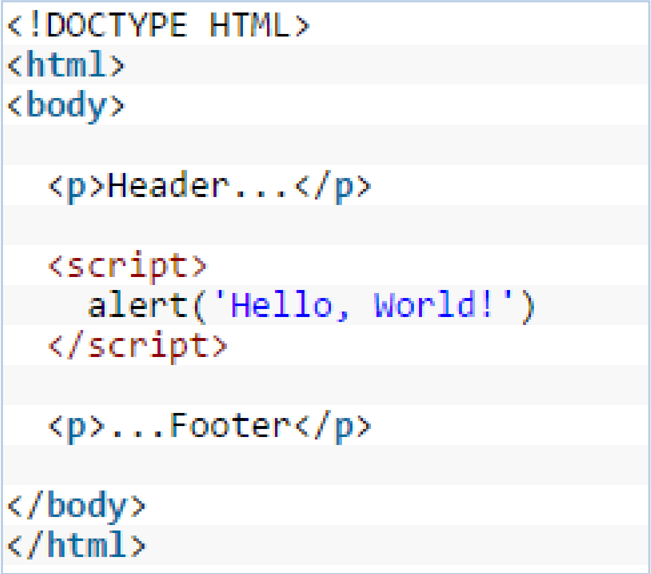
Client side

JAVASCRIPT



```
<!DOCTYPE HTML>
<html>
<body>

  <p>Header...</p>

  <script>
    alert('Hello, World!')
  </script>

  <p>...Footer</p>

</body>
</html>
```

- Javascript is
  - A scripting programming language
    - client side
    - object oriented
    - interpreted not compiled
    - untyped (var x = 1, var y = 'Hello')
  - Executed in the web browser
  - Allows you to make *interactive* web pages
  - Supported (*understood*) by the vast majority of browsers

- **Do not confuse Java and JavaScript !!!**

Java ≠ JavaScript

JQUERY

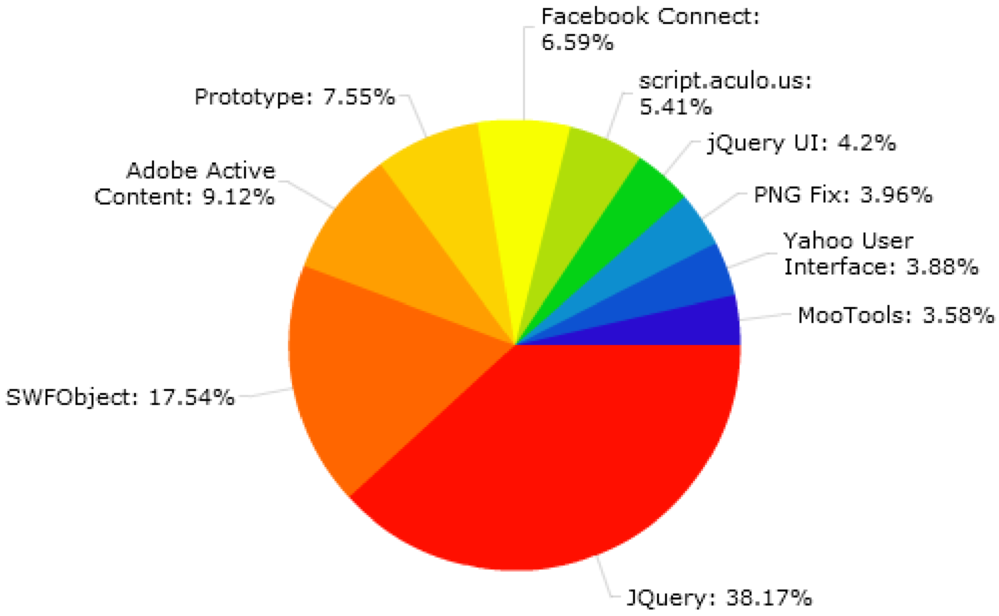
- JQuery is a JavaScript library.
- The goal of jQuery is to make it much easier to use JavaScript on your site
  - Simplification of orders
    - For the same task
      - javascript line number >>> jQuery line number
  - Simplification of DOM manipulation
  - Simplifying AJAX calls

JQUERY

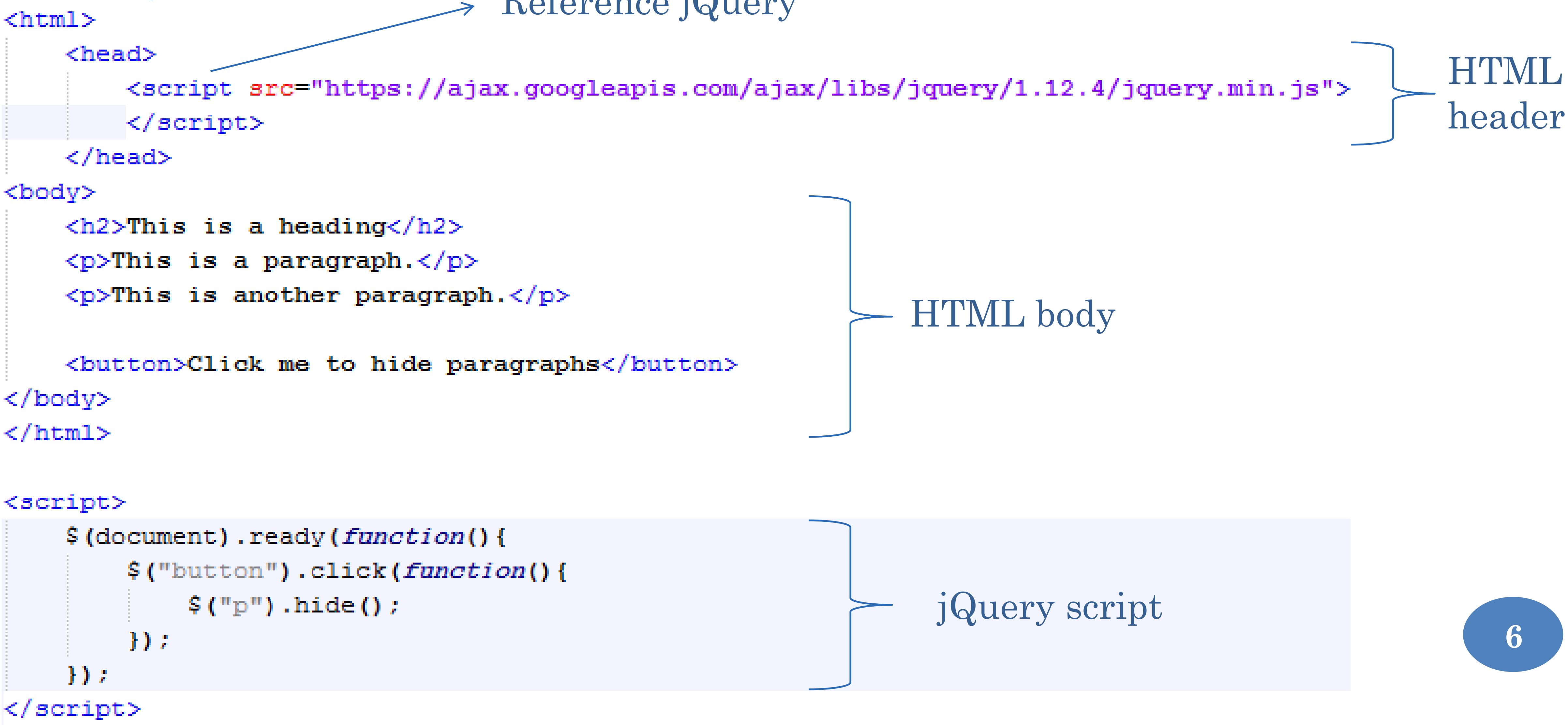
- The jQuery library contains the following features:
  - HTML / DOM manipulation
  - CSS manipulation
  - HTML event methods
  - effects and animations
  - AJAX (*Asynchronous Javascript and Xml)*
  - …

JQUERY

- Why jQuery?
  - Simple, easy to learn
  - Open source
  - The most popular
  - The most extensible
  - Multi-browser (cross-browsers)
  - Up-to-date documentation
  - Used by many of the largest companies
    - Google
    - Microsoft
    - IBM
    - Netflix
    - …



JQUERY



Reference jQuery

```html
<html>
    <head>
        <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js">
        </script>
    </head>
<body>
    <h2>This is a heading</h2>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>

    <button>Click me to hide paragraphs</button>
</body>
</html>

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

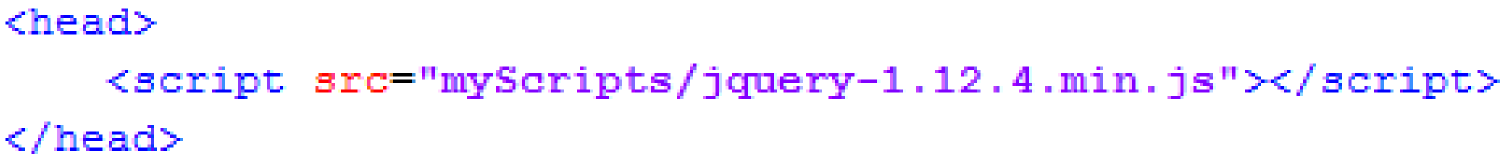
HTML header

HTML body

jQuery script

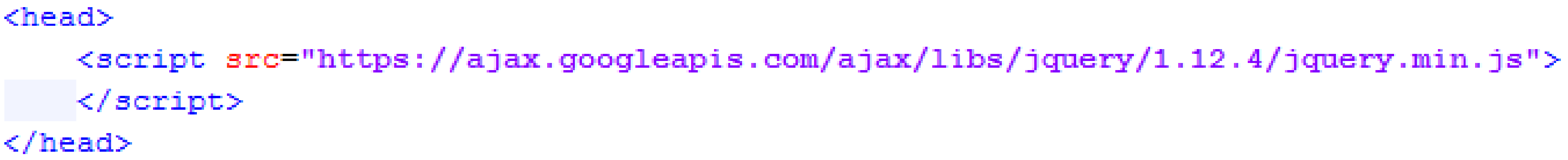
JQuery

- The reference to jQuery is placed in the 'header' of an HTML page
  - Can be downloaded from jQuery.com and placed in the project directory (ie / myScripts / ...)



