions.

|  |  |
| --- | --- |
| **Version No.** | **Release Date** |
| 1.0 | 26,August,2006 |
| 1.1 | 14,January,2007 |
| 1.2 | 10, September, 2007 |
| 1.3 | 14, January, 2009 |
| 1.4 | 14, January, 2010 |
| 1.5 | 31, January, 2011 |
| 1.6 | 3, May, 2011 |
| 1.7 | 3, November, 2011 |
| 1.8 | 9, August, 2012 |
| 1.9 | 15, January, 2013 |
| 1.10 | 24,May, 2013 |
| 1.11 | 24, January, 2014 |
| 2.0 | 18, April, 2013 |
| 2.1 | 24, January, 2014 |

**jQuery** is the easiest and the most popular **javascript library**. A library is just a collection of tools that allows us to implement functionality or effects without the need to write all the code that is needed to perform a certain task.

2. The purpose of jQuery is make it **easier** for you to use javascript on your website.

3. jQuery is a light-weight javascript library which means **“write less, do more”**.

4. jQuery library allows you to do following things:

* HTML/DOM(Document Object Model) Manipulation.
* CSS manipulation.
* Provides event methods to trigger and respond to a events on an html page such as mouse click, keypress etc.
* Simplifies AJAX calls.

## 2. Prerequisite

To learn jQuery faster and with less effort, you must have a basic knowledge of CSS, HTML & JavaScript.

## 3. jQuery Get Started

In this section, we will learn how to install and use jQuery in your project.

### 3.1 How to Add jQuery to your website

There are two ways you can add jQuery to your web pages.  
1. Download the jQuery library from jQuery.com -> **Not recommended method**.  
2. Include jQuery to your project using the CDN provided by Google, Microsoft etc -> **Recommended Method**.

#### 3.1.1 Download the jQuery library from jQuery.com

To download the jQuery visit this official page: [http://jquery.com/download/](https://jquery.com/download/). Here you will find two versions.

**Production version:** Download this version, if you intend to use the downloaded library for a live website.  
**Development version:** Download this version for testing and development purposes.

**After download:**  
Once you have downloaded the jQuery library. You need to **add the reference to it** from every webpage of your website, to do this add the link to the jQuery library inside <script> tag and in the <head> section of the webpage.

Place the downloaded jQuery file in the **same folder** where you have placed all of your webpages and then add the reference to the jQuery file in head section of every webpage like this:

<head>

<script src="jquery-3.4.0.min.js"></script>

</head>

#### 3.1.2 Use jQuery CDN

In this method, **you do not have to download and host the jQuery file**, you can use the jQuery file provided by Google and Microsoft CDN (Content Delivery Network). These **CDNs are fast** as they deliver the file from your **nearest hosting server** which is why I have recommended this method. All the tutorials that I have written on jQuery, I have used the CDN provided by Google.

**jQuery file provided by Google CDN:**  
You refer the file like this in the head section of your webpage:

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js"></script>

</head>

## 4. First jQuery Example

In the following example we are using the jQuery to hide all the paragraphs when the button is clicked. We have added to the reference to jQuery CDN file in the <script> tag inside <head>.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("p").hide();

});

});

</script>

</head>

<body>

<h2>jQuery tutorial on Beginnersbook.com</h2>

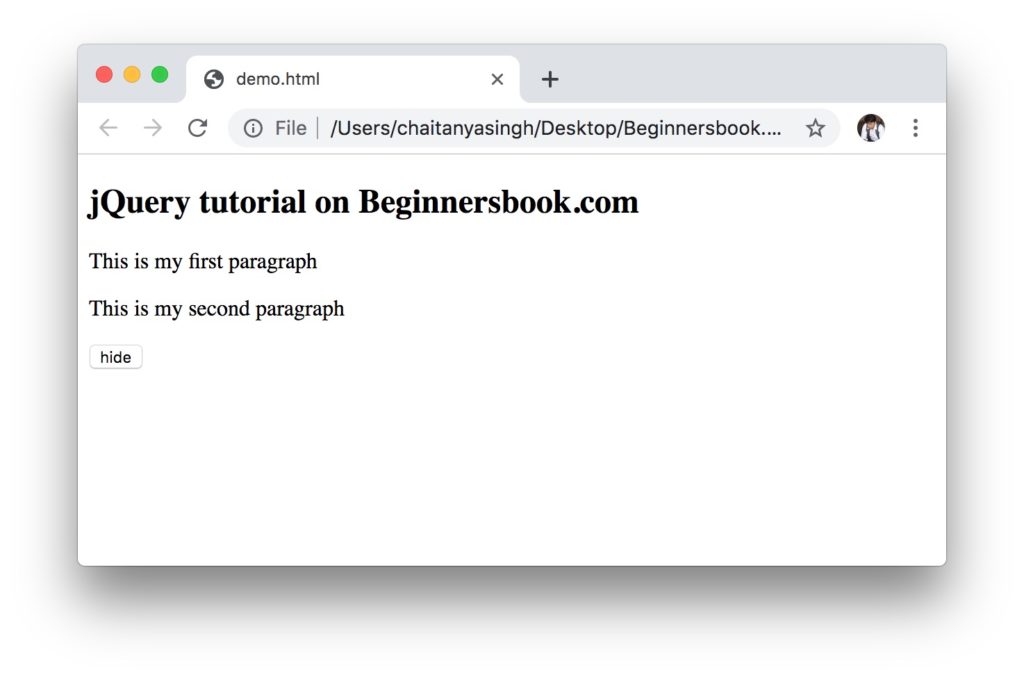
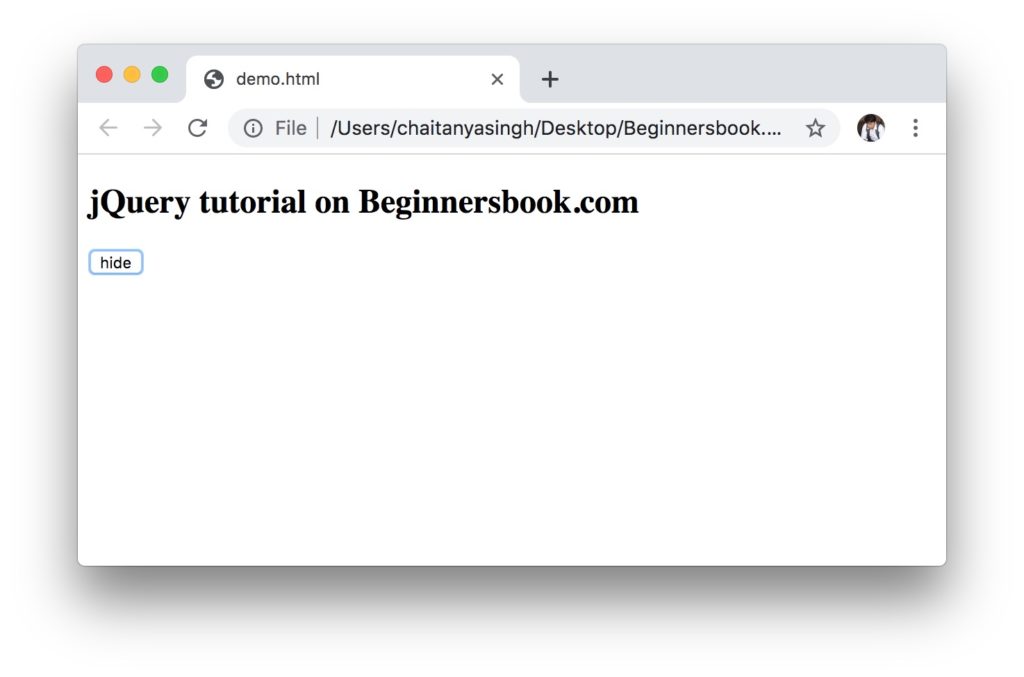
<p>This is my first paragraph</p>

<p>This is my second paragraph</p>

<button>hide</button>

</body>

</html>

**Output:**  
**Before the button is clicked:**  
  
**After the button is clicked:**  


There are couple of things you may be wondering about in the above example. Don’t worry, we will discuss everything in detail in the separate tutorials. I have added the links to those tutorials at the end of this article. For now, just discuss little bit about everything so we get to know what is happening here.

