

jQuery – Introduction

jQuery is a lightweight, "**write less, do more**", JavaScript library. jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation. Features are:

- HTML/DOM manipulation
- CSS manipulation
- HTML event methods
- Effects and animations
- AJAX
- Utilities

jQuery – Environment Setup

- Download the jQuery library from jQuery.com
- Include jQuery from a CDN, like Google

There are **two versions** of jQuery available for downloading:

Production version - this is for your live website because it has been minified and compressed

Development version - this is for testing and development (uncompressed and readable code)

The jQuery library is a single JavaScript file, and you reference it with the HTML `<script>` tag (notice that the `<script>` tag should be inside the `<head>` section):

jQuery – Reference Models

Hosted Model

```
<head>  
<script src="jquery-3.6.1.min.js"></script>  
</head>
```

CDN Model

```
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.1/jquery.min.js"></script>  
</head>
```

jQuery – Syntax

The jQuery syntax is tailor-made for **selecting** HTML elements and performing some **action** on the **element(s)**.

Basic syntax is: **\$(selector).action()**

- A \$ sign to define/access jQuery
- A (**selector**) to "query (or find)" HTML elements
- A jQuery **action()** to be performed on the element(s)

```
$(this).hide() - hides the current element.
```

```
$("p").hide() - hides all <p> elements.
```

```
$(".test").hide() - hides all elements with class="test".
```

```
$("#test").hide() - hides the element with id="test".
```

jQuery – Sample

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.1/jquery.min.js">
</script>
<script>
$(document).ready(function(){
  $("p").click(function(){
    $(this).hide();
  });
});
</script>
</head>
<body>

<p>If you click on me, I will disappear.</p>
<p>Click me away!</p>
<p>Click me too!</p>

</body>
</html>
```

If you click on me, I will disappear.

Click me away!

Click me too!

If you click on me, I will disappear.

Click me too!

Click me too!

jQuery – Selectors

- jQuery selectors allow you to select and manipulate HTML element(s).
- jQuery selectors are used to "find" (or select) HTML elements based on their name, id, classes, types, attributes, values of attributes and much more. It's based on the existing **CSS Selectors**, and in addition, it has some own custom selectors.
- All selectors in jQuery start with the dollar sign and parentheses: **\$()**.

jQuery – Selectors

Selector	Syntax	Example	Description
Element	\$(elementName)	\$("p")	Selects the “ p ” element
Id	\$(#id)	\$("#test")	Selects the element with the id test
Class	\$(".class")	\$(".c1")	Selects the element with the class name c1
Universal	\$("*")	\$("*")	Selects all element
Current	\$(this)	\$(this)	Selects the current element
Button	\$(":button")	\$(":button")	Selects all button
Even rows of table	\$("tr:even")	\$("tr:even")	Selects all even rows of <tr> element
Odd rows of table	\$("tr:odd")	\$("tr:odd")	Selects all odd rows of <tr> element

jQuery – Event Methods

An **event** represents the precise moment when something happens.

Examples:

- moving a mouse over an element
- selecting a radio button
- clicking on an element

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

jQuery – Focus and blur event

```
<script>
$(document).ready(function(){
    $("input").focus(function(){
        $(this).css("background-color", "yellow");
    });
    $("input").blur(function(){
        $(this).css("background-color", "green");
    });
});
</script>
</head>
<body>

Name: <input type="text" name="fullname"><br>
Email: <input type="text" name="email">

