

ENUNCIADO – ACTIVIDAD PRÁCTICA PREPARACIÓN EVALUACIÓN NRO. 1 POO

Una empresa de transportes marítimos necesita administrar los 2 barcos con los que cuenta, y adicionalmente administrar la carga que estos llevan. Cada uno de los barcos tiene asociado un código.

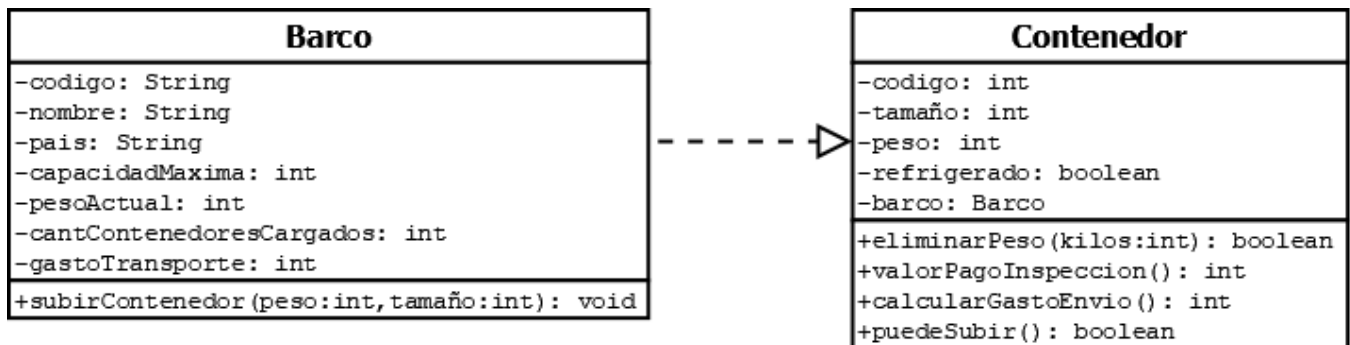
Usted es contratado para desarrollar una serie de clases para la nueva aplicación de gestión de los barcos que la empresa se encuentra desarrollando.

Hecho el análisis preliminar de los requerimientos, el analista de sistemas de la empresa le entrega un análisis de las clases. El analista define las siguientes clases:

Barco: del cual necesitamos guardar el código, nombre, el país al que pertenece, capacidad máxima, peso actual, cantidad de contenedores que tiene cargado y el costo total por concepto de transporte hasta el puerto siguiente.

Contenedor: código, marca, capacidad máxima (peso), tamaño (20 o 40 pies), peso actual de la carga, si es o no refrigerado y el buque al cual pertenece.

Ambas clases deben tener constructores, accesadores, mutadores y un método toString que devuelva una cadena con todos los valores de sus atributos.



El diagrama de clases es el siguiente:

Para cada clase además se han implementado las siguientes reglas de negocio.

Clase Barco:

- El código debe tener una longitud de 5 caracteres.
- La capacidad máxima debe ser un valor entre 10000 y 20000 kilos.

Clase Contenedor:

- El peso debe ser mayor a 0 y no puede superar la capacidad máxima de carga del barco.
- El tamaño de un contenedor puede ser 20 ó 40 pies.

Además, se deben implementar los siguientes métodos personalizados:

calcularGastoEnvio: Cada barco tiene un gasto de transporte hasta el siguiente puerto el cual debe ser dividido por la cantidad de contenedores que puede cargar. La cantidad máxima de contenedores está definida en contenedores de 20 pies, si subes contenedores de 40 pies ocupas el doble de espacio. Adicionalmente se le debe agregar un valor de 3500 pesos si es un contenedor de 20 pies o 9000 pesos si es de 40 pies, esto para pagar los gastos de aduana.

eliminarPeso: este método elimina una cierta cantidad de peso del contenedor, el valor no puede ser negativo o mayor al peso que ya tiene el contenedor.

valorPagoInspeccion: La aduana ha determinado que todos los contenedores deben ser inspeccionados, para esto se asigna personal que debe revisar el contenido del contenedor. Programe un método que permita calcular el valor a pagar por la inspección si se sabe que el precio que se cobra es de 5 pesos por kilo de carga revisado.

puedeSubir: este método devuelve un valor verdadero o falso si el peso del contenedor que se quiere subir puede subirse sin pasarse del límite máximo del barco.

subirContenedor: Cada vez que se cree un contenedor y se asocie a un barco, se debe cambiar la cantidad de contenedores cargados en el barco, teniendo la precaución de si el contenedor es de 20 o 40 pies.

Desde la clase que contiene el *main* debe:

- Crear 2 barcos.
- Crear 4 contenedores y asociarlos a un barco. EL cuarto contenedor que cree, lo debe crear con el peso máximo que puede cargar un barco.
- Mostrar la cantidad de contenedores (expresada en contenedores de 20 pies) sobre el barco y la cantidad de espacio disponible para el primer barco.
- Mostrar el valor a pagar por conceptos de inspección para el contenedor 3.
- Mostrar el valor que debe pagar cada uno de los contenedores creados por conceptos de gastos de envío.
- Mostrar si al contenedor 4 puedo subir una caja con mercadería que pesa 25000 kilos.
- Quitar 200 kilos desde el primero de los contenedores y mostrar el peso actual.

Hay que recordar que se debe programar en Java aplicando todas las convenciones aprendidas en clases y **encapsulando** la clase.