



## Buenas Prácticas JavaScript

Convención de nombres - <a href="https://google.github.io/styleguide/javascriptguide.xml">https://google.github.io/styleguide/javascriptguide.xml</a>	
<b>Clases / Módulos</b>	<ul style="list-style-type: none"><li>• Nombres en PascalCase (<b>C</b>alculadora<b>R</b>omana).</li><li>• Nombres simples y descriptivos.</li><li>• Colecciones nombrarlas en plural (<b>P</b>hotos = Backbone.Collection).</li></ul>
<b>Funciones / Constructores</b>	<ul style="list-style-type: none"><li>• Funciones en camelCase (<b>s</b>um<b>F</b>loat<b>N</b>umbers)</li><li>• Constructores en PascalCase (<b>U</b>ser<b>A</b>dmin).</li><li>• Nombres simples y descriptivos.</li></ul>
<b>Atributos/Variables</b>	<ul style="list-style-type: none"><li>• Nombres de atributos y variables en camelCase (<b>var</b> <b>b</b>asic<b>I</b>nformation = {}).</li><li>• Guion bajo '_' delante de propiedades privadas (<b>var</b> <b>_</b>myPrivateVariable).</li><li>• <b>_this</b> para guardar referencia de this (<b>var</b> <b>_</b>this = this).</li><li>• Signo \$ delante de variables que almacenan datos de jQuery. <pre>var myVar = 'foo',     \$myLayer = \$('#layer-one'),     \$myLayer2 = \$('#layer-two');</pre></li><li>• Sentencia única para múltiples declaraciones. <pre>var myVar = 'foo',     myVar2 = 'bar',     myVar3 = 'foobar';</pre></li></ul>
<b>Constantes</b>	<ul style="list-style-type: none"><li>• Nombres de constantes con todas las letras en mayúsculas separando palabras con signo '_' (<b>var</b> MY_CONST_NAME).</li></ul>
<b>Ficheros</b>	<ul style="list-style-type: none"><li>• Nombres de ficheros completamente en minúsculas incluyendo los signos '-' o '_' (<b>betacompany_util.js</b> o <b>my-exampleHelper.js</b>).</li></ul>
<b>String</b>	<ul style="list-style-type: none"><li>• Definir SIEMPRE cadenas de texto con comillas simples NO dobles (<b>var</b> myStr = 'Este es mi texto';).</li></ul>
Ordenación	
Variables	<ul style="list-style-type: none"><li>• Las variables <i>privadas</i> siempre al <b>inicio</b> de mi fichero .js si son globales o al inicio de mi función si son internas y luego las <i>públicas</i>. <pre>function myFunction() {     var _myPrivateVar = null;     ...     var myPublicVar = '123';     //aquí la implementación }</pre></li></ul>
funciones	<ul style="list-style-type: none"><li>• Las funciones <i>privadas</i> siempre al <b>inicio</b> de mi fichero .js y luego las <i>públicas</i>. <pre>//private functions function myPrivateFunction1() {...} function myPrivateFunction2() {...}   //public functions function myPublicFunction1() {...} function myPublicFunction2() {...}</pre></li></ul>



Podéis encontrar toda la información en  
<https://quark.everis.com/confluence/pages/viewpage.action?pageId=811303050>



**Y recordad amiguitos!!!  
No olvidéis mantener un  
bajo acoplamiento y una  
alta cohesión!!**