```
<head>
    <script src="myScripts/jquery-1.12.4.min.js"></script>
</head>
```

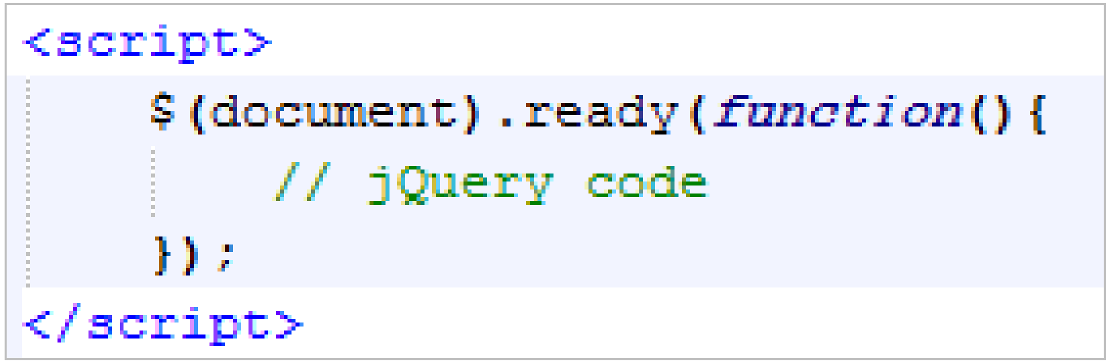
  - Can be included (as a reference) from a CDN (google, microsoft, ...)



```
<head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js">
    </script>
</head>
```

JQUERY

- The jQuery code
  - Starts with an opening tag <script> and ends with the closing tag </ script>
  - Can be included anywhere in an HTML page, preferably at the end
  - Should be placed after the *$ (document) .ready ()* *event* to prevent the code from running before the page finishes loading



```
<script>
    $(document).ready(function(){
        // jQuery code
    });
</script>
```

JQUERY

- Tools
  - Any text editor
    - notepad, **notepad ++** , dreamweaver, NetBeans, ...
- File extension
  - js
- Online reference:
  - https://jquery.com/
  - http://www.w3schools.com
  - https://www.udemy.com/jquery/
  - …

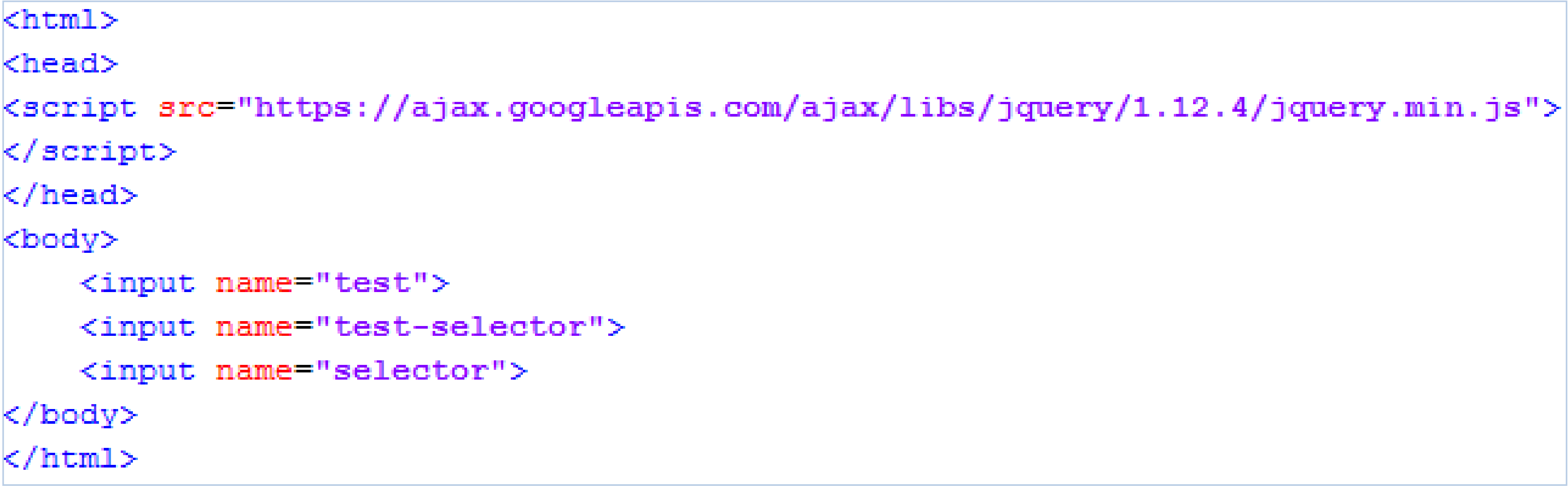
jQuery - Selectors

- JQuery selectors are used to "find" (or select) HTML elements based on their:
  - name,
  - type
  - identifier (id),
  - class
  - attributes,
  - …
- All jQuery selectors start with the $ sign and parentheses: $ ()

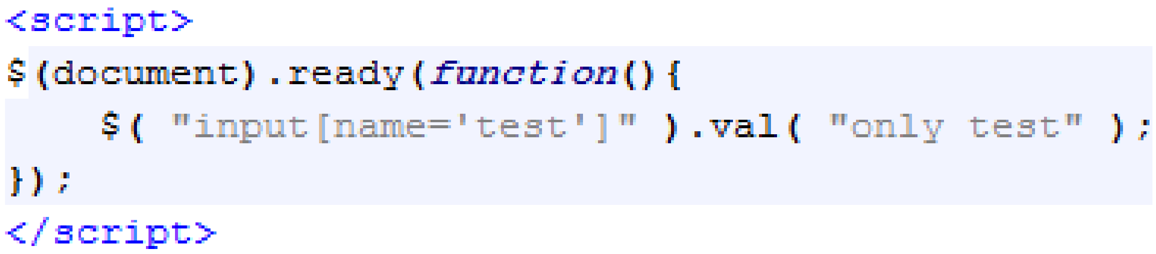
jQuery - Selectors

- Name
  - $ ("input [name = 'test']")
    - Returns all entries (html input) whose name 'test‘
  - $ ("input [name * = 'test']")
    - Returns all entries (html input) whose name contains: 'test'

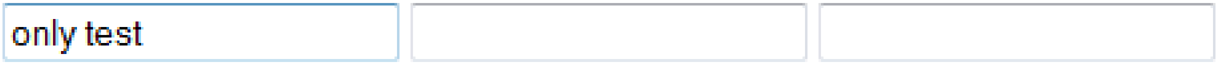
JQUERY - SELECTORS



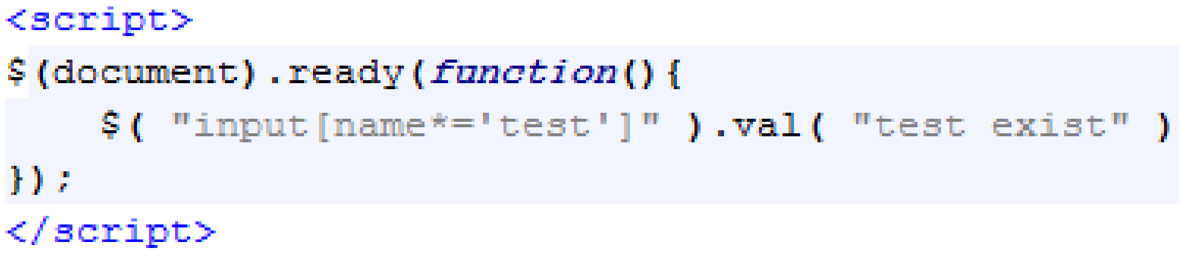
```
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js">
</script>
</head>
<body>
    <input name="test">
    <input name="test-selector">
    <input name="selector">
</body>
</html>
```



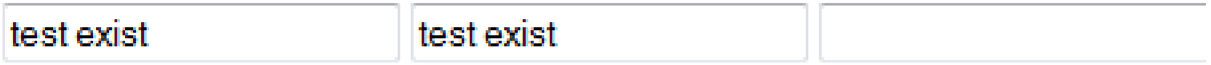
```
<script>
$(document).ready(function(){
    $( "input[name='test']" ).val( "only test" );
});
</script>
```



only test



