

Introduction to **jQuery** *write less, do more.*

Maurizio Tesconi

May 13, 2015



What is ?

write less, do more.

- Most popular, cross-browser JavaScript library
- Focusing on making client-side scripting of HTML simpler
- Open-source, first released in 2006
- Current release is 1.11 and 2.1
 - Same API
 - 2.X branch doesn't support IE 6, 7 and 8
 - Recommended to use 1.X for public sites





Secrets of the
**JavaScript
Ninja**

John Resig
Bear Bibeault

 MANNING



Features

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



You are not alone!

- Many LARGE companies use jQuery for their sites, including:



WORDPRESS



amazon.com.

mozilla



match.com
START YOUR LOVE STORY



More at http://docs.jquery.com/Sites_Using_jQuery



Cross Browser

```
var d = Date.parse("Wed Jun 25 12:05:12 +0000 2012");  
document.write(d);
```



1277467512000



1277467512000



NaN



Adding jQuery to Your Web Pages



Download jQuery
v1.11.2 or v2.1.3

```
<head>  
<script src="jquery-1.11.2.min.js"></script>  
</head>
```

Google CDN:

```
<head>  
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js">  
</script>  
</head>
```

~~<script type="text/javascript" ...~~



jQuery Syntax

- Basic syntax is:
 - `$(selector).action()`
- Examples:
 - `$(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"`.



First Example – Hello World!

```
<!DOCTYPE html>
<html>
  <head>
    <script src="//code.jquery.com/jquery-1.9.1.min.js">
    </script>
    <script>
      $(document).ready(function(){
        $("body").append("<p>Hello World!</p>");
      });
    </script>
  </head>
  <body>
  </body>
</html>
```



The Document Ready Event

```
$(document).ready(function(){  
    // jQuery methods go here...  
});
```

...or...

```
$(function(){  
    // jQuery methods go here...  
});
```



The noConflict() Method

```
$.noConflict();  
jQuery(document).ready(...)
```

...Or...

```
var jq = $.noConflict();  
jq(document).ready(function(){  
    jq("button").click(function(){  
        jq("p").text("jQuery is still working!");  
    });  
});
```



JQuery Selectors

```
$("#p")
```

```
$(".menu")
```

```
$("#header")
```

```
$("#ul a")
```

```
$("#ul > li")
```

```
$("#ul.menu > li")
```

```
.css( "color", "red" );
```



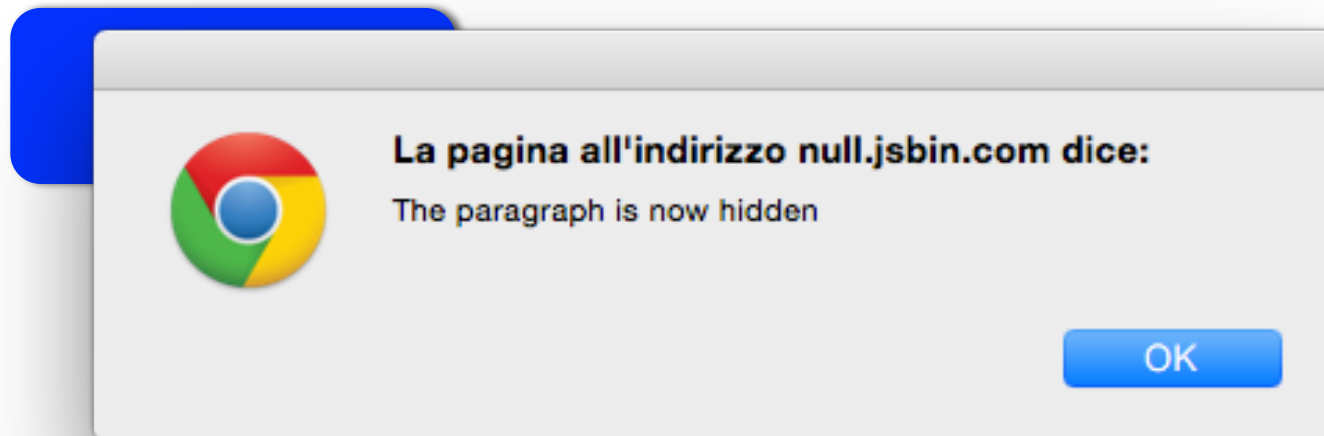
CSS
SELECTORS

DEMO LIVE



CallBack functions

```
$("#button").click(function(){  
    $("#p").hide("slow", function(){  
        alert("The paragraph is now hidden");  
    });  
});
```



