

Reglas del juego

- Tal como se le dijo antes, 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
Primer problema:	1.5
Completar el diagrama de clases para implementar modelo vista control	0.1
Pregunta de falso y verdadero y justificación si es necesaria	0.3
Completar el diagrama de clases para calcular precio de venta	0.3
Descripción del método vender e implementación de su código	0.8
Segundo problema:	2.0
Diagrama de clases completo	1.5
Descripción del método que permite registrar los materiales recolectados (entradas, salidas, parámetros, retorno y proceso)	0.3
Descripción del método o los métodos necesarios para mostrar las recomendaciones de clasificación (entradas, salidas, parámetros, retorno y proceso)	0.5
Tercer problema	1.5
Identificación de puntos en el código donde podrían presentarse excepciones	0.7
Completar el código para controlar las excepciones	0.8

Punto 1

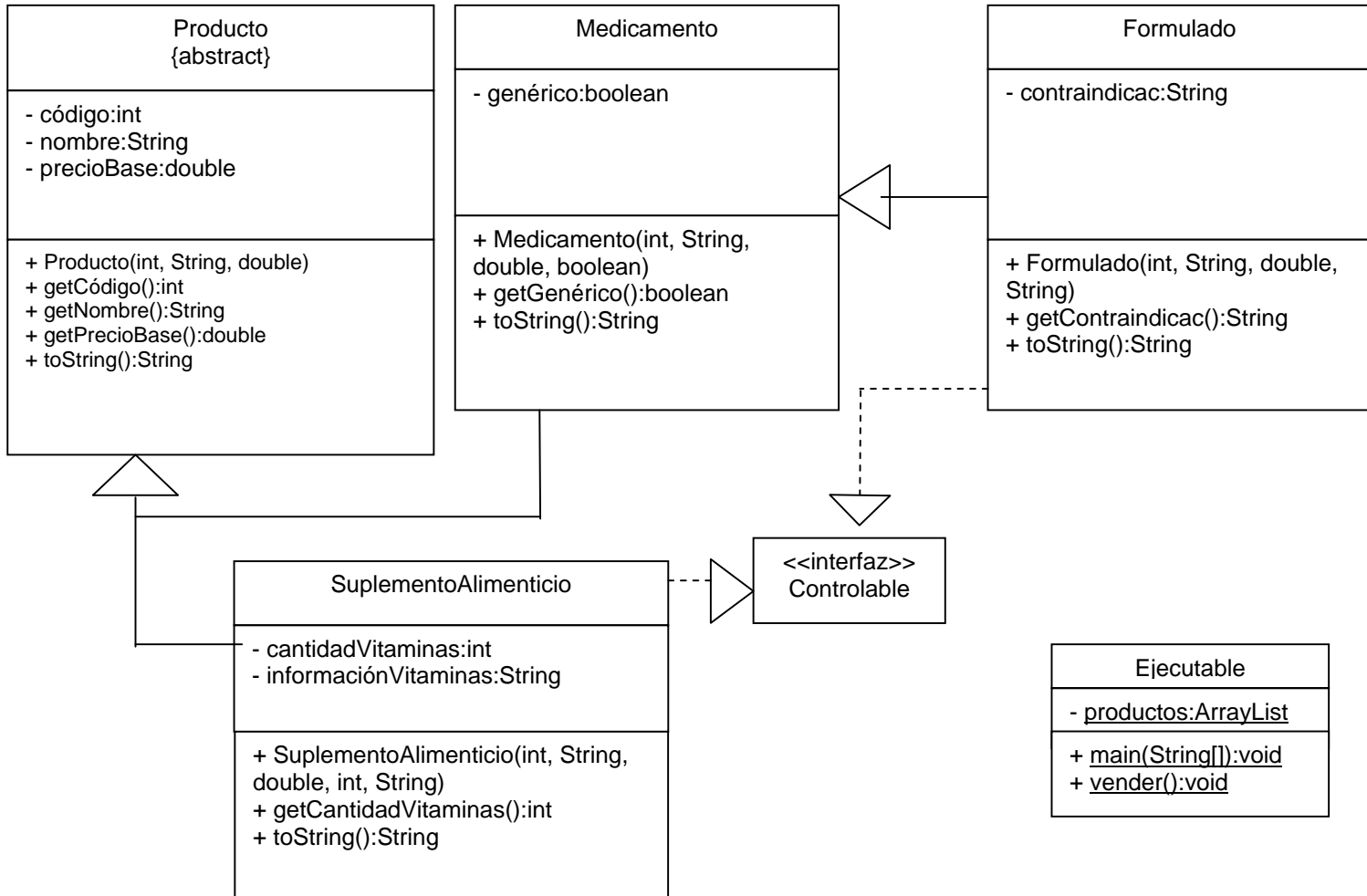
El diagrama de clases presentado a continuación ha sido desarrollado por un estudiante, a quien una importante cadena de farmacias de la ciudad, le ha solicitado el desarrollo de una aplicación en Java que le permita registrar la venta de sus productos.

