

## **OTROS EJERCICIOS EXTRAS (Guía POO)**

### **Ejercicio1**

Crear una clase llamada fracción que contenga 4 atributos (denominador y numerador de cada fracción). Crear la clase servicio que contenga métodos para sumar, restar, multiplicar y dividir fracciones. Mostrar por pantalla un menú de opciones para que el usuario decida que ver.

### **Ejercicio 2**

Definir la clase Tiempo, la cual tendrá la hora, minutos y segundos. Definir dos constructores: un constructor vacío y otro con la hora, minutos y segundos ingresado por el usuario. Deberá definir además, los métodos getters y setters correspondientes, y el método imprimirHoraCompleta(). Recordar crear una validación sobre las horas, minutos y segundos ingresados por el usuario. Te animas a simular el paso del tiempo en consola?????

### **Ejercicio 3**

Crear una clase Pass, donde declaremos los siguientes atributos:

- pass : Del tipo String
- nombre: Del tipo String
- dni: Del tipo int.

Métodos para desarrollar:

- Ingresar datos de usuarios (usando el constructor. HACER INGRESO DE DATOS FIJOS (no por teclado)
- Crear un método para ingresar una contraseña (**crearPass**). En este método, debemos validar que la longitud sea correcta. (10). En caso de ser correcto, almaceno la contraseña. Si ingreso a esta opción del menú, indefectiblemente deberá dejar su contraseña asignada.
- Crear un método para analizar la contraseña(**analizarPass**). Donde:
  - o Si Existe al menos una letra z : Es nivel MEDIO
  - o Si Existe al menos una letra z y al menos 2 letras a: Es nivel ALTO
  - o Si ninguna condición se cumple es nivel BAJO
- Crear los métodos correspondientes para modificar el nombre o DNI. Atención! Primero debe ingresar el pass para poder realizar dicha gestión. Caso contrario se impedirá el acceso a modificar los datos.
- A través de un menú de opciones:
  - o A) ingresar la contraseña, recordar que antes de guardarla debe ser ingresada una contraseña valida).
  - o B) Mensaje al usuario que tipo de NIVEL es su contraseña
  - o C) Modificar contraseña, donde primero debe poner su contraseña anterior para dar permiso
  - o C) Modificar nombre, donde primero debe poner su contraseña para dar permiso, SINO impedir cambios
  - o D) Modificar DNI, donde primero debe poner su contraseña para dar permiso, SINO impedir cambios

### **Ejercicio 4**

Definir la clase Fecha. La Clase entidad tendrá solo 3 atributos (día, mes, año). No se podrá utilizar ningún tipo de objeto de "clase fechas". Inicializando desde la entidad los valores en 1, 1, 1900 respectivamente.

Se deberá en métodos independientes:

- Verificar que el año ingresado por el usuario este entre 1900 y 2021, caso contrario, el año tomara el valor por defecto.
- Conocer la cantidad de días totales que tiene el mes elegido por el usuario. Ej: USTED INGRESO MES 10 (OCTUBRE), QUE TIENE 31 días

- Mostrar día anterior a la fecha ingresada (Hay que considerar que sucede con inicio de cada mes)
- Mostrar día posterior a la fecha ingresada (Hay que considerar que sucede con fin de cada mes)
- Crear un método para verificar si el año ingresado es bisiesto

### **Ejercicio 5**

Definir una clase triangulo que modelara triángulos isósceles . Definir los atributos necesarios para operar. Crear los métodos correspondientes a la clase Entidad.

Crear un arreglo con 4 objetos de la clase

Crear los siguientes métodos:

- Calcular área
- Calcular perímetro
- Mostrar los datos del triangulo que tenga el área de mayor superficie

### **Ejercicio 6**

Crear una clase Sopa de letras que contenga un atributo matriz, y otro palabra a encontrar.

Llenar una matriz de 10x10 con palabras de 5 caracteres (todas de forma horizontal, en orden) inicializándola por defecto con "palabras preestablecidas".

Crear métodos:

- Que el usuario ingrese una palabra y la busque en la matriz. Deberá retornar en que posición de la matriz inicia la palabra.
- Imprimir la sopa de letras en pantalla
- Que el usuario pueda reemplazar una palabra específica de la matriz ya pre -cargada (Es decir, por ejemplo, que si en posición 0.0 estaba perro, cambiarla a polo). Para realizar esta gestión, el método recibirá la palabra a buscar y a reemplazar sus datos
- Imprimir la sopa de letras, invertida (es decir, filas por columnas)

### **Ejercicio 7**

Crea una clase Auto. Como atributos tendrá nombre del dueño, fecha vencimiento carnet, color del vehículo, modelo y KM en motor (deberá ser inicializado por defecto en 7500km),

Crear métodos que permitan:

- Que un usuario cargue su ficha del auto.
- Que el usuario, modifique la titularidad del vehículo.
- Indicar un trayecto recorrido, modificando de este modo los KM del motor
- Crear un método que indique si es necesario realizar Service al vehículo, considerando que al llegar a los 10.000km corresponde realizarlo.