

Dado que disponéis de todo el material de clase ,internet etc.

Se pide funcionalidad, de modo que si el apartado no funciona su puntuación será 0, si funciona en algun caso pero en otros no, como máximo podrá alcanzarse ¼ de la puntuación del apartado.

Se penalizará código ineficiente, nombres no significativos.

Ejercicio2.- Duración del ejercicio2 : 45 minutos

(5 puntos)

Un transbordador espacial, es capaz de trasportar material desde la Tierra hasta la estación orbital Mxx Sea la clase *Material*, que representa el nombre, peso por unidad (en gramos) , unidades de un determinado material transportado, y si el material es peligroso o no lo es. **La clase Material no debes modificarla, ni añadirle nada, solo usarla.**

```
public class Material {
    private String nombre;
    private double peso;
    private int unidades;
    private boolean peligrosidad;
    public Material(String nombre, double peso, int unidades, boolean peligrosidad) {
        this.nombre = nombre;
        this.peso = peso;
        this.unidades = unidades;
        this.peligrosidad = peligrosidad;
    }
    public String getNombre() {
        return nombre;
    }
    public double getPeso() {
        return peso;
    }
    public int getUnidades() {
        return unidades;
    }
    public boolean isPeligrosidad() {
        return peligrosidad;
    }
}
```

Los materiales se mandan guardados en contenedores estancos. Por motivos de seguridad, las condiciones del contenedor son (*): en un contenedor **no se puede incluir más de un material peligroso, ni 2 materiales con el mismo nombre.**

Se pide **completar** la clase *Contenedor*, que representa los materiales existentes en un contenedor, y cuyo constructor es:

```
public class Contenedor {
    private Material[] materials;
    private int cont = 0; // contador de materiales diferentes en el contenedor
    private boolean peligroso = false; // indicador de peligrosidad del contenedor, será true si hay un material peligroso en el contenedor
    public Contenedor(int numMaximo) { // numMaximo es el número máximo de materiales diferentes que puede tener el contenedor
        materials = new Material[numMaximo];
    }
}
```

con los métodos que se piden a continuación, y solo esos métodos (no cambies nada del código proporcionado , solo debes usarlo y completarlo) :

1.desarrollando el método add, para añadir un nuevo material al contenedor si es posible con las condiciones expresadas en la descripción del contenedor (*):

la firma del método debe ser **public boolean add(Material material)**

2.5ptos

2.desarrollando el método deMasDe, que devuelve un array de materiales cuyas unidades son superiores a las unidades que entran como parámetro, su firma es **public Material[] deMasDe(int unidades)** **1 pto**

3.desarrollando el método composición que visualiza la composicion exacta de contenedor, indicando en la primera linea: contenedor peligroso o contenedor no peligroso de <<numMaximo>> materiales como maximo, según proceda y luego linea a linea nombre del material, material peligroso o no, unidades y peso, la firma del método es :

public void composicion()

1pto

Como solución debes mandar, **en un solo archivo comprimido**

1) El archivo **Contenedor.java** (código fuente)

2) El archivo **PruebaContenedor.java** que pruebe tu desarrollo, usando la clase Material y la clase Contenedor y los métodos desarrollados anteriormente, instanciando un **contenedor de 3 materiales**, que muestre :

a) que se puede añadir un material al contenedor si se dan las condiciones para ello

b) que no se puede añadir un material al contenedor si ya existe ese material en el contenedor

c) que no se puede añadir un nuevo material peligroso en el contenedor si ya existe un material peligroso en él.

d) que no se puede añadir un nuevo material al contenedor si ya hay 3 materiales diferentes

e) la composición exacta del contenedor.

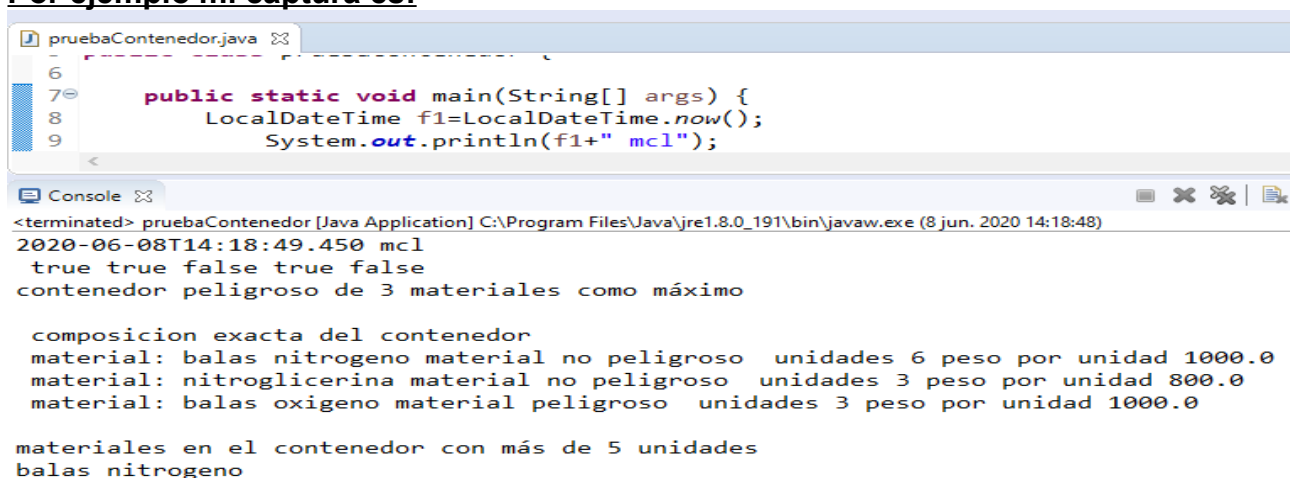
f) el nombre de los materiales contenidos en un array con los materiales transportados cuyas unidades sean superiores a 5. 1pto

3) **Captura de pantalla, de la prueba del desarrollo**, esta captura debe tener la hora de ejecución y tu nombre, con `LocalDateTime f1=LocalDateTime.now()` :

`System.out.println(f1+" el nombre del alumno");`

Sin la captura de pantalla de la ejecución con hora y nombre este apartado no puntúa

Por ejemplo mi captura es:



```
pruebaContenedor.java
6
7 public static void main(String[] args) {
8     LocalDateTime f1=LocalDateTime.now();
9     System.out.println(f1+" mcl");
}

Console
<terminated> pruebaContenedor [Java Application] C:\Program Files\Java\jre1.8.0_191\bin\javaw.exe (8 jun. 2020 14:18:48)
2020-06-08T14:18:49.450 mcl
true true false true false
contenedor peligroso de 3 materiales como máximo

composicion exacta del contenedor
material: balas nitrogeno material no peligroso unidades 6 peso por unidad 1000.0
material: nitroglicerina material no peligroso unidades 3 peso por unidad 800.0
material: balas oxigeno material peligroso unidades 3 peso por unidad 1000.0

materiales en el contenedor con más de 5 unidades
balas nitrogeno
```