Taller Principios

SRP – OCP – LCP

**(Single Responsibility Principle / Open Closed Principle / Liskov Principle)**

# Objetivos:

* Identificar incumplimiento a los principios SRP – OCP – LCP.
* Corregir código que incumpla los principios SRP – OCP – LCP.
* Utilizar Github como herramienta colaborativa para cargar los proyectos.
* Lengua de programación JAVA

# Instrucciones

* En taller se debe realizar individualmente.
* Usted trabajará con el proyecto en un repositorio de Github. El proyecto está compuesto por 6 paquetes (2 por cada principio, “Wrong” y “Good”).
* Descargue el proyecto desde el repositorio llamado: https://github.com/neortiz1
* Los paquetes que contienen **‘mal’** en su nombre, contiene clases e interfaces que incumplen un principio. Las tres primeras letras del nombre del paquete le indican el principio que se está incumpliendo.
* Analice el código y reestructure las clases (Refactorizar), de modo que el principio SOLID se esté cumpliendo. Agregue su propuesta de la solución en los sub-paquetes **‘Good’** respectivos.

# Conteste

Explique brevemente como cada principio se incumplió en el código analizado.

* SRP: La función “remainigFuel” de la clase vehículo no cumplía con solo una responsabilidad, puesto que solo se debería dar una por clase y si no es así procedemos a refactorizar.
* OCP: Se incumple cuando necesitemos añadir otra clase debido a que el diseño es cerrado - cerrado, entonces 2 clases cambiarán “DrivingMode” y “EventHandler”.
* LSP: No existe una herencia jerárquica entre las clases “Bird” con “Ostrich” y “FlyingBird” y a su vez “FlyingBird” con “Duck”.

# Entregable

* Enlace al repositorio de GitHub donde colocaron su solución a cada principio y este archivo Word.