The following is the jQuery part in the above example:

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("p").hide();

});

});

</script>

Here the first line is the reference to the jQuery CDN, this gives us access to the **jQuery library** which is full of tools that we can use to implement functionality or effect in our webpage without writing code. For example, here we use the jQuery method hide() to hide all the paragraphs, the actual implementation of hide() method is available in the file which you are **referring using the CDN**.

### 4.1 The Document Ready function

We always write the jQuery code inside document ready function like this:

$(document).ready(function(){

// jQuery code

});

We do this to prevent any jQuery code from running before the document is finished loading. It is always best to wait for the document to be fully loaded before any of the jQuery code runs, this prevents the unexpected behaviour that may arise if write the jQuery code outside document ready function.

|  |  |  |
| --- | --- | --- |
| LINK | EXAMPLE | DESCRIPTION |
| [\* Selector](https://beginnersbook.com/2019/04/jquery-selector-select-all/) | $(“\*”) | Selects all the elements of an html page. |
| [#id Selector](https://beginnersbook.com/2019/04/jquery-id-selector/) | $(“#myid”) | Selects the element with id value equals to  “myid”. |
| [.class Selector](https://beginnersbook.com/2019/04/jquery-class-selector/) | $(“.myclass”) | Selects all the elements with class “myclass”. |
| [Multiple class Selector](https://beginnersbook.com/2019/04/jquery-multiple-classes-selector/) | $(“.myclass1,.myclass2”) | Selects all the elements with class “myclass1”  or “myclass2”. |
| [Element Selector](https://beginnersbook.com/2019/04/jquery-element-selector/) | $(“p”) | Selects all the paragraphs |
| [Multiple elements Selector](https://beginnersbook.com/2019/04/jquery-multiple-elements-selector/) | $(“h2,p,a”) | Selects all the h2, p and a elements of an  html page |

## What can we do using jQuery Selectors

You can select html elements based on their name, class, id, attribute, value etc. and then you can apply behaviour to them.

**For example** lets say you have an html page with a paragraph and a button, you can apply behaviour to the button using jQuery selector to hide the paragraph when the button is clicked, or you can change the background colour of the paragraph when the button is clicked, you can do whatever you want, basically jQuery selectors give you full control over the html elements.

## Types of jQuery Selectors

There are number of jQuery Selectors which you can use to manipulate the html elements. Lets take few examples to understand the various types of selectors.

jQuery selector starts with $ sign followed by parenthesis. For example $(p) selects all the paragraphs of the html page.

### jQuery Element Name Selector

jQuery element name selector syntax is $(element\_name). It selects the html element based on the element name. We provide the element name inside parenthesis to select it.

In the following example we have an html page that contains few elements such as head, body, h2, p, button etc. In the script section of the web page we have added the jQuery function that selects the h2 element and hide it when the button is clicked.

We have used the selector on h2 element and we are calling hide function like this: $(“h2”).hide(); this will select all the h2 element of the html page and hide them when the button is clicked.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("h2").hide();

});

});

</script>

</head>

<body>

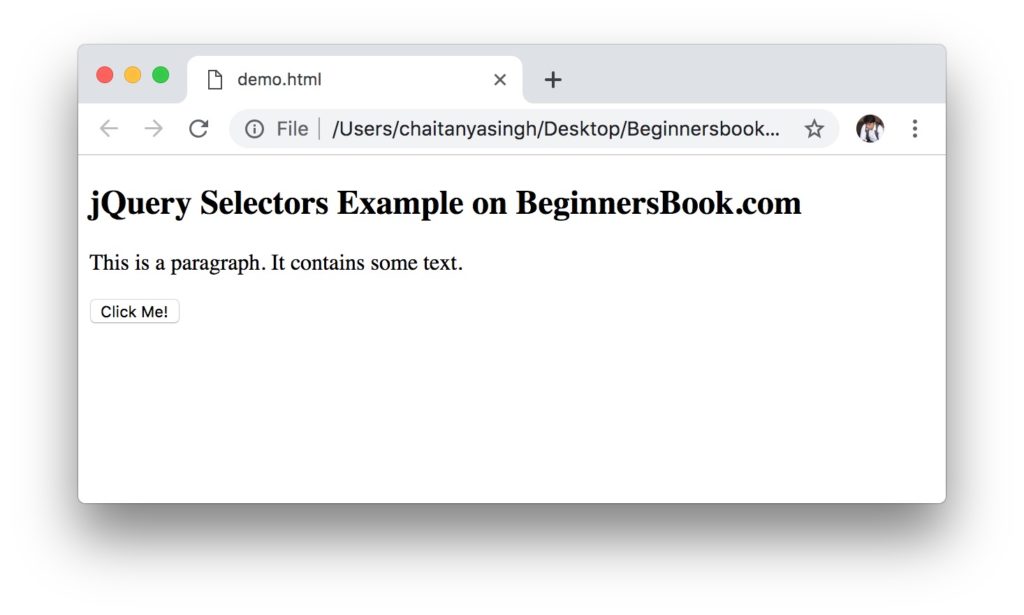
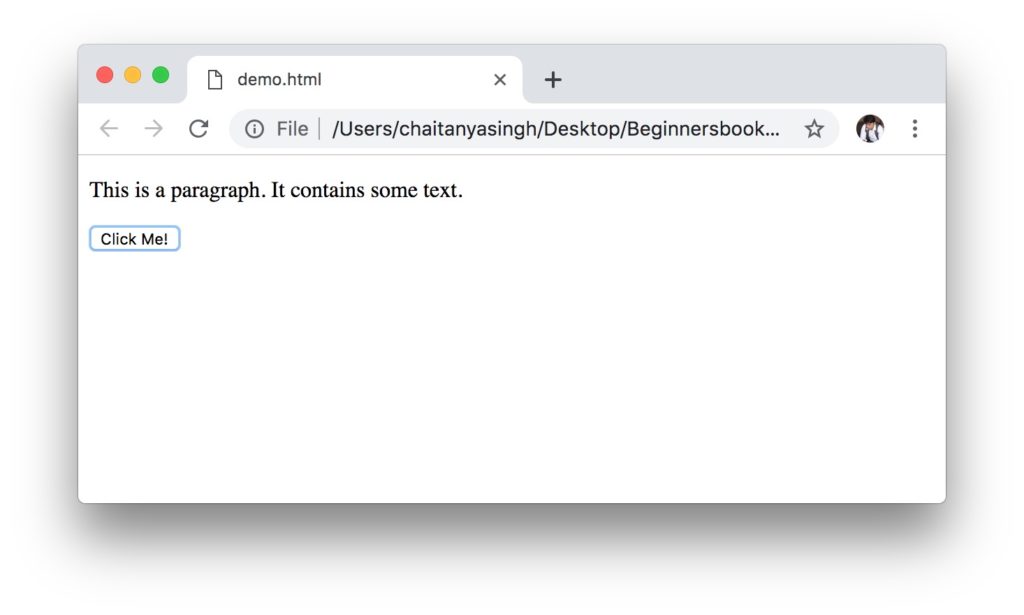
<h2>jQuery Selectors Example on BeginnersBook.com</h2>

<p>This is a paragraph. It contains some text.</p>

<button>Click Me!</button>

</body>

</html>

**Output:**  
**Before button is clicked:**  
This is when the above html page is loaded in the browser, we have not yet clicked the button.  
  
**After button is clicked:**  
Once the button is clicked it selected all the h2 element of the html page and hidden them.  


### jQuery Element #id Selector

In the above example we have seen how to select an html element based on the name, however the downside of using name selector is that it selects all the elements of the html page, for example $(p) selects all the paragraphs of the html page.

What if we want to select only a particular paragraph and not all the paragraphs of an html page, we can do this by assigning an id to that particular paragraph and then select the para based on the id.

The syntax of element #id selector is:

$("#my\_id")

This will select the element that has the id my\_id.