```
<script>
$(document).ready(function(){
    $( "input[name*='test']" ).val( "test exist" )
});
</script>
```

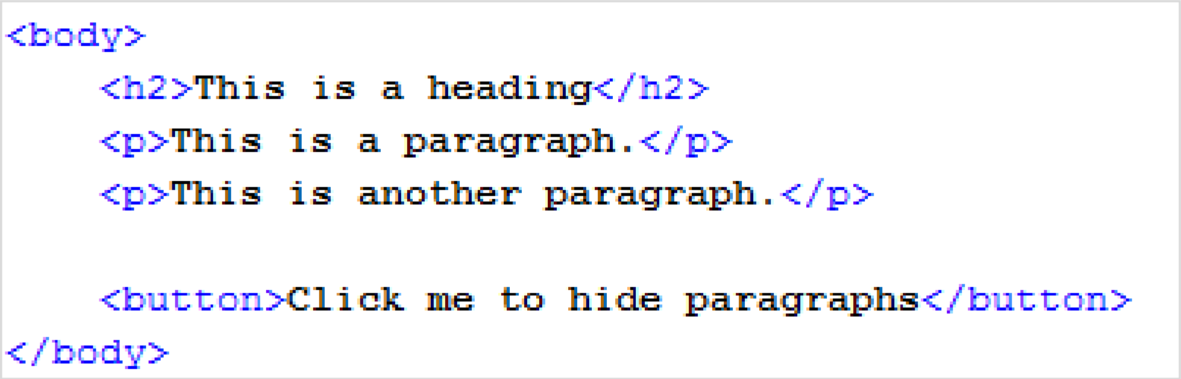


test exist

test exist

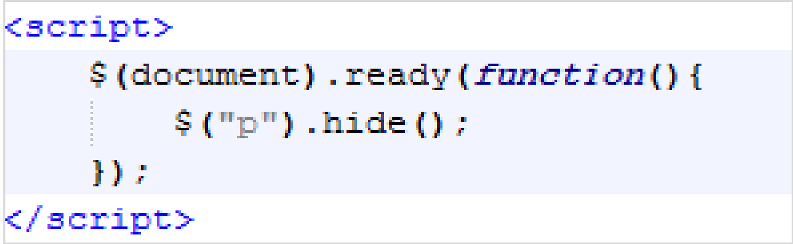
JQUERY - SELECTORS

- Type
  - Returns all elements of a certain type, ie:
    - $ ("p"): return all elements of type p
    - $ ("H1"): returns all elements of type H1
    - $ ("label"): returns all elements of the type label
    - ...



```
<body>
    <h2>This is a heading</h2>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>

    <button>Click me to hide paragraphs</button>
</body>
```

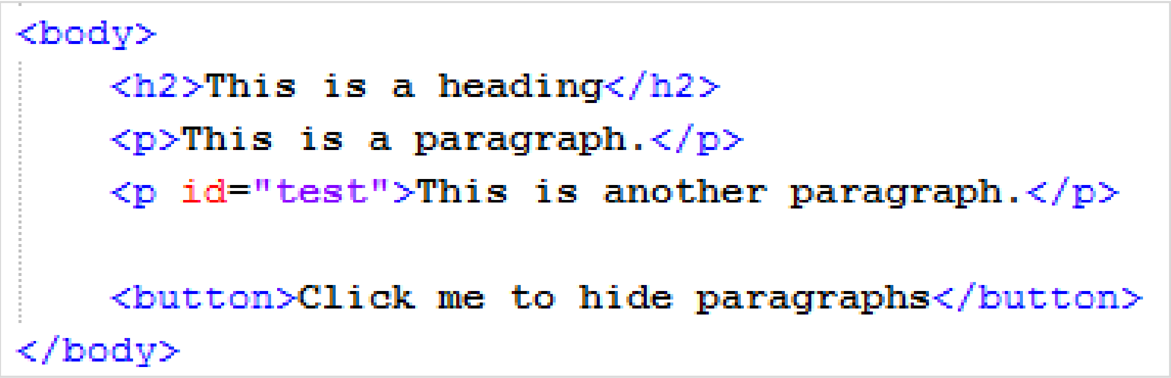


```
<script>
    $(document).ready(function(){
        $("p").hide();
    });
</script>
```

After running the script, the 2nd and 3rd elements will be invisible

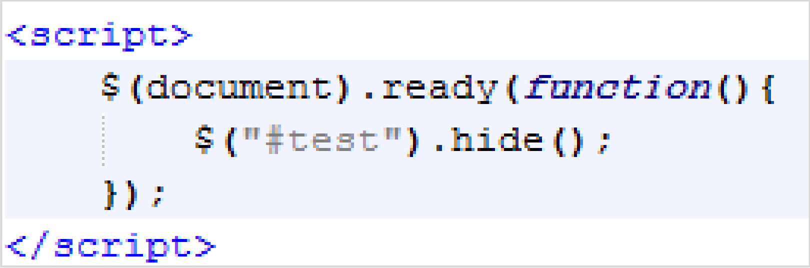
JQUERY - SELECTORS

- Identifier *$("#Id")*
  - Returns an element with a given identifier
    - $ ("# test"): returns the element having 'test' as Id
    - an identifier must be unique per element



```
<body>
    <h2>This is a heading</h2>
    <p>This is a paragraph.</p>
    <p id="test">This is another paragraph.</p>

    <button>Click me to hide paragraphs</button>
</body>
```

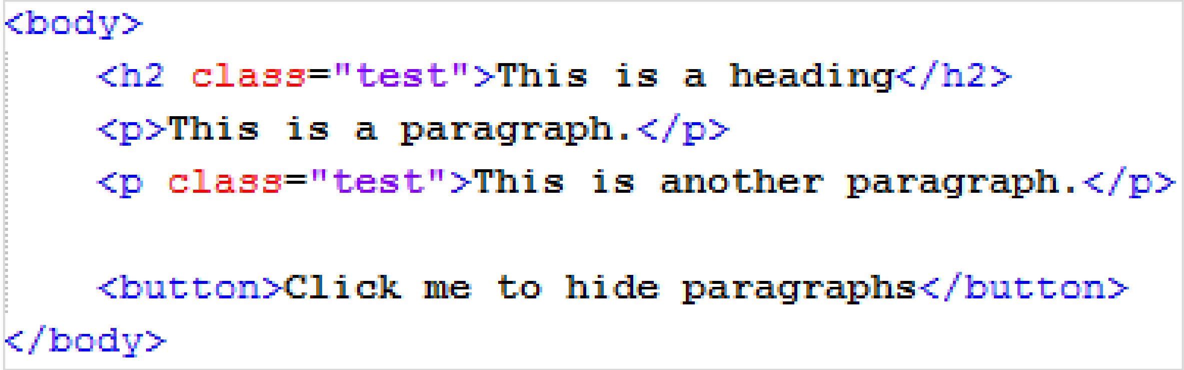


```
<script>
    $(document).ready(function(){
        $("#test").hide();
    });
</script>
```

After running the script, only the 3rd elements will be invisible

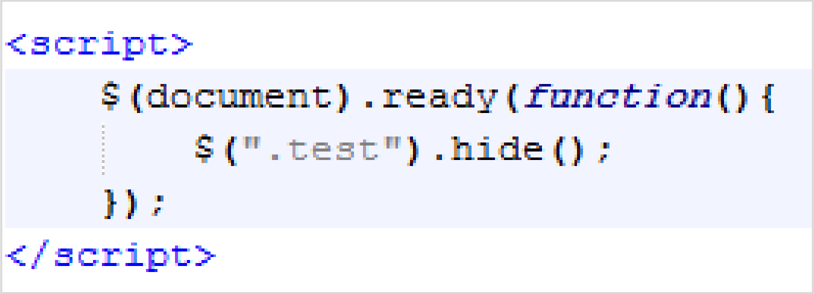
JQuery - Selectors

- *$* Classes (*“.test"*)
  - Returns all items with a given class
    - *$* (*“.test"*) : returns elements with the class 'test'



```
<body>
    <h2 class="test">This is a heading</h2>
    <p>This is a paragraph.</p>
    <p class="test">This is another paragraph.</p>

