

Es habitual usarlo para que una clase sepa a cuál pertenece (por ejemplo, a qué casa pertenece una puerta), porque de lo contrario, cada clase "conoce" a los objetos que contiene (la casa "conoce" a sus puertas), pero no al contrario.

Ejercicios propuestos:

- **(7.5.1)** Crea una versión ampliada del ejercicio 7.4.2, en la que el constructor sin parámetros de la clase "Trabajador" se apoye en otro constructor que reciba como parámetro el nombre de esa persona. La versión sin parámetros asignará el valor "Nombre no detallado" al nombre de esa persona.

7.6. Sobrecarga de operadores

Los "operadores" son los símbolos que se emplean para indicar ciertas operaciones. Por ejemplo, el operador "+" se usa para indicar que queremos sumar dos números.

Pues bien, en C# se puede "sobrecargar" operadores, es decir, redefinir su significado, para poder sumar (por ejemplo) objetos que nosotros hayamos creado, de forma más cómoda y legible. Por ejemplo, para sumar dos matrices, en vez de hacer algo como "matriz3 = suma(matriz1, matriz2)" podríamos hacer simplemente "matriz3 = matriz1 + matriz2"

No entraremos en detalle, pero la idea está en que redefiniríamos un método llamado "operador +", y que devolvería un dato del tipo con el que estamos (por ejemplo, una Matriz) y recibiría dos datos de ese mismo tipo como parámetros:

```
public static Matriz operator +(Matriz mat1, Matriz mat2)
{
    Matriz nuevaMatriz = new Matriz();

    for (int x=0; x < tamanyo; x++)
        for (int y=0; y < tamanyo; y++)
            nuevaMatriz[x, y] = mat1[x, y] + mat2[x, y];

    return nuevaMatriz;
}
```

Desde "Main", calcularíamos una matriz como suma de otras dos haciendo simplemente

```
Matriz matriz3 = matriz1 + matriz2;
```

Ejercicios propuestos:

- **(7.6.1)** Desarrolla una clase "Matriz", que represente a una matriz de 3x3, con métodos para indicar el valor que hay en una posición, leer el valor de una posición, escribir la matriz en pantalla y sumar dos matrices.