Lets take an example.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("#myid").hide();

});

});

</script>

</head>

<body>

<h2 id="myid">jQuery element id selector example</h2>

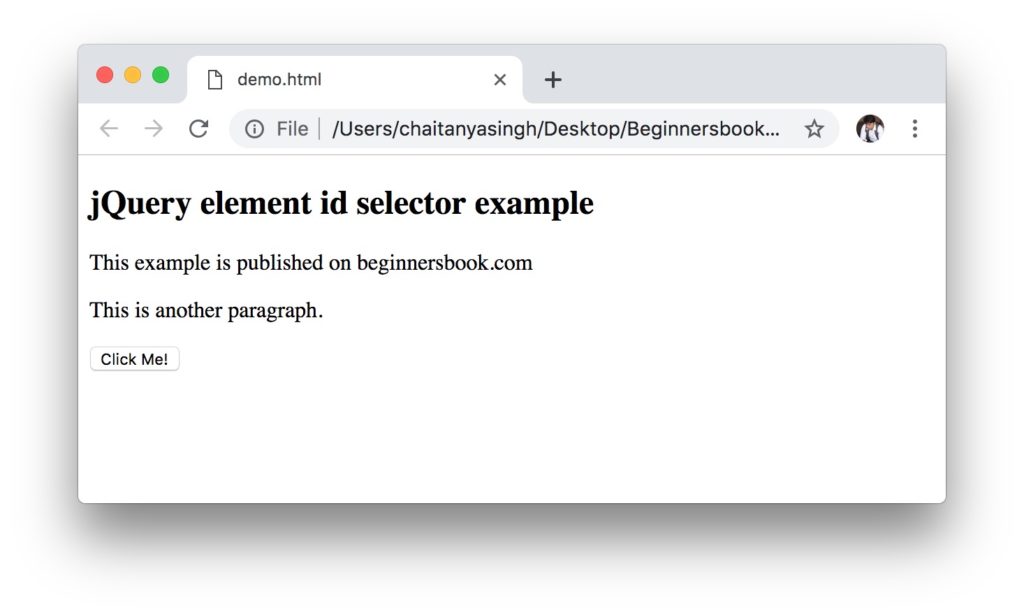
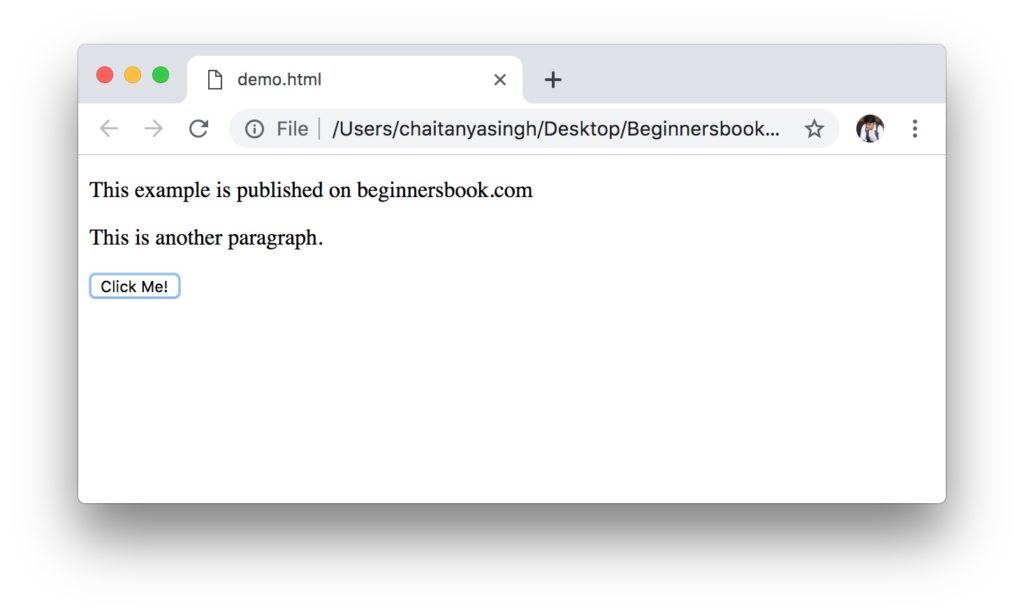
<p>This example is published on beginnersbook.com</p>

<p>This is another paragraph.</p>

<button>Click Me!</button>

</body>

</html>

**Output:**  
**Before button is clicked:**  
  
**After button is clicked:**  


### jQuery Element .class Selector

In the above example, we have seen the element id selector. The limitation of id selector is that an id is unique and two html elements cannot have the same id.

What if we want to hide a particular h2 element and a particular para element on a button click event. We can achieve this with the help of **element class selectors**, we can assign the same class to that h2 element and p element and call the hide function on the element class selector. The syntax of element class selector is:

$(".myclass")

This will select all html elements that has the class myclass.

Lets take an example:  
In the following example we have assigned a class myclass to three html elements, h2, p and button element. We have called the hide function on the element class selector where we passed the class name as myclass. This should hide all these three elements on the button click event.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$(".myclass").hide();

});

});

</script>

</head>

<body>

<h2 class="myclass">jQuery element class selector example</h2>

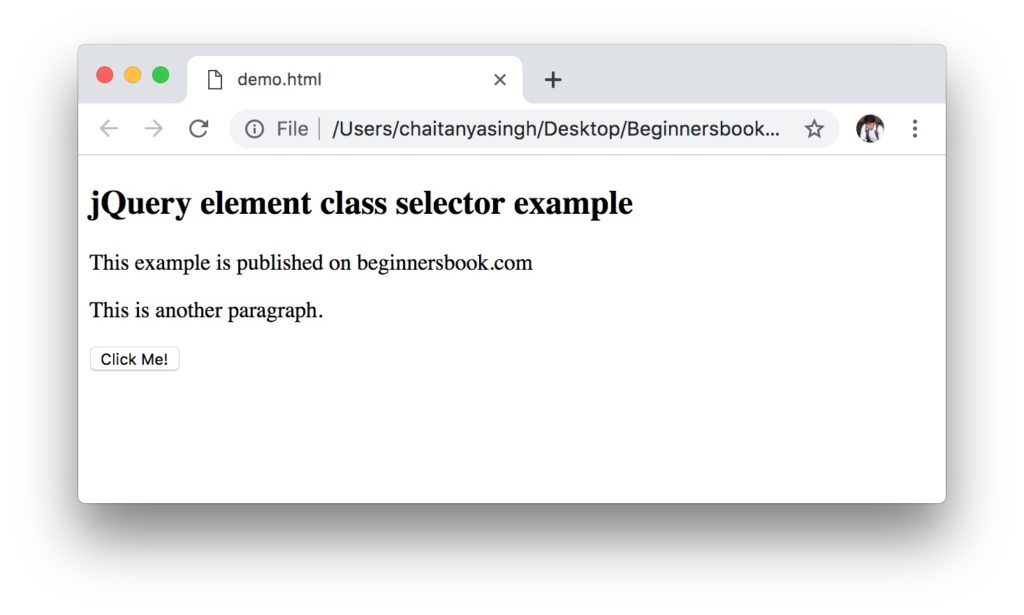
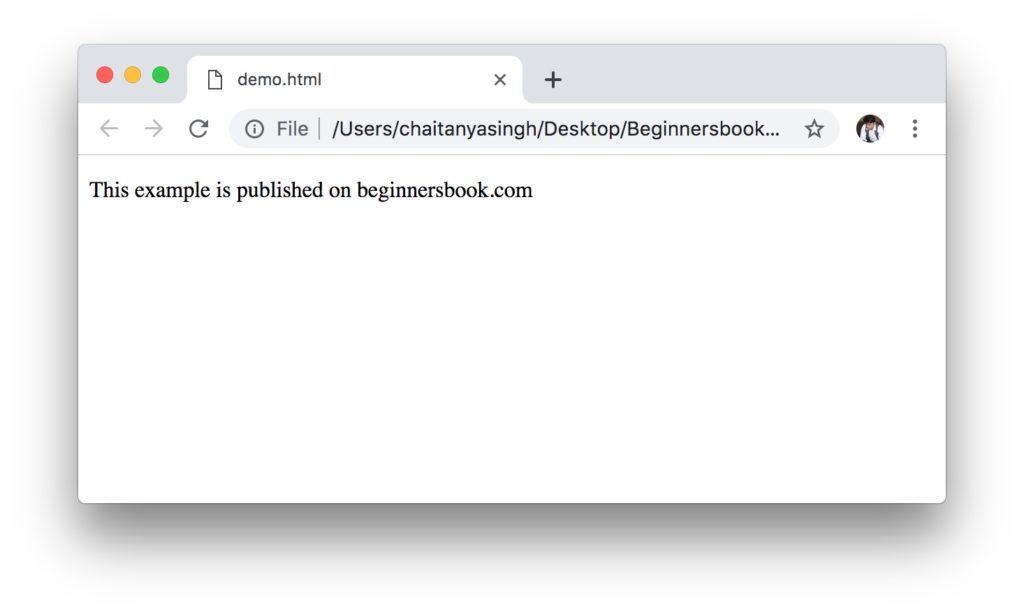
<p>This example is published on beginnersbook.com</p>

<p class="myclass">This is another paragraph.</p>

<button class="myclass">Click Me!</button>

</body>

</html>

**Output:**  
**Before button is clicked:**  
  
**After button is clicked:**  


**jQuery’s event API** provides several useful methods that can be used to create dynamic web pages. Events are nothing but actions that you can perform on a web page such as mouse click, mouse hover, double click etc. **In jQuery we can detect such events** and can write a custom code (function) for each of these events. This gives us freedom to do whatever we want to do on an occurrence of an event, such custom functions are called **event handlers**. In this guide, we will discuss various events provided by jQuery event API.

