0485.- Programación.

U4U5 - Examen - 1920 - Tarde

EXAMEN UNIDADES 4 y 5 CURSO 19-20 TURNO TARDE

Debemos crear una clase Minecraft. Esta clase contendrá un array de Materiales que gestionará y que nunca podrá exceder de 10.
Los Materiales son una clase no instanciable de los que guardaré:

- El nombre
- La masa (entre 0 y 1000)
- La capacidad de quemarse (entre 0 y 100)
- La capacidad de diluirse (entre 0 y 100).
- Si son movibles o no.

En relación al array de Materiales debo de tener en cuenta que:

- En cualquier momento puedo añadir un Material pero nunca podré tener más de 10.
- En el array de materiales nunca puedo tener un material que no tenga masa (la masa alcanza un valor menor o igual a cero). Si existe en la lista debo de actualizarla para borrarlo. Para ello, debo implementar el método borrarMaterialSinMasa().
- Puedo mostrar el estado de todos los materiales por pantalla mediante el método mostrarEstado().
- Puedo mostrar el array de materiales ordenados por defecto por masa, aunque adicionalmente puedo mostrarlos ordenados por capacidad de quemarse o de diluirse.
- Debo de tener también un método ultimoMaterialQueQueda() que me diga, cuando hay un sólo (uno solo en todo el array) material sin masa, y que muestre toda su información por pantalla.

Aunque los materiales no son instanciables tengo a su vez distintos tipos:

- Cristal de los que guardaré también el tipo (TRANSPARENTE o TRANSLUCIDO).
- Metal de los que guardaré si es imantable o no.
- Rocas de los que guardaré la dureza.
- Y otra clase no instanciable especial de materiales, llamada Herramientas de la que derivan Sierra de la que me guardo el diámetro de la sierra y Pico del que me guardo el grosor del mismo en milímetros.

Todas estas clases deben implementar su propio método toString() para mostrar toda la información.

No todos los materiales se comportan igual:

- Cristal, Metal y Rocas implementarán la interfaz Mezclar que contiene un método MezclarConMaterial() que recibe un Material como parámetro. Cuando se ejecute el método, sacará por pantalla el nuevo material que se puede formar. Por ejemplo "Cristal de ópalo".
- Pico implementará la interfaz Minar que contiene dos funciones Hacer() y Deshacer() que reciben un Material como parámetro y cuya implementación por defecto es que reciben 100 de masa en Hacer y en Deshacer se le resta 100 de masa.

Todos los atributos deben ser privados. Deben realizarse los correspondientes set/get.

Crear en el Main un ejemplo que contenga al menos 5 materiales, que muestre al menos 2 mezclas, una con mismo material y otra distintos materiales, 11 procesos de Hacer y Deshacer y que tras todo eso, muestre la lista de Materiales atendiendo a todos los criterios de ordenación solicitados.

Instruciones para la Entrega

Entregar un **archivo comprimido (zip o rar)** que contenga cada uno de los ficheros correspondientes a los ejercicios (EjX.java siendo X el número del ejercicio)

EL fichero comprimido se llamará **Apellido1_Apellido2_U4-U5_Examen.zip** (o rar)

Evita ñ y acentos en el nombre.

Deberan subirse a gitHub los ficheros en la carpeta correspondiente (U5_Examen) que se encontrará dentro de la carpeta U5