Add Elements

- **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



Create elements

```
$("#body").append("<p>paragraph</p>")
```

```
function appendText() {  
    var txt1 = "<p>Text.</p>"; // Create element with HTML  
    var txt2 = $("#<p></p>").text("Text."); // Create with jQuery  
    var txt3 = document.createElement("p"); // Create with DOM  
    txt3.innerHTML = "Text."  
    $("#body").append(txt1, txt2, txt3); // Append the new elements  
}
```



Remove Elements

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

```
$("#div1").remove();
```

```
$("p").remove(".italic");
```



Get/Sets Content and Attributes

- Content
 - **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***
- Attributes
 - **attr()** - method is used to get or set ***attribute*** values.

[DEMO live](#)



Get and Set CSS Classes

- **addClass()** - Adds one or more classes to the selected elements
- **removeClass()** - Removes one or more classes from the selected elements
- **toggleClass()** - Toggles between adding/removing classes from the selected elements

[live DEMO](#)



css() Method

- **css()** - Sets or returns the style attribute
 - **css**("propertyname");
 - **css**("propertyname", "value");
 - **css**({"propertyname": "value", "propertyname": "value", ...});

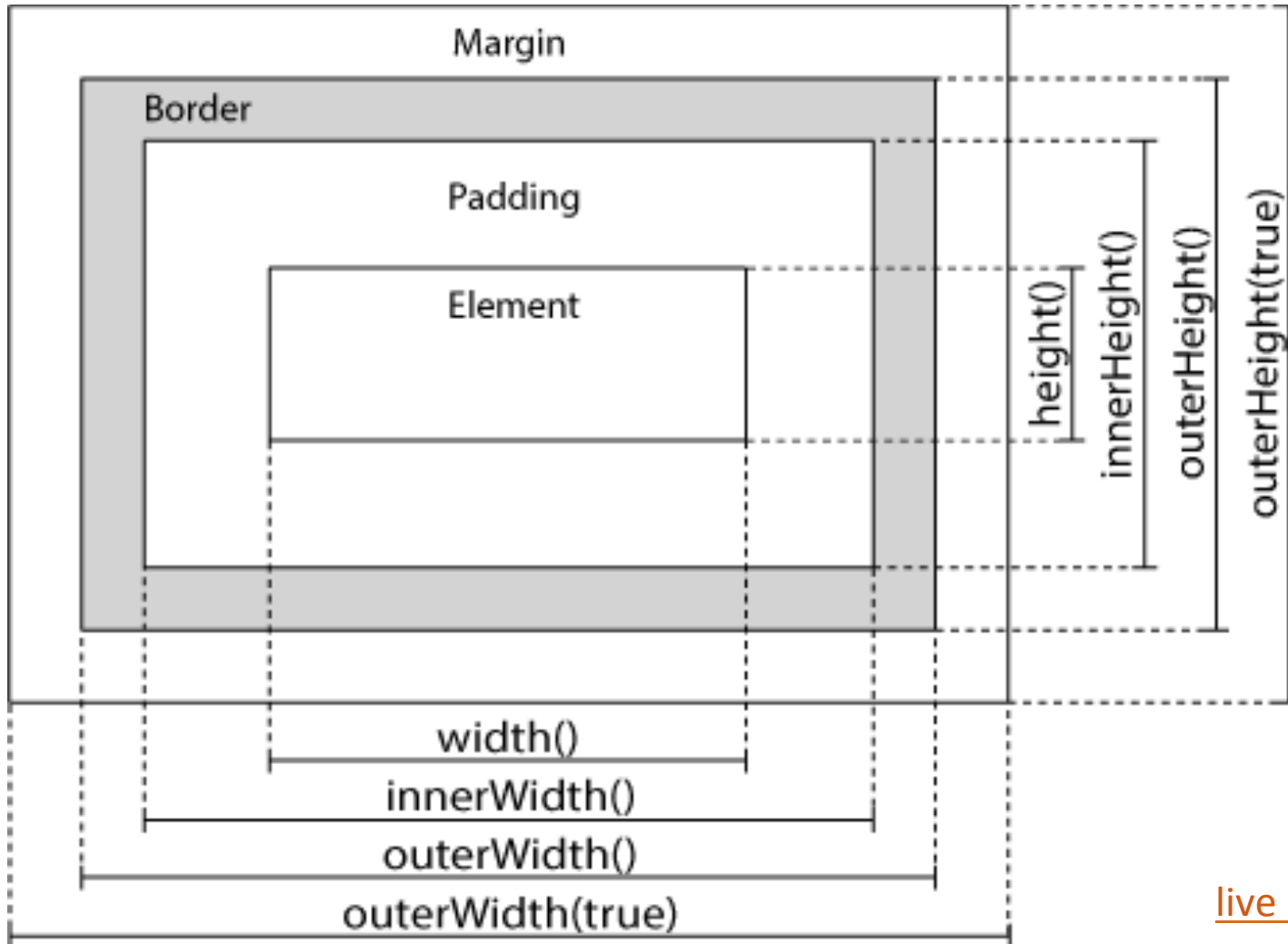
```
$("#p").css("background-color", "yellow");
```

```
$("#p").css({  
    "background-color": "yellow",  
    "font-size": "200%"
```

```
}  
);
```



