

Destructuring

¿Qué es?

- *Destructuring* nos da una nueva manera de extraer datos de objetos o arreglos y asignarlos a variables.

Ejemplo 1

```
const array = ['pato', 'ganzo', 'pollo'];
```

```
var primer = array[0];
```

```
var segundo = array[1];
```

```
var tercero = array[2];
```

```
const array = ['pato', 'ganzo', 'pollo'];
```

```
const [a,b,c] = array;
```

```
// a = 'pato'
```

```
// b = 'ganzo'
```

```
// c = 'pollo'
```

```
const [a,,c] = array;
```

```
// a = 'pato'
```

```
// c = 'pollo'
```

Ejemplo 2

- Intercambio de valores entre dos variables:

```
let x = 2;  
let y = 5;  
  
[x,y] = [y,x];  
  
// x = 5  
// y = 2
```

Ejemplo 3

- **Multiples valores de retorno**
- Gracias a destructuring assignment, las funciones pueden retornar multiples valores. Aunque en JavaScript siempre ha sido posible regresar un arreglo de una función, esto agrega más flexibilidad.

```
function f() { return [1, 2]; }
```

Ejemplo 4

- **Recoger Multiples valores de retorno**
- En variables independientes

```
1 | var a, b;  
2 | [a, b] = f();  
3 | console.log("A es " + a + " B es " + b);
```

- En un array

```
1 | var a = f();  
2 | console.log("A es " + a);
```