Práctica 4

10 de octubre del 2019

Estructura de Datos

Número de Control – Nombre:

18131209 – ADAME SANDOVAL JOSE MISAEL

Descripción:

COMPETENCIA 11.

1. Escribe un metodo en la clase Cola, que elimine al ultimo elemento de la cola. NO USES PROPIEDADES NI METODOS DE LA CLASE COLA..

2. Escribe un metodo en la clase Cola, que elimine a todos los elementos de la cola menos al primero. NO USES PROPIEDADES NI METODOS DE LA CLASE COLA.

3.Escribe un metodo en la clase Cola, que elimine a todos los elementos de la cola menos al penultimo. NO USES PROPIEDADES NI METODOS DE LA CLASE COLA.

COMPETENCIA 12.

1. Agrega un boton, que elimine al ultimo elemento de la cola.

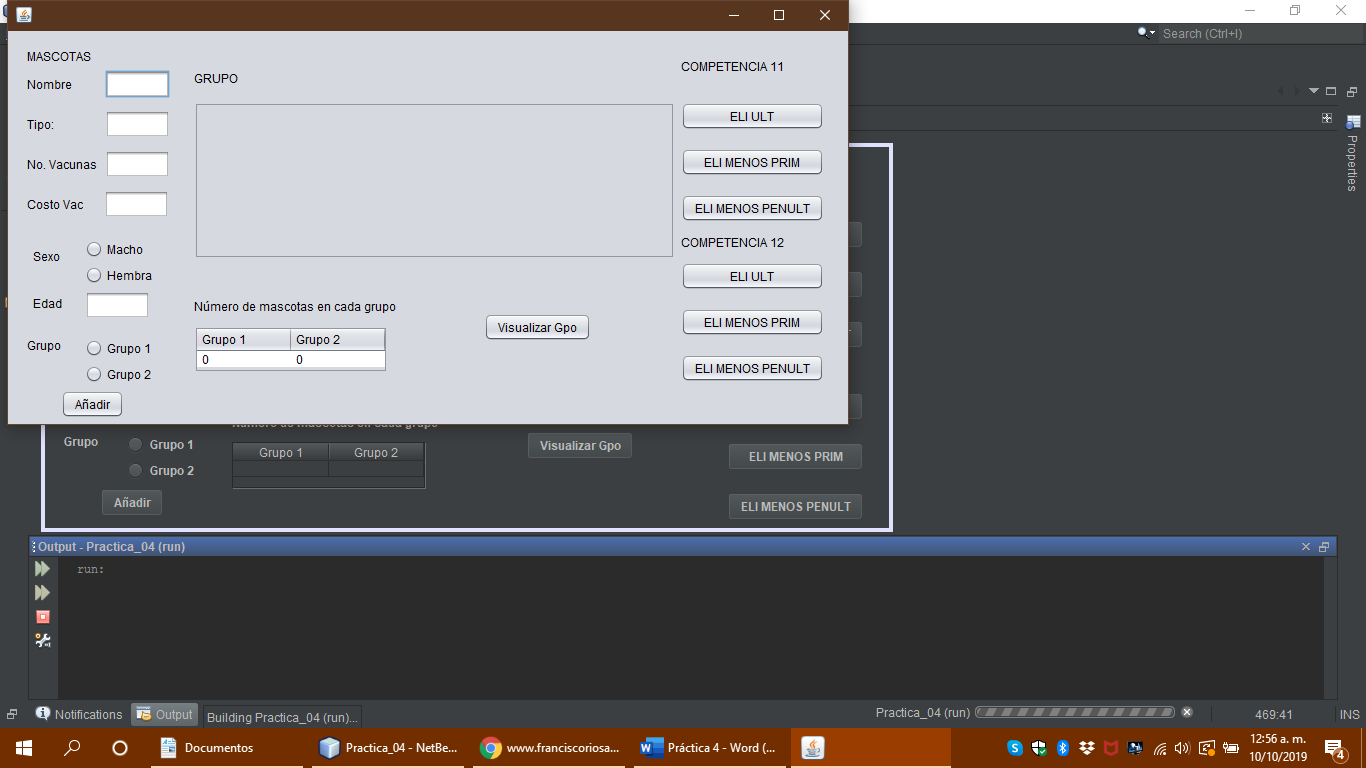
2. Agrega un boton que elimine a todos los elementos de la cola menos al primero

3.Agrega un boton que elimine a todos los elementos de la cola menos al penultimo.

