Ejercicios Adicionales - Módulo Objetos **Taller de Programación 2021**

Ejercicio 1. Un Laboratorio realiza experimentos para evaluar la eficacia de un nuevo fármaco para la diabetes y para ello conformó dos grupos de pacientes: Grupo Alfa y Grupo Beta.

Ambos grupos registran la información de a lo sumo 10 pacientes (identificados de 1 a 10). De cada paciente se guarda: nombre, último resultado de glucosa (double) y última dosis recibida de fármaco (double). Sin embargo, los grupos difieren en la forma de aplicar el fármaco a los pacientes *(esto se detalla más adelante)*

1) Genere las clases necesarias, cada una con los constructores, estado, getters y setters adecuados. Tenga en cuenta que los grupos inicialmente no tienen pacientes.
2) Agregue a la clases que corresponda los métodos necesarios para:
   a) Agregar un paciente *P* al grupo y retornar su número identificatorio en el grupo (ID).
   b) Obtener un paciente del grupo dado un ID válido (1 a 10).
   c) Aplicar una dosis a un paciente. Se recibe una dosis *D* (double) y se debe modificar su última dosis recibida a D y disminuir la glucosa en un valor aleatorio entre 0 y 1.
   d) Aplicar una dosis *D* (double) de fármaco a los pacientes del grupo, teniendo en cuenta que: en el Grupo Alfa se le aplica la dosis *D* a todos los pacientes; en el Grupo Beta se le aplica la dosis *D* a los pacientes cuya glucosa supera el valor 2.5.
   e) Obtener la representación string del grupo, la cual se compone por el ID,. nombre, última glucosa y última dosis de todos los pacientes del grupo
3) Realice un programa que instancie un Grupo Alfa y un Grupo Beta. Llene cada grupo con pacientes (el primero con 3 y el segundo con 4). Aplique una dosis *D* de fármaco (leída por teclado) a los pacientes de cada grupo. Imprima la representación string de cada grupo.

Ejercicio 2. Queremos representar estanterías de libros. Una estantería mantiene sus libros organizados en N estantes cada uno con lugar para M libros. Un libro posee título, peso, cantidad de páginas y su autor (del que se registra nombre y biografía).

a) Implemente las clases de su modelo, con sus atributos y getters/setters adecuados.

b) Provea constructores para iniciar autores y los libros a partir de toda su información.

c) Provea un constructor para iniciar la estantería para N estantes y lugar para M libros por estante (inicialmente no debe tener libros cargados).

d) Provea un constructor para iniciar la estantería para 5 estantes y lugar para 10 libros por estante (inicialmente no debe tener libros cargados).

e) Implemente los siguientes métodos:

- *almacenarLibro*: recibe un libro, un nro. de estante y nro. de lugar válidos y guarda al libro en la estantería. Asuma que dicho lugar está disponible.

- *sacarLibro*: recibe el título de un libro, y saca y devuelve el libro con ese título, quedando su lugar disponible. Tenga en cuenta que el libro puede no existir.

- *calcularLibroMasGrande*: calcula y devuelve el libro con más páginas de la estantería.

- *calcularEstanteMasPesado*: calcula y devuelve el número del estante más pesado (teniendo en cuenta el peso de sus libros).

d) Realice un programa que instancie una estantería para 5 estantes y 3 libros por estante. Almacene 7 libros en la estantería. A partir de la estantería: saque el libro “2001 Odisea del Espacio” e informe su representación String; luego, informe

- el título del libro más grande
- el número del estante más pesado.