

Defina las clases para implementar en java una solución orientada a objetos para el siguiente problema

La casa de comidas para llevar "El Mosquito", desea sistematizar sus menús y la gestión de sus clientes. Cada comida posee un nombre, un tiempo de preparación, un precio, calorías, y un listado de ingredientes.



Las comidas pueden agruparse para formar menús compuestos. Por ejemplo, el menú compuesto con nombre "comidas serranas" está formado por las comidas "Picada clásica", "Pan de campo", "Aceitunas rellenas". A su vez, estos menús compuestos pueden conformarse con otros menús compuestos y/o simples. En un menú compuesto, su tiempo de preparación es el mayor de los tiempos de sus elementos contenidos, el precio es la suma de sus partes menos un 10%. Las calorías son la suma de las calorías de sus elementos. Los ingredientes de un menú compuesto son la unión de los ingredientes de sus elementos (sin repetidos).

Existen también menús especiales o con restricciones, los cuales se comportan de manera similar a los menús compuestos. Algunos casos de menús especiales son: menú "Vegetariano", que no permite que se incorporen elementos que dentro de sus ingredientes contengan "Carne"; menú "Bajas calorías", que no permite que se incorporen elementos que tengan más de 400 calorías; menú "Económico" que no permite que se incorporen elementos cuyo precio sea mayor a \$550. Estos son algunos ejemplos de posibles menús, a futuro se puede agregar nuevos menús especiales (add Elemento ()) (como txt).

Además, La casa de comidas desea poder buscar entre sus comidas aquellas que cumplan con ciertos requisitos. Por ejemplo: buscar (filtro +)

- Comidas cuyo costo es menor a \$1200.
 - Comidas cuyo tiempo de preparación sea menor a 30 minutos.
 - Comidas que contengan como ingrediente "Azúcar".
 - Cualquier combinación lógica de las anteriores.
- como se copia

Nota: en el caso de los menús compuestos, no se debe devolver el menú como un todo, sino que se seleccionan las comidas del menú que cumplan con los requisitos (teniendo en cuenta todos los niveles de anidamiento).

filtrar por nombre de comida (comida)

De los clientes de la casa de comidas se conoce su nombre, un listado de comidas (de cualquier tipo) que ha solicitado, las calorías máximas permitidas y sus preferencias. El sistema debe proveer un sistema de recomendación que dada una comida, ya sea simple, compuesta o restringida el cliente determine si le gusta o no. Al cliente le gusta una comida si NO la solicitó antes y cumple con sus preferencias, a continuación se lista algunos tipos de preferencias posibles:

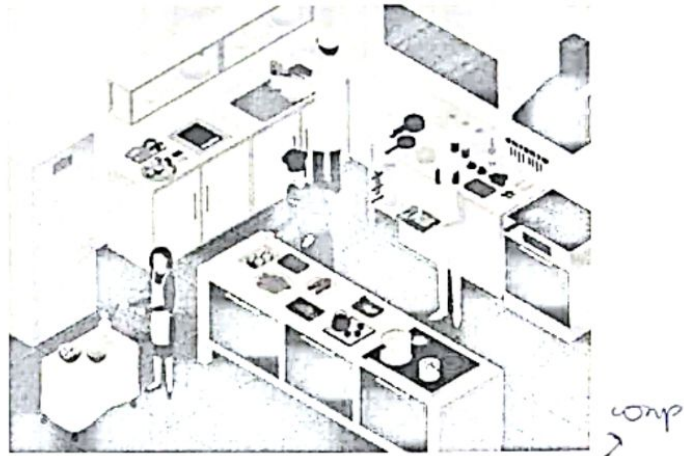
- Al cliente solo le gustan las comidas cuyo costo sea menor a \$2000
- Al cliente solo le gustan comidas que contengan "Mayonesa"
- Al cliente solo le gustan comidas que su costo sea menor a \$3400.

Los anteriores son solo algunos ejemplos de preferencias, pueden surgir nuevas o combinaciones lógicas de los mismos.

reducir
ejemplos de menú y como

Programación 2
Tecnatura Universitaria en Desarrollo de Aplicaciones Informáticas
FINAL 22-03-2022

1) Un conocido restaurante desea informatizar el despacho de comidas de su cocina. En este restaurante, cuando se ordena un pedido en una determinada mesa, se registra el conjunto de comidas, el número de mesa y el nombre del mozo que atendió la mesa. Una comida puede ir de algo tan simple como un tomate a algo tan complejo como lentejas rellenas con mousse de champagne y portobellos. Cada comida posee un nombre, un tipo (ensalada, postre, plato principal, tapas, vegetal, fruta, etc), un modo de preparación principal (frito, al horno, al microondas, solo, etc), las calorías, un precio y el tiempo necesario para su preparación (expresado en minutos).