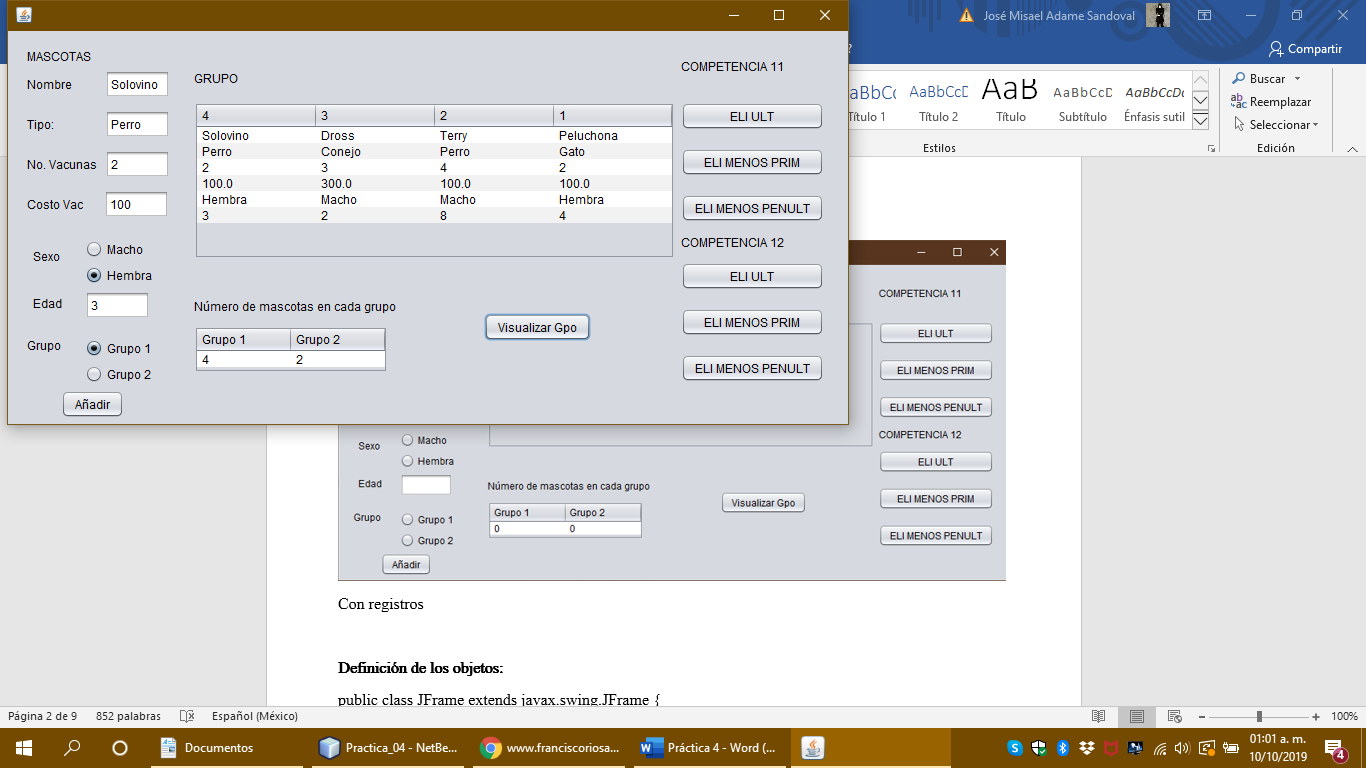
**Desarrollo:**

Interfaz Gráfica

al inicio



Con registros (se muestra el grupo 1)



**Definición de los objetos:**

public class JFrame extends javax.swing.JFrame {

Cola grupo1 = new Cola();

Cola grupo2 = new Cola();

public JFrame() {

initComponents();

}

**Clases:**

**Mascota**

public class Mascota {

private String nombre;

private String tipo;

private int noVacunas;

private double costoVacunas;

private String sexo;

private int edad;

public Mascota(String nombre, String tipo, int noVacunas,

double costoVacunas, String sexo, int edad) {

this.nombre = nombre;

this.tipo = tipo;

this.noVacunas = noVacunas;

this.costoVacunas = costoVacunas;

this.sexo = sexo;

this.edad = edad;

}

public Mascota() {

// Solo reservo espacio de memoria

}

public String getNombre() {

return nombre;

}

public String getTipo() {

return tipo;

}

public int getNoVacunas() {

return noVacunas;

}

public double getCostoVacunas() {

return costoVacunas;

}

public String getSexo() {

return sexo;

}

public int getEdad() {

return edad;

}

}

**Cola**

public class Cola {

private ArrayList<Mascota> \_elems;

private int \_capacidad;

public Cola() {

\_elems = new ArrayList<>();

\_capacidad = 0; // cola sin limites

}

public Cola(int dimension) {

\_elems = new ArrayList<>();

\_capacidad = dimension > 0? dimension : 0;

}

public boolean Empty() {

return \_elems.isEmpty();

}

public boolean Full() {

return \_capacidad > 0 && \_elems.size() == \_capacidad;

}

public int Length() {

return \_elems.size();

}

public void Insert(Mascota elem) {

if (!Full())

\_elems.add(elem);

}

public Mascota Remove() {

return !Empty()? \_elems.remove(0) : new Mascota();

}

public void EliminarUltimo() {

if(\_elems.size() > 0)

\_elems.remove(\_elems.size()-1);

}

public void EliminarMenosPrimero() {

while(\_elems.size() > 1)

\_elems.remove(1);

}

public void EliminarMenosPenultimo() {

if(\_elems.size() > 0) {

\_elems.remove(\_elems.size()-1);

while(\_elems.size() > 1)

\_elems.remove(0);