## jQuery Event Method Syntax

This is a button click event, it will trigger when we click on a button on a webpage.

$("button").click();

Now that we have detected a particular event using above code, our next step to write a custom code that executes when a button click event occurs. For this we have to write a custom function called event handler as shown below:

$("button").click(function(){

// custom code for event handler

});

## $(document).ready() method

Before we start learning various events, lets first understand the use of **most commonly used method** in jQuery. $(document).ready() method executes when a document (a web page) is **fully loaded in the browser**. We almost always write event handlers inside this method because we do not want to trigger an event when our page is not fully loaded.

We write a **event handler inside $(document).ready() method** like this: In the following example we are hiding all the paragraphs on a button click event. We have written the button click event handler inside $(document).ready() method so that the event only triggers when the page is fully loaded.

$(document).ready(function(){

$("button").click(function(){

$("p").hide();

});

});

## jQuery Events

### jQuery Mouse Events

|  |  |
| --- | --- |
| TUTORIAL LINK | DESCRIPTION |
| [click](https://beginnersbook.com/2019/04/jquery-click-event/) | jQuery click event occurs when a user clicks on an html element. |
| [dblclick](https://beginnersbook.com/2019/04/jquery-dblclick-event/) | jQuery dblclick event occurs when a user double clicks on an html element. |
| [mouseenter](https://beginnersbook.com/2019/04/jquery-mouseenter-event/) | jQuery mouseenter event occurs when the mouse pointer enters the html element. |
| [mouseleave](https://beginnersbook.com/2019/04/jquery-mouseleave-event/) | jQuery mouseleave event occurs when the mouse pointer leaves the html element. |
| [mousedown](https://beginnersbook.com/2019/04/jquery-mousedown-event/) | jQuery mousedown event occurs when any of the mouse button is pressed down while the mouse pointer is over an html element. |
| [mouseup](https://beginnersbook.com/2019/04/jquery-mouseup-event/) | jQuery mouseup event occurs when any of the mouse button is released while the mouse pointer is over an html element. |
| [hover](https://beginnersbook.com/2019/04/jquery-hover-event/) | jQuery hover event is a combination of mouseenter() and mouseleave() methods. First event handler function executes when mouse pointer enters the html element and the second event handler function executes when the mouse pointer leaves the html element. |

### jQuery Form Events

|  |  |
| --- | --- |
| TUTORIAL LINK | DESCRIPTION |
| [focus](https://beginnersbook.com/2019/05/jquery-focus-event/) | jQuery focus event occurs when a form field gets focus. |
| [blur](https://beginnersbook.com/2019/05/jquery-blur-event/) | jQuery blur event occurs when a form field loses focus. |
| [submit](https://beginnersbook.com/2019/05/jquery-submit-event/) | jQuery submit event occurs when a form is submitted. |
| [change](https://beginnersbook.com/2019/05/jquery-change-method/) | jQuery change event occurs when the value of an element has been changed. The applicable elements for this event are: <input>, <textarea> and <select> |

### jQuery Keyboard Events

|  |  |
| --- | --- |
| TUTORIAL LINK | DESCRIPTION |
| [keypress](https://beginnersbook.com/2019/05/jquery-keypress-event/) | jQuery keypress event occurs when a button is pressed down. The keypress event is not triggered for special keys such as ALT, CTRL, SHIFT, ESC. |
| [keydown](https://beginnersbook.com/2019/05/jquery-keydown-event/) | jQuery keydown event occurs when a keyboard key is pressed down. Unlike keypress event, this event is triggered for ALT, CTRL, SHIFT, ESC keys. |
| [keyup](https://beginnersbook.com/2019/05/jquery-keyup-event/) | jQuery keyup event occurs when a keyboard key is released |

### jQuery Window Events

|  |  |
| --- | --- |
| TUTORIAL LINK | DESCRIPTION |
| [resize](https://beginnersbook.com/2019/05/jquery-resize-event/) | jQuery resize event occurs when the browser window is resized. |
| [scroll](https://beginnersbook.com/2019/05/jquery-scroll-event/) | jQuery scroll event occurs when user scrolls in the specified html element. |

In the following example, we have associated the click method to the [**id selector**](https://beginnersbook.com/2019/04/jquery-id-selector/), we have provided the button id, so when we click on the button, the click event is triggered. In the click event function we are hiding the h2 elements using hide() method.

You can see in the output screenshots that when you click on the “Hide Heading” button, the heading of the document gets hidden.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("#myid").click(function(){

$("h2").hide();

});

});

</script>

</head>

<body>

<h2>jQuery click event example</h2>

<p>This example is published on beginnersbook.com. The main heading

will be hidden when you click on the "Hide Heading" button</p>

<button id="myid">Hide Heading</button>

</body>

</html>

# jQuery dblclick() Method

BY CHAITANYA SINGH | FILED UNDER: [**JQUERY**](https://beginnersbook.com/category/jquery/)

**jQuery dblclick() Method** attaches a double click event handler function to an html element. This event handler function executes when a user double clicks on the attached html element.

## jQuery dblclick() Method Syntax

$(selector).dblclick(function(){

//event handler code. This code will execute

// when a user double clicks on the selected html element.

});

Here $(selector) is a [**jQuery selector**](https://beginnersbook.com/2019/04/jquery-selectors/) that selects an html element and the **dbclick() method** attaches that selected element to the event handler function.

## jQuery dblclick() Example

In the following example we have attached the double click event handler function to the jQuery selector that selects h2, p and button elements, which means whenever a user double clicks on any of these elements, they will be hidden because in event handler function we are calling [**hide() method**](https://beginnersbook.com/2019/04/jquery-effects-show-and-hide/) like this: $(this).hide(); that hides the current element.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("h2, p, button").dblclick(function(){

$(this).hide();

});

});

