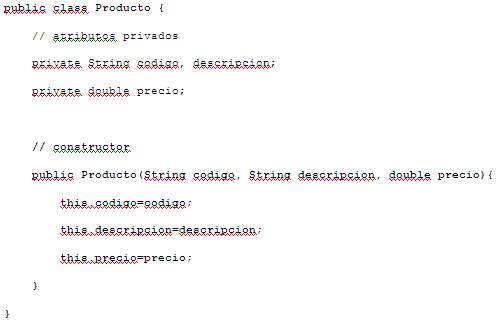
**GUIA DE LABORATORIO 3**

**Clases administradoras con arreglos de objetos**

**Ejercicio 1**

Cree un proyecto nuevo de nombre **P03E01**. Cree un paquete nuevo de nombre **p03e01**. Copie y pegue en el paquete creado, desde el proyecto P02E01 la clase **Producto** que tiene los siguientes atributos y constructor:



Diseñe una nueva clase de nombre **ArregloProductos**, que tenga como atributos privados un arreglo de objetos y un contador de objetos que se guardan en el arreglo. Considere un tamaño fijo de 50 para el arreglo. Considere dentro de la funcionalidad de la clase los métodos necesarios de administración del arreglo de objetos para: agregar un nuevo objeto, recuperar, actualizar, buscar, eliminar.

Clic derecho en el paquete **p03e01**, seleccionamos **New Class**, le damos el nombre **ArregloProductos** y escribimos sus atributos y constructor:

public class ArregloProductos {

**// atributos**

private Producto a[]; // arreglo de objetos

private int n; // contador de objetos

private final int tamaño=50;

**// constructor**

public ArregloProductos(){

a = new Producto[tamaño];

n=0;

}

Con la ayuda de NetBeans, generamos los métodos get-set para sus atributos, excepto el tamaño.

Antes de la última llave de cierre escribimos los métodos adicionales de administración del arreglo de objetos:

**// métodos de administración**

public void agrega(Producto p){ // agrega un nuevo objeto

if(n<tamaño){

a[n] = p;

n++;

}

}

public Producto getProducto(int i){ // recupera un objeto

return a[i];

}

public void actualiza(Producto p, int i){ // actualize un objeto

a[i] = p;

}

public int busca(String codigo){

// busca y retorna la posición de un objeto, a través de su codigo

for(int i=0; i<n; i++){

if(getProducto(i).getCodigo().equals(codigo))

return i;

}

return -1; // no lo encuentra

}

public void elimina(int p){ // elimina un objeto de la posición indicada

for(int i=p; i<n-1; i++){

a[i] = a[i+1];

}

n--;

}

public double mayorPrecio(){

double m=a[0].getPrecio();

for(int i=1; i<n; i++){

if(getProducto(i).getPrecio()>m)

m =getProducto(i).getPrecio();

}

return m;

}

public double menorPrecio(){

double m=a[0].getPrecio();

for(int i=1; i<n; i++){

if(getProducto(i).getPrecio()<m)

m =getProducto(i).getPrecio();

}

return m;

}

public double precioPromedio(){

double s=0;

for(int i=1; i<n; i++){

s +=getProducto(i).getPrecio();

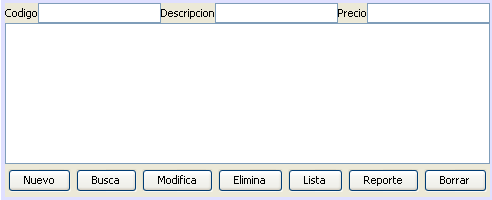
}

return s/n;

}

}

Diseñe una nueva clase gestora tipo **Panel** de nombre **PanelPrincipal** con la interface gráfica de usuario necesaria para gestionar, sobre el arreglo de objetos, los procesos considerados en el siguiente diseño:



Vaya a la ficha **Source** para escribir lo que está en negrita:

public class PanelPrincipal extends javax.swing.JPanel {

**// atributo**

**ArregloProductos ap;**

public PanelPrincipal() {

initComponents();

**ap = new ArregloProductos();**

}

Regrese al diseño y con doble clic en cada botón programe su acción considerando lo siguiente:

En el botón **Nuevo**, crea un objeto de la clase **Producto**, lo guarda en el arreglo de objetos y muestra un listado de todos los objetos. Evita la duplicidad de códigos.