La cadena de farmacias maneja dos tipos de productos: medicamentos y suplementos alimenticios. Existe un tipo particular de medicamento denominado "medicamentos formulados" de los cuales se almacenan sus contraindicaciones. Para los suplementos vitamínicos es importante almacenar la información referente a las vitaminas que contienen.

Cuando se realiza una venta el sistema debe solicitar el código único del producto y si se trata de un medicamento formulado o un suplemento alimenticio debe indicarle al vendedor que es indispensable solicitar la fórmula médica.

El precio de venta de un producto se calcula dependiendo de si es un medicamento o un suplemento vitamínico. Para los medicamentos, el precio de venta se calcula aplicando un recargo al precio base. Si se trata de un medicamento genérico no se aplica recargo, mientras que, si es un medicamento NO genérico se aplica un 20% de recargo.

En cuanto a los suplementos vitamínicos, el precio se incrementa en un 2% por cada vitamina que contiene.



Teniendo en cuenta el diagrama de clases y la situación anterior, resuelva cada una de las siguientes preguntas:

- (Vale 0.1 puntos)** Complete el diagrama de clases implementando el patrón modelo vista control.
- (Vale 0.3 puntos) Responda Falso o Verdadero (justificar si su respuesta es falso):** La interfaz Controlable está mal definida, no se han especificado los métodos que deben implementarse en las clases que la Implementan.
- (Vale 0.3 puntos)** Complete el diagrama de clases de tal forma que el sistema pueda calcular el precio de venta de cada producto, según la descripción dada por la cadena de farmacias.
- (Vale 0.8 puntos)** El estudiante no describió completamente los métodos, realice el análisis del método `+vender():void` de la clase Ejecutable y el código java del mismo.

Punto 2

Un grupo de estudiantes está trabajando en un proyecto ecologista que les permitirá recibir diferentes tipos de basura y realizar un proceso de reciclaje.

El proceso es el siguiente: cada estudiante perteneciente al grupo realiza un recorrido diferente por la ciudad y pasa por casa de sus mejores amigos recogiendo la basura. Cuando finaliza el recorrido llega al cuartel general y se debe encargar de clasificarla.

El grupo de estudiantes está muy interesado en que usted realice una aplicación que le permita al estudiante que llega con la basura seleccionar el tipo de basura, registrar algunos datos básicos dependiendo del tipo seleccionado y finalmente desplegar en pantalla las indicaciones de clasificación a seguir cuando se trate de materiales reciclables.

Si el material recolectado es reciclable la aplicación debe mostrar un mensaje indicando el color de la bolsa de basura en el que debe ser depositado el material. De acuerdo con la siguiente tabla:

Material	Color bolsa de reciclaje
Papel y Cartón	Azul
Vidrio	Verde
Plásticos y metales	Amarillo

Del material recibido es importante para el grupo registrar alguna información adicional:

- Papel: indispensable almacenar el gramaje y si viene o no limpio de ganchos de grapadoras, clips, etc.
- Cartón: se requiere diferenciar si es cartón paja o cartón normal.
- Vidrio: es indispensable registrar si es un envase con contenido consumible o por el contrario no lo es.
- Plástico: es indispensable almacenar el color
- Metales: importante almacenar el peso en kilogramos
- Desechos orgánicos: sólo se registra el peso.

Si los materiales recolectados son desechos orgánicos deben almacenarse en un contenedor especial para su posterior venta a empresas productoras de fertilizantes. Tenga en cuenta que estos materiales no son reciclables.

Es importante que la aplicación almacene el nombre del material y el nombre de la persona que lo suministró.

Usted debe:

1. **(Vale 1.5 puntos)** Realizar el diagrama de clases completo teniendo en cuenta que la aplicación debe permitir:
 - Registrar los materiales recolectados
 - Conocer el tipo de bolsa en el que debe ser clasificada la basura
2. **(Vale 0.5 puntos)** Realizar la descripción del método que permite registrar los materiales recolectados y el método o los métodos necesarios para mostrar las indicaciones de clasificación.