Asimismo, para comidas complejas, puede ser que se requiera de otras recetas para su preparación. Por ejemplo, para poder realizar una "pizza margarita", es necesario contar con ingredientes específicos (muzzarella, hojas de albahaca), con la receta "masa base para pizza", y con la receta de "salsa de tomates". En este caso las calorías de la comida compleja se calculan como la suma de las calorías de cada una de sus partes, al igual que el tiempo de preparación y el precio.

La cocina del restaurante está organizada en estaciones de trabajo que serán las encargadas de la elaboración de las comidas. Dado que cada estación tiene especialidades diferentes, cuando un pedido arriba a la cocina, las comidas ordenadas en el mismo, se van asignando a las diferentes estaciones, hasta que alguna la acepte. Una estación de trabajo aceptará preparar una comida, si su especialidad lo permite según los siguientes criterios:

- Solo aceptar recetas cuyo modo de preparación principal sea "al horno".
- Solo se aceptan recetas que sean del tipo "ensalada".
- Solo se aceptan recetas que se preparen en menos de 15 minutos.
- También pueden existir combinaciones de especialidades, por ejemplo aceptar solo comidas del tipo "ensalada" y que el tiempo de preparación sea menor a 20 minutos; o aceptar recetas que se hagan al horno y no contengan menos de 400 calorías.

Frito prepara (comida) no menciona precio.

Las especialidades de cada estación de trabajo puede variar en cualquier momento (en tiempo de ejecución). Si no hay ninguna estación de trabajo que pueda elaborar la comida, la misma no se realiza (no se asigna a ninguna estación).

strategy, la estación tiene un filtro como att

calculador de precio.

costo del pedido = costo Comida

El costo total de un pedido es la suma de los costos de todas sus comidas más un adicional que aplica la cocina del restaurante, en base a algún criterio, a cada comida. Ejemplos de posibles formas de calcular el costo de un pedido son:

- Si es cocinada al horno y en menos de 15 minutos suman \$50; en otro caso se suman \$130.
- Si es una ensalada: \$10; en otro caso \$50.
- Sumar 100\$ si las calorías son mayores a 500, sino nada.
- Sumar siempre el 10% del precio de la comida en calidad de propina para los mozos

double costo a sumar filtro condicion

NOTA: La forma de calcular el costo de un pedido la establece la cocina del restaurante y es la misma para todos los pedidos. Sin embargo, cada día la cocina del restaurante puede establecer una forma diferente.

Cocina

Calculador c

→ el filtro/precio viene en condiciones?

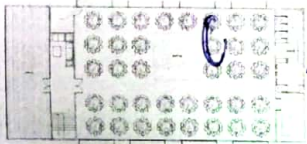
Tecnatura Universitaria en Desarrollo de Aplicaciones Informáticas
Programación 2 - Parcial 29/10/2022 - Lobería

Defina las clases para implementar una solución orientada a objetos para el siguiente problema e implemente en Java.

LoberiaAlquila una empresa dedicada al alquiler de un sin fin de elementos, desea organizar sus elementos disponibles en alquiler. Los tipos de elementos que dispone en alquiler son los siguientes:



- **Elementos de Decoración**, de los cuales se registra el valor ^{\$}, antigüedad ^{int} (en meses), código de identificación ^{int} y una descripción ^{String}.
- **Muebles Para Fiesta**, de los cuales se registra el tiempo ^{int} que hace que los tiene la empresa (en meses) el valor del mueble ^{\$} un código de identificación ^{int} y una descripción ^{String}.
- **Salones de Eventos**, donde se registra un código de identificación ^{int}, una descripción ^{String}, cuantos meses ^{int} hace que se incorporó a la empresa y el valor del mismo ^{\$}.



- **Paquetes Festivos**, un paquete posee un código de identificación ^{int} y los elementos que lo conforman (Elementos de Decoración, muebles, Salones, paquetes y cualquier otro tipo de elementos que la empresa Lobería Alquila puede alquilar en un futuro). El valor total de un paquete queda determinado por la suma de todos los elementos que posee. La antigüedad de un paquete queda determinada por la mayor antigüedad de sus elementos. ^{f. vencimiento}

- **Elemento con Vencimiento**, está pensado para cuando los elementos se van a vencer en cierta cantidad de meses de antigüedad (por ejemplo 12) y puede hacer referencia a cualquier tipo de elemento existente. Tienen un código de identificación propio y el valor del elemento con vencimiento es el valor del elemento contenido siempre que la antigüedad sea menor que los meses de vencimiento, en caso de ser mayor se descuenta, al día de hoy, un 2% por mes de diferencia. La empresa está evaluando el 2% y piensa cambiarlo a un 3% o 5% en un futuro. ^{heuda del simple y sobrescribe un por}

Se desea listar todos los elementos de la empresa que cumplan con determinado criterio. A continuación se listan algunos ejemplos. ^{busqueda}

- Listar todos los elementos que en la descripción contenga la palabra "Globo",
- Listar todos elementos que tengan más de 3 meses de antigüedad y el valor sea menor a \$10.000,
- Listar todos los elementos cuyo código de identificación sea igual a 35,
- Se pueden agregar nuevos tipos de búsquedas así como también combinaciones lógicas de las mismas.

