

### CONTENIDOS

- Implementación de POO
- Separación de responsabilidades

1) En el mismo repositorio del TP1 (<https://classroom.github.com/a/yIIZTnUy>) a partir de la rama TP2 cree una rama que se llame TP3 y trabaje en ella para el presente trabajo práctico

2) Migración del Sistema para Cadetería

El cliente presentó como nuevo requisito la posibilidad de acceder via web, por lo que se necesita adaptar el código actual para independizarlo de la interfaz de usuario, como requisito previo a la migración a un servidor web.

Para poder cumplir con dicho requisito se propuso las siguientes modificaciones:

- Modifique todos los métodos públicos de la clase cadetería de manera tal que ésta sólo reciba datos primitivos (datos del tipo string, double, int, etc)
- Elimine todas las llamadas a métodos vinculados a la interfaz de usuario que pudiera tener la cadetería

Ejemplo.

Si posee un método `MostrarNombreCadeteria` definido así:

```
public void MostrarNombreCadeteria()  
{  
  
    Console.WriteLine(this.Nombre);  
}
```

Reemplácelo por:

```
public string MostrarNombreCadeteria()  
{  
  
    return this.Nombre;  
}
```

3) Opcional (trabajo de investigación)

Desde una nueva rama llamada Biblioteca-de-clases, migre todas las clases de la cadetería a una biblioteca de clases y reimplemente la interfaz de usuario para que consuma esta biblioteca de clases desde una segunda aplicación de consola.

Para realizar dicha tarea puede utilizar la siguiente documentación:

<https://learn.microsoft.com/es-mx/dotnet/core/tutorials/library-with-visual-studio-code?pivots=dotnet-7-0>