

## Compras\OrdenCompra

###

```
1. package compras;
2.
3. import gestionHumana.*;
4. import java.util.ArrayList;
5.
6. public class OrdenCompra {
7.
8.     public int codigo;
9.     "default" String tipo;
10.    public Empleado comprador;
11.    "default" ArrayList<Producto> productos;
12.
13.    public OrdenCompra(int codigo, String tipo, Empleado comprador,
14.        ArrayList<Producto> productos) {
15.        this.codigo = codigo;
16.        this.tipo = tipo;
17.        this.comprador = comprador;
18.        this.productos = productos;
19.        Producto.totalProductosPedidos += productos.size();
20.    }
21.
22.    public void agregarProducto(Producto producto) {
23.        if (producto.tipo.equals(tipo)) {
24.            productos.add(producto);
25.            Producto.totalProductosPedidos++;
26.        }
27.    }
28.
29.    public void retirarProducto(Empleado empleado, Producto producto) {
30.        if (!empleado.tengoPermiso()) {
31.            return;
32.        }
33.        retirarProducto(producto);
34.    }
35.
36.    private void retirarProducto(Producto producto) {
37.        for (int i = 0; i < productos.size(); i++) {
38.            if (producto.getCodigo() == productos.get(i).getCodigo()) {
39.                productos.remove(i);
40.                producto.totalProductosPedidos--;
41.                producto.imprimirNombre();
42.                System.out.println(" retirado");
43.                break;
44.            }
```

```

45.     }
46. }
47.
48. public void descontar() {
49.     Producto.totalProductosPedidos -= productos.size();
50. }
51.
52. }
###

```

## Compras\Producto

```

###
1. package compras;
2.
3. public class Producto {
4.
5.     private final int codigo;
6.     "default" String nombre;
7.     "default" String tipo;
8.     "default" static int totalProductosPedidos;
9.
10.    public Producto(int codigo, String nombre, String tipo) {
11.        this.codigo = codigo;
12.        this.nombre = nombre;
13.        this.tipo = tipo;
14.    }
15.
16.    public void imprimirNombre() {
17.        System.out.print(nombre);
18.    }
19.
20.    public void setCodigo(int codigo) {
21.        "No se puso nada"
22.    }
23.
24.    public int getCodigo() {
25.        return codigo;
26.    }
27.
28.    public static int getTotalProductosPedidos() {
29.        return totalProductosPedidos;
30.    }
31. }
###

```

## gestionHumana\Empleado

###

```
1. package gestionHumana;  
2.  
3. public class Empleado {  
4.  
5.     public final long cedula;  
6.     public String nombre;  
7.     "default" String cargo;  
8.  
9.     public Empleado(long cedula, String nombre, String cargo) {  
10.         this.cedula = cedula;  
11.         this.nombre = nombre;  
12.         this.cargo = cargo;  
13.     }  
14.  
15.     public boolean tengoPermiso() {  
16.         return cargo.contains("Administrador");  
17.     }  
18. }
```

###

## objtaller3\objTaller3

###

```
1. package objtaller3;  
2.  
3. import compras.*;  
4. import gestionHumana.Empleado;  
5. import java.util.ArrayList;  
6.  
7. public class ObjTaller3 {  
8.  
9.     public static void main(String[] args) {  
10.         Producto p1 = new Producto(1, "Escoba", "Aseo");  
11.         Producto p2 = new Producto(2, "Camisa", "Ropa");  
12.         Producto p3 = new Producto(3, "Trapera", "Aseo");  
13.         Producto p4 = new Producto(4, "Pantalon", "Ropa");  
14.         Producto p5 = new Producto(5, "Jabon", "Aseo");  
15.         Empleado emp1 = new Empleado(405, "Juan", "Ingeniero");  
16.         ArrayList<Producto> productos1 = new ArrayList<>();  
17.         productos1.add(p1);  
18.         productos1.add(p3);  
19.         OrdenCompra orden1 = new OrdenCompra(101, "Aseo", emp1, productos1);  
20.         System.out.println(Producto.getTotalProductosPedidos());  
21.         orden1.agregarProducto(p4);
```

```

22.     System.out.println(Producto.getTotalProductosPedidos());
23.     orden1.agregarProducto(p5);
24.     System.out.println(Producto.getTotalProductosPedidos());
25.     System.out.println("Orden " + orden1.codigo + " creada");
26.
27.     Empleado emp2 = new Empleado(128,"Susana", "Administradora de sucursal");
28.     ArrayList<Producto> productos2 = new ArrayList<>();
29.     productos2.add(p2);
30.     productos2.add(p4);
31.     OrdenCompra orden2 = new OrdenCompra(202, "Ropa", emp2, productos2);
32.     System.out.println(Producto.getTotalProductosPedidos());
33.     System.out.println(emp2.cedula + " va a retirar producto");
34.     orden2.retirarProducto(emp2, p4);
35.     System.out.println(Producto.getTotalProductosPedidos());
36.     orden2.retirarProducto(emp1, p2);
37.     System.out.println(Producto.getTotalProductosPedidos());
38. }
39. }

