**Caso de Estudio**

Se considera un programa que permite establecer el costo de un producto en base a los materiales de fabricación, y el costo de cada proceso de producción. Cada proceso se puede descomponer en diferentes subprocesos, cada uno con su costo particular.



El diseño del programa comenzaría como:



Se solicita:

* Codifique en Java las diferentes relaciones entre clases.
* Establezca algún mecanismo para la creación de cada subproceso de tal forma que queden establecidas las relaciones como en la figura.
* Codifique el método *getCosto()* en *Producto,* el cual debe retornar el costo del producto en base a las materias primas utilizadas y a la sumatoria de costos de cada proceso y subproceso. Agregue los métodos que considere necesarios en cada clase.

RESOLUCIÓN

**Importante!!!**

**Antes de ver la resolución intenten codificar SOLOS lo que se solicita.**

**Tener en cuenta también que puede haber más de una solución válida.**

|  |
| --- |
| ***interface MateriaPrima{***  ***public float materia1=10.80f;***  ***public float materia2=11.20f;***  ***}***  *//Esta clase tiene una relación consigo misma*  ***class Proceso {***  ***private float valor;***  ***private String nombre;***  ***private Proceso subprocesos[] = new Proceso[4];*** *//cada proceso puede tener hasta 4 subprocesos.*      ***public Proceso(float v, String n) {*** *//constructor recibe un valor y un nombre*  ***valor = v;***  ***nombre = n;***  ***}***    ***public Proceso[] getSubprocesos(){*** *//como subprocesos es privada...*  ***return subprocesos;***  ***}***    *//METODO IMPORTANTE: cada proceso retorna su valor + el valor de todos sus hijos*  ***public float getValor(){***  ***float acu = valor;*** *//acumulo el valor de este proceso*    ***for(int i=0; i<subprocesos.length; i++)*** *//recorro los hijos obteniendo el valor*  ***if(subprocesos[i] != null)*** *//solo si esta celda del arreglo tiene un objeto Proceso*  ***acu += subprocesos[i].getValor();*** *//cada hijo, retornara tambien la sumatoria de los nietos*    ***return acu ;***  ***}***  ***}***  ***//Clase Producto. Tiene la referencia solamente al proceso principal (ver figura y UML)***  ***class Producto implements MateriaPrima{***  ***private Proceso procesos;*** *//agregacion...*  ***public float flete = 100.0f;***    ***public Proceso getProcesos(){***  ***return procesos;***  ***}***  ***public void setProcesos(Proceso p){***  ***procesos = p;***  ***}***    *//Este metodo devuelve el costo total del producto: materiales + procesos*  ***public float getCostoTotal(){***  ***float acumulador = 0;***    ***acumulador += materia1;***  ***acumulador += materia2;***  ***acumulador += flete;***    *//Empiezo a recorrer procesos y subprocesos*  ***acumulador += procesos.getValor();***    ***return acumulador;***  ***}***    ***}***  *//PROGRAMA/CLASE PRINCIPAL*  ***public class Main {***  ***public static void main(String[] args) {***  *//mètodo principal*  ***Producto a1 = new Producto();***    *//TRATAMOS DE REPLICAR EL ESQUEMA DE LA FIGURA: procesos y subprocesos*  *//con sus costos y nombres*    *//PRIMERO: creo todos los procesos en forma separada*  ***Proceso \_1311 = new Proceso(2 , "1.3.1.1");***  ***Proceso \_131 = new Proceso(5 , "1.3.1");***  ***Proceso \_132 = new Proceso(5 , "1.3.2");***  ***Proceso \_13 = new Proceso(5 , "1.3");***    ***Proceso \_12 = new Proceso(1 , "1.2");***    ***Proceso \_111 = new Proceso(5 , "1.1.1");***  ***Proceso \_112 = new Proceso(5 , "1.1.2");***  ***Proceso \_11 = new Proceso(5 , "1.1");***    ***Proceso \_1 = new Proceso(15 , "1");***      *//SEGUNDO: Asigno subproceos para formar el arbol (ver figura).*  ***\_1.getSubprocesos()[0] = \_11;*** *//el proceso 1 tiene TRES procesos hijos*  ***\_1.getSubprocesos()[1] = \_12;***  ***\_1.getSubprocesos()[2] = \_13;***    ***\_11.getSubprocesos()[0] = \_111;*** *//el proceso 1.1 tiene DOS procesos hijos*  ***\_11.getSubprocesos()[1] = \_112;***    ***\_13.getSubprocesos()[0] = \_131;*** *//el proceso 1.3 tiene DOS procesos hijos*  ***\_13.getSubprocesos()[1] = \_132;***    ***\_131.getSubprocesos()[0] = \_1311;*** *//el proceso 1.3.1 tiene UN proceso hijo*      *//TERCERO: Asigno toda la estructura de procesos al producto*  ***a1.setProcesos( \_1 );***      *//CUARTO: pregunto cuanto cuesta el producto*  ***float costoTotal = a1.getCostoTotal();***  ***System.out.println("Costo Total del Producto: $" + costoTotal);***  ***}***  ***}*** |