</script>

</head>

<body>

<h2>jQuery double click event example</h2>

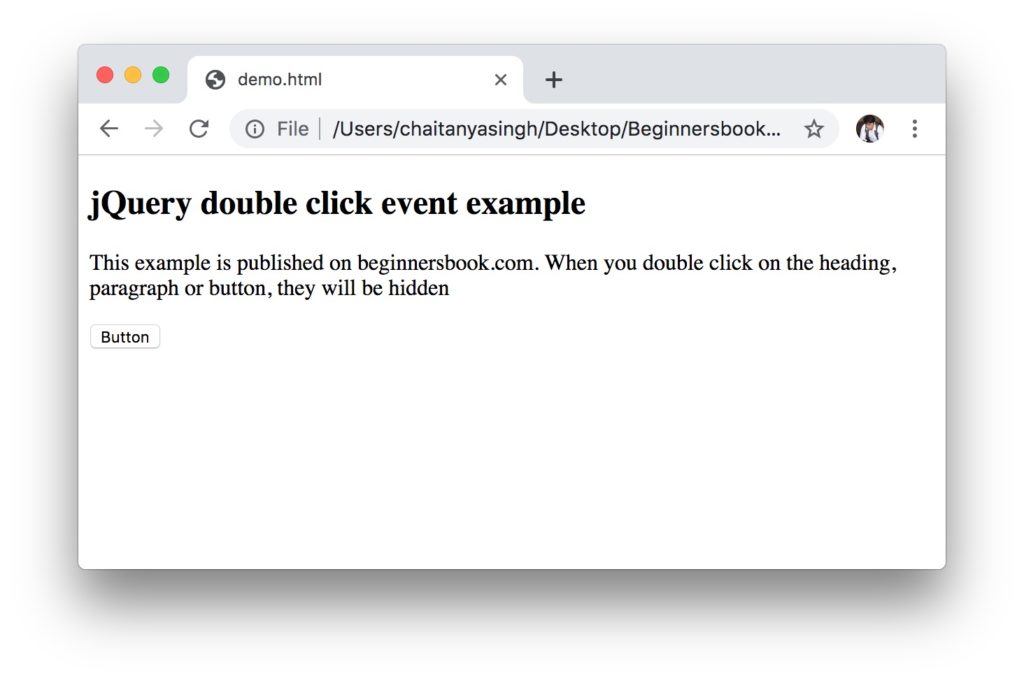
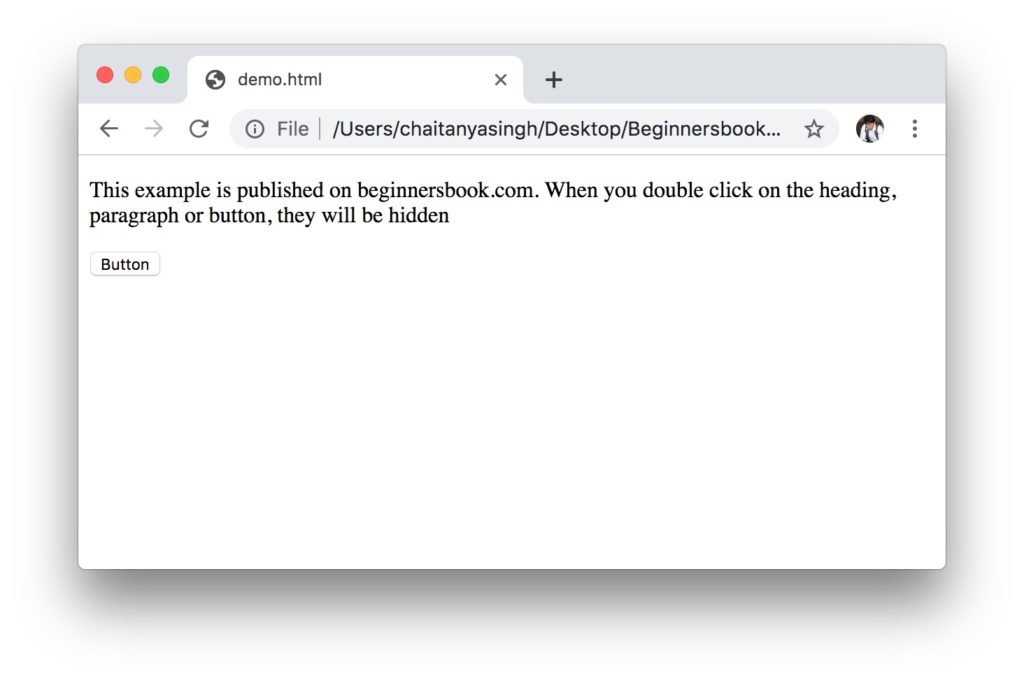
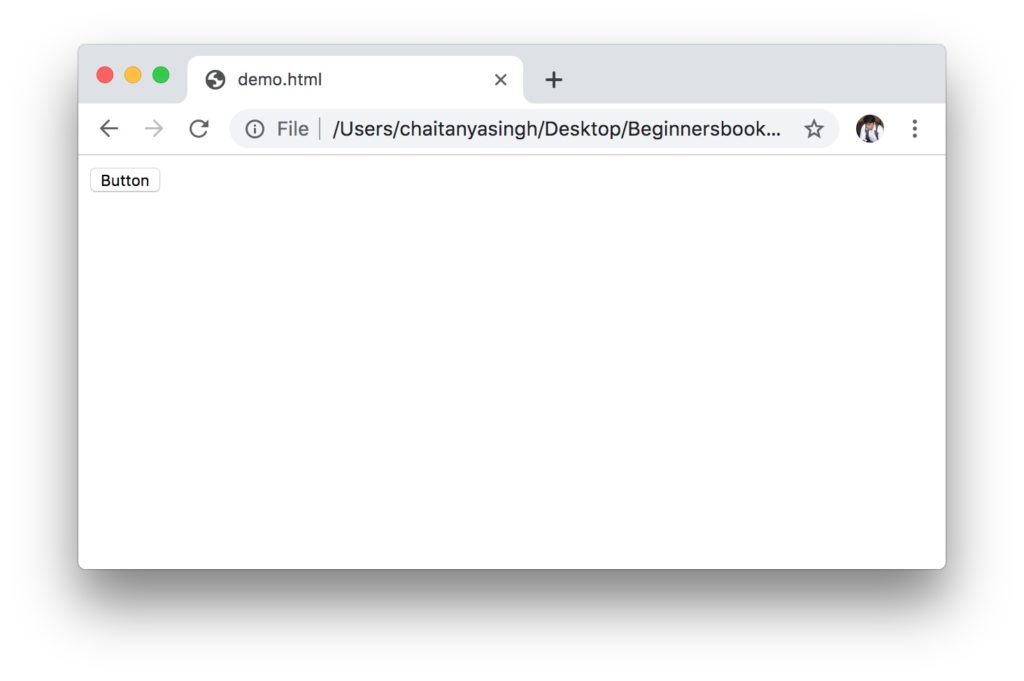
<p>This example is published on beginnersbook.com. When you double

click on the heading, paragraph or button, they will be hidden</p>

<button>Button</button>

</body>

</html>

**Output:**  
**Before a user double clicks on any html element:**  
  
**After a user double clicks on the h2 heading:**  
  
**After a user double clicks on the paragraph:**  


#### jQuery Form Events

|  |  |
| --- | --- |
| TUTORIAL LINK | DESCRIPTION |
| [focus](https://beginnersbook.com/2019/05/jquery-focus-event/) | jQuery focus event occurs when a form field gets focus. |
| [blur](https://beginnersbook.com/2019/05/jquery-blur-event/) | jQuery blur event occurs when a form field loses focus. |
| [submit](https://beginnersbook.com/2019/05/jquery-submit-event/) | jQuery submit event occurs when a form is submitted. |
| [change](https://beginnersbook.com/2019/05/jquery-change-method/) | jQuery change event occurs when the value of an element has been changed. The applicable elements for this event are: <input>, <textarea> and <select> |

# jQuery focus() method

BY CHAITANYA SINGH | FILED UNDER: [**JQUERY**](https://beginnersbook.com/category/jquery/)

**jQuery focus() method** attaches an event handler function to html elements. This event handler function executes when a **form field gets focus**.

## jQuery focus() method Syntax

$("input").focus(function(){

//code to execute when the focus event is triggered

});

## jQuery focus() Example

In the following example we have two input form fields, we want to change the background colour of the form field **when the field gets focus** (when we click inside the box), to do this, we have attached the event handler function where we are changing the background colour of html elements, to the input html element. This way when any of the form field gets focus, its background colour gets changed to colour value #444.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("input").focus(function(){

$(this).css("background-color", "#444");

});

});

</script>

</head>

<body>

<h2>jQuery focus event example on Beginnersbook.com</h2>

Your Name: <input type="text" name="name"><br>

Your Age: <input type="text" name="age">

</body>

</html>

**jQuery blur() method** attaches an event handler function to html elements. This event handler function executes when a form field loses focus. This method works just opposite to the [**focus() method**](https://beginnersbook.com/2019/05/jquery-focus-event/). The difference between focus() and blur() method is that the focus event is triggered when a form field gets focus while blur event is triggered when a form field loses focus.

## jQuery blur() method Syntax

$("input").blur(function(){

//code to execute when a form field loses focus.

});

Here blur() method attaches the event handler function where we write our custom code to the input html elements. This custom code inside event handler function executes when the form field loses focus.

## jQuery blur() Example

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("input").blur(function(){

$(this).css("background-color", "#eee");

});

});

</script>

</head>

<body>

<h2>jQuery blur event example on Beginnersbook.com</h2>

Your Name: <input type="text" name="name"><br>

Your Age: <input type="text" name="age">

</body>

</html>

**jQuery submit() Method** triggers the submit event when the **form is submitted**. The submit() method attaches an event handler function to the “form”, this event handler function executes when the submit event is triggered.

## jQuery submit() Method Syntax

$("form").submit(function(){

//code to execute when the form is submitted.

