

# CS2013: Programación III

## Laboratorio 1B

José Chávez

UTEC

# Contenido

## Ejercicios

# Contenido

## Ejercicios

# Ejercicio 1: La clase Personaje

- Utilizar la clase Personaje para simular un videojuego de rol.
- La clase debe tener atributos como: nombre, vida, ataque, defensa, magia. Utilice valores enteros para estos atributos.
- Implementar el método `cambiarNombre` para modificar el nombre de un personaje.
- Cree un personaje con determinados atributos. Luego, utilizando el constructor de copia, cree otro objeto con los mismos atributos y luego cambie su nombre.

# Ejericio 1: La clase Personaje

Solución: [Link a GitHub](#)

## Ejercicio 2: La clase Auto

- Implemente la clase Auto, donde cada objeto tenga tres atributos: color, precio y kilometraje (inicializado en 0). Utilice el constructor para configurar el color y precio.
- Crear el método viajar que reciba como parámetro la cantidad de kilómetros (entero) a recorrer y actualice el atributo kilometraje. El método debe imprimir un mensaje indicando la cantidad de kilómetros recorridos hasta el momento.
- Utilizar encapsulamiento para los atributos de la clase Auto.
- Crear el constructor copia y la función amiga mostrar que reciba como parámetro un objeto de la clase Auto y muestre por pantalla sus atributos.

# Ejercicio 2: La clase Auto

Solución: [Link a GitHub](#)

## Ejercicio 3: Jugar a los Dados

- Implemente la clase Dados para simular un juego de dados. El constructor de la clase debe recibir un número entero, indicando la cantidad dados a utilizar.
- El jugador debe escoger la cantidad de dados y almacenar sus resultados en un array dinámico.
- Si el jugador logra obtener un promedio mayor o igual a 4 gana.
- El método lanzar() debe generar un valor aleatorio para cada uno de los dados y verificar si ganó el jugador.
- En la interfaz de debe solicitar al jugar la cantidad de dados y crear dos jugadores adicionales. Cada uno con los mismo resultados, pero uno utilizando el constructor de copia y otro el de movimiento.



# Ejercicio 3: Jugar a los Dados

Solución: [Link a GitHub](#)