

CÓDIGO: _____ **NOMBRE:** _____

Reglas del juego

- Durante este examen usted no puede pedir ABSOLUTAMENTE nada prestado a sus compañeros, ni hablar con ellos.
- Su nombre y su firma a la aceptación del compromiso de no hacer fraude, en la hoja de respuestas a este examen, deben ir en lapicero. Si emplea más de una hoja márkelas TODAS de igual forma.
- Por ningún motivo puede salir del salón, antes de terminar el examen. De manera que si se retira se considerará que terminó su trabajo.
- Puede consultar sus notas, el libro o apuntes; pero se le recomienda que lo haga cuando esté seguro de qué quiere buscar, en caso contrario estas consultas se convierten en una distracción que le hace perder tiempo.
- No se resolverá ninguna duda durante el examen, así que si algo no le queda claro y usted requiere hacer algún tipo de supuesto, ESCRÍBALO y de una breve explicación de la razón por la cual lo supone. Tenga presente que un supuesto no puede contradecir lo que se le está diciendo.
- Debe escribir con letra clara.

La forma de evaluar este parcial es la siguiente:

Ítem	Valor
Punto 1	0.5
Asociación correcta entre cada definición y concepto	0.05
Punto 2	3.0
Identificación de clases con sus atributos y métodos básicos	1.0
Identificación de todos los métodos (diferentes a los básicos)	0.5
Identificación de relaciones	1.0
Interfaces	0.5
Punto 3	1.5
Completar el diagrama de clases	0.3
Describir el método solicitado	0.6
Código del método solicitado	0.6

Punto 1 (Vale 0.5 puntos)

A continuación encontrará una tabla que presenta definiciones y conceptos sobre los temas estudiados en el curso. Relacione cada una de las definiciones de la columna A con el concepto apropiado en la columna B. (Se presentan más conceptos que definiciones, quedarán conceptos sin asociar; sin embargo, cada concepto de la columna B debe asociarse a una y solo una definición de la columna A)

COLUMNA A		COLUMNA B
1. Mecanismo que permite detener la ejecución de un hilo A mientras que otro hilo B termina su ejecución.	()	Excepciones que se derivan de la clase Exception pero no de RuntimeException.
2. Condiciones indeseables que pueden presentarse en un programa y que deben evitarse.	()	FileReader
3. Realiza procesamiento adicional cuando se realiza una lectura de datos desde un archivo.	()	+ join():void
4. Mecanismo que permite establecer una especie de "tipo común" a un conjunto de clases que no pertenecen a una misma jerarquía de herencia.	()	Colección
5. Tipo de colección que no facilita el acceso a cualquiera de sus objetos dada su posición.	()	Interfaces
6. Excepciones que se verifican durante la fase de compilación del código.	()	Serialización
7. Permite la implementación de mecanismos que permiten conectarse a un archivo de texto para escribir información.	()	Listas Enlazadas
8. Objeto con capacidad de contener referencias individuales a muchos objetos	()	BufferedReader
	()	Interfaces bandera
	()	Excepciones que se derivan de la clase Error o de la clase RunTimeException.

Punto 2 (Vale 3.0 puntos)

Un supermercado está necesitando una aplicación orientada a objetos, desarrollada en Java, que permita gestionar los procesos de administración de la información de sus productos, clientes, proveedores y el registro de las ventas. El sistema también debe permitir la generación de ciertos listados. El administrador del supermercado suministró la siguiente descripción de los procesos del negocio, que esperan sean soportados por el sistema de información:

“Necesitamos un sistema que nos permita registrar y consultar información relacionada con nuestros clientes, productos, ventas y proveedores. Empecemos hablando de nuestros proveedores. Los proveedores son las empresas que nos suministran los productos que vendemos; para que una empresa pueda ser un proveedor nuestro, deberá registrarse primero, indicando su NIT (número de identificación), razón social, dirección, teléfono y correo electrónico; no es posible hacer negocios con proveedores que no forman parte de nuestro registro; para el supermercado es primordial que el listado de proveedores pueda consultarse de forma ordenada según su número de identificación. Ahora pasemos a nuestros clientes. Los clientes del supermercado son las personas que compran nuestros productos; lo que queremos es que el sistema permita el registro de clientes, indicando su número de identificación, nombre completo teléfono, dirección, email, profesión y fecha de cumpleaños. El sistema debe permitir el registro de productos con sus proveedores a partir de parejas clave valor, e implementar mecanismos para realizar búsquedas rápidas de dichos productos. Un producto está asociado a uno y solo un proveedor, aunque un proveedor puede vendernos más de un producto.

Todos los productos que vendemos en este supermercado se encuentran clasificados bajo una de las siguientes categorías: alimentos, productos de aseo y productos de hogar. Cada producto se identifica por un código de barras, tiene un nombre, un precio unitario, la cantidad de unidades disponibles en inventario, el inventario máximo permitido y un indicador que nos permite saber si está exento o no del impuesto de IVA. Con los productos de aseo siempre tenemos mucho cuidado, pues algunos de estos son tóxicos y deben ser ubicados en estantes del supermercado que no sean fácilmente accesibles por los niños; queremos que el sistema nos permita consultar aquellos productos de aseo que son tóxicos para asignarles una ubicación especial. En la categoría de productos de hogar tenemos muchos subtipos. No obstante, el sistema debe dar un tratamiento especial a los electrodomésticos, pues estos deben incluir el tiempo durante el cual incluyen garantía y si son importados o nacionales; y a la ropa, pues es importante saber si se trata de una prenda para dama, caballero, niña o niño

Otro proceso importante es la venta. Necesitamos que cada venta que realizamos quede registrada en el sistema con su número de factura, fecha de venta, el cliente a quien se hace la venta (en caso de que el cliente se encuentre registrado en el sistema) y el listado de productos que el cliente compra. Además de registrar la venta, el sistema debe poder calcular el valor a pagar, el cual debe registrarse y desplegarse por pantalla. Finalmente, es importante que el sistema nos permita realizar consultas de alimentos ordenados por fecha de vencimiento (los próximos a vencerse primero). ¡Ahh!, casi lo olvido, el sistema debe poder almacenar la información cada vez que termine su ejecución y recuperarla, la siguiente vez que se inicie”.

Teniendo en cuenta la descripción de procesos suministrada por el administrador del supermercado, usted debe **diseñar el diagrama de clases completo, de los paquetes modelo y control**, para dar solución a las necesidades del negocio.

Punto 3 (vale 1.5 puntos)

El administrador del supermercado para el cual usted realizó el diseño del diagrama de clases, quedó muy contento con su trabajo. Gracias a su diseño de clases, el sistema pudo ser desarrollado sin problemas.

Ahora ellos quieren que usted diseñe y desarrolle la funcionalidad para que el sistema permita almacenar en un archivo cada vez que el administrador del almacén lo desee, la información de los pedidos que deben realizarse para los productos cuyo nivel de inventario disponible se encuentre por debajo de 10 unidades. Este archivo será procesado por otro sistema que posee el supermercado, el cual se encarga de enviar un mensaje de correo electrónico solicitando cada producto a cada proveedor. El archivo debe incluir para cada producto a comprar: el código de producto, el número de unidades que deben ser compradas (teniendo en cuenta el inventario máximo), el número de identificación del proveedor, su razón social y su correo electrónico. La estructura del archivo debe ser algo como:

```
codproducto1numUnidadesidProveedorrasonSocialcorreoElectronico
codproducto2numUnidadesidProveedorrasonSocialcorreoElectronico
...
codproductoNnumUnidadesidProveedorrasonSocialcorreoElectronico
```

Uste debe:

1. Modificar el diagrama de clases para que el sistema pueda realizar la nueva funcionalidad solicitada.
2. Describir el método que genera el archivo que debe ser procesado por el otro sistema.
3. Escribir el código java para el método descrito.