```

###

## EXPLICACIÓN.

### Clase **OrdenCompra** (Paquete **compras**)

1. **Línea 3:** acceder a clases del paquete gestionHumana.
2. **Línea 6:** OrdenCompra necesita ser accesible desde otros paquetes, como objtaller3.
3. **Línea 8, 9, 10, 11:** los atributos son de tipo public, default, public, default: unos pueden accederse desde cualquier clase y otros solo desde las clases dentro del paquete.
4. **Línea 22:** Como agregarProducto debe ser llamado desde otros paquetes (como objtaller3), necesita ser public.
5. **Línea 29, 31:** método público. Si no se cumple el if, el return para la ejecución y sigue con otro bloque de código.
6. **Línea 37:** asegura que i no supere el tamaño de la lista productos y no hayan problemas en el rango pues cada indice será válido.
7. **Línea 48:** no devuelve nada, solo actualiza el contador.

### Clase **Producto** (Paquete **compras**)

1. **Línea 5,6,7,8:** los atributos son de tipo private, default, default, default: el primero solo se puede acceder desde dentro de la clase Producto, y el resto son accesibles solo dentro del paquete.
2. **Línea 10:** es public para que otras clases puedan crear productos y trabajar con ellos.

3. **Línea 16:** es público porque otras clases (como OrdenCompra) necesitan invocarlo para imprimir el nombre de un producto. Si este método fuera privado, no se podría llamar desde fuera de la clase Producto.
  4. **Línea 21:** El atributo codigo está declarado como final, lo que significa que su valor no puede ser modificado después de su inicialización en el constructor. Aunque el método setCodigo() está declarado, no puede cambiar el valor del atributo.
  5. **Línea 24:** El método devuelve un valor de tipo int, que es el tipo del atributo codigo.
  6. **Línea 28:** Es público para que cualquier clase que use productos (como OrdenCompra o ObjTaller3) pueda acceder al conteo total de productos pedidos.
- 

## Clase **Empleado** (Paquete **gestionHumana**)

1. **Línea 1:** ese es el paquete al que pertenece la clase, debe definirse.
  2. **Línea 5,6,7:** los atributos son public, public, default: los dos primeros son accesibles desde otras clases, el ultimo no es necesario ser expuesto fuera del paquete.
  3. **Línea 15:** Otras clases necesitan verificar si un empleado tiene permisos para realizar acciones específicas (como retirar productos en OrdenCompra).
- 

## Clase **ObjTaller3** (Paquete **objtaller3**)

1. **Línea 3:** se importan todas las clases dentro del paquete compras ( OrdenCompra y Producto).
2. **Línea 5:** ArrayList es una clase del paquete java.util y no está disponible automáticamente a menos que se importe explícitamente. Es necesario por la definición de productos1 y productos2.
3. **Línea 9:** así se definen los argumentos en el método main. es el punto de entrada estándar para cualquier programa Java.
4. **Línea 19:** 101 es el código de la orden de compra, es la salida esperada.
5. **Línea 23:** Llama a este método para actualizar la lista de productos de la orden (productos1) y el contador global (totalProductosPedidos) en la clase Producto.
6. **Línea 36:** emp1 representa al empleado que intenta retirar el producto. Aunque emp1 no tiene permisos, la salida requerida no se ve afectada.