}

}

}

**Métodos de la aplicación:**

private void formWindowOpened(java.awt.event.WindowEvent evt) {

jTabNoMascotas.setValueAt(grupo1.Length(), 0, 0);

jTabNoMascotas.setValueAt(grupo2.Length(), 0, 1);

}

private void actualizarNoMascotasGpos() {

jTabNoMascotas.setValueAt(grupo1.Length(), 0, 0);

jTabNoMascotas.setValueAt(grupo2.Length(), 0, 1);

}

private void jButAñadirActionPerformed(java.awt.event.ActionEvent evt) {

Cola grupoSel = rb1.isSelected()? grupo1 : grupo2;

if(jTextNombre.getText().isEmpty() || jTextTipo.getText().isEmpty() ||

jTextNoVacunas.getText().isEmpty() || jTextCosto.getText().isEmpty()

|| jTextEdad.getText().isEmpty()) {

JOptionPane.showMessageDialog(this, "NO SE PUDO GUARDAR LA MASCOTA"

+ " , DEBE LLENAR TODOS LOS CAMPOS",

"ERROR", JOptionPane.ERROR\_MESSAGE); }

else if(! grupoSel.Full()) {

Mascota m = new Mascota(jTextNombre.getText(), jTextTipo.getText(),

Integer.parseInt(jTextNoVacunas.getText()),

Double.parseDouble(jTextCosto.getText()),

rbM.isSelected()? "Macho" : "Hembra",

Integer.parseInt(jTextEdad.getText()));

grupoSel.Insert(m);

jTabNoMascotas.setValueAt(grupoSel.Length(), 0, rb1.isSelected()? 0 : 1);

} else {

JOptionPane.showMessageDialog(this, "YA ESTA LLENO EL GRUPO",

"ERROR", JOptionPane.PLAIN\_MESSAGE);

}

}

private void jButVisualizarGpoActionPerformed(java.awt.event.ActionEvent evt) {

Cola grupoSel = rb1.isSelected()? grupo1 : grupo2;

Cola aux = new Cola(grupoSel.Length());

DefaultTableModel modelo = (DefaultTableModel) jTabMascotas.getModel();

modelo.setColumnCount(0);

for(int i=grupoSel.Length();i > 0;i--)

modelo.addColumn(i);

int col = grupoSel.Length() - 1;

while (!grupoSel.Empty()) {

Mascota m = grupoSel.Remove();

jTabMascotas.setValueAt(m.getNombre(),0,col);

jTabMascotas.setValueAt(m.getTipo(),1,col);

jTabMascotas.setValueAt(m.getNoVacunas(),2,col);

jTabMascotas.setValueAt(m.getCostoVacunas(), 3, col);

jTabMascotas.setValueAt(m.getSexo(), 4, col);

jTabMascotas.setValueAt(m.getEdad(), 5, col--);

aux.Insert(m);

}

while (!aux.Empty())

grupoSel.Insert(aux.Remove());

}

private void jButEliUltClaseActionPerformed(java.awt.event.ActionEvent evt) {

Cola grupoSel = rb1.isSelected()? grupo1 : grupo2;

grupoSel.EliminarUltimo();

actualizarNoMascotasGpos();

}

private void jButEliMenosPrimClaseActionPerformed(java.awt.event.ActionEvent evt) {

Cola grupoSel = rb1.isSelected()? grupo1 : grupo2;

grupoSel.EliminarMenosPrimero();

actualizarNoMascotasGpos();

}

private void jButEliMenosPenultClaseActionPerformed(java.awt.event.ActionEvent evt) {

Cola grupoSel = rb1.isSelected()? grupo1 : grupo2;

grupoSel.EliminarMenosPenultimo();

actualizarNoMascotasGpos();

}

private void jButEliUltFrameActionPerformed(java.awt.event.ActionEvent evt) {

Cola grupoSel = rb1.isSelected()? grupo1 : grupo2;

if(!grupoSel.Empty()) {

Cola aux = new Cola();

while(grupoSel.Length() > 1)

aux.Insert(grupoSel.Remove());

grupoSel.Remove();

while(!aux.Empty())

grupoSel.Insert(aux.Remove());

actualizarNoMascotasGpos();

}

}

private void jButEliMenosPrimFrameActionPerformed(java.awt.event.ActionEvent evt) {

Cola grupoSel = rb1.isSelected()? grupo1 : grupo2;

if(grupoSel.Length() > 1) {

Cola aux = new Cola();

aux.Insert(grupoSel.Remove());

while(!grupoSel.Empty())

grupoSel.Remove();

grupoSel.Insert(aux.Remove());

actualizarNoMascotasGpos();

}

}

private void jButEliMenosPenultFrameActionPerformed(java.awt.event.ActionEvent evt) {

Cola grupoSel = rb1.isSelected()? grupo1 : grupo2;

if(!grupoSel.Empty()) {

Cola aux = new Cola();

while(grupoSel.Length() > 2)

grupoSel.Remove();

aux.Insert(grupoSel.Remove());

grupoSel.Remove();

grupoSel.Insert(aux.Remove());

actualizarNoMascotasGpos();

}

}