Todos los resultados se devuelven siempre ordenados por código de identificación, y en caso de igualdad por antigüedad. ^{Elem es Comparable} ^{solo incluye Elementos.}

NOTA: en el caso de los paquetes el resultado de la búsqueda incluye cualquier elemento que posea y cumple con la búsqueda (No se incluyen paquetes en los resultados de una búsqueda). En el caso del elemento con vencimiento, se retorna el elemento al que hace referencia siempre y cuando cumpla con el criterio de búsqueda y la antigüedad del mismo no haya superado la pre establecida como vencimiento.

AlquiloLoberia posee una política de promocionar elementos. A veces se promocionan los elementos que NO tengan la palabra "deluxe" en su descripción, otras que su valor sea menor a \$3.500, o condiciones similares e incluso combinaciones de condiciones. Proveer un servicio en la empresa que dado un artículo responda a la consulta de si está o no promocionado, así como también mecanismos que permitan cambiar la política de promoción de la empresa. También proveer un método que retorne todos sus elementos promocionados.

~~promocionar (condiciones)~~ ^{LoberiaAlquila} ^{esta Promocionada (Elem.c)}
^{filas promocionados}

Programación 2 - TUDAI - Final 4 de Marzo de 2022

1) Librería

Una librería requiere un sistema que le aconseje qué libros puede ofrecerle a un determinado cliente. Un libro se compone de un nombre, autor (solo uno), un resumen y un listado de géneros literarios que abarca. Para que un libro pueda ser ofrecido a un cliente, alguno de sus géneros literarios (al menos uno) debe estar incluido en los géneros por los que el cliente tiene preferencia. Es decir que el cliente acepta la recomendación siempre y cuando el libro contenga alguno de sus géneros preferidos. Por ejemplo, si a David le gustan los géneros "ciencia ficción", "aventuras", y "novelas románticas", y se le pregunta si acepta el libro "Mujercitas" cuyos géneros son "Novela Romántica", "Clásico", y "Drama", él lo va a aceptar porque incluye uno de sus géneros preferidos ("novela Romántica").

Se debe proveer servicios que permitan: (Libros Obtenidos) comparemos pedidos:

1. Conocer la antigüedad de un cliente
2. Obtener el monto total gastado por un cliente en la librería
3. Conocer la cantidad de libros comprados por un cliente en la librería
4. Obtener un listado de clientes que cumplan con determinadas condiciones, por ejemplo, aquellos que:
 - han comprado más de 10 libros
 - tienen más de 8 años de antigüedad
 - Gastaron más \$20000
 - han comprado más de 5 libros y gastaron más \$25000
 - Combinaciones de las anteriores.

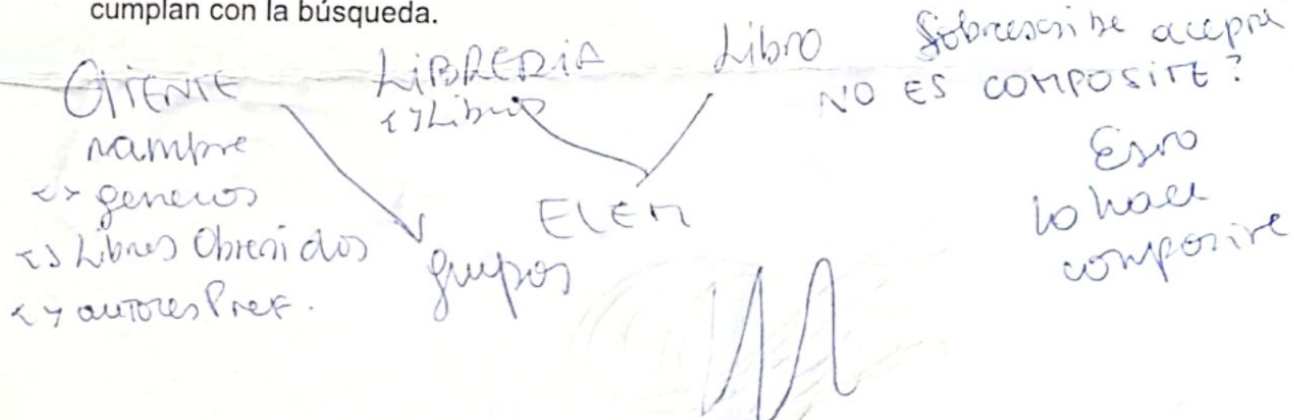
Incorporar al diseño la posibilidad de modelar clientes exigentes, que solo aceptan un libro si el autor está en su lista de preferidos y el libro contiene TODOS sus géneros preferidos.

