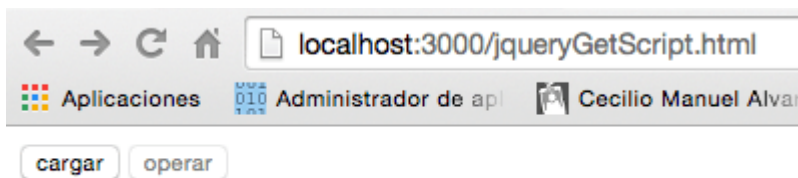


Cada día necesitamos más realizar la carga dinámica de ficheros de Javascript. Tecnologías como [Require.js](#) abordan esta problemática. Pero hay muchas situaciones cotidianas que se pueden resolver de forma más sencilla con algo como jQuery getScript.

## Utilizando jQuery getScript

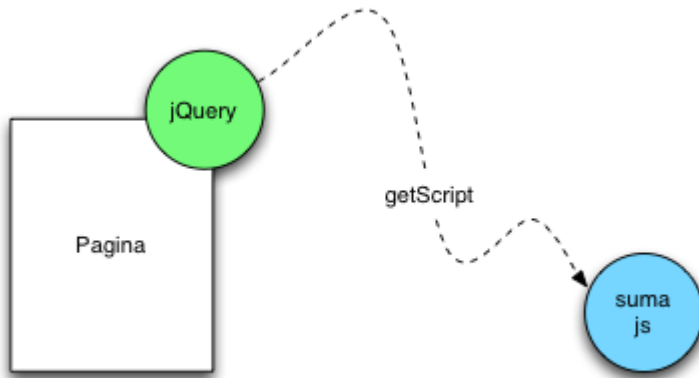
En el siguiente ejemplo se muestran dos botones “cargar” y “operar”. El primero de ellos aparece habilitado cuando se carga la página.



El segundo de ellos aparece deshabilitado ya que necesita de un fichero de JavaScript que realiza una operación de sumar y que no hemos cargado inicialmente con la página.

```
function suma (a,b) {  
  
return a+b;  
}
```

Para poder habilitar el segundo botón y pulsarle hay que cargar previamente el script de la suma.



Para realizar esta operación hay que usar jQuery y su método getScript.

```
<html>
<head>
<script src="jquery-1.11.1.js"></script>
<script type="text/javascript">

$(document).ready(function() {

$("#operar").attr("disabled",true);

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

$.getScript("suma.js",function() {

console.log("script cargado");
```

```
$("#operar").attr("disabled",false);

});

});

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

alert (suma (2,2));
})

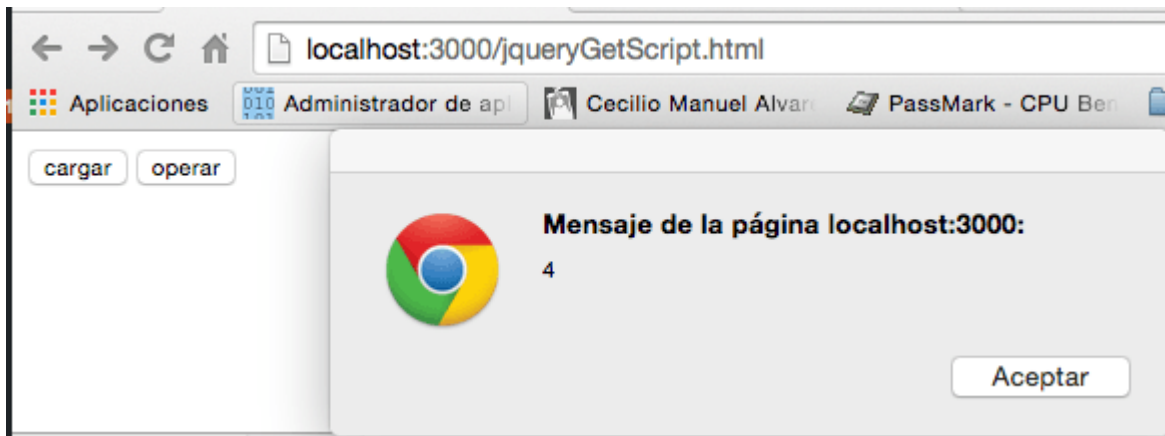
});

</script>
</head>
<body>
<input type="button" value="cargar" id="cargar" />

<input type="button" value="operar" id="operar"/>

</body>
</html>
```

De esta forma el script de la suma se cargará de forma dinámica . Para ello se ha utilizado jQuery getScript realizando una petición AJAX y cargando dinámicamente el contenido de la función suma.Una vez cargado se habilita el botón de operar y el usuario puede pulsar y realizar la suma como si el script hubiera sido cargado al iniciar la página.



Otros artículos relacionados: [Utilizando jQuery Data](#) , [jQuery On/Off](#) , [jQuery Sizzle](#)