JavaScript for in vs for of es una pregunta muy habitual hoy en día . Poco a poco vamos usando cada día más JavaScript ES6 y estos tipos de bucles son muy habituales. Vamos a explicarlo un poco las diferencias entre ambos . Normalmente cuanto trabajamos con JavaScript es muy habitual utilizar colecciones de objetos, en este caso partiremos de una lista de personas.

```
var listaPersonas = [
  {"nombre": "pepe", "apellidos": "perez"},
  {"nombre": "ana", "apellidos": "gomez"},
  {"nombre": "almudena", "apellidos": "blanco"}
];

// La forma más común de imprimir todos los datos por pantalla es
recorrer la lista con un bucle for.
for (var i = 0; i < listaPersonas.length; i++) {
  console.log(listaPersonas[i].nombre);
  console.log(listaPersonas[i].apellidos);
}
```

La forma más común de imprimir todos los datos por pantalla es recorrer la lista con un bucle for.

```
// La forma más común de imprimir todos los datos por pantalla es
recorrer la lista con un bucle for.
for (var i = 0; i < listaPersonas.length; i++) {
  console.log(listaPersonas[i].nombre);
  console.log(listaPersonas[i].apellidos);
```

```
}
```

El resultado es:

```
pepe
perez
ana
gomez
almudena
blanco
```

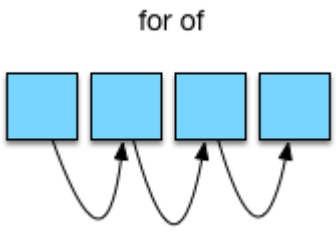
Recordemos que tambien podríamos haber utilizado programación funcional y forEach

JavaScript for of

Ahora bien a partir de JavaScript ES6 existe otra forma de recorrer las colecciones de objetos. Podemos usar un bucle for of que tiene una sintaxis mucho más natural.

```
for (var persona of listaPersonas) {

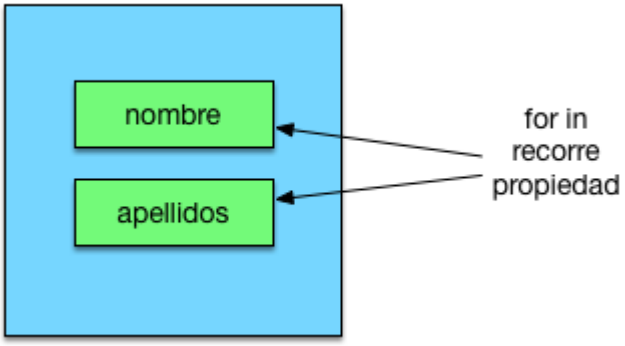
console.log(persona.nombre);
console.log(persona.apellidos);
}
```



El resultado será idéntico. De esta forma lo estaremos recorriendo como si fuera una estructura forEach de otros lenguajes de programación como por Java. Hemos tenido que esperar hasta JavaScript ES6 para tenerlo a nuestra disposición.

For in

Ahora bien ¿Para que sirve el bucle for in?.El bucle for in es el encargado de recorrer la colección de propiedades de un objeto.



Podríamos modificar el código del principio para hacerlo más flexible utilizando esta estructura que esta disponible desde JavaScript ES5.

```
  for (var propiedad in persona) {
    // Imprime el valor de cada propiedad
    console.log(persona[propiedad]);
  }
}
```

La ventaja que tiene este anidamiento de bucles es que nos permite sacar por pantalla las propiedades de cualquier lista de objetos que pasemos y no solo las de la persona. Hemos ganado en flexibilidad y claridad usando for in y for of.

Otros artículos relacionados :

- Introducción a JavaScript ES6
- JavaScript console time y rendimiento ,
- JavaScript Promise y la programación asíncrona