

Programación Básica 1 - Parcial 2 C03 Q2 TN 2023

Gestión de ceramicas

Nos contrataron para desarrollar un producto software que soporte algunas funcionalidades respecto a la venta de cajas de cerámica.

Las cajas de cerámica vienen de un tamaño y cantidad. Actualmente contamos con 5 medidas y se nos provee de esta información.

Representaremos una sucursal de venta que permite, además de armar pedidos, calcular la cantidad de cajas necesarias para cubrir un área dada.

Es necesario resolver las siguientes funcionalidades:

Clase de prueba GestionDeCeramicas

Será la clase que contenga al método main y deberemos desarrollar lo siguiente para garantizar el correcto funcionamiento:

- **private static void calcularCantidadDeCajasDeCeramicaPorAreaDeCobertura(Sucursal sucursal):** Se deberá ingresar el area (cantidad de metros cuadrados) a cubrir y la caja de cerámicas deseada ((ver metodo obtenerCajaDeCeramica())). Luego calcular la cantidad de cajas necesarias (ver en Sucursal) e informarla.
- **private static void agregarCajasDeCeramicaAPedido(Sucursal sucursal):** Se deberá ingresar el número de pedido, la caja de cerámicas que se agregaran (ver metodo obtenerCajaDeCeramica()) y la cantidad de cajas a agregar. Con estos tres datos, se deberán agregar las cajas al pedido de la sucursal y mostrar un mensaje en caso de éxito u otro en caso de no lograr completar la operación.
- **private static void obtenerMontoTotalDeLosPedidosUtilizados(Sucursal sucursal):** obtener el monto total de los pedidos utilizados en la sucursal y mostrarlo
- **private static void obtenerPromedioDePrecioDeCajasDeCeramicaDeUnTipoDeCeramicaEnUnPedido(Sucursal sucursal):** Se deberá solicitar el ingreso del numero de pedido el cual no puede ser menor que cero, ni mayor que la cantida de pedidos que puede tener la sucursal. En caso de no ser válido, se deberá continuar solicitando el ingreso del dato, hasta que se ingrese un dato válido. Se deberá solicitar el ingreso del tipo de cerámica para el cual se calculara el promedio (ver metodo ingresarTipoDeCeramica()) y luego se deberá obtener y mostrar el promedio de

precio de cajas de cerámica, en el pedido ingresado, para el tipo de cerámica ingresado.

- **private static CajaDeCeramica obtenerCajaDeCeramica(Sucursal sucursal):** Mostrar las cajas de cerámica disponibles ordenadas por tipo de cerámica de manera ascendente, solicitar el ingreso del código de la caja de cerámica y con este buscar la caja en la sucursal y devolverla.
- **private static MenuPrincipal obtenerOpcionDeEnumParaMenuPrincipal() y private static TipoDeCeramica ingresarTipoDeCeramica():** mostrar las opciones del menú correspondiente(ver métodos para tal fin), solicitar el ingreso de la opción deseada, obtenerla desde el enum y devolverla.

Clase Sucursal

Esta clase contiene la lógica principal de las operaciones que soportará la sucursal. Completar el constructor, getter, setters y método necesarios para asegurar el correcto funcionamiento, además de los siguientes métodos:

- **public CajaDeCeramica buscarCajaDeCeramicaPorCodigo(String codigoCajaDeCeramica):** Buscar y devolver la caja de cerámica que cumpla con el código suministrado. Es necesario no discernir entre mayúsculas y minúsculas (el usuario puede ingresar el código tanto con mayúsculas como con minúsculas).
- **public int calcularCantidadDeCajasDeCeramicaPorAreaDeCobertura(double areaACubrir, CajaDeCeramica cajaDeCeramica):** Calcular y devolver la cantidad de cajas necesarias para cubrir el área especificada. Cada caja conoce el área de cobertura según el tamaño de las cerámicas. Necesitamos entonces dividir el área a cubrir, con el área de cobertura de una caja, para conocer cuántas cajas se necesitan. En caso de que la división contenga decimales, se deberá redondear al siguiente número entero.
- **public boolean agregarCajasDeCeramicaAPedido(int numeroPedido, CajaDeCeramica cajaDeCeramica, int cantidadCajas):** Agregar al pedido la cantidad de cajas de cerámica especificada como parámetro. Cuando se agregan cajas de cerámica a un pedido, el mismo se debe marcar como utilizado. Devuelve verdadero en caso de poder agregar todas las cajas. Caso contrario, devuelve falso.
- **public double obtenerMontoTotalDeLosPedidosUtilizados():** Calcular y devolver el monto total de los pedidos utilizados.
- **public double obtenerPromedioDePrecioDeCajasDeCeramicaDeUnTipoDeCeramicaEnUnP**