</body>
</html>
```

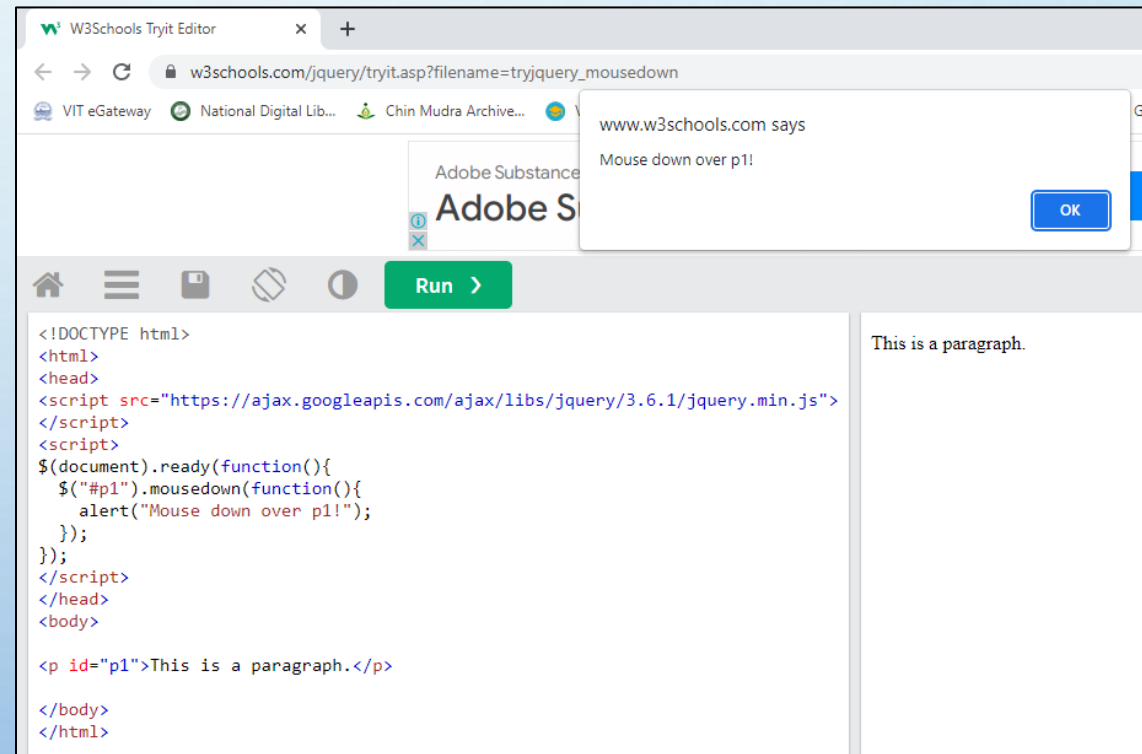
Name:
Email:

Name:
Email:

Name:
Email:

jQuery – Mousedown Event

```
<script>
$(document).ready(function(){
    $("#p1").mousedown(function(){
        alert("Mouse down over p1!");
    });
});
</script>
```



jQuery – Effects

jQuery **effects** add an **X factor** to the website interactivity. jQuery provides a trivially simple interface for doing various kind of amazing effects like show, hide, fade-in, fade-out, slide-up, slide-down, toggle etc. jQuery methods apply commonly used effects quickly with a minimum configuration.

- **Hide/show** – either hides or shows the element. Hide() and show() are methods
- **Fade** – can fade elements in and out of visibility. Various methods are fadeIn(), fadeout(), fadeToggle() and fadeTo().
- **Slide** – slide elements up and down. slideDown(), slideUp() and slideToggle()

jQuery – Sample

```
<script>
$(document).ready(function(){
  $("#hide").click(function(){
    $("p").hide();
  });
  $("#show").click(function(){
    $("p").show();
  });
});
</script>
</head>
<body>

<p>If you click on the "Hide" button, I will disappear.</p>

<button id="hide">Hide</button>
<button id="show">Show</button>
```

If you click on the "Hide" button, I will disappear.

Hide Show

Hide Show

If you click on the "Hide" button, I will disappear.

Hide Show

jQuery – Sample

```
<script>
$(document).ready(function(){
  $("tr:even").css("background-color", "yellow");
});
</script> </head> <body>