Defina las clases para implementar una solución orientada a objetos para el problema e implemente en Java todos los métodos necesarios para resolver los servicios solicitados.

Solo final libre:

2) Extienda la solución anterior para incluir el siguiente requerimiento

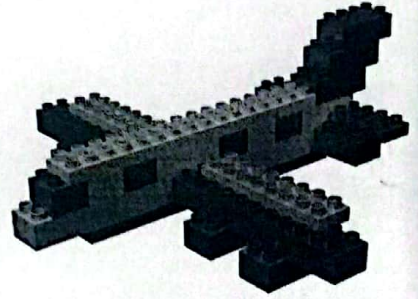
Incorporar Grupos de clientes, los cuales se componen de varios clientes (incluso sub grupos) y aceptan un libro si al menos alguno de sus integrantes acepta el libro. Si un grupo compra un libro la compra se la agrega al grupo. La cantidad de libros comprados y lo gastado se define como la suma de las cantidades y lo gastado de sus integrantes. En el caso de la antigüedad se establece como la máxima antigüedad entre sus miembros. Para las búsquedas, si un grupo cumple EL con lo solicitado se agrega a la búsqueda, en caso contrario se agregan los elementos por el contenido (en cualquier nivel de profundidad) que cumplan con la búsqueda.



Programación 2 - Olavarría
Tecnatura Universitaria en Desarrollo de Aplicaciones Informáticas
Parcial 04/06/2022

Defina las clases para implementar una solución orientada a objetos para el siguiente problema e implemente en Java.

Una juguetería se especializa en la venta de juegos de ladrillos (LEGO, RASTI, Mis ladrillos, etc) y requiere de un sistema que le permita administrar piezas, combos y paquetes de ladrillos. De cada pieza se guarda la marca, peso, costo, color y cantidad de encastrés. Por ejemplo:



Ladrillo Lego 9 gramos, \$10, rojo, 6 encastrés → to Spring 11?

Ladrillo Rasti, 3 gramos, \$2, azul, 2 encastrés

Un combo agrupa piezas o conjuntos de piezas (pueden ser piezas, Combos o Paquetes). Cada Combo/Paquete guarda su marca; el peso del mismo es la suma de los pesos de sus elementos; y el color se determina como el color del último de sus elementos. La cantidad de encastrés es el promedio de todos los encastrés de sus elementos. En el combo, el costo es la suma de los costos menos un 10% de descuento, mientras que en los paquetes el costo es la suma de los costos menos un 15% de descuento. *clase vs. instancia*

La juguetería mantiene un listado de todos los productos que comercializa (tanto piezas como Combo o Paquetes). Para facilitar la búsqueda de los productos, necesita brindar listados en los cuales se incluyan todas las piezas, combos o paquetes que cumplen con una determinada condición. A continuación, se listan algunos ejemplos de listados:

- Listar todos los elementos de color rojo
- Listar todos los elementos de color rojo y de peso menor a 4 gramos.
- Listar todos los elementos de color rojo o azul y con menos de 100 encastrés.
- Se pueden agregar nuevos tipos de búsquedas así como también combinaciones lógicas de las mismas.

NOTA: Los listados se retornan ordenados por marca, en el caso de ser de la misma marca por costo, luego por cantidad de encastrés y por último por color.

Combotemático (Filmo +) sobreescribe agrega

La juguetería desea incorporar combos temáticos por ejemplo:

- Combo Star Wars: solo permite que se le incorporen elementos de color negro, blanco o gris.
- Paquete Liviano: solo permite que el peso de los elementos que se le incorporen no supere los 3 gramos.
- Combo Lego: solo permite que los elementos que se le incorporen sean de la marca Lego.

Los anteriores son algunos ejemplos de combos especiales que desea agregar la juguetería pero a futuro puede ser que se combinen estos combos o incluso se cambien la forma de aceptación, por ejemplo un Combo Star Wars de Lego que solo acepta de color negro, blanco o gris y que sean de la marca LEGO. NOTA: en caso de cambiar el criterio por el que se forma un Combo, los elementos existentes no se revisan, y solo aplica para nuevos elementos que se agreguen al mismo.



Tener en cuenta todos los mecanismos de la Programación Orientada a Objetos que se fueron viendo a lo largo de la materia

COMBO
marca

PIEZA
marca
peso
costo
color
cantidad encastrés

ELEM
marca