Dimensions

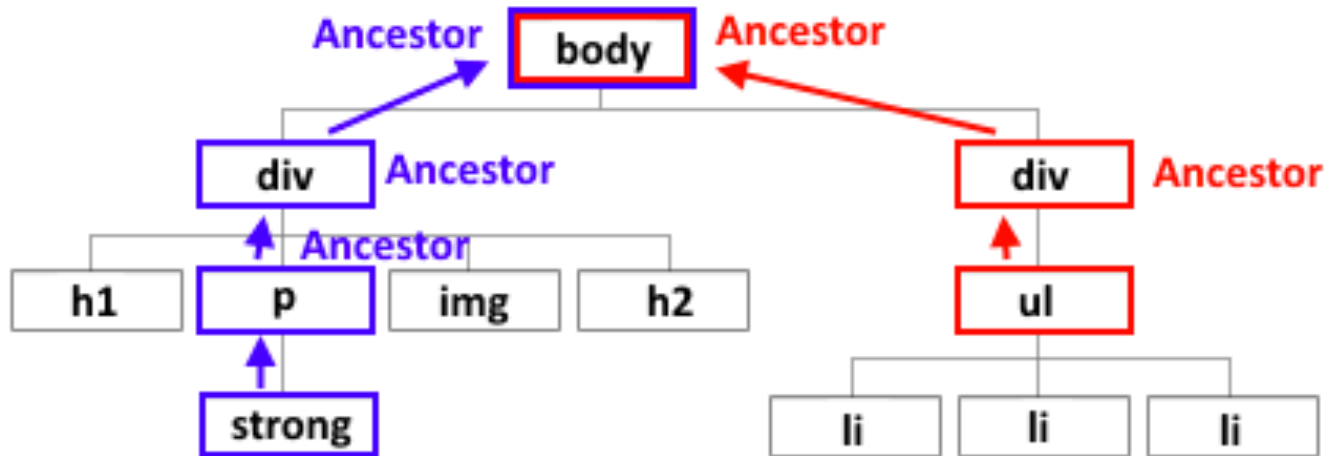


[live DEMO](#)

