

## Control 2

**Competencia a evaluar:** El estudiante debe ser capaz de construir: modelo del dominio, diagrama de clases del dominio de la aplicación y código java del problema a resolver

### Problema

Se tiene que manejar información los productos que están presentes en las canastas que está entregando el gobierno por la pandemia del coronavirus. De cada producto interesa su código y nombre

La aplicación debe leer desde un archivo llamado “Productos.txt” la información de los productos, donde cada registro tiene el código y nombre del producto. Suponga que no vienen productos repetidos

Su aplicación también debe:

- Desplegar todos los productos por pantalla
- Buscar un producto cuyo código se lee desde pantalla y desplegar sus datos

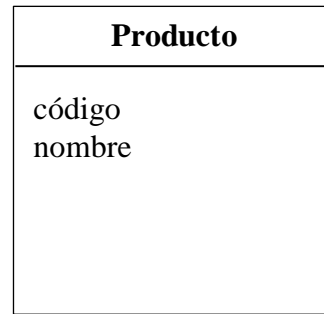
### Se pide:

- a. Modelo del dominio
- b. Diagrama de clases del dominio de la aplicación
- c. Código Java (todo, excepto la clase Producto)

**Nota:** Escriba el código para toString() de ser necesario

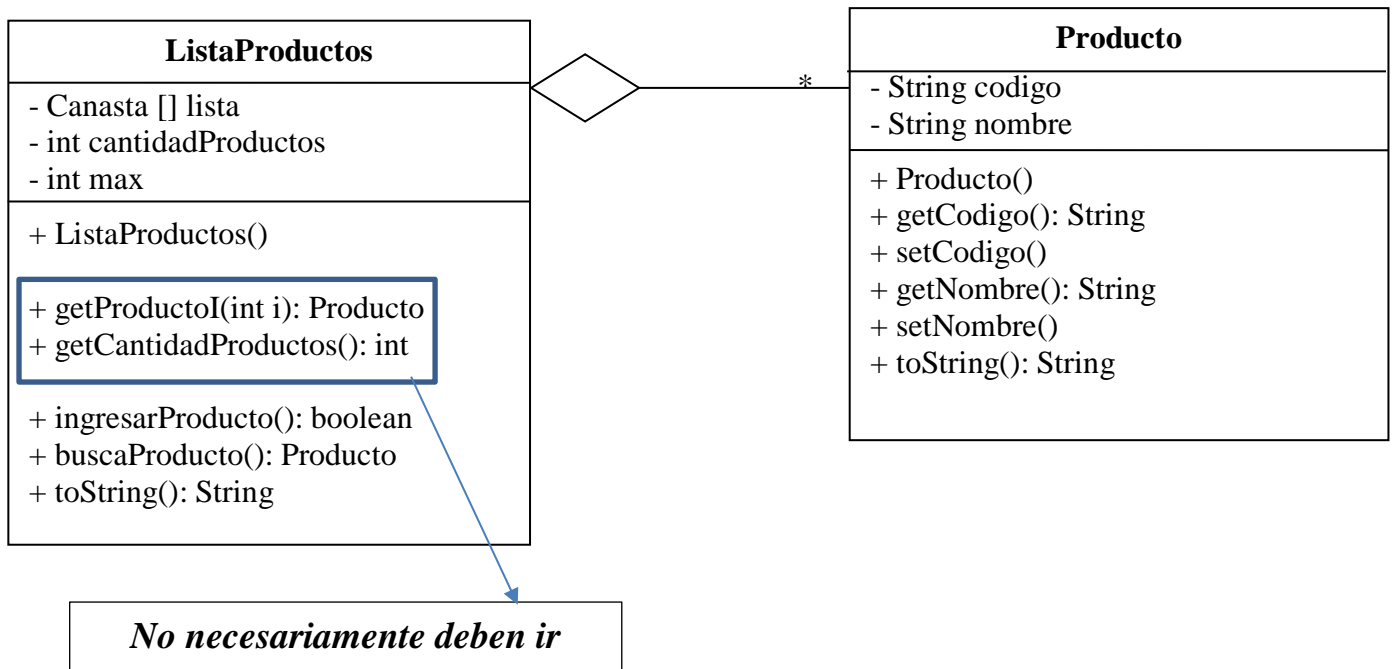
## 1. Modelo del dominio

*0.5 puntos*



## 2. Diagrama de clases del dominio de la aplicación

*1 punto*



```
public class Producto {  
    private String nombre;  
    private String código;  
  
    public Producto(String codigo, String nombre) {  
        this.codigo = codigo;  
        this.nombre = nombre;  
    }  
  
    public String getNombre() {  
        return nombre;  
    }  
  
    public void setNombre(String nombre) {  
        this.nombre = nombre;  
    }  
  
    public String getCodigo() {  
        return codigo;  
    }  
  
    public void setCodigo(String codigo) {  
        this.codigo = codigo;  
    }  
  
    @Override  
    public String toString() {  
        return "nombre producto: " + this.nombre +  
               ", código producto: " + this.codigo;  
    }  
}
```

*No se pide*

```

public class ListaProductos {
    private int max;
    private int cantidadProductos;
    private Producto [] lista;

    public ListaProductos (int max) {
        this.max = max;
        cantidadProductos = 0;
        lista = new Producto[max];
    }

    public boolean ingresarProducto(Producto producto) {
        if (cantidadProductos < max ) {
            lista[cantidadProductos]= producto;
            cantidadProductos++;
            return true;
        }
        else {
            return false;
        }
    }

    public Producto buscarProducto(String codigo) {
        int i= 0;
        while(i<cantidadProductos && !lista[i].getCodigo().equals(codigo)) {
            i++;
        }
        if(i == cantidadProductos) {
            return null;
        }
        else {
            return lista[i];
        }
    }

    public int getCantidadProductos() {
        return cantidadProductos;
    }

    public String toString() {
        String salida = "Lista Productos \n";
        for(int i = 0; i< cantidadProductos; i++ ) {
            salida = salida + lista[i].toString()+"\n";
        }
        return salida;
    }
}

```

*1,5 puntos*

*No necesita implementarse*

```

import ucn.StdIn;
import ucn.Stdout;

public class App {

    public static void leerProductos (ListaProductos listaProductos){
        ArchivoEntrada archivo = new ArchivoEntrada("Productos.txt");
        boolean ingreso = true;
        While(ingreso &&!archivo.isEndFile()) {
            Registro registro = archivo.getRegistro()
            String codigo = StdIn.readString();
            String nombre = StdIn.readString();

            Producto producto = new Producto(código, nombre);

            ingreso = listaProductos.ingresarProducto(producto);
        }
    }

    public static desplegarProductoBuscado (ListaProductos listaProductos){
        StdOut.print("código producto:");
        String codigo = StdIn.readString();
        Producto producto = listaproduktos.buscarProducto(codigo);
        if(producto!= null) {
            StdOut.println(producto.toString());
        }
        else {
            StdOut.println("producto no existe");
        }
    }

    public static void main(String[] args) {
        ListaProductos listaProductos = new ListaProductos(100);

        //Lectura de los productos
        leerProductos(listaProductos);

        //Despliegue de los productos
        listaProductos.toString();

        //Busqueda de un producto
        desplegarProductoBuscado(listaProductos);
    }

} //Fin App

```

*1 punto*

*1 punto*

*1 punto*