

*Leé completo y con cuidado el enunciado. Pensá bien la estrategia de resolución antes de comenzar el desarrollo de lo que te solicitan. El objetivo de este examen es evaluar la correcta aplicación de los conceptos y técnicas vistos hasta el momento:*

* *Correcta implementación de constructores.*
* *Modularización reutilizable y mantenible con uso de métodos con correcta parametrización y correcto encapsulamiento, publicando setters y getters sólo cuando corresponda.*
* *Manejo de clases y colecciones*
* *Manejo de Asociación, Agregación y Composición.*
* *Manejo de relaciones jerárquicas. Herencia, Clases Abstractas e Interfaces.*
* *Manejo de Arreglos uni y bi dimensionales.*
* *Manejo de TDAs – Pilas, Colas y Listas ordenadas*
* *Importación y Exportación de proyectos Java desde Eclipse*

Para realizar el examen

* Descargar el archivo TP1-2022-C1-Parcial2-ApellidoNombre.zip del aula virtual
* Importar en Eclipse el proyecto descargado y renombrarlo con tu apellido y nombre. Por ejemplo: TP1-2022-C1-Parcial1-PerezJuan
* Al finalizar el examen, exportarlo en un archivo ZIP con el mismo nombre del proyecto y subirlo en el aula virtual en el link correspondiente al examen, el archivo a subir debe llamarse igual que el proyecto.

**NOTA: DEBE RENOMBRAR EL PROYECTO INMEDIATAMENTE LUEGO DE IMPORTARLO EN ECLIPSE**

**Enunciado**

Una juguetería nos pide el desarrollo de un programa que permita gestionar los productos que expone en el local.

La juguetería almacena sólo tres tipos de juguetes: Pelotas, Títeres y Peluches. Todos los juguetes tienen varios datos en común y otros particulares para cada tipo de juguete. Los comunes son:

* Un id (entero, no puede ser 0 ni negativo);
* Una marca (del tipo texto, no puede ser nula ni vacía);
* Un precio (número real, no puede ser menor a 1).

Cada tipo de juguete tendrá, además, esta información extra:

* Los títeres, el tamaño en centímetros (entero), que no puede ser menor a 30 ni mayor a 70.
* Los peluches, un modelo (texto, no puede ser ni nulo ni vacío).
* Las pelotas, un tipo específico (el deporte para el que están diseñadas: Fútbol, Tenis o Golf).

Para guardar sus productos la juguetería usa una estantería con 6 niveles y 3 estantes en cada nivel. En cada estante de cada nivel se almacenará **sólo un tipo de juguete** (pelotas, títeres o peluches) y un máximo de juguetes que se colocan en fila del fondo hacia adelante. Aunque nos avisan que en un futuro la cantidad de juguetes a guardar en cada estante podría cambiar según el tamaño del producto, en una primera implementación siempre será de 4 (cuatro). Por diseño, cada estante puede ser accedido solo por su lado frontal, es decir que sólo es visible/accesible el último juguete allí almacenado.

Un detalle técnico a destacar es que tanto la **juguetería** como los **estantes** implementan **ContenedorInteligente**, lo que los habilita a guardar elementos según su tipo y luego recuperarlos basándose en el ID del elemento a retirar.

**Para aprobar el examen se deberá realizar lo siguiente:**

1. Implementar las clases **Juguete**, **Titere, Pelota** y **Peluche** agregando los atributos correspondientes y aplicando las validaciones indicadas.
2. Implementar el método ***mostrar()*** definido en la interfaz **Mostrable** donde considere pertinente en base a la salida esperada.
3. Completar la clase **Jugueteria**, agregando los atributos y métodos necesarios para cumplir con el funcionamiento
   1. Constructor deberá recibir por parámetro la profundidad de la estantería.
   2. Método ***void guardar(Juguete juguete) throws RuntimeException*** deberá intentar el guardado el juguete recibido en la primera ubicación de la fila que corresponda a su tipo y que tenga lugar. Si no hay lugar, deberá arrojarse una excepción indicando el problema.
   3. Método ***boolean existeJuguete(int idJuguete)*** deberá indicar si hay guardado un juguete con el identificador recibido.
   4. Método ***Juguete recuperarPorId(Integer id)*** deberá sacar el juguete de la estantería. Si no se encuentra se debe devolver null.
4. Completar la clase **Estante**, agregando los atributos y métodos necesarios.
   1. Constructor deberá recibir por parámetro la capacidad máxima del estante.
   2. Método ***guardar(Juguete juguete) throws RuntimeException*** deberá intentar agregar el juguete en el estante, y si no hay lugar deberá arrojar una excepción para indicar la situación. El juguete deberá ser agregado de forma tal que se pueda retirar primero el último que se agregó.
   3. Método de ***Juguete recuperarPorId(Integer id)*** que debe retirar un juguete en base al ID indicado. Si el juguete no está deberá devolver null. El estante debe quedar ordenado como estaba antes de la operación.
5. Completar la gestión de errores en la clase **Parcial2**. Solo se debe modificar lo que se considere pertinente para la gestión de errores, no modificar el resto de la clase.

Para probar el funcionamiento del programa se dispone de una clase de prueba **Parcial2** que contiene la creación de la juguetería y dispone de un lote de prueba con juguetes que serán almacenados.

**NOTA**: Esta clase no deberá ser modificada por los alumnos, salvo la gestión de errores.

De funcionar, el programa emite la siguiente salida por consola...

| Guardando juguetes en la jugueteria  --------------------------------------------  Peluche depositado id: 10  Error depositando titere: Tamanio de titere invalido.  Pelota depositada id: 12  Error depositando Peluche: La marca no puede ser nula ni vacia  Error depositando titere: Tamanio de titere invalido.  Peluche depositado id: 15  Error depositando titere: Tamanio de titere invalido.  Pelota depositada id: 17  Error depositando Peluche: Modelo de peluche invalido.  Error depositando titere: Tamanio de titere invalido.  Error depositando pelota: La superficie no puede ser nula  Pelota depositada id: 21  Pelota depositada id: 22  Pelota depositada id: 23  Pelota depositada id: 24  Pelota depositada id: 25  Pelota depositada id: 26  Pelota depositada id: 27  Pelota depositada id: 28  Pelota depositada id: 29  Pelota depositada id: 30  Error depositando pelota: No se pudo depositar el juguete.  Peluche depositado id: 32  Fin del guardado  --------------------------------------------  El producto id 55 NO esta depositado  El juguete id 23 esta depositado  --------------------------------------------  Se comienza a comprar juguetes  --------------------------------------------  Se compro el producto id: 12  La pelota cuesta 1234.50 pesos, es del tipo Futbol ademas es de la marca Wilson y tiene un id 12  No se pudo comprar producto id: 13  Se compro el producto id: 28  La pelota cuesta 1234.50 pesos, es del tipo Tenis ademas es de la marca Wimbledon y tiene un id 28  Se compro el producto id: 15  El peluche cuesta 2300.50 es de modelo All Team es de la marca Marvel y tiene un id 15  Final del compras  --------------------------------------------  El total de ventas fue de 4769.5  -------------------------------------------- |
| --- |

**La entrega del examen debe completarse antes de cumplido el plazo establecido y debes asegurarte de terminar el intento y completar la entrega con “Enviar todo y Terminar”.**