<h1>Welcome to My Web Page</h1>

<table border="1">
  <tr>
    <th>Company</th> <th>Country</th>    </tr>
  <tr>
    <td>Alfreds Futterkiste</td> <td>Germany</td>    </tr>
  <tr>
    <td>Berglunds snabbköp</td><td>Sweden</td>    </tr>
  <tr>
    <td>Centro comercial Moctezuma</td> <td>Mexico</td>    </tr>
  <tr>
    <td>Ernst Handel</td><td>Austria</td>    </tr>
  <tr>
    <td>Island Trading</td><td>UK</td> |    </tr>
</table>
```

Welcome to My Web Page

Company	Country
Alfreds Futterkiste	Germany
Berglunds snabbköp	Sweden
Centro comercial Moctezuma	Mexico
Ernst Handel	Austria
Island Trading	UK

jQuery – DOM Manipulation

jQuery comes with a bunch of **DOM** related methods that make it easy to access and manipulate elements and attributes. jQuery methods for DOM manipulation are:

Get Content Methods:

text() - Sets or returns the text content of selected elements

html() - Sets or returns the content of selected elements (including HTML markup)

val() - Sets or returns the value of form fields

jQuery – DOM(Get Content)

```
<script>
$(document).ready(function(){
  $("#btn1").click(function(){
    alert("Text: " + $("#test").text());
  });
  $("#btn2").click(function(){
    alert("HTML: " + $("#test").html());
  });
});
</script>
```

```
</head>
<body>
```

```
<p id="test">This is some <b>bold</b> text in a paragraph.</p>
```

```
<button id="btn1">Show Text</button>
```

```
<button id="btn2">Show HTML</button>
```

This is some **bold** text in a paragraph.

Show Text

Show HTML

www.w3schools.com says

Text: This is some bold text in a paragraph.

OK

www.w3schools.com says

HTML: This is some bold text in a paragraph.

OK

jQuery – DOM (Get Content)

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    alert("Value: " + $("#test").val());
  });
});
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<p>Name: <input type="text" id="test" value="Mickey Mouse"></p>
```

```
<button>Show Value</button>
```

Name:

Show Value

www.w3schools.com says

Value: Mickey Mouse

OK

Name:

Show Value

www.w3schools.com says

Value: Web Technology Course

OK

jQuery – Set Content

Can use the same three methods of get content (text(), html() and val()) to set the content

```
<script>
$(document).ready(function(){
  $("#btn1").click(function(){
    $("#test1").text("Hello world!");
  });
  $("#btn2").click(function(){
    $("#test2").html("<b>Hello world!</b>");
  });
  $("#btn3").click(function(){
    $("#test3").val("Dolly Duck");
  });
});
</script>
</head>
<body>

<p id="test1">This is a paragraph.</p>
<p id="test2">This is another paragraph.</p>
<p>Input field: <input type="text" id="test3"
value="Mickey Mouse"></p>
<button id="btn1">Set Text</button>
<button id="btn2">Set HTML</button>
<button id="btn3">Set Value</button>
```

This is a paragraph.

This is another paragraph.

Input field:

Set Text

Set HTML

Set Value

This is a paragraph.

Hello world!

Input field:

Set Text

Set HTML

Set Value

Hello world!

This is another paragraph.

Input field:

Set Text

Set HTML

Set Value

This is a paragraph.

This is another paragraph.

Input field:

Set Text

Set HTML

Set Value

jQuery – Get Attributes

The jQuery **attr()** method is used to get/set attribute values.

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    alert($("#w3s").attr("href"));
  });
});
</script>
</head>
<body>

<p><a href="https://www.w3schools.com" id="w3s">W3Schools.com</a></p>

<button>Show href Value</button>
```

[W3Schools.com](https://www.w3schools.com)

Show href Value

www.w3schools.com says

https://www.w3schools.com

OK

jQuery – Set Attribute

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("#w3s").attr("href", "https://www.javatpoint.com/jquery-tutorial");
  });
});
</script>
</head>
<body>
```

```
<p><a href="https://www.w3schools.com" id="w3s">W3Schools.com</a></p>
```

```
<button>Change href Value</button>
```

Mouse over the link (or click on it) to see that the value of the href attribute has changed.</p>

[W3Schools.com](https://www.w3schools.com)

Change href Value

Mouse over the link (or click on it) to see that the value of the href attribute has changed.

```
$("button").click(function(){
  $("#w3s").attr({
    "href" : "https://www.w3schools.com/jquery/",
    "title" : "W3Schools jQuery Tutorial"
  });
});
```

jQuery – Call Back Function

All of the three jQuery methods: **text()**, **html()**, **val()**, and **attr()** come with a callback function.