En el botón **Busca**, ubica el objeto que tiene el código ingresado y muestra su información. En caso que no lo encuentre muestra un mensaje de error en una ventana.

En el botón **Modifica**, actualiza la información de objeto que tiene el código ingresado y muestra un listado de todos los objetos.

En el botón **Elimina**, elimina el objeto que tiene el código ingresado y muestra un listado de todos los objetos.

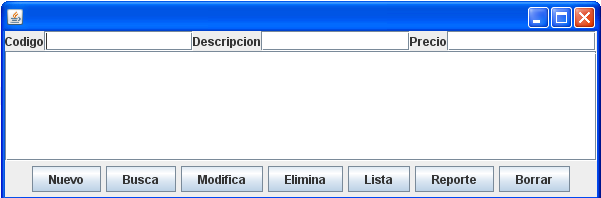
En el botón **Lista**, muestra un listado de todos los objetos.

En el botón **Reporte**, muestra la siguiente información: precio más alto, precio promedio, precio más bajo.

En el botón **Borrar** borra la información del área de texto, de las cajas de texto y enfoca el ingreso en la caja de texto del código.

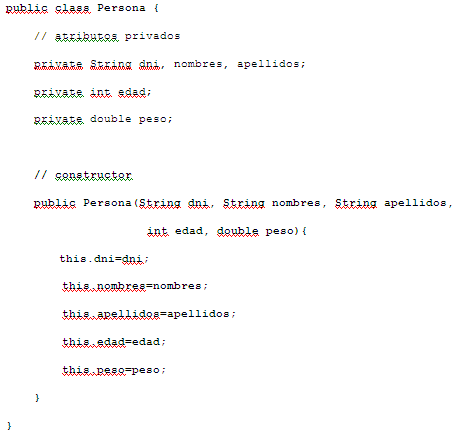
Copie la clase de aplicación de nombre **Principal** y péguelo en éste paquete.

Ejecute su aplicación.



**Ejercicio 2**

Cree un proyecto nuevo de nombre **P03E02**. Cree un paquete nuevo de nombre **p03e02**. Diseñe una nueva clase **Persona** con los siguientes atributos y constructor:



Con ayuda de NetBeans, genere los métodos get-set para todos sus atributos.

Diseñe una nueva clase administradora de nombre **ArregloPersonas**, que tenga como atributos privados un arreglo de objetos y un contador de objetos que se guardan en el arreglo. Considere un tamaño fijo de 50 para el arreglo. Considere dentro de la funcionalidad de la clase los métodos necesarios de administración del arreglo de objetos para: agregar un nuevo objeto, recuperar, actualizar, buscar, eliminar.

Diseñe una nueva clase gestora tipo **Panel** de nombre **PanelPrincipal** con la interface gráfica de usuario necesaria para gestionar, sobre el arreglo de objetos, los procesos considerados en el siguiente diseño:



Vaya a la ficha **Source** para declarar y crear el objeto de tipo **ArregloPersonas**

Regrese al diseño y con doble clic en cada botón programe su acción considerando lo siguiente:

En el botón **Nuevo**, crea un objeto de la clase **Persona**, lo guarda en el arreglo de objetos y muestra un listado de todos los objetos. Evita la duplicidad de DNI.

En el botón **Busca**, ubica el objeto que tiene el DNI ingresado y muestra su información. En caso que no lo encuentre muestra un mensaje de error en una ventana.

En el botón **Modifica**, actualiza la información de objeto que tiene el DNI ingresado y muestra un listado de todos los objetos.

En el botón **Elimina**, elimina el objeto que tiene el DNI ingresado y muestra un listado de todos los objetos.

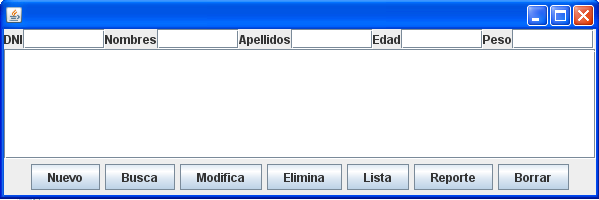
En el botón **Lista**, muestra un listado de todos los objetos.

En el botón **Reporte**, muestra la siguiente información: mayor edad, edad promedio, menor edad, mayor peso, peso promedio, menor peso.