});

## jQuery submit() Example

In the following example we have a form with two input fields. The value of these two input fields are already populated, we want an **alert message to displayed when the form is submitted**. To do this, we have set an alert message inside submit event handler function, this event handler function executes when the form is submitted.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("form").submit(function(){

alert("form is submitted.");

});

});

</script>

</head>

<body>

<h2>jQuery submit event example on Beginnersbook.com</h2>

<form action="">

Your Name: <input type="text" name="name" value="Chaitanya"><br>

Your Age: <input type="text" name="age" value="31"><br><br>

<input type="submit" value="Submit">

</form>

</body>

</html>

## jQuery change() Method Syntax

$(selector).change(function(){

//code that executes when the change event is occurred.

});

Here $(selector) can be $(“input”), $(“textarea”) or $(“select”).

## jQuery change() Example

In the following example, we have a textarea and we have **attached the “textarea” to event handler function** using change() method, which means when the value of the textarea changes, the event handler function executes.

To **demonstrate the change event**, we have typed a sentence in “textarea” and then clicked outside of the “textarea”, this triggered the change event and the event handler function executes, in the event handler function we are displaying an alert message which you see in the output screenshots.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("textarea").change(function(){

alert("The text in textarea has been changed.");

});

});

</script>

</head>

<body>

<h2>jQuery change event example on Beginnersbook.com</h2>

<textarea type="text"></textarea>

<p>Write something in the textarea and then click outside of the textarea box.

You will see an alert message that the value of the textarea has been changed.</p>

</body>

</html>

#### jQuery Keyboard Events

|  |  |
| --- | --- |
| TUTORIAL LINK | DESCRIPTION |
| [keypress](https://beginnersbook.com/2019/05/jquery-keypress-event/) | jQuery keypress event occurs when a button is pressed down. The keypress event is not triggered for special keys such as ALT, CTRL, SHIFT, ESC. |
| [keydown](https://beginnersbook.com/2019/05/jquery-keydown-event/) | jQuery keydown event occurs when a keyboard key is pressed down. Unlike keypress event, this event is triggered for ALT, CTRL, SHIFT, ESC keys. |
| [keyup](https://beginnersbook.com/2019/05/jquery-keyup-event/) | jQuery keyup event occurs when a keyboard key is released |

HandsOn for KeyBoard Events

## jQuery keypress() Method Syntax

$("input").keypress(function(){

//this code executes when the keypress event occurs.

});

Here we have selected input text field **$(“input”)** using [**jQuery selector**](https://beginnersbook.com/2019/04/jquery-selectors/) and the keypress() method attaches this html element to the keypress event handler function, which means the keypress event will only trigger when a key is pressed inside the text box field.

## jQuery keypress() Example

In the following example, we are using keypress event to **count the number of characters in the input text field**. To do this we have attached the input text field to the event handler function using **keypress() method**. This triggers the keypress event each time a key is pressed inside the text field, in the event handler function we are increasing the counter each time a key is pressed and displaying the same in the “span” area.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

count = 0;

$(document).ready(function(){

$("input").keypress(function(){

$("span").text(count = count+1);

});

});

</script>

</head>

<body>

<h2>jQuery keypress event example on Beginnersbook.com</h2>

Write anything here: <input type = "text">

<p>The current characters count in the textbox is: <span>0</span></p>

</body>

</html>

## keypress() vs keydown()

The [**keypress() Method**](https://beginnersbook.com/2019/05/jquery-keypress-event/) doesn’t trigger keypress event when any of the special keys such as ALT, CTRL, SHIFT, ESC are pressed, if you want to check for these keys then use keydown() method instead.

## jQuery keydown() Method Syntax

$("input").keydown(function(){

//this code executes when a key is pressed down.

});

Here we have attached the **event handler function** to input text field ($(“input”)) using keydown() method, this way the keydown event will only trigger when the key is pressed inside the text field. If you like, you can select any other html element using [**jQuery selector**](https://beginnersbook.com/2019/04/jquery-selectors/).

## jQuery keydown() Example

In the following example we have **attached the input text field to event handler function using keydown() method**. This makes the keydown event to trigger when a key is pressed down inside the input field.

Here we are changing the background colour of the input field inside the event handler function, which means when the keydown event triggers, the background colour of input field changes.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("input").keydown(function(){

$("input").css("background-color", "green");

});

});

</script>

</head>

<body>

<h2>jQuery keydown event example on Beginnersbook.com</h2>

Write anything here: <input type = "text">

<p>The background color of input text field will change to green, when you

press a key inside the input field.</p>

</body>

</html>

## jQuery keyup() Method Syntax

$("input").keyup(function(){

//this code executes when the keyup event occurs

});

Here we have attached the input text field to the event handler function using keyup() method, which means the keyup event will trigger when a **key is released inside input text field**.

## jQuery keyup() Example

In the following example we are using both the methods, **[keydown()](https://beginnersbook.com/2019/05/jquery-keydown-event/)** and keyup(). Here we have two event handler functions, the first event handler function is attached to input text using keydown() method and the second event handler function is attached to input text field using **keyup() method**.

Here we are changing the background colour of input text field to green when keydown event occurs and the background colour changes to yellow when keyup event occurs. **See the screenshots in the output**.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("input").keydown(function(){

$("input").css("background-color", "green");

});

$("input").keyup(function(){

$("input").css("background-color", "yellow");

});

});

</script>

</head>

<body>

<h2>jQuery keyup event example on Beginnersbook.com</h2>

Write anything here: <input type = "text">

<p>The background color of input text field will change to green, when you

press a key inside the input field and the color will change to yellow when you release the key inside the input text field.</p>

</body>

</html>

#### jQuery Window Events

|  |  |
| --- | --- |
| TUTORIAL LINK | DESCRIPTION |
| [resize](https://beginnersbook.com/2019/05/jquery-resize-event/) | jQuery resize event occurs when the browser window is resized. |
| [scroll](https://beginnersbook.com/2019/05/jquery-scroll-event/) | jQuery scroll event occurs when user scrolls in the specified html element. |

## jQuery resize() Method Syntax

$(window).resize(function(){

//code that executes when the resize event occurs.

});

## jQuery resize() Example

In the following example we have attached the window element to the event handler function using **resize() method**. This method triggers the resize event when the **browser window is resized**. When the resize event is triggered, the event handler function executes, inside the event handler function we have setup an alert message. This alert message will be displayed whenever we change the size of the browser window. **See the screenshots in the output**.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$(window).resize(function(){

alert("Browser window is resized.")

});

});

</script>

</head>

<body>

<h2>jQuery resize event example on Beginnersbook.com</h2>

<p>An alert message "Browser window is resized" will be displayed

when you change the size of the browser window.</p>

</body>

</html>

## jQuery scroll() Method Syntax

$(window).scroll(function(){

//this code executes when the scroll event occurs.

});

Here we have attached the **window object $(window)** to the event handler function, however you can attach **any scrollable element(for example “div”)** to the event handler function using scroll() method.

## jQuery scroll() Example

In the following example we have a “div” element and we have attached this “div” element to the event handler function using scroll() method. When the user scroll inside this “div” element, the scroll counter, following the “div” element is updated simultaneously. Here we are **counting the number of times user scroll inside the “div” element**.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

var count = 0;

$(document).ready(function(){

$("div").scroll(function(){

$("span").text( count= count+1);

});

});

</script>

</head>

<body>

<h2>jQuery scroll event example on Beginnersbook.com</h2>

<div style="border:1px solid grey;width:250px;height:80px;overflow:scroll;">

I am just writing this text to fill the div box, whenever you scroll inside

this div box, the scroll count should be displayed below.

<br>

The count changes simultaneously as you scroll the div element</div>

<p>Scroll counter: <span>0</span></p>

</body>

</html>

|  |  |
| --- | --- |
| TUTORIAL LINK | DESCRIPTION |
| [jQuery show and hide](https://beginnersbook.com/2019/04/jquery-effects-show-and-hide/) | jQuery show() and hide() methods explained with examples. |
| [jQuery Fading effects](https://beginnersbook.com/2019/04/jquery-fading-effects/) | jQuery fadeIn(), fadeOut(), fadeToggle() and fadeTo() methods explained with examples |
| [jQuery sliding effects](https://beginnersbook.com/2019/04/jquery-sliding-effects/) | jQuery slideDown(), slideUp() and slideToggle() methods explained with examples. |
| [jQuery animate](https://beginnersbook.com/2019/04/jquery-animate-effect/) | jQuery animate() method with examples. |
| [jQuery stop()](https://beginnersbook.com/2019/04/jquery-effects-stop-animations/) | jQuery stop() method explained with examples. |
| [jQuery Callback](https://beginnersbook.com/2019/04/jquery-callback-function/) | jQuery callback function with examples. |
| [jQuery chaining](https://beginnersbook.com/2019/04/jquery-chaining-multiple-effects/) | jQuery chaining multiple methods/actions. |

## jQuery hide()

jQuery hide() method hides the selected html element. In the following example, we are hiding the selected h2 element.

