

Tenemos nuestra clase Edificio con las siguientes propiedades:

- calle.
- número.
- código postal.
- plantas del edificio (dentro de cada planta tendremos un número de puertas y para cada puerta almacenaremos el nombre del propietario).

Las propiedades calle, número, código postal son propiedades simples, de un solo valor. La propiedad plantas es un array, una lista. El edificio puede tener más de una planta (lista de plantas) y, para cada planta un número n de puertas (lista de puertas) y, además para cada puerta queremos almacenar el nombre del propietario.

El enunciado dice: “Cada vez que instanciamos un edificio le pasaremos la calle, número y código postal como parámetros”. No nos dicen que le pasemos el número de plantas, por lo tanto, no podemos crear el array con un número de elementos definido.

De esta forma, si lo definiésemos como un objeto literal, suponiendo que lo creamos sin datos de las plantas tal y como dice en el enunciado.

```
const edificio1={  
  calle: 'Hospital',  
  numero: 5,  
  cp: 15003,  
  plantas:[], // o bien plantas: new Array()  
}  
  
// Para mostrar la información utilizamos el objeto  
  
console.log('Edificio1:', edificio1);
```

Supongamos a hora que queremos añadirle a nuestro objeto un método que nos muestre la información del edificio con el número de plantas que tiene:

Como lo hemos definido en la notación literal, nuestro objeto completo quedaría de la forma:

```
const edificio1={  
  calle: 'Hospital',  
  numero: 5,  
  cp: 15003,  
  plantas:[],  
  
  // método mostrar datos del edificio, indicando el número de plantas  
  muestraInfo(){  
    console.log(`Edificio calle ${this.calle}, nº ${this.numero}, ${this.cp}, tiene ${this.plantas.length} plantas`);  
  }  
}
```

```
}
```

```
// llamada al método:
```

```
edificio1.muestraInfo();
```

¿Qué pasa cuando le añadimos las plantas?

Ejemplo: nuestro edificio1 es de la forma

	<i>Puerta A / índice 0</i>	<i>Puerta B / índice 1</i>	<i>Puerta C / índice 2</i>
<i>Planta 1 / índice 0</i>	<i>Jenaro Pardo Vélez</i>	<i>María Souto González</i>	<i>Ana Varela Varela</i>
<i>Planta 2 / Índice 1</i>	<i>Isabel Abelenda Lado</i>	<i>Luis Caramés Pampín</i>	<i>José Carballo Calero</i>
...	...		

Método agregarPlantasYPuertas(numplantas, puertas)

Si nos fijamos en el ejemplo tenemos una tabla. Una tabla no es más que un array de dos dimensiones. Eso significa que, por un lado las plantas son un array y, a su vez, en cada planta como hay varias puertas vamos a tener un array de puertas (esa sería la segunda dimensión).

Ejemplo:

```
Planta1 = ['Puerta A', 'Puerta B', 'Puerta C']
```

Pero, **¿Quién es 'Planta 1' ?**

'Planta 1' es un elemento del array plantas del edificio, es decir, si queremos hacer referencia a la primera planta del edificio "edificio1" de nuestro ejemplo, se haría

```
edificio1.plantas[0]
```

O bien, si las plantas tienen nombre:

```
edificio1.plantas['Planta 1']
```

Y, **¿Cuál es el valor que se almacena en 'Planta 1'?**

En 'Planta 1' guardaremos las puertas que tiene la planta, que según el ejemplo que hemos puesto son: 'Puerta A', 'Puerta B', 'Puerta C', es decir, nos encontramos de nuevo con un array.

```
edificio1.plantas[0] = []
```

O lo que es lo mismo

```
edificio1.plantas[0] = new Array();
```

En este método todavía no nos dicen nada del contenido de la tabla, solo nos piden crear la estructura de tabla, es decir, crear una tabla vacía de tantas filas como numplantas y tantas columnas como puertas.

¿Cómo se traduce esto a la hora de añadir las plantas a nuestro edificio?

EL enunciado nos dice dos cosas:

- “Se le pasa el **número de plantas** que queremos **crear** en el piso(planta) y el número de puertas por planta. ”
- “Cada vez que se llame a este método, **añadirá** el número de plantas y puertas indicadas en los parámetros, a las que ya están creadas en el edificio”.

Si el edificio aún no tuviese plantas:

Crear un bucle para llenar el array de plantas (la propiedad plantas), desde 0 hasta el parámetro numplantas serían las filas y para cada una de las filas (plantas) un bucle interno para las puertas. De momento con valores vacíos, es decir, cada puerta de la planta sin datos.

Pero puede que ya tengamos plantas en nuestro edificio y, en ese caso nos pide **añadir** ¿Cómo sé cuántas plantas tiene mi edificio edificio1?

Al ser un array:

edificio1.plantas.length

Esto qué quiere decir, que para nuestro bucle de creación de plantas y puertas, en lugar de empezar en la planta 0, empezaríamos en la última planta (*edificio1. Plantas.length*) y, *cuál sería el fin???...*

Y, **¿cuál es el contenido de cada puerta de la planta?**

En cada una de las puertas lo que se guarda es el propietario

Planta1 = [*Jenaro Pardo Vélez*, *María Souto González*, *Ana Varela Varela*']

O lo que es lo mismo:

`edificio1.plantas[0][0] = 'Jenaro Pardo Vélez'`

`edificio1.plantas[0][1] = 'María Souto González'`

...

Espero que esto os ayude a seguir, he utilizado una definición de objeto literal, tendríais que traducirla a una clase lógicamente para que podamos construir un edificio de cada vez a través del constructor de la misma.