    <button>Click me to hide paragraphs</button>
</body>
```



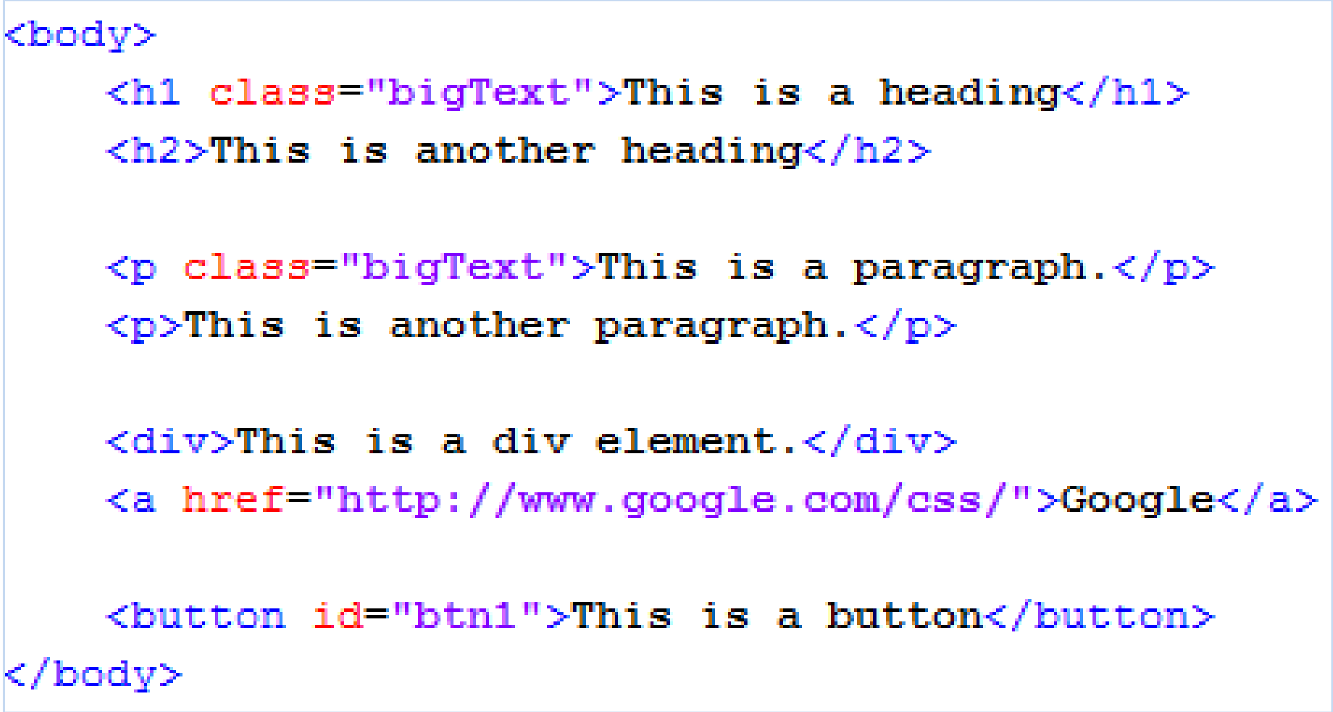
```
<script>
    $(document).ready(function(){
        $(".test").hide();
    });
</script>
```

After running the script,
1st and 3rd elements will be
invisible

JQUERY - SELECTORS

- Attributes
  - $ ("*"): all elements of the HTML page
  - $ ("p: first"): the first element P
  - $ ("ul li: first"): the first element 'li' of the first list 'ul'
  - $ ("ul li: first-child"): the first element 'li' of all the lists
  - $ ("[href]"): the links
  - ...

JQUERY – EXERCICE 1

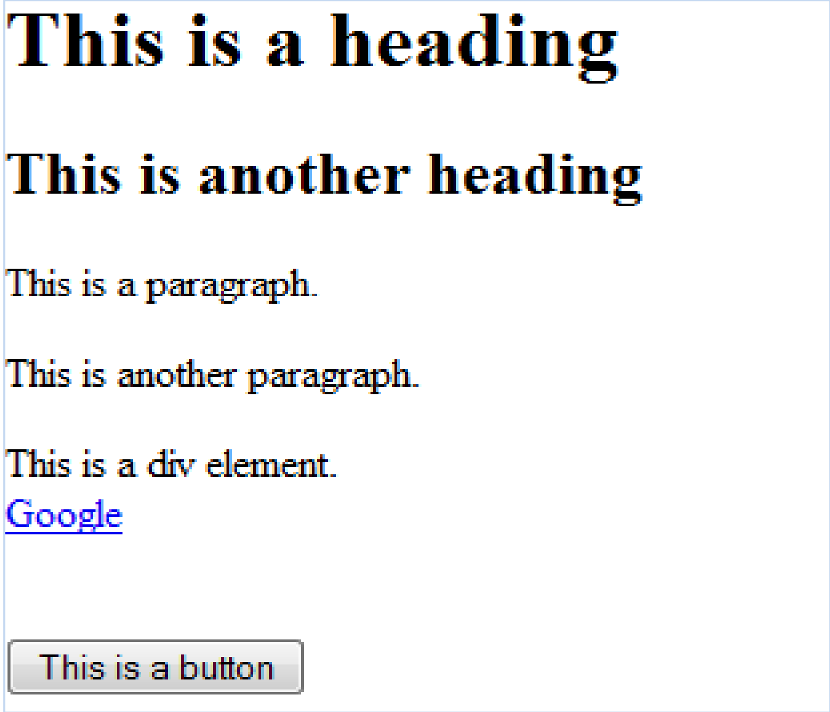


```
<body>
    <h1 class="bigText">This is a heading</h1>
    <h2>This is another heading</h2>

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

    <div>This is a div element.</div>
    <a href="http://www.google.com/css/">Google</a>

    <button id="btn1">This is a button</button>
</body>
```



This is a heading

This is another heading

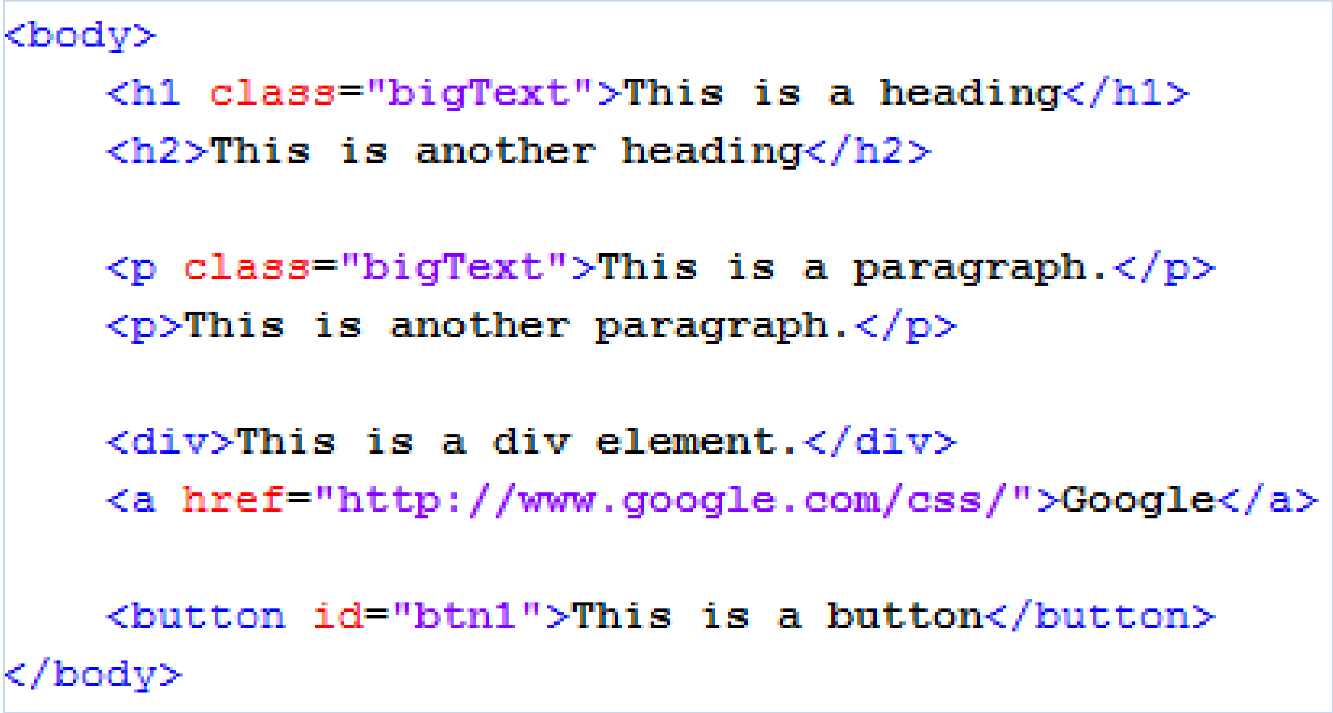
This is a paragraph.

This is another paragraph.

This is a div element.
Google

- Write the script that allows the click on 'btn1' button to:
  1. Hide all p items
  2. Hide all div elements
  3. Hide all items that have as 'bigText' class
  4. Hide all links (href)
  5. Hide all elements of the page

JQuery – Exercice 1

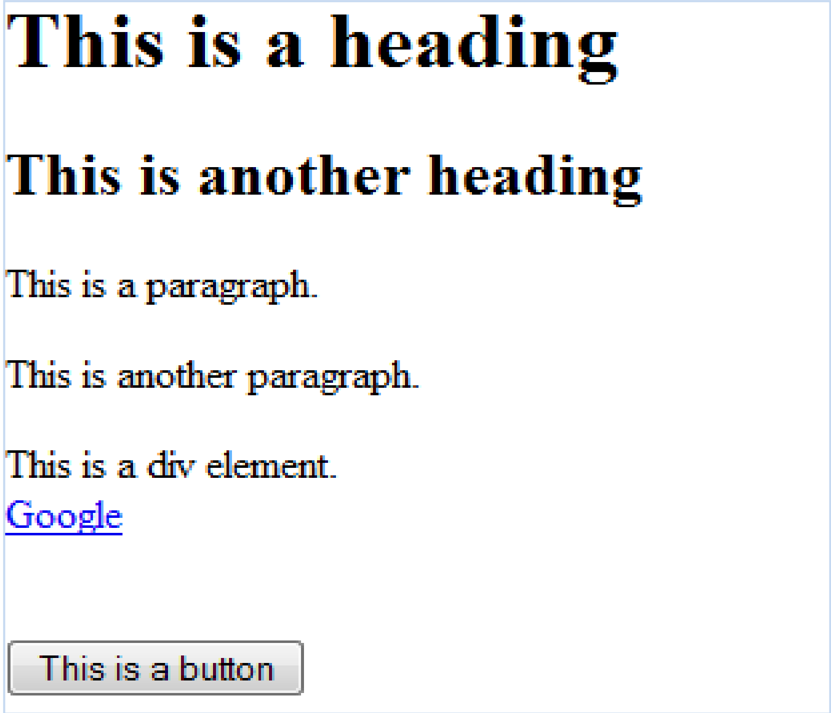


```html
<body>
    <h1 class="bigText">This is a heading</h1>
    <h2>This is another heading</h2>

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

    <div>This is a div element.</div>
    <a href="http://www.google.com/css/">Google</a>