We are calling $(this).hide(); inside this $("h2").click(function(), the $("h2").click(function() method runs when we click an h2 element because we have passed h2 in the jQuery selector $(“h2”). The method $(this).hide(); runs inside the click method and it hides the currently clicked element.

Since this code will only run when an h2 element is clicked, this is why in the following example when you click on an h2 heading, it gets hidden but nothing happens when you click other elements such as paragraphs and button.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("h2").click(function(){

$(this).hide();

});

});

</script>

</head>

<body>

<h2>jQuery hide() example</h2>

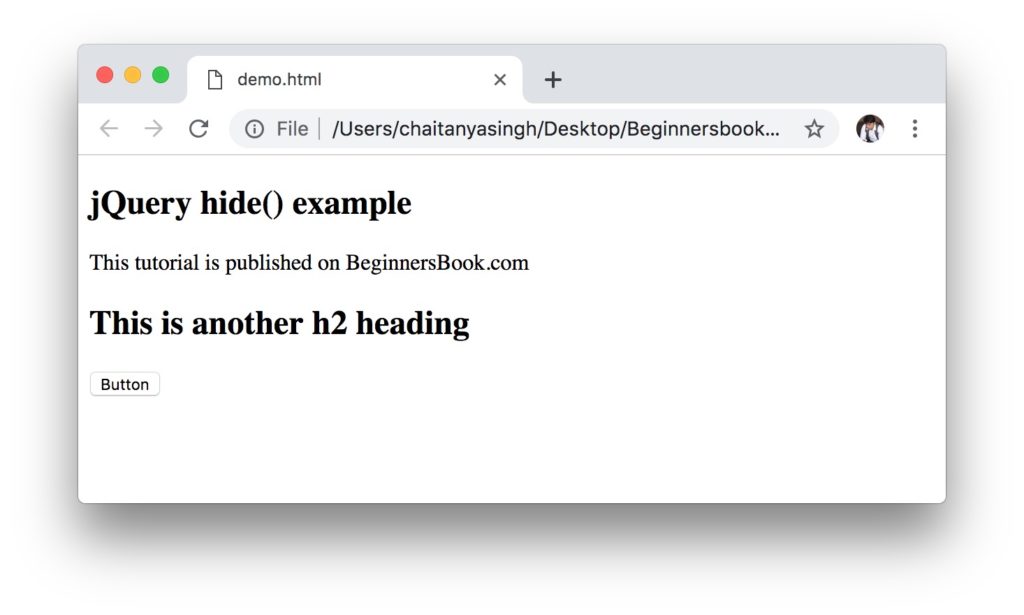
<p>This tutorial is published on BeginnersBook.com</p>

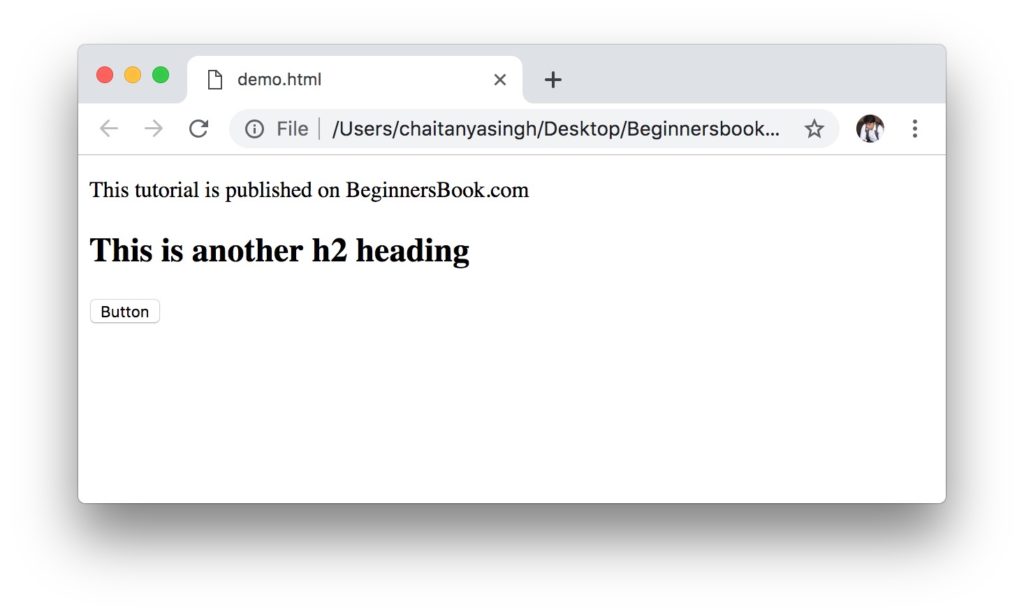
<h2>This is another h2 heading</h2>

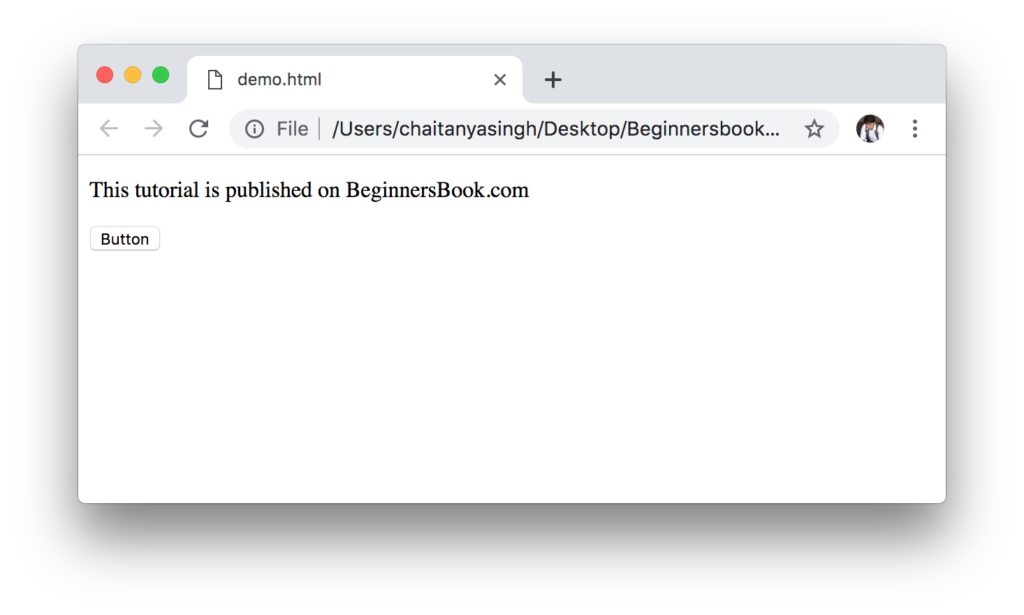
<button>Button</button>

</body>

</html>

**Output:**  
When the page is originally loaded, we have not clicked anything yet:  


After the first h2 element is clicked:  


After the second h2 element is clicked:  


## jQuery show and hide example

In the following example, we have assigned ids “hide” and “show” to the two buttons hide and show respectively and we are calling click function on these ids, we are using [**jQuery id selector**](https://beginnersbook.com/2019/04/jquery-id-selector/) here.

On the click event on hide button we are hiding the paragraph using $("p").hide();, here we have used [**element name selector**](https://beginnersbook.com/2019/04/jquery-element-selector/) to select all the paragraphs.

On the click event on show button we are displaying the hidden paragraph using show() method like this: $("p").show();.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("#hide").click(function(){

$("p").hide();

});

$("#show").click(function(){

$("p").show();

});

});

</script>

</head>

<body>

<h2>jQuery show and hide example</h2>

<p>This tutorial is published on BeginnersBook.com</p>

<button id="hide">hide</button>

<button id="show">show</button>

</body>

</html>

jQuery has four useful methods which you can use to fade html elements in and out of visibility.

