Informe del Tp N6 de Programacion II

\*Enunciado P1: Imagina que estás desarrollando un sistema para gestionar una red de dispositivos IoT

(Internet of Things). Los dispositivos pueden ser sensores de temperatura, cámaras de

seguridad, o altavoces inteligentes, entre otros:

• Un dispositivo se pude encender, apagar y obtener su Estado.

• Implementa por lo menos tres dispositivos concretos.

• Se debe registrar el historial de estado del dispositivo y el mismo debe ser utilizado por

cada dispositivo.

Como yo lo realice:

Primero defini una clase abstracta llamada "Dispositivo" a la cual le puse como atributo un String nombre, boolean estado y un List<String> historial, todos con "protected", a su vez defini su respectivo constructor y en el caso del estado lo iguale a un false cosa que por defecto empiece apagado, luego agregue los public void encender que cuando este es igual a "true", es porque esta encendido y un public void apagar que cuando es "false" es porque esta apagado, luego utilice las siguientes librerias: (java.time.LocalDateTime y java.time.format.DateTimeFormatter) para poder mas adelante poder mostrar la hora en la que se realizo el compilado. Obviamente agregue los Getter de cada atributo para mas adelante hacer un menu cosa que quede mas comodo y prolijo el desarrollo del problema, por ultimo le hago una sobreescritura de String cosa que me devuelva la informacion que quiero mostrar. Ahora defino otra clase llamada "AltavozInteligente" que hereda atributos de mi clase padre "Dispositivo" y como atributo extra le elijo colocar un int volumen, tambien le quise poner un int bateria pero al final me decidi por no utilizarlo pero lo deje como comentario, despues hice su respectivo constructor, su setter para asignarle un valor a este atributo mas tarde y su respectiva sobreescritura con string para que me devuelva la informacion que quiero mostrar. Cree otra clase mas llamada "Freezer" la cual hereda tambien de su clase padre "Dispositivo" y como atributo extra le coloque un boolean congelar, si es "true" congela, si es "false" no congela, a su vez, le agregue su respectivo constructor con su Setter y su sobreescrituca de string. Por ultimo hice una ultima clase(no vi la necesidad de usar interfaces en este punto) esta clase la llame "Termometro" que hereda de su clase padre "Dispositivo" donde como atributo le coloque un "double temperatura", a su vez, tambien le hago su respectivo constructor, su setter y su sobreescritura con string, ahora viene lo complicado, mi Main, de por si este punto me tomo 3 dias en poder realizarlo, despues de hacer mi menu con arrayslist, scanner, println, con do wile, switch, no me alcanzan los dedos para decir cuantas veces no me salia bien el codigo, cuando solucionaba algo, otro problema salia y asi estuve todo el dia viernes, fue un estres total, al final salio bien como esperaba, creo que tiene margen de mejora tambien porque esta lejos de ser perfecto.

\*Enunciado P2: Desarrolla un sistema para gestionar tareas en un proyecto colaborativo:

• Una tarea tiene una fecha de inicio, fecha de fin y responsable al que se le asigna la

tarea.

• Implementa por lo menos tres tipos de tareas concretas.

• Se debe poder calcular el tiempo total invertido en cada tarea y generar un informe de

progreso.

Como yo lo realize:

Primero defini una clase abstracta como "TareaBase" a la cual como atributo le coloque un String titulo, double estimacionhoras y un string responsable, su respectivo constructor y sus getter para mas tarde me devuelva el valor, tambien unas sobreescritulas de double estimacionhoras y una sobreescritura de string, con la intencion de que mas adelante me devuelvan el valor que yo les asigne. esta clase abstracta implementara 3 interfaces llamadas "Progresable", "informable","Calculable " que extiende a mi clase abstracta. Tambien cree 3 clases mas llamadas "TareaAvance ", "TareaDesarrollo" y "Tareadeprueba", todas heredan de su clase padre "TareaBase" en TareaAvance como atributo tengo un int avanceHecho y un int avanceTotal, a su vez le hago su constructor y sus Getter para usarlos mas adelante, tambien una sobreescritura de double porcentajeAvance donde utilizo una formula para sacar el porcentaje del avance de la tarea, en mi otra clase "TareaDesarrollo" como atributos le puse un int moduloCompletos y un int moduloTotales, tambien les hice sus constructores y sus Getter, por ultimo una sobreescritura de un double porcentajeAvance con su respectiva formula para que despues me devuelva el valor que le estoy pidiendo.Por ultimo mi ultima clase "Tareadeprueba" le coloque atributos como int casosHechos y int casosTotales a los cuales le coloque un constructor y tambien sus Getters y una sobreescritura con double porcentajeAvance para que me devuelva el porcentaje de los casos hechos y totales. Ya en mi clase main, volvi a aplicar la misma idea del punto anterior que fue la de realizar un menu, casualmente tambien fue donde mas tuve inconvenientes ya que de primera mano no me compilaba por un error en las interfaces, despues que habia puesto mal la formula de porcentaje lo que hacia que me de 0% de progreso, despues tuve que rehacer mis tres clases "TareaAvance ", "TareaDesarrollo" y "Tareadeprueba" porque en TareaDesarrollo no me salia los modulos completos y totales, en TareaAvance no me salia los avances hechos y los totales y en Tareadeprueba no me salia los casos hechos y casos totales, fue mucho estres el segundo punto, despues de unas horas pude estabilizar y hacer andar el codigo.

Enunciado P3:

Una empresa tiene un sistema de gestión de productos en su almacén que maneja diferentes tipos de productos, como productos alimenticios, electrónicos y ropa. Cada tipo de producto tiene características y comportamientos únicos que deben ser gestionados de manera adecuada. Se requiere que los productos compartan algunas funcionalidades comunes, pero también que tengan funcionalidades específicas según su tipo.

• Todos los productos tienen un nombre y un precio. Estas propiedades deben estar presentes en todos los tipos de productos.

• Se espera que todos los productos puedan calcular su precio y mostrar sus detalles (nombre, precio y características específicas).

• Se debe implementar un mecanismo para verificar si un producto alimenticio está próximo a vencer.

• Los productos electrónicos tienen un período de garantía que debe ser mostrado junto con los detalles del producto.

• La ropa tiene un talle que debe ser mostrado junto con los detalles del producto.

Como yo lo realice:

Para este punto armé una clase abstracta Producto con los atributos básicos (nombre y precio) y usé interfaces para que todos los productos puedan calcular su precio y mostrar sus detalles.

Después cree 3 clases (Alimenticio, Electronico, Ropa que agrego sus respectivos atributos como vencimiento, garantía o talle. (todas heredando de su clase padre)

Lo más fácil fueron los electrónicos y la ropa, porque era solo mostrar un dato extra. Lo que más me costó fue manejar la fecha de vencimiento de los alimentos; al principio me enredé con “Date”, pero lo solucioné usando LocalDate, que me permitió fijarme mas fácil si un producto estaba por vencer, luego en mi main realize otro menu guiandome de los puntos anteriores, El menú lo hice con un Scanner y un do-while, con opciones simples. Usé Instanceof para distinguir el tipo de producto.

Al final quedó ordenado con herencia, interfaces y ya casi sin margen de error debido a que fue bastante similar al anterior punto solo que agregandole un par de datos extra.