En el botón **Borrar** borra la información del área de texto, de las cajas de texto y enfoca el ingreso en la caja de texto del dni.

Copie la clase **Principal** del ejercicio anterior y péguelo en éste paquete.

Ejecute su aplicación.

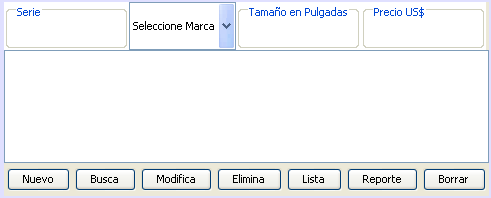


**Ejercicio 3**

Cree un proyecto nuevo de nombre **P03E03**. Cree un paquete nuevo de nombre **p03e03**. Copie y pegue en el paquete creado, desde el proyecto P02E03 la clase **TV.**

Diseñe una nueva clase administradora de nombre **ArregloTV**, que tenga como atributos privados un arreglo de objetos y un contador de objetos que se guardan en el arreglo. Considere un tamaño fijo de 50 para el arreglo. Considere dentro de la funcionalidad de la clase los métodos necesarios de administración del arreglo de objetos para: agregar un nuevo objeto, recuperar, actualizar, buscar, eliminar.

Diseñe una nueva clase gestora tipo **Panel** de nombre **PanelPrincipal** con la interface gráfica de usuario necesaria para gestionar, sobre el arreglo de objetos, los procesos considerados en el siguiente diseño:



Vaya a la ficha **Source** para declarar y crear el objeto de tipo Arreglo TV

Regrese al diseño y con doble clic en cada botón programe su acción considerando lo siguiente:

En el botón **Nuevo**, crea un objeto de la clase **TV**, lo guarda en el arreglo de objetos y muestra un listado de todos los objetos. Evita la duplicidad de serie.

En el botón **Busca**, ubica el objeto que tiene la serie ingresada y muestra su información. En caso que no lo encuentre muestra un mensaje de error en una ventana.

En el botón **Modifica**, actualiza la información de objeto que tiene la serie ingresada y muestra un listado de todos los objetos.

En el botón **Elimina**, elimina el objeto que tiene la serie ingresada y muestra un listado de todos los objetos.

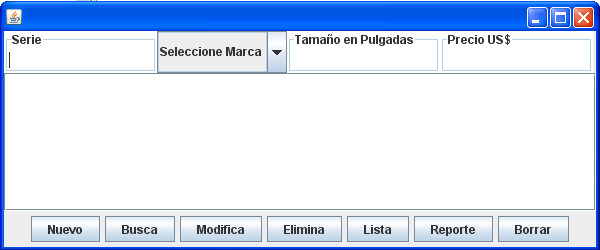
En el botón **Lista**, muestra un listado de todos los objetos.

En el botón **Reporte**, muestra la siguiente información: mayor precio, precio promedio, menor precio. Todos los precios en soles y dólares.

En el botón **Borrar** borra la información del área de texto, de las cajas de texto y enfoca el ingreso en la caja de texto de la serie.

Copie la clase **Principal** del ejercicio anterior y péguelo en éste paquete.

Ejecute su aplicación.

