

course: COMP6210 week: 01 lesson: 01 topic: solid-principle-1-single-responsibility-principle-exercises

[DOWNLOAD PDF](#) [CLICK HERE](#)

Exercise 01

Looking at the code below, refactor this code to comply with the Single Responsibility Principle.

Answer the following questions as a class or as the tutor comes around:

- How many things is this method doing?
- What is a key element that is missing for the program output

Your Task:

- Refactor the code.
- You are to create a new `Class` that holds all of the logic code.
- The Program is **only** allowed to hold the input and output of the program.
- Note that anything that is not UI related needs to be put into methods, no matter how small it is - so that it potentially can be used somewhere else.

```
private static void Main(string[] args)
{
    List<string> names = new List<string> { "Lisa", "Bob", "Stephen", "Ricky", "Christine" };

    Console.WriteLine("The number of people in this array is:");
    Console.WriteLine(names.Count);

    Console.WriteLine("Please add another name:");
    string name = Console.ReadLine();

    names.Add(name);

    Console.WriteLine("The names in the array list are:");

    foreach(string x in names)
    {
        Console.WriteLine(x);
    }

    Console.WriteLine("The number of people in this array is:");
    Console.WriteLine(names.Count);
}
```

Exercise 02

Looking at the code below refactor the code according to the Single Responsibility Principle

Answer the following questions as a class or as the tutor comes around:

- What errors can you see when it comes to UI and Business Logic Separation
- Do all the methods fit into this class according to SRP?
- What is missing from the Person class? (1 or more things...)

Your Task:

- Refactor the code.
- You have to keep the Person class, but you are free to add others

```
class Person
{
    public string Fname { get; set; }
    public string Lname { get; set; }
    public string Email { get; set; }

    public void Introduction(string name)
    {
        Console.WriteLine("Hello, my name is" + name);
    }

    public int DisplayMyAge()
    {
        return 36;
    }

    public void HasBirthday()
    {
        Console.WriteLine(36 + 1);
    }

    public void ShowStudentId()
    {
        Console.WriteLine("1003004");
    }
}

class Program
{
    static void Main(string[] args)
    {
        Person Billy = new Person();
        Billy.Fname = "Billy";
        Billy.Lname = "Simpson";
        Billy.Email = "billy.simpson@acme.com";

        Billy.Introduction(Billy.Fname);
        Billy.DisplayMyAge();
        Billy.HasBirthday();
        Billy.DisplayMyAge();
        Billy.ShowStudentId();
    }
}
```