The callback function has two parameters: the **index** of the current element in the list of elements selected and the **original (old) value**.

Return the string you wish to use as the new value from the function.

jQuery – Call Back Functions

```
<script>
$(document).ready(function(){
  $("#btn1").click(function(){
    $("#test1").text(function(i, origText){
      return "Old text: " + origText + " New text: Hello world! (index: " + i + ")";
    });
  });

  $("#btn2").click(function(){
    $("#test2").html(function(i, origText){
      return "Old html: " + origText + " New html: Hello <b>world!</b> (index: " + i +
    });
  });
});
</script>
</head>
<body>

<p id="test1">This is a <b>bold</b> paragraph.</p>
<p id="test2">This is another <b>bold</b> paragraph.</p>

<button id="btn1">Show Old/New Text</button>
<button id="btn2">Show Old/New HTML</button>
```

This is a **bold** paragraph.

This is another **bold** paragraph.

Show Old/New Text

Show Old/New HTML

Old text: This is a bold paragraph. New text: Hello world! (index: 0)

Old html: This is another **bold** paragraph. New html: Hello **world!** (index: 0)

Show Old/New Text

Show Old/New HTML

jQuery – Call Back Functions

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("#w3s").attr("href", function(i, origValue){
      return origValue + "/jquery/";
    });
  });
});
</script>
</head>
<body>

<p><a href="https://www.w3schools.com" id="w3s">W3Schools.com</a></p>

<button>Change href Value</button>

<p>Mouse over the link (or click on it) to see that the value of<br>
the href attribute has changed.</p>
```

W3Schools.com

Change href Value

Mouse over the link (or click on it) to see that the value of the href attribute has changed.

jQuery – Add Methods

jQuery methods that are used to add new content:

append() - Inserts content at the end of the selected elements

prepend() - Inserts content at the beginning of the selected elements

after() - Inserts content after the selected elements

before() - Inserts content before the selected elements

jQuery – append

```
<script>
$(document).ready(function(){
  $("#btn1").click(function(){
    $("p").append(" <b>Appended text</b>.");
  });

  $("#btn2").click(function(){
    $("ol").append("<li>Appended item</li>");
  });
});
</script>
</head>
<body>

<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
<ol>
  <li>List item 1</li>
  <li>List item 2</li>
  <li>List item 3</li>
</ol>
<button id="btn1">Append text</button>
<button id="btn2">Append list items</button>
```

This is a paragraph.

This is another paragraph.

1. List item 1
2. List item 2
3. List item 3

Append text

Append list items

This is a paragraph. **Appended text.**

This is another paragraph. **Appended text.**

1. List item 1
2. List item 2
3. List item 3
- 4. Appended item**
- 5. Appended item**

Append text

Append list items

jQuery – prepend

```
<script>
$(document).ready(function(){
  $("#btn1").click(function(){
    $("p").prepend(" <b>Prepended text</b>.");
  });

  $("#btn2").click(function(){
    $("ol").prepend("<li>Prepended item</li>");
  });
});
</script>
</head>
<body>

