

7) En este ejercicio, se requiere extender el tipo `int[]` con algunos métodos de extensión. Se presenta el código del método de extensión `Print(this int[] vector, string leyenda)` que imprime en la consola los elementos del vector precedidos por una leyenda que se pasa como parámetro. Se requiere codificar el método de extensión `Seleccionar(...)` que recibe como parámetro un delegado de tipo `FuncionEntera` y devuelve un nuevo vector de enteros producto de aplicar la función recibida como parámetro a cada uno de los elementos del vector. El siguiente programa debe producir la salida indicada.

```
-----Program.cs-----
int[] vector = [1, 2, 3, 4, 5];
vector.Print("Valores iniciales: ");
var vector2 = vector.Seleccionar(n => n * 3);
vector2.Print("Valores triplicados: ");
vector.Seleccionar(n => n * n).Print("Cuadrados: ");

-----FuncionEntera.cs-----
delegate int FuncionEntera(int n);
```

#### Salida por consola

```
Valores iniciales: 1, 2, 3, 4, 5
Valores triplicados: 3, 6, 9, 12, 15
Cuadrados: 1, 4, 9, 16, 25
```

Para ello, completar el código de la siguiente clase estática `VectorDeEnterosExtension`

```
static class VectorDeEnterosExtension
{
    public static void Print(this int[] vector, string leyenda)
    {
        Console.WriteLine(leyenda + string.Join(", ", vector));
    }
    public static int[] Seleccionar(. . . ) {
        . . .
    }
}
```