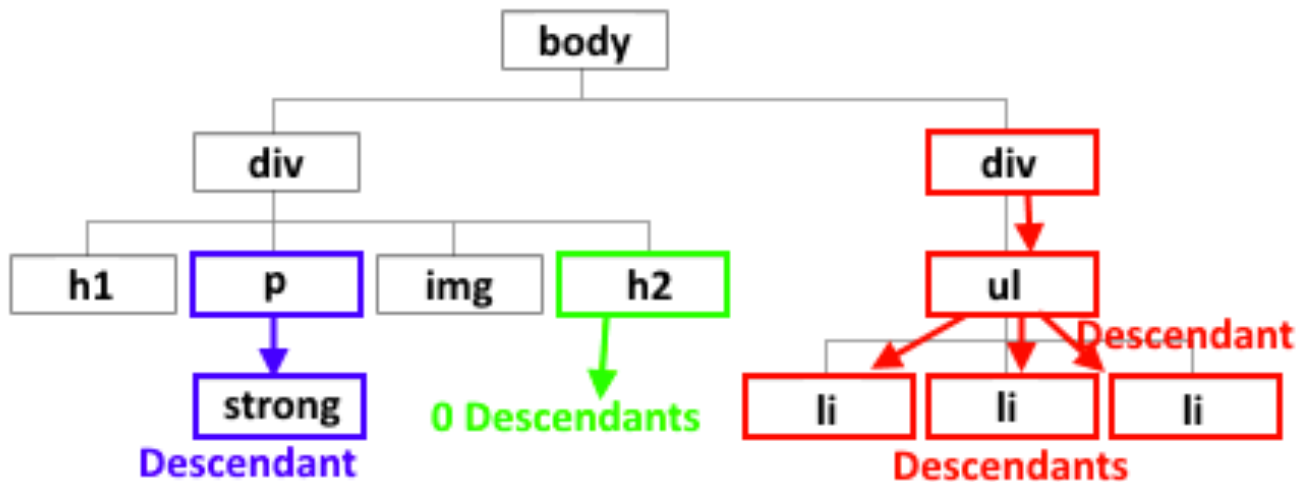


Traversing (1)

`.parents()`

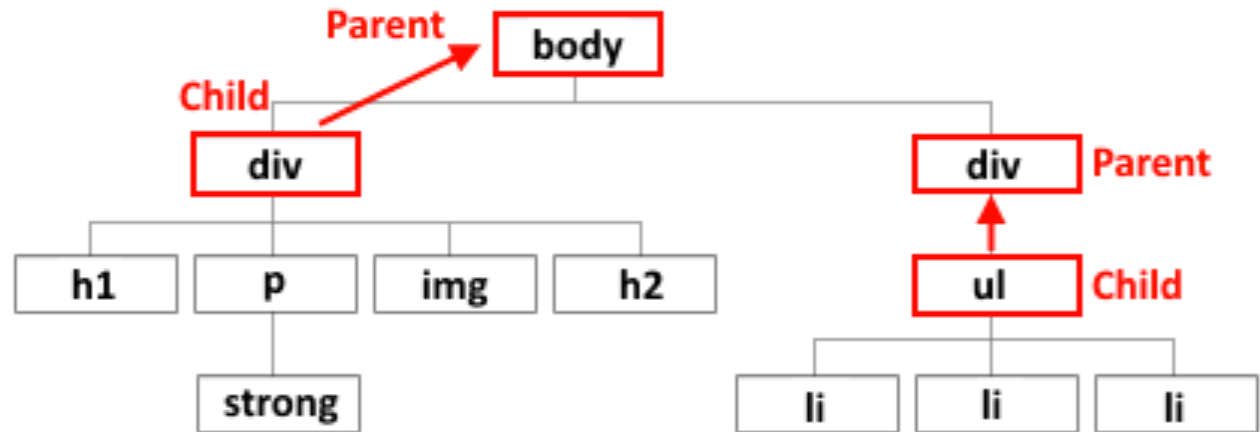


`.find()`



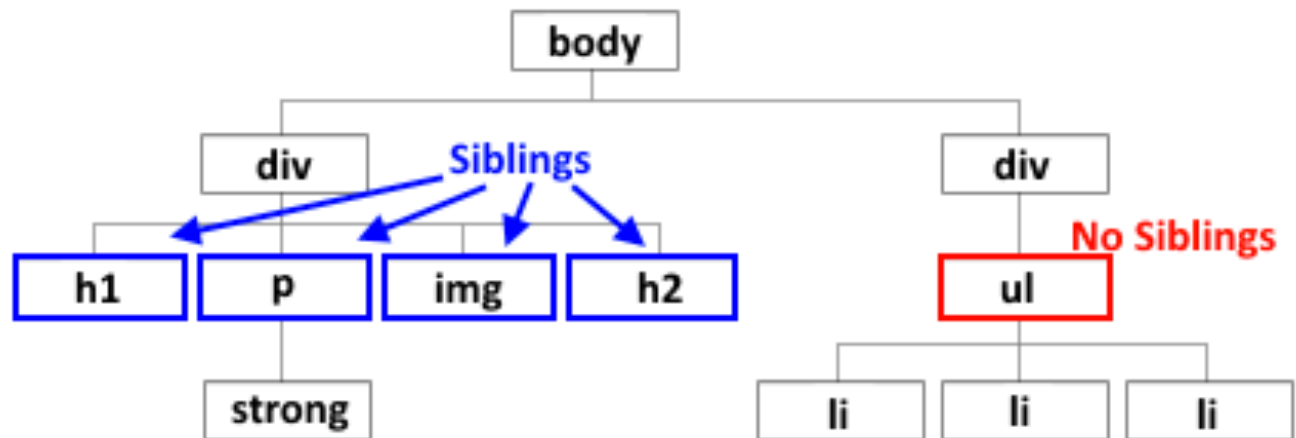
Traversing (2)

`.parent()`



`.children()`

`.siblings()`



Events

- An event represents the precise moment when something happens.