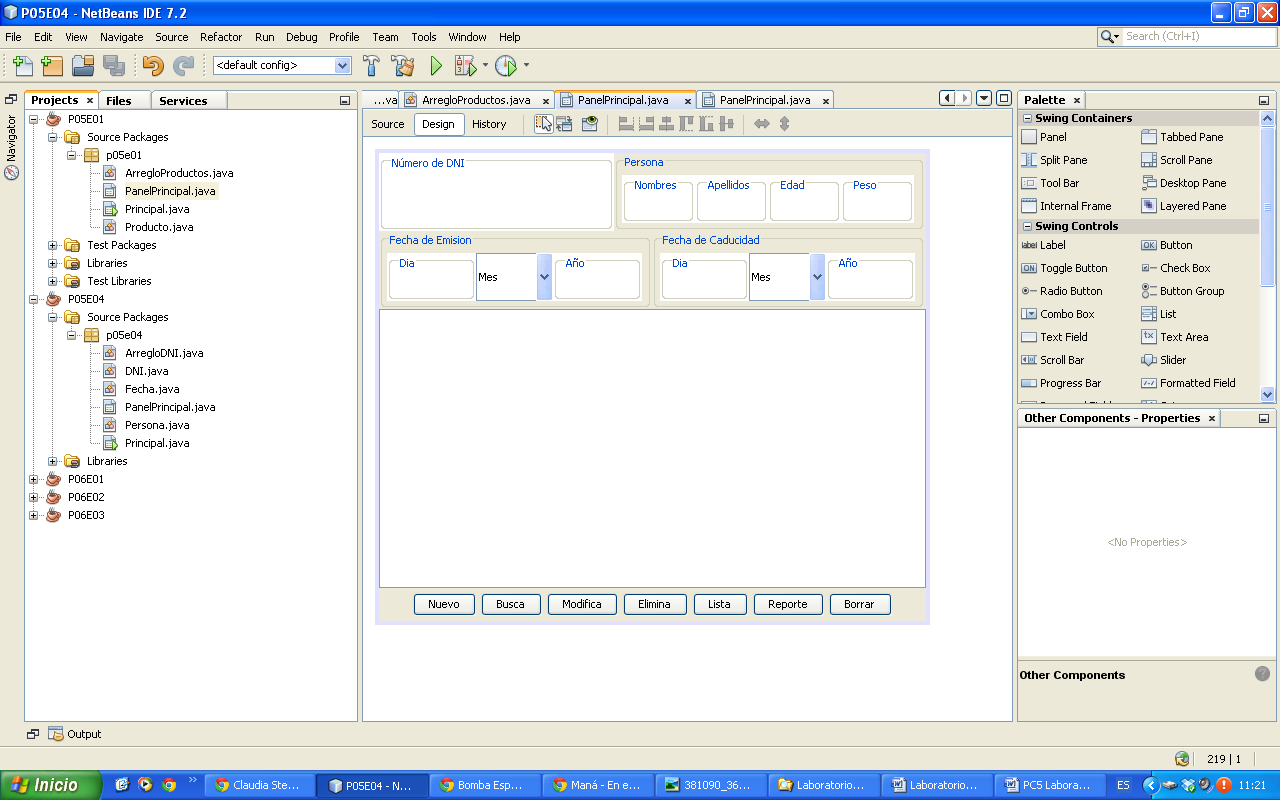


**Ejercicio 4**

Cree un proyecto nuevo de nombre **P03E04**. Cree un paquete nuevo de nombre **p03e04**. Copie y pegue en el paquete creado, desde el proyecto P02E04 la clase **DNI.**

Diseñe una nueva clase administradora de nombre **ArregloDNI**, que tenga como atributos privados un arreglo de objetos y un contador de objetos que se guardan en el arreglo. Considere un tamaño fijo de 50 para el arreglo. Considere dentro de la funcionalidad de la clase los métodos necesarios de administración del arreglo de objetos para: agregar un nuevo objeto, recuperar, actualizar, buscar, eliminar.

Diseñe una nueva clase gestora tipo **Panel** de nombre **PanelPrincipal** con la interface gráfica de usuario necesaria para gestionar, sobre el arreglo de objetos, los procesos considerados en el siguiente diseño:



Vaya a la ficha **Source** para declarar y crear el objeto de tipo Arreglo DNI

Regrese al diseño y con doble clic en cada botón programe su acción considerando lo siguiente:

En el botón **Nuevo**, crea un objeto de la clase **DNI**, lo guarda en el arreglo de objetos y muestra un listado de todos los objetos. Evita la duplicidad de serie.

En el botón **Busca**, ubica el objeto que tiene la serie ingresada y muestra su información. En caso que no lo encuentre muestra un mensaje de error en una ventana.

En el botón **Modifica**, actualiza la información de objeto que tiene la serie ingresada y muestra un listado de todos los objetos.

En el botón **Elimina**, elimina el objeto que tiene la serie ingresada y muestra un listado de todos los objetos.

En el botón **Lista**, muestra un listado de todos los objetos.

En el botón **Reporte**, muestra la siguiente información: mayor precio, precio promedio, menor precio. Todos los precios en soles y dólares.

En el botón **Borrar** borra la información del área de texto, de las cajas de texto y enfoca el ingreso en la caja de texto de la serie.

Copie la clase **Principal** del ejercicio anterior y péguelo en éste paquete.

Ejecute su aplicación.