    <button id="btn1">This is a button</button>
</body>
```



This is a heading

This is another heading

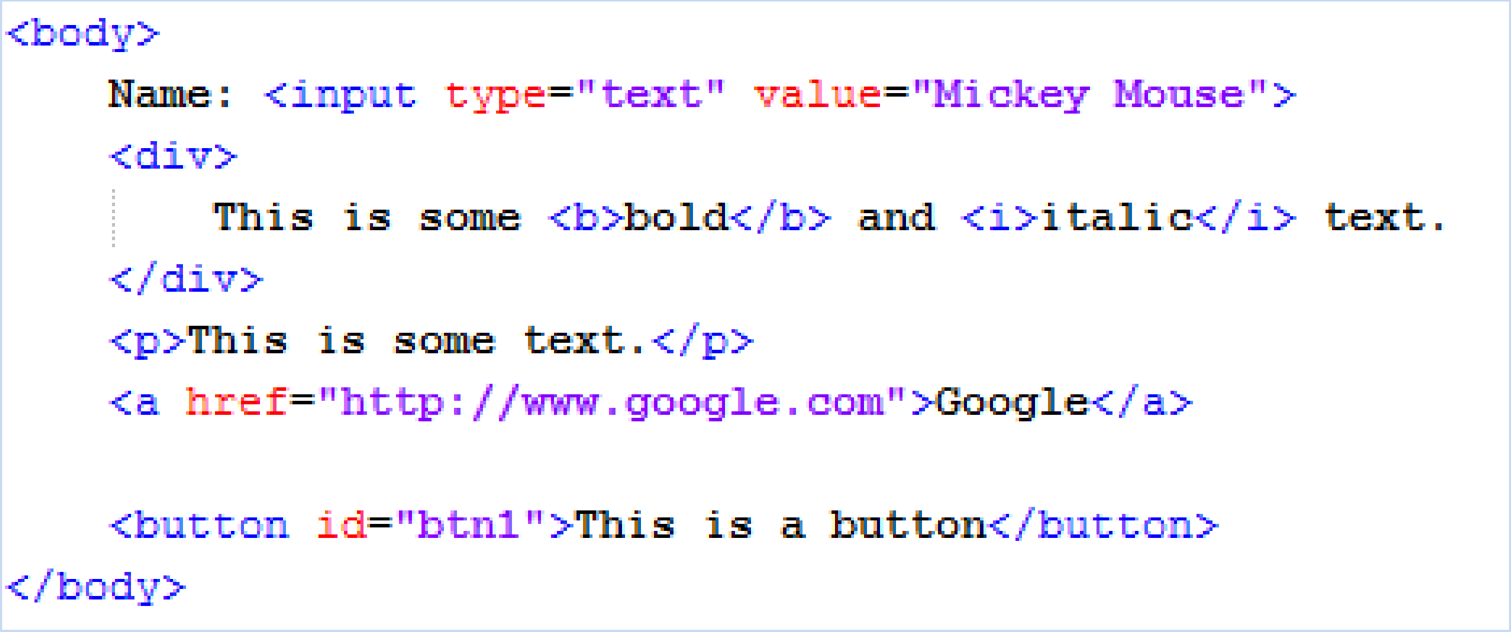
This is a paragraph.

This is another paragraph.

This is a div element.
Google

```
<script>
$(document).ready(function(){
    $("#btn1").click(function(){
            $("p").hide();          // 1
            $("div").hide();        // 2
            $(".bigText").hide();   // 3
            $("[href]").hide();     // 4
            $("*").hide();          // 5
    });
}); </script>
```

jQuery – Exercice 2

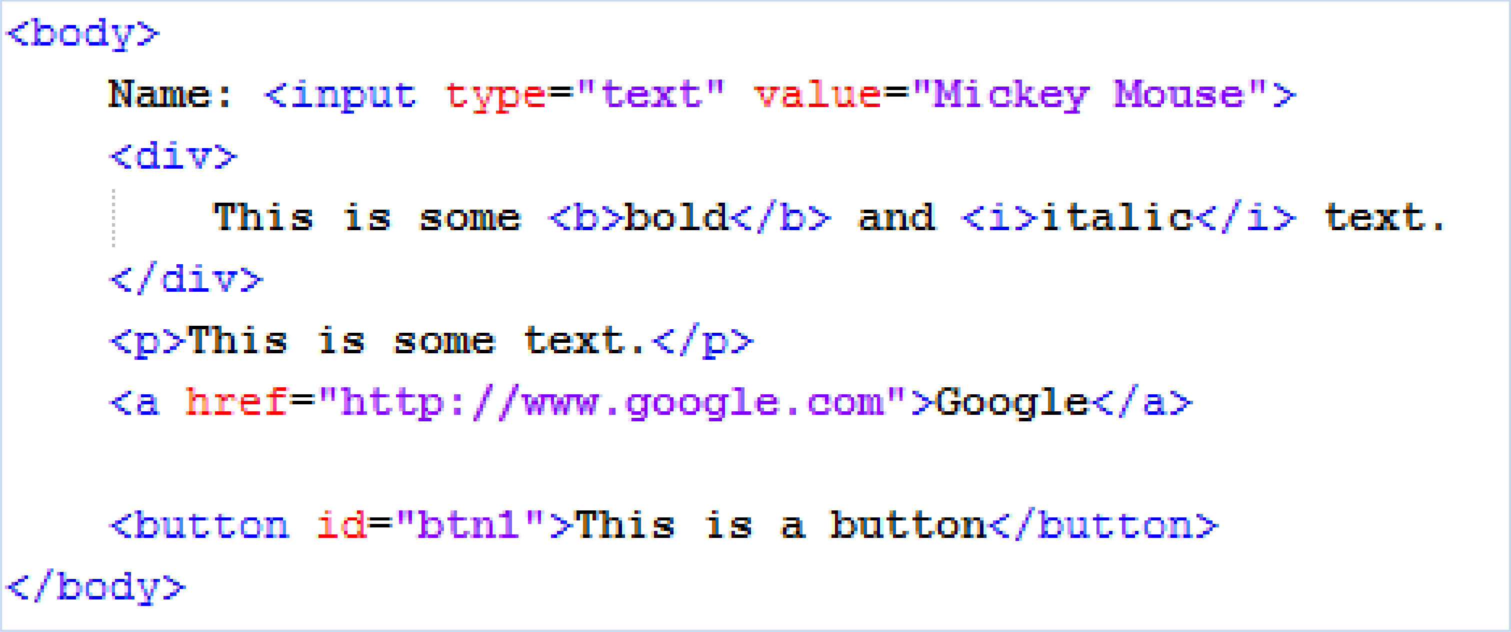


```
<body>
    Name: <input type="text" value="Mickey Mouse">
    <div>
        This is some <b>bold</b> and <i>italic</i> text.
    </div>
    <p>This is some text.</p>
    <a href="http://www.google.com">Google</a>

    <button id="btn1">This is a button</button>
</body>
```

- Which result produces the following operations:
  - $("input").val();
  - $("div").html();
  - $("p").text();
  - $("a").attr("href");

JQUERY – EXERCICE 2



```
<body>
    Name: <input type="text" value="Mickey Mouse">
    <div>
        This is some <b>bold</b> and <i>italic</i> text.
    </div>
    <p>This is some text.</p>
    <a href="http://www.google.com">Google</a>

    <button id="btn1">This is a button</button>
</body>
```

- Which result produces the following operations:
  - $("input").val(); → Mickey Mouse
  - $("div").html(); → This is some <b>bold</b> and <i>italic</i> text.
  - $("p").text(); → This is some text.
  - $("a").attr("href"); → http://www.google.com

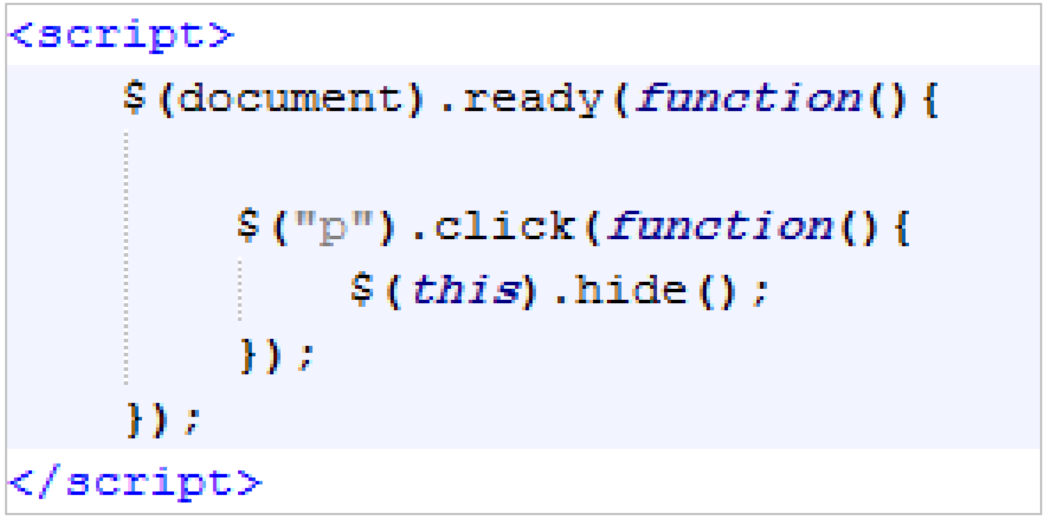
JQUERY - EVENTS

- All actions performed by a user that a browser can process, in a web page, triggers events

- Examples:
  - moving a mouse on an element
  - the selection of a radio button
  - click on an item
  - …

JQUERY - EVENTS

- $(document).ready()
  - allows you to execute a function when the document is fully loaded
- click()
  - The function is executed when the user clicks on the HTML element.

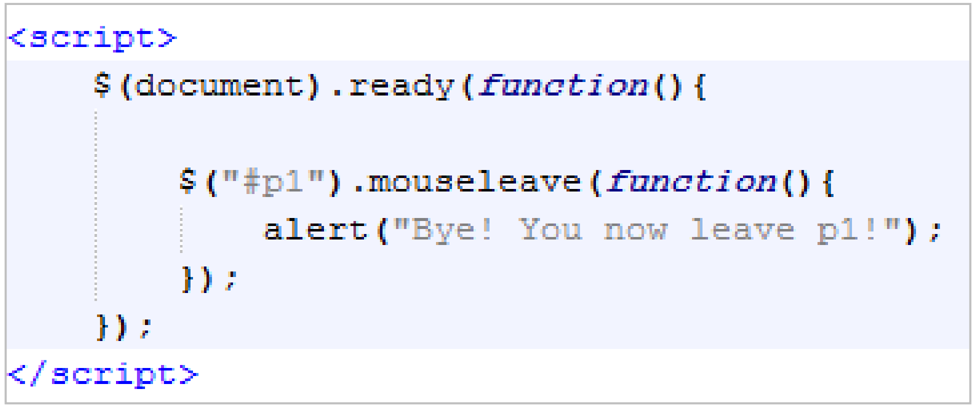