Examples:

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



Events (2)

- `click()`
- `hover()`
- `keypress()`
- `change()`
- `scroll()`
- `resize()`

```
$("p").click(function() {  
    // action goes here!!  
});
```



Effects (1)

- `$(selector).hide(speed,callback);`
- `$(selector).show(speed,callback);`
- `$(selector).toggle(speed,callback);`

```
$("#button").click(function() {  
    $("#p").toggle(1000);  
});
```



Effects (2)

- Fading Methods
 - `fadeIn()`
 - `fadeOut()`
 - `fadeToggle()`
 - `fadeTo()`
- Sliding Methods
 - `slideDown()`
 - `slideUp()`
 - `slideToggle()`

[live DEMO](#)



Animations

`animate()` method used to create custom animations.

- `$(selector).animate({params},speed,callback);`
- `$(selector).stop();`

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

```
$("#stop").click(function(){
  $("#div").stop();
});
```

[live DEMO](#)



Chaining

- run multiple jQuery methods within a single statement.
- one after the other

```
$("#p1").css("color", "red").slideUp(2000).slideDown(2000);
```

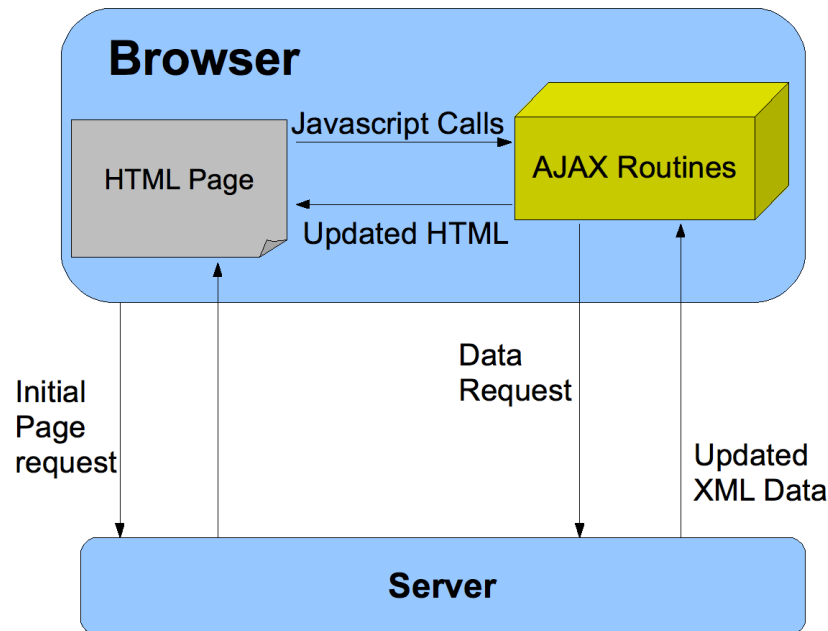
or

```
$("#p1") .css("color", "red")  
        .slideUp(2000)  
        .slideDown(2000);
```



AJAX Introduction

- AJAX = Asynchronous JavaScript and XML.
- request in background without reloading the whole page.
- different browsers have different syntax for AJAX implementation



load() Method

- `$(selector).load(URL,data,callback);`

demo_test.txt

```
<h2>jQuery and AJAX is FUN!!!</h2>  
<p id="p1">This is some text in a paragraph.</p>
```

```
$("#div1").load("demo_test.txt");
```



AJAX get() and post() Methods

- `$.get(URL, callback);`
- `$.getJSON(URL, data, callback);`
- `$.post(URL, data, callback);`

```
$.get("demo_test.php", function(data, status){  
    alert("Data: " + data + "\nStatus: " + status);  
});
```

```
$.post("demo_test_post.php",  
    { name: "Donald Duck",  
      city: "Duckburg" },  
    function(data, status){  
        alert("Data: " + data + "\nStatus: " + status);  
    });
```



