

Se necesita administrar un depósito de computación para lo cual contamos con la siguiente información.

Cada elemento de computación cuenta con: modelo, marca, número de serie, identificador (este valor no puede cargarse, ya que es el resultado de unir modelo-marca-número serie)."

A su vez contamos con las siguientes tipificaciones:

- **Monitores:** del cual conocemos su año de fabricación, si es nuevo o usado (esto es determinado por el año de fabricación = año actual), pulgadas (esta información es un número entero pero en algunos casos puede no tener valor).
- **Computadoras:** del cual conocemos la descripción del procesador, la cantidad de memoria RAM (que es un valor entre los siguientes; 2GB, 4GB, 8GB, 16GB) y el nombre del fabricante.

Se necesita desarrollar una clase global que tenga una ÚNICA instancia (validada por la arquitectura). La clase debe compartir con la instancia las listas de productos que hay en el depósito.

Además esta clase global debe permitir a la instancia creada:

1. Agregar un producto a la lista. Generar una sobrecarga de métodos para cada subtipo de producto.
2. Eliminar un producto de la lista pasando como parámetro el identificador. Validar que el identificador exista antes de eliminar el objeto de la lista.
3. Obtener de cada producto una descripción del estilo:
 - a. Monitores: "MONITOR {marca} - {modelo} {pulgadas}" ". En el caso que no tenga definido un valor para pulgadas no debería mostrar nada.
 - b. Computadoras: "PC {modelo} - {marca} - {cpu} {ram} RAM - {fabricante}"
4. Obtener una única lista de objetos del mismo tipo que contenga las dos listas anteriores ordenada por tipo de producto.

Tanto en la adición de productos como en la eliminación, al momento de consumarse el hecho, se deberá lanzar un solo evento de "producto agregado/eliminado" con la información del tipo de producto y el identificador. Estos métodos no deben retornar ningún valor.

Todos estos métodos deben estar contenidos en el proyecto de **Lógica**.

En la aplicación de consola, además de permitir el ingreso de datos por teclado para agregar un elemento nuevo (elegir la opción de ingreso de datos que les parezca conveniente) se debe:

Crear un método de extensión para validar que el valor ingresado como memoria RAM esté en el rango establecido por el enumerador anterior. Usar este método en la carga de datos.

Crear un manejador de evento "producto agregado/modificado" que imprima en pantalla la descripción de cada producto (método 3) resultado del método de obtención de productos así como también la cantidad de productos actuales por tipo de producto y el porcentaje de los mismos (método 4).

EJ: "Producto agregado X, Total de Monitores: 20, Total de Computadoras: 40, Monitores: 33%, Computadora: 66%"

Como este método retorna la lista ordenada por tipo de producto se necesita identificar al último producto ingresado, para lo cual vamos a imprimirlo con un color distinto a los demás.

- Para esto usar *Console.ForegroundColor*