<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
<ol>
  <li>List item 1</li>
  <li>List item 2</li>
  <li>List item 3</li>
</ol>
<button id="btn1">Prepend text</button>
<button id="btn2">Prepend list items</button>
```

This is a paragraph.

This is another paragraph.

1. List item 1
2. List item 2
3. List item 3

Prepend text

Prepend list items

Prepended text.This is a paragraph.

Prepended text.This is another paragraph.

1. Prepended item
2. Prepended item
3. List item 1
4. List item 2
5. List item 3

Prepend text

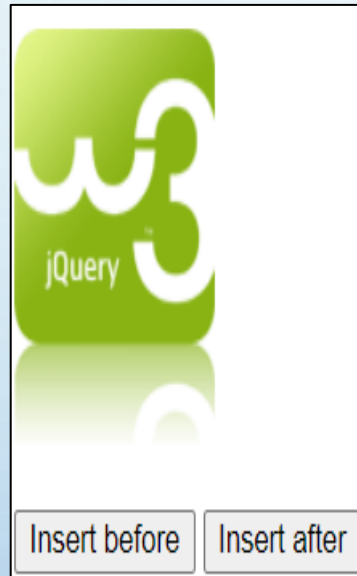
Prepend list items

jQuery – before and after

```
<script>
$(document).ready(function(){
  $("#btn1").click(function(){
    $("img").before("<b>Before</b>");
  });

  $("#btn2").click(function(){
    $("img").after("<i>After</i>");
  });
});
</script>
</head>
<body>
<br><br>

<button id="btn1">Insert before</button>
<button id="btn2">Insert after</button>
```



jQuery – Remove Elements

To remove elements and content, there are mainly two jQuery methods:

remove() - Removes the selected element (and its child elements)

empty() - Removes the child elements from the selected element

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("#div1").remove();
  });
});
</script>
</head>
<body>

<div id="div1" style="height:100px;width:300px;
border:1px solid black;background-color:yellow;">
```

This is some text in the div.

<p>This is a paragraph in the div.</p>

<p>This is another paragraph in the div.</p>

</div>

This is out of div para

<button>Remove div element</button>

This is some text in the div.

This is a paragraph in the div.

This is another paragraph in the div.

This is out of div para

Remove div element

This is out of div para

Remove div element

jQuery – Empty

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("#div1").empty();
  });
});
</script>
</head>
<body>

<div id="div1" style="height:100px;width:300px;
border:1px solid black;background-color:yellow;">

This is some text in the div.
<p>This is a paragraph in the div.</p>
<p>This is another paragraph in the div.</p>

</div>
This is out of div para<br>

<button>Empty the div element</button>
```

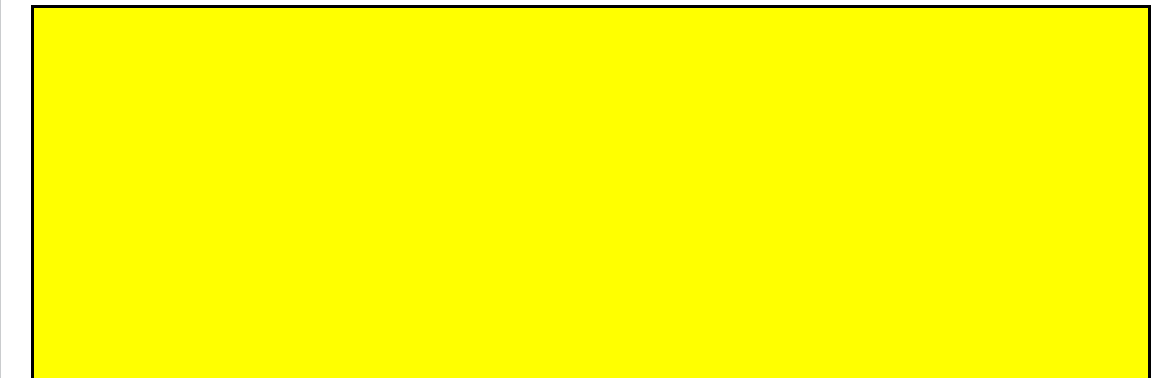
This is some text in the div.

This is a paragraph in the div.

This is another paragraph in the div.

This is out of div para

Empty the div element



This is out of div para

Empty the div element

jQuery – Remove with Filter

The jQuery **remove()** method also accepts one parameter, which allows to **filter** the elements to be removed. The parameter can be any of the jQuery selector syntaxes.

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("p").remove(".test");
  });
});
</script>

</head>
<body>

<p>This is a paragraph.</p>
<p class="test">This is paragraph with id test.</p>
<p class="test">This is another paragraph with id test.</p>

<button>Remove all p elements with class="test"</button>
```

This is a paragraph.