NOTA: ES INDISPENSABLE LA IMPLEMENTACIÓN DE INTERFACES PARA DAR SOLUCIÓN A ESTE EJERCICIO

Punto 3

El código que se presenta a continuación corresponde a una aplicación que fue creada para comprobar si una clase hace parte de la jerarquía de la clase Collection.

```

public static void main(String[] args) throws NullPointerException {
    Scanner lector=new Scanner(System.in);
    int opción=0;
    opción=lector.nextInt();
    switch(opción){
        case 1: System.out.println("\nPor favor digite el nombre completo de la clase");
        nombreClase=lector.nextLine();
        instanciaDeLaClase(nombreClase);
        break;
        default: System.out.println("\La opción solicitada no está disponible");
    }
}
/*Método instanciaDeLaClase
 * parámetros: Un objeto String con el nombre de la clase
 * proceso: Crea un objeto Class con el nombre de la clase que recibe como parámetro.
 * crea un objeto Object con el objeto Class
 * si el objeto Object es de tipo Collection o de alguna de sus hijas lo dice, de lo contrario
 * dice que no lo es
 */

public static void instanciaDeLaClase(String nombreClase) throws NullPointerException {
    Class clase=Class.forName(nombreClase);
    Object objeto=clase.newInstance();
    if(objeto instanceof Collection)
        System.out.println("La clase digitada por el usuario pertenece a la jerarquía de la
        clase Collection");
    else
        System.out.println("La clase NO digitada por el usuario pertenece a la jerarquía de
        la
        clase Collection ");
}

```

Usted debe:

1. **(Vale 0.7 puntos)** Identificar claramente en el código donde pueden presentarse excepciones comprobadas y no comprobadas que se puedan presentar.
2. **(Vale 0.8 puntos)** Complete el código para controlar las excepciones identificadas en el punto anterior. Tenga en cuenta que el mensaje que visualiza el usuario al presentarse la excepción debe explicar la causa de la misma en un lenguaje claro para él.

A continuación se le presenta documentación sobre excepciones que puede serle útil:

```

public Object newInstance()
    throws InstantiationException,
           IllegalAccessException
    Creates a new instance of the class represented by this Class object. The class is instantiated as if by a new
    expression with an empty argument list. The class is initialized if it has not already been initialized.

public static Class<?> forName(String className)
    throws ClassNotFoundException
    Returns the Class object associated with the class or interface with the given string name.

public int nextInt()
    Scans the next token of the input as an int.

An invocation of this method of the form nextInt() behaves in exactly the same way as the invocation
nextInt(radix), where radix is the default radix of this scanner.

```

Returns:

the int scanned from the input

Throws:

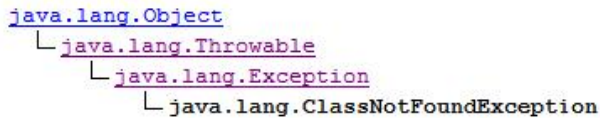
[InputMismatchException](#) - if the next token does not match the *Integer* regular expression, or is out of range

[NoSuchElementException](#) - if input is exhausted

[IllegalStateException](#) - if this scanner is closed

java.lang

Class `ClassNotFoundException`

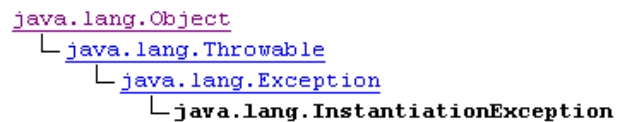


All Implemented Interfaces:

[Serializable](#)

java.lang

Class `InstantiationException`



java.util

Class `InputMismatchException`

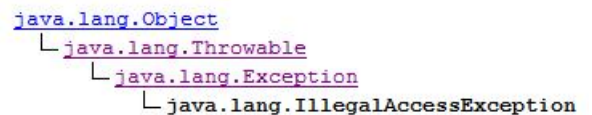


All Implemented Interfaces:

[Serializable](#)

java.lang

Class `IllegalAccessException`

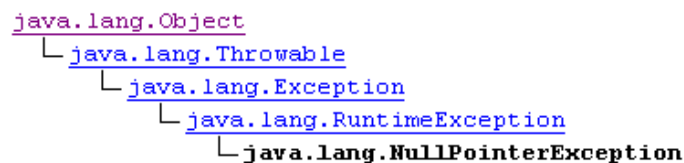


All Implemented Interfaces:

[Serializable](#)

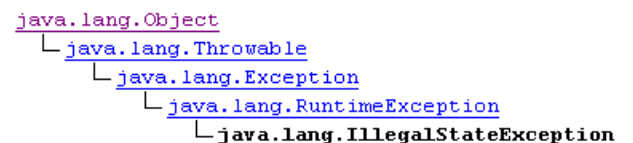
java.lang

Class `NullPointerException`



java.lang

Class `IllegalStateException`

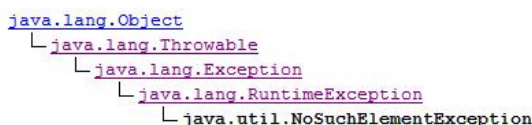


All Implemented Interfaces:

[Serializable](#)

java.util

Class `NoSuchElementException`



All Implemented Interfaces:

[Serializable](#)

Direct Known Subclasses:

[InputMismatchException](#)