

Guión Javascript – Ajax - JSON

Objetos literales de Javascript

- Es otra forma de crear objetos, que consiste en definir un array asociativo. (probar en editor)

```
avion = [ "Boeing" , "747" , 450 ];
```

```
avion[0] -> "Boeing"  
avion[1] -> "747"  
avion[2] -> 450
```

```
avion = { marca:"Boeing" , modelo:"747" , pasajeros:450 }
```

Esto sería equivalente a haber definido el objeto así:

```
function Avion () {  
    this.marca;  
    this.modelo;  
    this.pasajeros;  
}  
  
var avion=new Avion();  
  
avion.marca="Boeing";  
avion.modelo="747";  
avion.pasajeros=450;
```

- Para acceder a cada propiedad del objeto se puede hacer: (probar en editor)

```
avion.marca  
  
avion["marca"]
```

- Para acceder a todas las propiedades podemos utilizar el siguiente bucle: (probar en editor)

```
for (var prop in avion) {  
  
    cad= prop + "= " + avion[prop] + "<br>";  
  
}
```

JSON

JSON es otro formato de intercambio de datos muy utilizado en AJAX alternativo a XML, que sigue la sintaxis de objeto literal de JavaScript.

Por ejemplo los datos del objeto avion creado anteriormente en formato se expresaría:

```
avion = { "marca":"Boeing" , "modelo":"747" , "pasajeros":450 }
```

Y se podría manejar como hemos manejado los objetos literales: (probar en editor)

Si recibimos los datos en formato JSON pero a través de un string debemos convertirlos a formato JSON con el siguiente método (probar en editor, poniendo los datos JSON en string, y parseando a JSON):

```
datosJSON=parse.JSON(texto)
```

Ejemplo

Mostrar los datos de mi IP.

Los datos van a ser recogidos en formato JSON.

Vamos a pedir por Ajax estos datos en formato JSON a la URL <http://ip-api.com/json/>