This is paragraph with id test.

This is another paragraph with id test.

Remove all p elements with class="test"

This is a paragraph.

Remove all p elements with class="test"

jQuery – Form (Serialize)

jQuery **serialize()** method is used to create a text string in standard URL-encoded notation. It is used in form controls like **<input>**, **<textarea>**, **<select>**, etc. It serializes the form values so that its serialized values can be used in the URL query string while making an AJAX requests.

Syntax: \$(selector).serialize()

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("div").text($("#form").serialize());
    });
});
</script>
</head>
<body>
<form action="">
    First name: <input type="text" name="FirstName" value="Sonoo"><br>
    Last name: <input type="text" name="LastName" value="Jaiswal"><br>
</form>
<button>Serialize form values</button>
<div></div>
```

First name:

Last name:

First name:

Last name:

FirstName=Sonoo&LastName=Jaiswal

First name:

Last name:

FirstName=VIT&LastName=University

jQuery – Form (SerializeArray)

The jQuery **serializeArray()** Method is used to create a JavaScript array of objects by serializing form values. It operates on a jQuery collection of forms and form controls. You can select one or more form elements such as `<input>`, `<textarea>` or the form element itself.

jQuery – serializeArray

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        var x = $("form").serializeArray();
        $.each(x, function(i, field){
            $("#results").append(field.name + ":" + field.value + " ");
        });
    });
});
</script>
</head>
<body>
<form action="">
    First name: <input type="text" name="FirstName" value="Ajeet"><br>
    Last name: <input type="text" name="LastName" value="Maurya"><br>
    Gender: <input type="radio" name="gen" value="Male"> Male
    <input type="radio" name="gen" value="Female"> Female
    <input type="radio" name="gen" value="Transgender"> Transgender<br><br>
    Qualification: <select name="qualification">
        <option value="Diploma">Diploma </option>
        <option value="UG">UG </option>
        <option value="PG">PG </option>
        <option value="Ph.D">Ph. D. </option>
        <option value="PDF">PDF </option>
    </select><br><br>
</form>
<button>Serialize form values</button>
<div id="results"></div>
```

First name:

Last name:

Gender: ☐ Male ☒ Female ☐ Transgender

Qualification:

Serialize form values

FirstName:Ajeet LastName:Maurya gen:Female qualification:UG

jQuery - Animation

animate() - The jQuery animate() method is used to create custom animations. it moves a <div> element to the right, until it has reached a left property of 250px:

```
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/
jquery.min.js"></script>
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("div").animate({left: '250px'});
  });
});
</script>
</head>
<body>
```

Start Animation

By default, all HTML elements have a static position, and cannot be moved. To manipulate the position, remember to first set the CSS position property of the element to relative, fixed, or absolute.



```
<button>Start Animation</button>
```

By default, all HTML elements have a static position, and cannot be moved. To manipulate the position, remember to first set the CSS position property of the element to relative, fixed, or absolute!

```
<div style="background:#98bf21;height:100px;width:100px;position:absolute;"></div>
```

```
</body>
</html>
```

JQUERY – ANIMATION

MANIPULATE MULTIPLE PROPERTIES

notice that multiple properties can be animated at the same time:

```
$("#button").click(function(){  
  $("#div").animate({  
    left: '250px',  
    opacity: '0.5',  
    height: '150px',  
    width: '150px'  
  });  
});
```

```
<html>  
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/  
jquery.min.js"></script>  
<script>  
$(document).ready(function(){  
  $("#button").click(function(){  
    $("#div").animate({  
      height: 'toggle'  
    });  
  });  
});  
</script>  
</head>  
<body>
```

<p>Click the button multiple times to toggle the animation.</p>

<button>Start Animation</button>

<p>By default, all HTML elements have a static position, and cannot be moved. To manipulate the position, remember to first set the CSS position property of the element to relative, fixed, or absolute!</p>