```
<script>
    $(document).ready(function(){

        $("p").click(function(){
            $(this).hide();
        });
    });
</script>
```

JQUERY - EVENTS

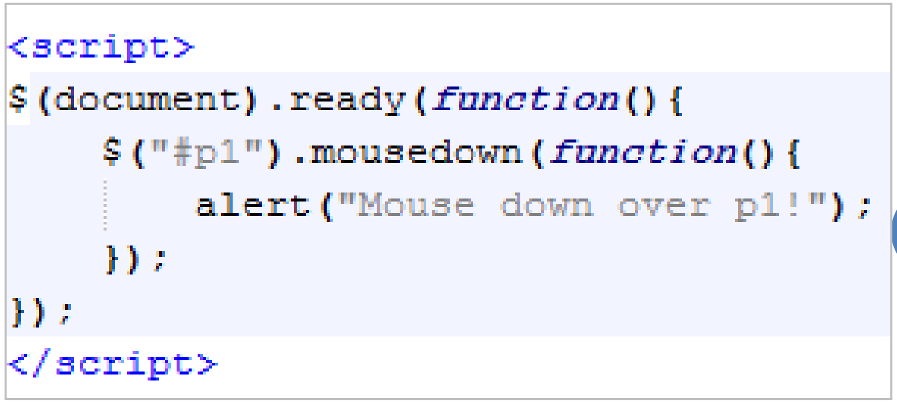
- mouseenter() / mouseleave()
  - executed when the mouse pointer enters (leaves) in the HTML element



```
<script>
    $(document).ready(function(){

        $("#p1").mouseleave(function(){
            alert("Bye! You now leave p1!");
        });
    });
</script>
```

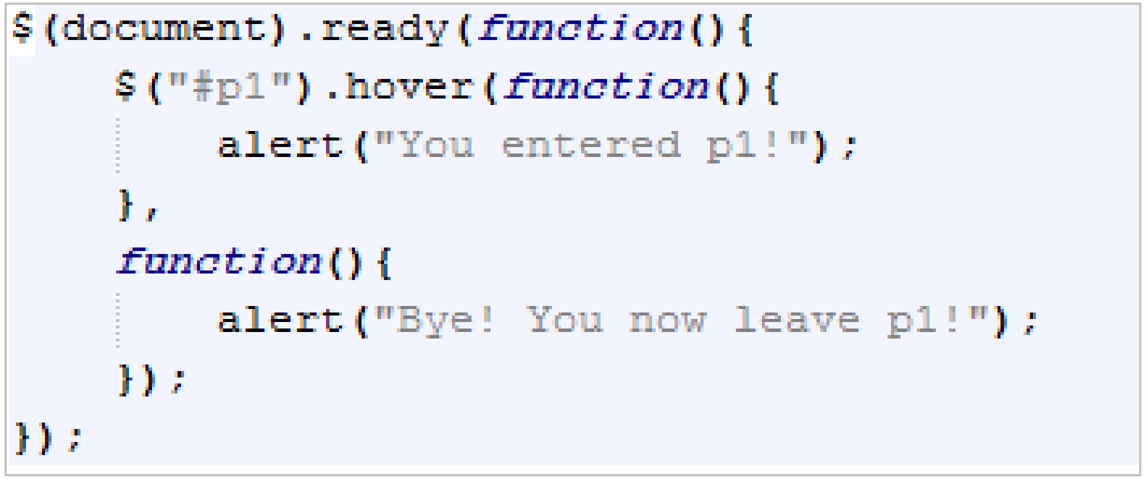
- mousedown()/mouseup()
  - executed when the right, left or middle mouse button is pressed down (released)



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

JQUERY - EVENTS

- hover()
  - combination of the mouseenter () and mouseleave () methods
    - the first function is executed when the mouse enters the HTML element,
    - and the second function is executed when the mouse leaves the HTML element
- …



```
$(document).ready(function(){
    $("#p1").hover(function(){
        alert("You entered p1!");
    },
    function(){
        alert("Bye! You now leave p1!");
    });
});
```

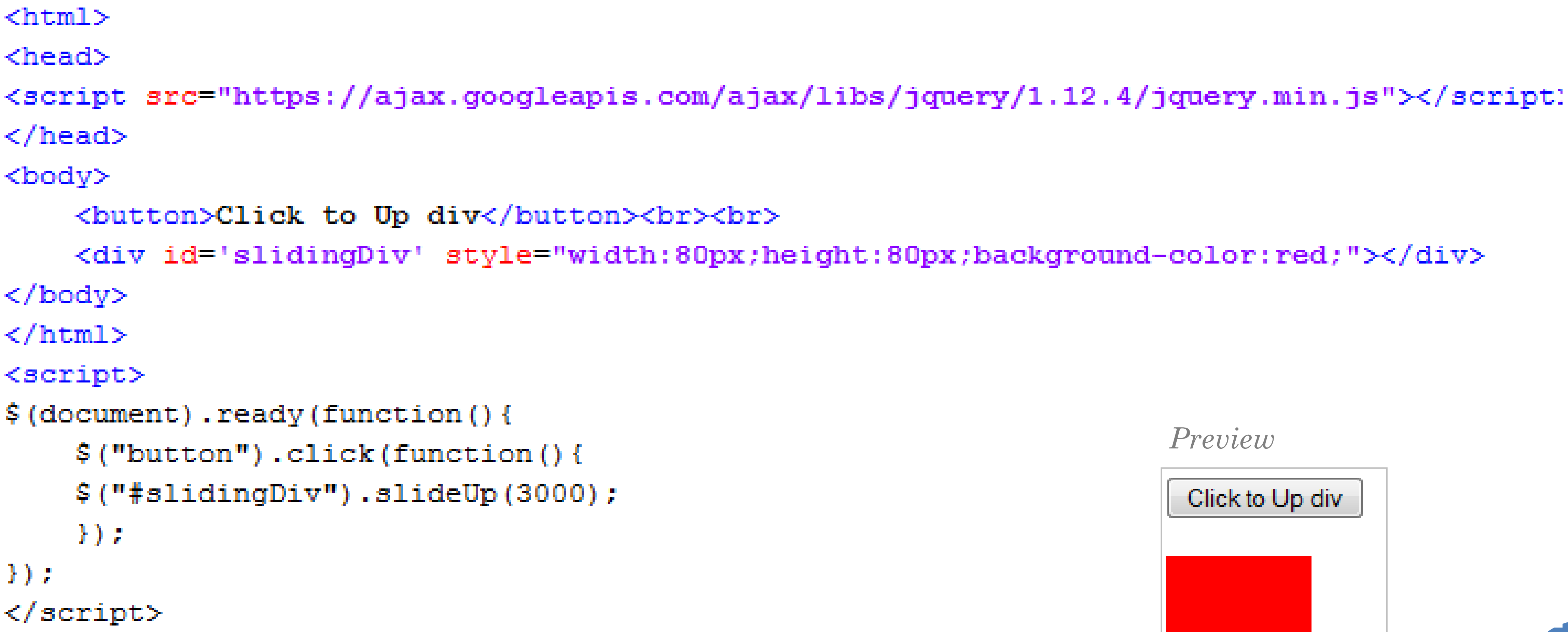
JQUERY – EFFECTS

- hide()
  - $(*selector*).hide(*speed,callback*);
- show()
  - $(*selector*).show(*speed,callback*);
- toggle()
  - $(*selector*).toggle(*speed,callback*);
- fadeIn()
  - $(*selector*).fadeIn(*speed,callback*);
- fadeOut()
  - $(*selector*).fadeOut(*speed,callback*);

JQUERY – EFFECTS

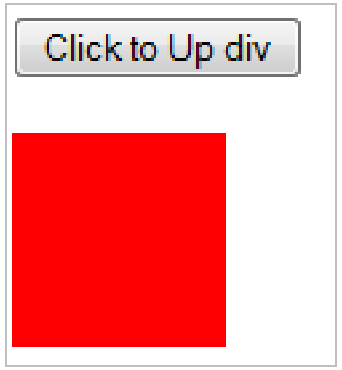
- slideDown()
    - $(*selector*).slideDown(*speed,callback*);
- slideUp()
    - $(*selector*).slideUp(*speed,callback*);
- animate()
    - $(*selector*).animate({*params*},*speed,callback*);
- …

JQUERY – EXERCICE 3



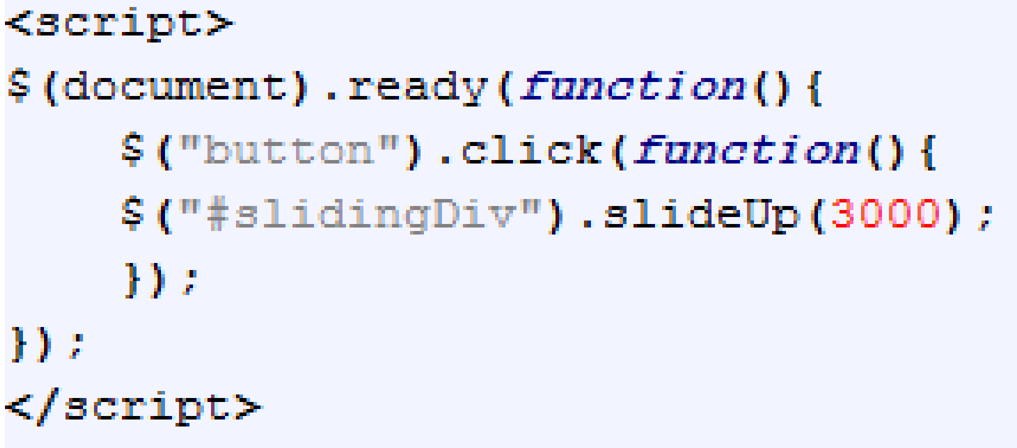
```
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script
</head>
<body>
    <button>Click to Up div</button><br><br>
    <div id='slidingDiv' style="width:80px;height:80px;background-color:red;"></div>
</body>
</html>
<script>
$(document).ready(function(){
    $("button").click(function(){
    $("#slidingDiv").slideUp(3000);
    });
});
</script>
```

Preview



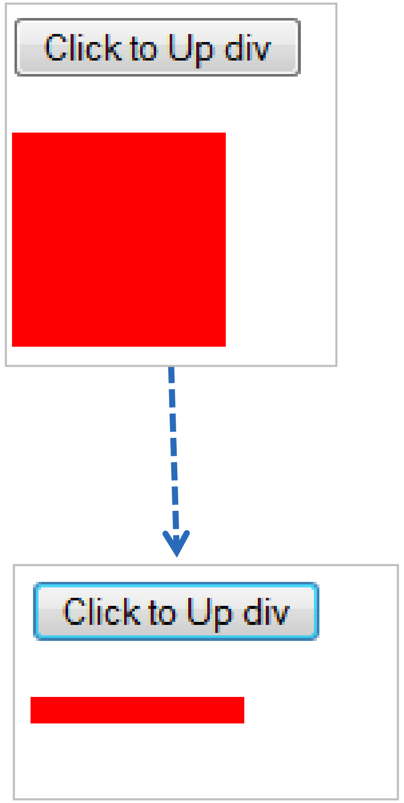
What result does this script produce?

JQUERY – EXERCICE 3