Ajax example

index.html

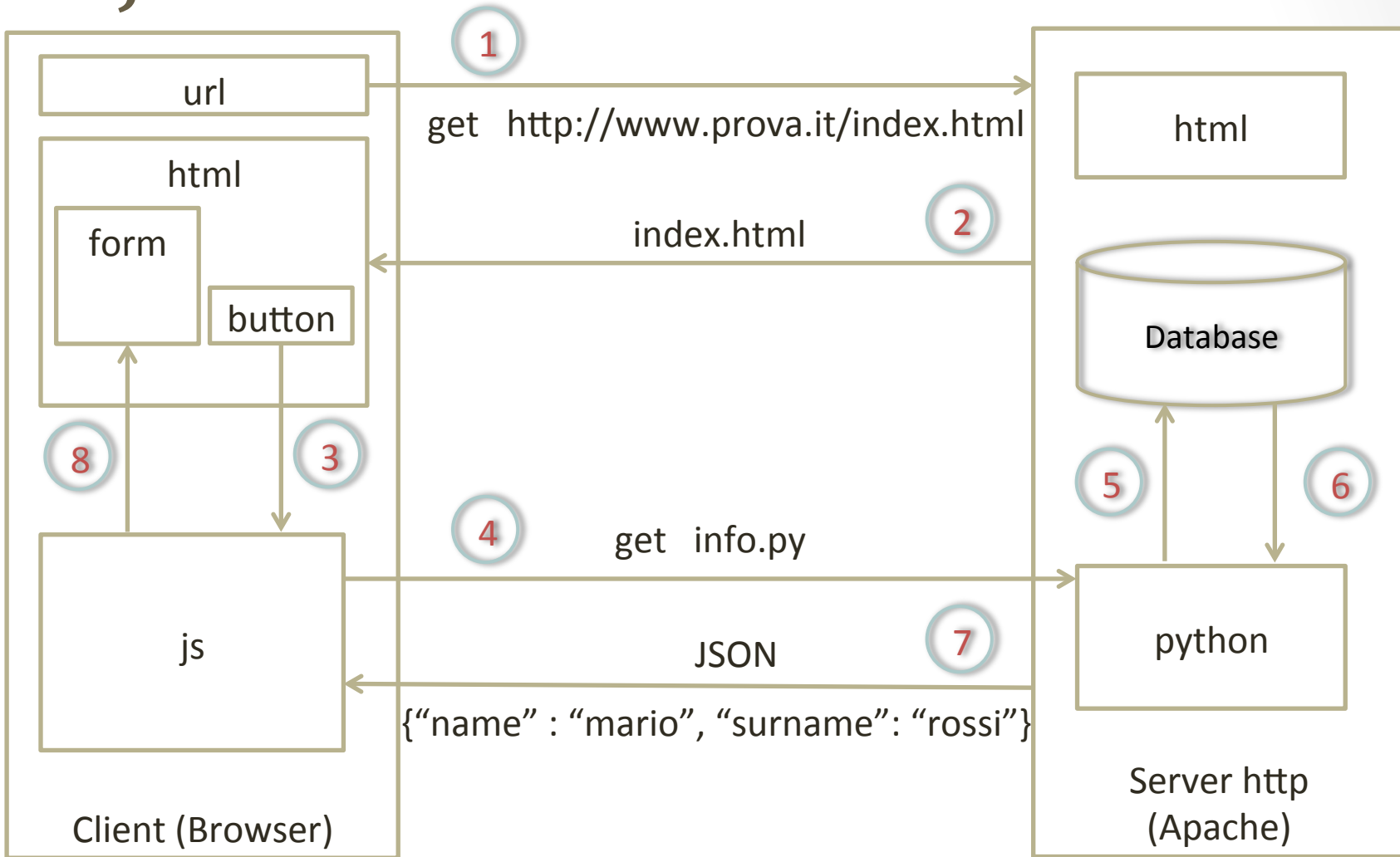
```
<button id="button">Button</button>
<div>Name: <span id="name"> </span></div>
<div>Surname: <span id="surname"> </span></div>
```

info.py -> JSON -> {"name": "mario", "surname": "rossi"}

```
$(document).ready(function() {
    $("#button").click(function(){
        $.getJSON("info.py", function(data) {
            $("#name").html(data.name);
            $("#surname").html(data.surnam);
        });
    });
});
```



Ajax



Examples

- [Simple Javascript calculator](#)
- [Adjust font-size dynamically](#)
- [Bar char](#)
- [Change by click, ver 2](#)

