

Clase 16

Miercoles, 11 Octubre 2023

[Indice](#)

Repaso de funciones

- **FUNCION:** Le paso una lista y me devuelve los dos valores mayores incluidos en ella.

```
,
public static List<int>int GetTwoMajors(List<int> list)
{
    for (int i = 0; i < list.Count - 1; i++)
    {
        for (int j = i + 1; j < list.Count; j++)
        {
            if(l[i] < l[j])
            {
                int aux;
                aux = l[i];
                l[i] = l[j];
                l[j] = aux;
            }
        }
    }
    List<int> listResult = new List<int>();
    listResult.Add(list[0]);
    listResult.Add(list[1]);

    return listResult;
}
```

- **FUNCION:** Le paso una lista y me devuelve solo los numeros pares.

```
,
public static List<int>int GetEvenNumbers(List<int> list)
{
    List<int> listResult = new List<int>();
    for (int i = 0; i < list.Count; i++)
    {
        if(Functions.IsEven(list[i]))
            listResult.Add(list[i]);
    }
    return listResult;
},
```

- **FUNCION:** Igual que la anterior, pero que funcione con arrays.

```
,
public static int[] GetEvenNumbers(int[] array)
{
    int count = 0;
    for (int i = 0; i < array.Length; i++)
    {
        if(Functions.IsEven(array[i]))
            count++;
    }

    int[] arrayResult = new int[count];
    for (int i = 0; i < array.Length; i++)
    {
        if(Functions.IsEven(array[i]))
            arrayResult[i] = array[i];
    }
    return arrayResult;
},
```

- Si tenemos una sentencia con una declaración demasiado larga y complicada y que además se repite constantemente, es bueno almacenar dicha sentencia dentro de una variable, para acceder a ella más facilmente.