```
<script>
$(document).ready(function(){
    $("button").click(function(){
    $("#slidingDiv").slideUp(3000);
    });
});
</script>
```

Preview

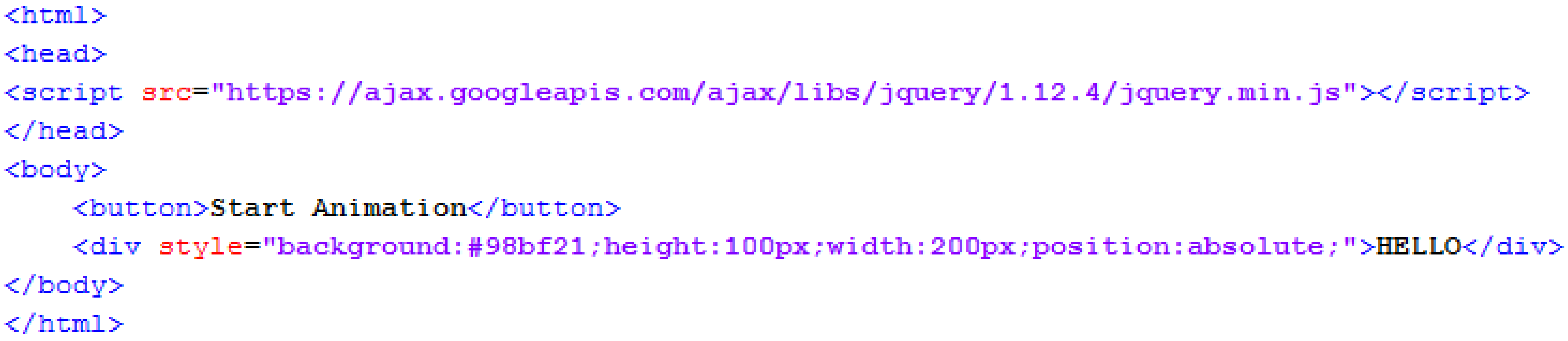


The slideUp method has the definition:

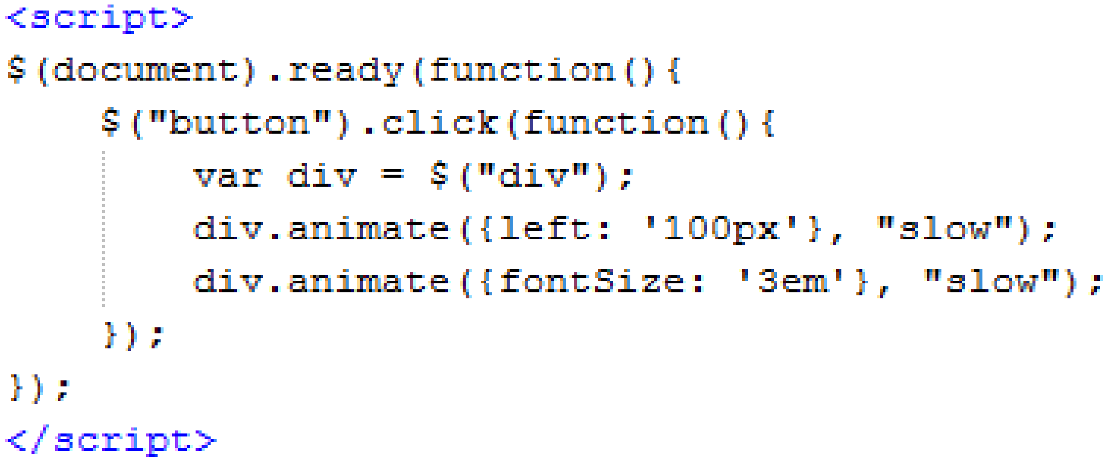
$(*selector*).slideUp(*speed*);

In that case,
selector : div having as Id: #slidingDiv
speed : 3000 represents the speed of the animation
The click on the button flushes the event and div glides upwards

JQUERY – EXERCICE 4

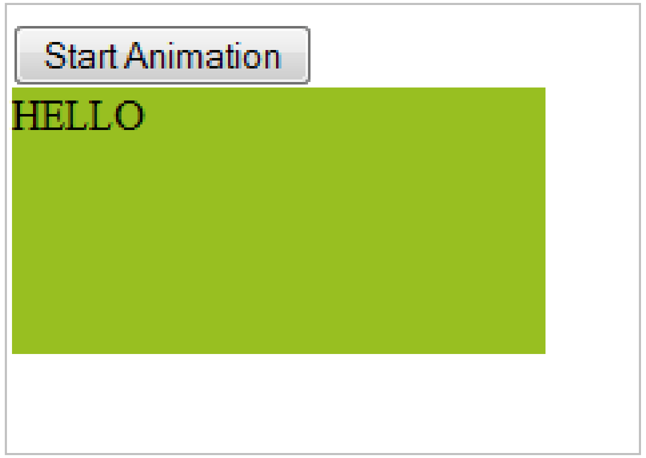