pedido(int numeroPedido, TipoDeCeramica tipoDeCeramica): Dado el número de pedido y el tipo de cerámica, obtener el promedio de precio de cajas de cerámica incluidas en dicho pedido que sean del tipo de cerámica especificado. Si un pedido no fue utilizado, devolver cero.

- **public CajaDeCeramica[] obtenerCeramicasDisponiblesOrdenadasPorTipoDeCeramicaAscendente():** Ordenar el array de cajas existente por tipo de ceramica ascendente y devolverlo.

Clase Pedido

Representa la abstracción de un pedido posible de realizar al cual se le agregarán cajas de cerámicas. Resolver el constructor, getters, setters y métodos necesarios, además de:

- **public boolean agregarCajaDeCeramica(CajaDeCeramica ceramicaDeCeramica):** Agregar una caja de ceramica al array de cajas de ceramicas.

Clase CajaDeCeramica

Serán los productos con los cuales realizaremos las operaciones. Se deberá completar el constructor, getters, setters y métodos necesarios para el correcto funcionamiento del software.

Lineamientos

- No es posible modificar la firma de los métodos existentes, pero es posible agregar todos los métodos que se consideren necesarios.
- Se pueden aplicar las mejoras que considere necesarias, siempre cumpliendo el punto anterior.
- Se deberán completar los métodos y código faltante para que el sistema funcione correctamente.
- Deberá ajustar el nombre del proyecto suministrado indicando sus apellidos y nombres en donde dice “ApellidosNombres”:
PBI2023Q2C03TN-Parcial2-**ApellidosNombres**.
- La versión de java del proyecto deberá ser ajustada a la del laboratorio.

Condiciones de aprobación

Se considerará desaprobado todo proyecto que:

- no compile, o,
- no se identifiquen los atributos necesarios para resolver lo solicitado, o,
- los tipos de datos elegidos para los atributos no sean los adecuados, o,
- no cumpla con las condiciones mínimas descritas a continuación.

Se considerará aprobado aquel examen que cumpla con el 70% de puntaje.

Clase de prueba GestionDeSucursal	private static void calcularCantidadDeCajasDeCeramicaPorAreaDeCobertura(Sucursal sucursal)	5%
	private static void agregarCajasDeCeramicaAPedido(Sucursal sucursal)	10%
	private static void obtenerMontoTotalDeLosPedidosUtilizados(Sucursal sucursal)	5%
	private static void obtenerPromedioDePrecioDeCajasDeCeramicaDeUnTipoDeCeramicaEnUnPedido(Sucursal sucursal)	5%
	private static CajaDeCeramica obtenerCajaDeCeramica(Sucursal sucursal)	5%
	private static MenuPrincipal obtenerOpcionDeEnumParaMenuPrincipal() y private static TipoDeCeramica ingresarTipoDeCeramica()	5%
Clase Sucursal	public CajaDeCeramica buscarCajaDeCeramicaPorCodigo(String codigoCajaDeCeramica)	5%
	public int calcularCantidadDeCajasDeCeramicaPorAreaDeCobertura(double areaACubrir, CajaDeCeramica cajaDeCeramica)	5%
	public boolean agregarCajasDeCeramicaAPedido(int numeroPedido, CajaDeCeramica cajaDeCeramica, int cantidadCajas)	15%

	public double obtenerMontoTotalDeLosPedidosUtilizados()	10%
	public double obtenerPromedioDePrecioDeCajasDeCeramicaDeUnTipoDeCeramicaEnUnPedido(int numeroPedido, TipoDeCeramica tipoDeCeramica)	15%
	public CajaDeCeramica[] obtenerCeramicasDisponiblesOrdenadasPorTipoDeCeramicaAscendente()	5%
Clase Pedido	public boolean agregarCajaDeCeramica(CajaDeCeramica ceramicaDeCeramica)	5%
Todas las clases	Constructores, getters, setters y métodos necesarios	5%
Total		100%