

Práctica 1

6 de septiembre del 2019

Estructura de Datos

Número de control – Nombre:

18131209 – ADAME SANDOVAL JOSE MISAEL

18131222 – CARREON REYES JOSE LUIS

Descripción:

ESCRIBE UNA APLICACION JAVA NETBEANS VISUAL QUE LEA Y VISUALICE A 2 GRUPOS DE MASCOTAS. ALMACENA A CADA GRUPO EN UN ARREGLO DINAMICO.

DE UNA MASCOTA INTERESA: SU NOMBRE, SU TIPO (PERRO, OSO, GATO, ETCETERA), EL NUMERO DE VACUNAS QUE LE HAN PUESTO Y EL COSTO TOTAL DE ESAS VACUNAS. ADEMAS INTERESA SI ES MACHO O HEMBRA, Y SU EDAD.

1. AGREGA UN BOTON QUE INTERCAMBIE LA PRIMER MASCOTA CON LA ULTIMA DEL GRUPO SELECCIONADO. UTILIZA EL METODO `set()` PARA OBJETOS `ARRAYLIST`
2. AGREGA UN BOTON QUE ELIMINE A TODAS LAS MASCOTAS CUYO COSTO DE VACUNAS SEAN MAYORES A 100 PESOS, DE AMBOS GRUPOS. UTILIZA UN METODO EN LA APLICACION.
3. AGREGA UN BOTON QUE INTERCAMBIE A LAS ULTIMAS MASCOTAS DE AMBOS GRUPOS. AYUDA: AMBOS GRUPOS DEBEN TENER AL MENOS UNA MASCOTA.

Desarrollo:

Interfaz Gráfica

AÑADIR MASCOTA

Nombre

Tipo

Vacunas

Costo

Sexo ☒ Macho ☐ Hembra

Edad

Grupo

NO. MASCOTAS EN CADA GRUPO:

Grupo 1	Grupo 2
0	0

#	Nombre	Tipo	# Vac	\$ Vac	Sexo	Edad
---	--------	------	-------	--------	------	------

1a mascota --> última / mismo grupo

Costo Vacunas > 100 / ambos grupos

últ masc <--> últ masc. / ambos grupos

Definición de los objetos

```
public class JFrame extends javax.swing.JFrame {  
    private ArrayList<Mascota> grupo1 = new ArrayList<>();  
    private ArrayList<Mascota> grupo2 = new ArrayList<>();  
  
    public JFrame() {  
        initComponents();  
    }  
}
```

Clases

```
public class Mascota {  
    // Atrib  
    private String nombre;  
    private String tipo;  
    private int noVacunas;
```

```
private double costoVacunas;  
private char sexo;  
private int edad;
```

// Get

```
public String getNombre(){  
    return nombre;  
}
```

```
public String getTipo(){  
    return tipo;  
}
```

```
public int getNoVacunas(){  
    return noVacunas;  
}
```

```
public double getCostoVacunas(){  
    return costoVacunas;  
}
```

```
public char getSexo(){  
    return sexo;  
}
```

```
public int getEdad(){  
    return edad;  
}
```

// Set

```
public void setNombre(String nom){  
    nombre = nom;  
}
```

```
public void setTipo(String tip){  
    tipo = tip;  
}
```

```
public void setNoVacunas(int noV){  
    noVacunas = noV;  
}
```

```
public void setCostoVacunas(double cosV){  
    costoVacunas = cosV;  
}
```

```
public void setSexo(char sex){  
    sexo = sex;  
}
```

```
public void setEdad(int ed){  
    edad = ed;  
}
```

// Met

```
public void Leer(String nom, String tip, int noV, double cosV, char sex, int ed){  
    nombre = nom;  
    tipo = tip;  
    noVacunas = noV;  
    costoVacunas = cosV;
```

facto
vittor

```
sexo = sex;
```

```
edad = ed;
```

```
}
```

```
}
```

Métodos de la aplicación

```
private ArrayList<Mascota> getGrupoSel() {  
    int noGrupoSel = jComboGrupo.getSelectedIndex();  
    switch(noGrupoSel){  
        case 0:  
            return grupo1;  
        case 1:  
            return grupo2;  
        default:  
            return grupo1;  
    }  
}
```

```
private void eliminarPorCosto(ArrayList<Mascota> grupo){  
    for (int i = 0; i < grupo.size(); i++){  
        if(grupo.get(i).getCostoVacunas() > 100){  
            grupo.remove(i);  
        }  
    }  
}
```

```
private void jButtonAnadirActionPerformed(java.awt.event.ActionEvent evt) {  
    ArrayList<Mascota> grupoSel = getGrupoSel();  
    Mascota masc = new Mascota();  
    masc.Leer(jTextNombre.getText(), jTextTipo.getText(),  
        Integer.parseInt(jTextNoVacunas.getText()),  
        Double.parseDouble(jTextCosto.getText()), jRadioM.isSelected() ? 'M' : 'F',
```



```
        Integer.parseInt(jTextEdad.getText()));  
        grupoSel.add(masc);  
        jTabNoMasc.setValueAt(grupoSel.size(), 0, jComboGrupo.getSelectedIndex());  
    }
```

```
private void formWindowOpened(java.awt.event.WindowEvent evt) {  
    jTabNoMasc.setValueAt(grupo1.size(), 0, 0);  
    jTabNoMasc.setValueAt(grupo1.size(), 0, 1);  
}
```

```
private void jButtVisGpoActionPerformed(java.awt.event.ActionEvent evt) {  
    ArrayList<Mascota> grupoSel = getGrupoSel();  
    jLabGrupoVis.setText("Grupo " + Integer.toString(jComboGrupo.getSelectedIndex() + 1));  
    DefaultTableModel modelo = (DefaultTableModel) jTabMascotas.getModel();  
    modelo.setRowCount(0);  
    int i = 0;  
    for(Mascota masc : grupoSel){  
        Object[] o = new Object[7];  
        o[0] = ++i;  
        o[1] = masc.getNombre();  
        o[2] = masc.getTipo();  
        o[3] = masc.getNoVacunas();  
        o[4] = masc.getCostoVacunas();  
        o[5] = masc.getSexo();  
        o[6] = masc.getEdad();  
        modelo.addRow(o);  
    }  
}
```

```
private void jButtonIntercambiar1ActionPerformed(java.awt.event.ActionEvent evt) {  
    ArrayList<Mascota> grupoSel = getGrupoSel();  
    Mascota aux = grupoSel.get(0);  
    grupoSel.set(0, grupoSel.get(grupoSel.size() - 1));  
    grupoSel.set(grupoSel.size() - 1, aux);  
}
```

```
private void jButtonIntercambiar3ActionPerformed(java.awt.event.ActionEvent evt) {  
    Mascota aux = grupo1.get(grupo1.size() - 1);  
    grupo1.set(grupo1.size() - 1, grupo2.get(grupo2.size() - 1));  
    grupo2.set(grupo2.size() - 1, aux);  
}
```

```
private void jButtonEliminar2ActionPerformed(java.awt.event.ActionEvent evt) {  
    eliminarPorCosto(grupo1);  
    eliminarPorCosto(grupo2);  
}
```