```
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
</head>
<body>
    <button>Start Animation</button>
    <div style="background:#98bf21;height:100px;width:200px;position:absolute;">HELLO</div>
</body>
</html>
```



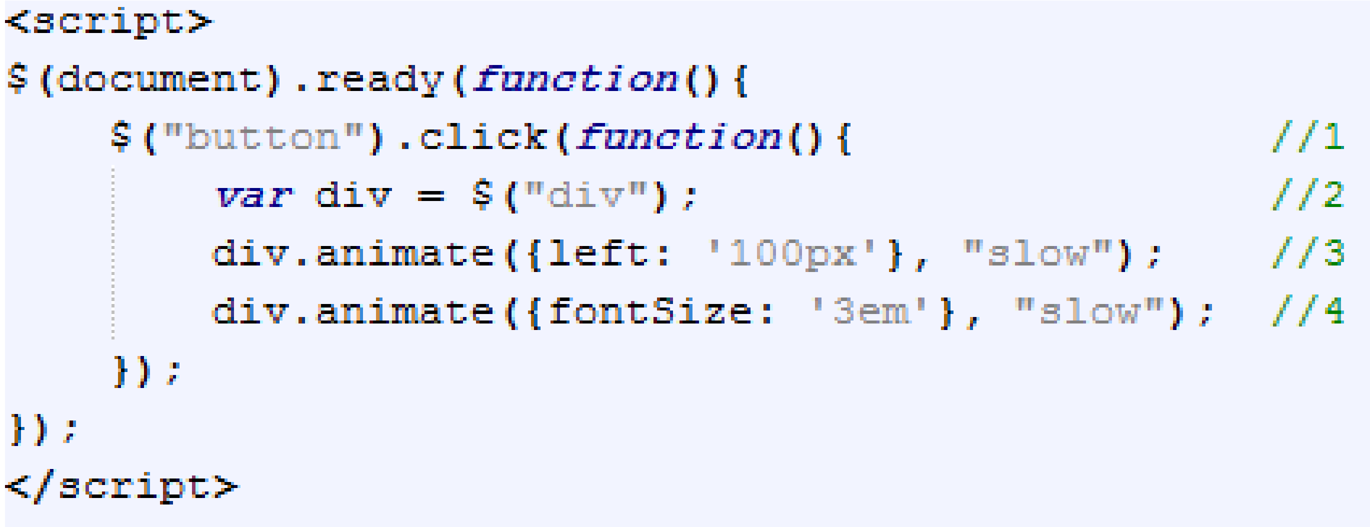
```
<script>
$(document).ready(function(){
    $("button").click(function(){
        var div = $("div");
        div.animate({left: '100px'}, "slow");
        div.animate({fontSize: '3em'}, "slow");
    });
});
</script>
```

Preview



What result does this script produce?

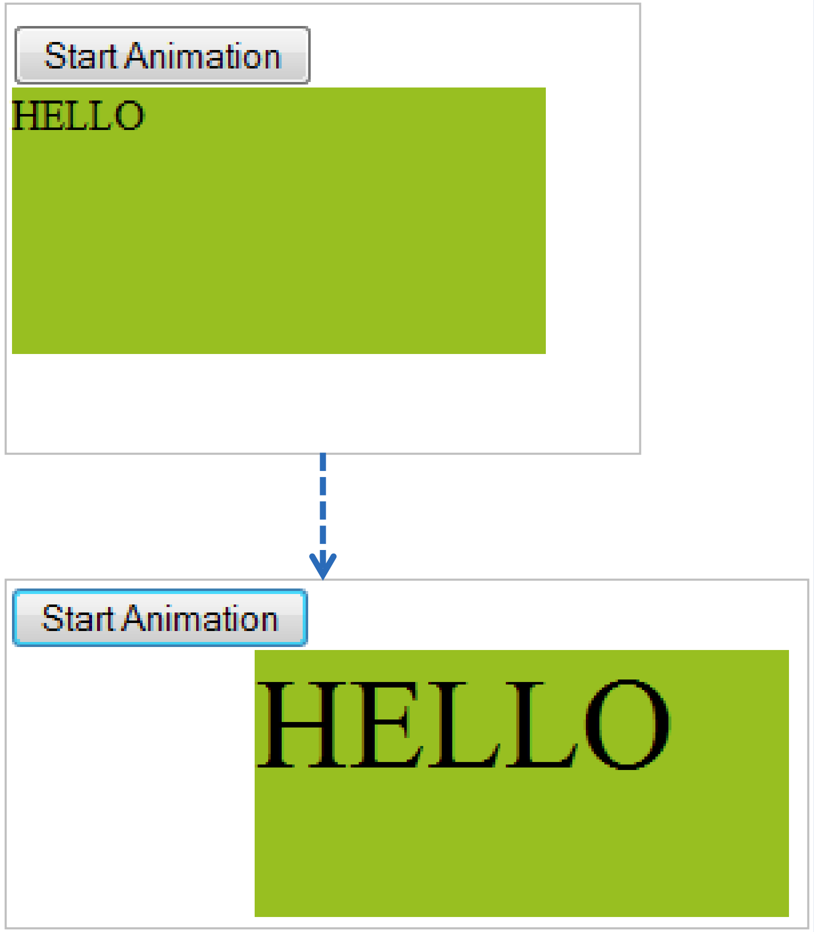
JQUERY – EXERCICE 4



```
<script>
$(document).ready(function(){
    $("button").click(function(){                    //1
        var div = $("div");                          //2
        div.animate({left: '100px'}, "slow");        //3
        div.animate({fontSize: '3em'}, "slow");      //4
    });
});
</script>
```

1. adding a click event to all the *'buttons'* of the page (here only one)
2. the variable div contains all the 'div' of the page (here only one)
3. the ***animate*** function :
   - Change the position of the div from 100px to the left and do the animation slowly (*slow*)
   - Then, change the size to 3em

Preview

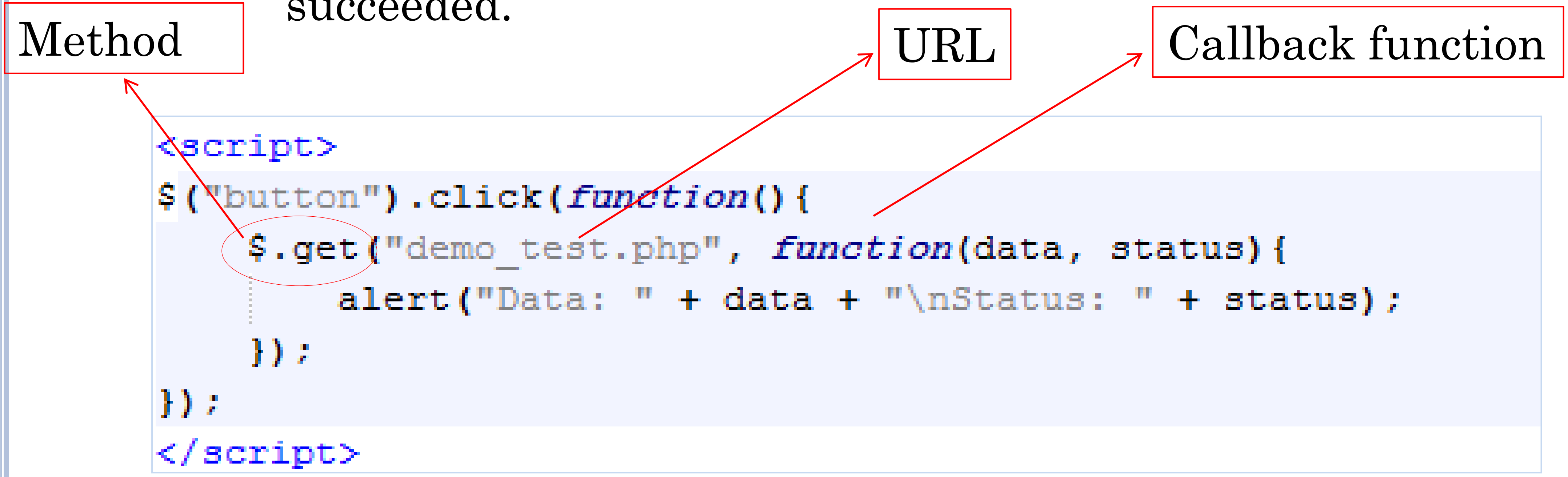


JQUERY – AJAX

- AJAX = Asynchronous JavaScript and XML.
- In short,
  - AJAX is used to
    - load data in the background
    - and post them on the webpage,
    - <u>without having</u> to reload the entire page.
- Examples of applications using AJAX:
    - Gmail, ie
      - send a new email
    - Youtube, ie
      - upload a new video
    - Facebook, ie
      - Post, like, …

JQuery – AJAX

- Get:Request data from the server using an HTTP GET request
  - $.get(*URL,callback*);
    - *URL:* specifies the requested URL (link)
    - *callback:* is the name of a function to execute if the request succeeded.



Method

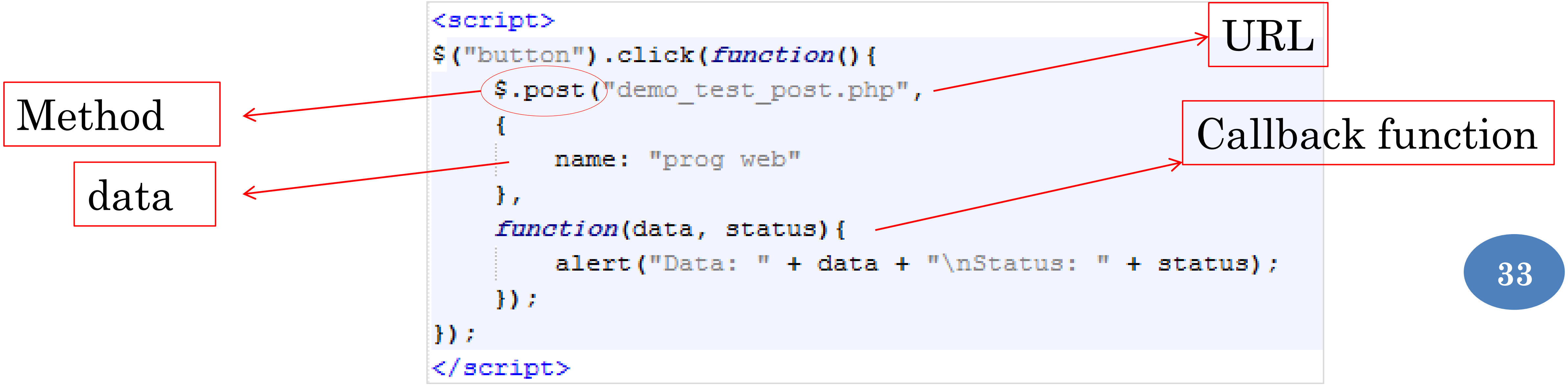
URL

Callback function

```
<script>
$("button").click(function(){
    $.get("demo_test.php", function(data, status){
        alert("Data: " + data + "\nStatus: " + status);
    });
});
</script>
```

JQUERY – AJAX

- Post: requests data from the server using an HTTP POST request
  - $.post(*URL,data,callback*);
    - *URL:* specifies the requested URL (link)
    - *data:* specifies the parameters to send with the request
    - *callback: i*s the name of a function to execute if the request succeeded



```
<script>
$("button").click(function(){
    $.post("demo_test_post.php",
    {
        name: "prog web"
    },
    function(data, status){
        alert("Data: " + data + "\nStatus: " + status);
    });
});
</script>
```