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

## jQuery fadeIn() syntax

selector.fadeIn( speed, [callback] );

**selector:** It is used to select a single or multiple html elements on which this fade in effect can be applied. To read more about selectors refer this guide: [**Selectors in jQuery**](https://beginnersbook.com/2019/04/jquery-selectors/).

**speed:** The speed is an optional parameter, fade in speed in milliseconds or in values such as “slow” or “fast”.

**callback:** callback function that executes when the fadein effect is complete. It is an optional parameter.

## jQuery fadeIn() Example

In the following example we have one paragraph and we have set the display to none for this paragraph. On the button click event, we are fading in this paragraph using fadeIn() method.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("p").fadeIn();

});

});

</script>

</head>

<body>

<h2>jQuery fadein effect example</h2>

<p style="display:none;">This tutorial is published on beginnersbook.com. This

paragraph will appear when the FadeIn button is clicked.</p>

<button>FadeIn</button>

</body>

</html>

## jQuery fadeOut() Syntax

selector.fadeOut( speed, callback\_function);

**selector:** It is used to select the html element on which this fade out effect is being applied.

**speed:** The speed of the fadeout effect in milliseconds. It is an optional parameter and can take values such as “slow” or “fast” or in milliseconds.

**callback\_function:** It is also an optional parameter. This callback function executes when the fade out effect is complete. You can use this to set alert or display a message on the screen when fadeout effect is complete.

## jQuery fadeOut() Effect Example

In the following example we are applying the fadeOut effect on the paragraphs by using paragraph selector ($(“p”)), we are calling the fadeOut() method inside button click function which means when the FadeOut button is clicked, the paragraph will fadeout as you can see in the output.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("p").fadeOut();

});

});

</script>

</head>

<body>

<h2>jQuery fadeOut effect example</h2>

<p>This tutorial is published on beginnersbook.com. This

paragraph will gradually fadeout when the FadeOut button is clicked.</p>

<button>FadeOut</button>

</body>

</html>

## jQuery fadeToggle() Syntax

$(selector).fadeToggle(speed, callback\_function);

**$(selector):** [**Selector**](https://beginnersbook.com/2019/04/jquery-selectors/) which is used to select html elements on which this effect is being applied.

**speed:** Optional Parameter. Values can be in milliseconds or string values such as “slow” or “fast”.

**callback\_function:** Optional parameter. This function executes when the toggle effect is complete.

## jQuery fadeToggle() Example

In the following example, we have used paragraph selector to apply the effects on the paragraphs. The effect takes place when the button is clicked because fadeToggle() method is called inside button click function. Since fadeOut() method toggles between fadein and fadeout, the button fades out the paragraph and when clicked again, it fades in the same paragraph.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("p").fadeToggle();

});

});

</script>

</head>

<body>

<h2>jQuery fadeToggle effect example</h2>

<p>This tutorial is published on beginnersbook.com. This

paragraph will fadeout when the FadeToggle button is clicked and

it will fadein when the same button is clicked again.</p>

<button>FadeToggle</button>

</body>

</html>

## jQuery fadeTo() Method Syntax

$(selector).fadeTo(speed, opacity, callback\_function);

**$(selector)** is any [**jQuery Selector**](https://beginnersbook.com/2019/04/jquery-selectors/), which is used to select the html elements on which this effect is being applied.

**speed** is a required parameter in fadeTo() method, it can take values in milliseconds or string values like “slow” or “fast”.

**opacity** is the required parameter that can take values between 0 and 1, 0 means completely invisible while 1 means completely visible. The values can be passed in decimal points such as 0.5, 0.25 etc.

**callback\_function** is an optional parameter which defines a function that executes when the effect is complete.

## jQuery fadeTo() Example

In the following example, we have adjusted the opacity of different html elements with different values using fadeTo() method. Highest opacity value is 1 (completely visible) and lowest opacity value is 0 (completely invisible). As you can see in the output screenshots that higher opacity element is more visible than lower opacity element.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("p").fadeTo("slow", 0.25);

$("h2").fadeTo("slow", 0.10);

$("button").fadeTo("fast", 0.80);

});

});

</script>

</head>

<body>

<h2>jQuery fadeTo effect example</h2>

<p>This tutorial is published on beginnersbook.com. We have adjusted

the opacity of different html elements on this page in the script part, lets

see how they look after the button click event.</p>

<button>FadeTo</button>

</body>

</html>

## jQuery sliding methods

1. jQuery slideDown()  
2. jQuery slideUp()  
3. jQuery slideToggle()

Hands On above 3

## jQuery slideDown() Syntax

$(selector).slideDown(speed, callback\_function);

**$(selector)** is to select the html element.

**speed** in milliseconds or “slow” or “fast” to adjust the speed of slide down effect. It is an optional parameter.

**callback\_function** is also an optional parameter and it executes when the slide down effect is complete.

## jQuery slideDown() example

In the following example we are applying the slide down effect on the paragraph, we have selected the paragraph using [**id selector**](https://beginnersbook.com/2019/04/jquery-id-selector/).

The paragraph is initially set to display none but once the h2 heading is clicked, the paragraph slowly appears on the screen with a slide down transition.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("h2").click(function(){

$("#para").slideDown("slow");

});

});

</script>

</head>

<body>

<h2>jQuery slideDown effect example, Click here</h2>

<p id = "para" style="display:none;">This tutorial is published on beginnersbook.com. This

para is not initially visible, when we click on the h2 heading then this para is visible</p>

</body>

</html>

## jQuery animate() Syntax

$(selector).animate({parameters}, speed, callback\_function);

[**$(selector)**](https://beginnersbook.com/2019/04/jquery-selectors/) is to select html element on which this animation effect is being applied.

**parameters** define the css properties that are used in animation.

**speed** is an optional parameter, it is used for adjusting the speed of animation. It can provided in milliseconds or “slow” or “fast”.

**callback\_function** is also an optional parameter, it is passed as a parameter to the animate() method. It executes when the animation is finished.

## jQuery animate() Example

In the following example we are animating a div block using **animate() method**, we have set the **css properties** of div block inside animate() method and we are calling this method inside button click event so when the button is clicked, the animation takes place.

**Important point to Note:** By default the position of all the elements in an html page is “static” which means they cannot be moved like we did in the example below, so **to change the position during animation you have to first set the position of the html element to relative, fixed, or absolute**. In the following example we have set the position to **absolute**.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("div").animate({

left: '300px',

width: '200px',

height: '200px'

});

});

});

</script>

</head>

<body>

<h2>jQuery Animation example</h2>

<p>This tutorial is published on beginnersbook.com. Here

we will animate the following div block using animate() method.

We will adjust the css properties of this block using animate()

method.</p>

<button>Animate the block</button>

<div style="background:green;height:80px;width:120px;position:absolute;"></div>

</body>

</html>

## jQuery stop() method Syntax

$(selector).stop(clearQueue, goToEnd);

**clearQueue** is an optional **boolean parameter**. By default it is false but when set to true, it clears the animation queue, stopping the current and all the queued animations to **stop immediately**.

**goToEnd** is an optional **boolean parameter**. By default it is set to false, but when set to true, it causes currently running animation to **immediately complete**.

## jQuery stop() method Example

In the following example we have two buttons “Start” and “Stop”. The Start button starts the animation, which gradually increases the font size of specified text to 10em, in between if we stop the animation by clicking on the Stop button, it stop the animation immediately and the font size never reaches 10em.

Here in the output screenshot you can see I have stopped the animation in between and the font size is not 10em.

The stop() method is generally useful when the **animation or effect speed is slow**, then only you get the chance to stop the animation in between, else the effect/animation will complete before you get a chance to stop them, this is the reason I have set the speed to 5000 milliseconds in the following example to slow down the animation speed.

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">

</script>

<script>

$(document).ready(function(){

$("#start").click(function(){

var para = $("p");

para.animate({fontSize: '10em'}, 5000);

});

$("#stop").click(function(){

$("p").stop();

});

});

</script>

</head>

<body>

<h2>jQuery Stop Animation example</h2>

<p>BeginnersBook.com</p>

<button id="start">Start</button>

<button id="stop">Stop</button>

</body>

</html>