

Nombre: Andrew Vilcacundo

Link del repositorio: <https://github.com/AndrewVilcacundo/PRUEBA2222.git>

```
Datos.java x registro.java x registro.form x
1  import java.io.Serializable;
   5 usages new *
2  public class Datos implements Serializable {
   4 usages
3      String nombre;
   4 usages
4      Double codigo;
   4 usages
5      int cedula;
   4 usages
6      String apellido;
7
8      2 usages new *
   public String getNombre() {
9          return nombre;
10     }
11
12     no usages new *
   public void setNombre(String nombre) {
13         this.nombre = nombre;
14     }
15
16     2 usages new *
   public Double getCodigo() {
17         return codigo;
18     }
```

```
Datos.java x registro.java x registro.form x
36     public void setApellido(String apellido) {
37         this.apellido = apellido;
38     }
39
40     1 usage new *
   public Datos(String nombre, Double codigo, int cedula, String apellido) {
41         this.nombre = nombre;
42         this.codigo = codigo;
43         this.cedula = cedula;
44         this.apellido = apellido;
45     }
46
47     new *
   @Override
48     public String toString() {
49         return "Datos{" +
50             "nombre=" + nombre + '\'' +
51             ", codigo=" + codigo +
52             ", cedula=" + cedula +
53             ", apellido=" + apellido + '\'' +
54             '}';
55     }
56 }
57
58
```

```
Datos.java x registro.java x registro.form x
1 import javax.swing.*;
2 import java.awt.event.ActionEvent;
3 import java.awt.event.ActionListener;
4 import java.io.FileOutputStream;
5 import java.io.IOException;
6 import java.io.ObjectOutputStream;
7 import java.util.ArrayList;
8
9 AndrewVilcacundo *
10 public class registro {
11     private JPanel registro;
12     private JTextField marca;
13     private JLabel vehiLabel;
14     private JLabel nombre;
15     private JLabel codigo;
16     private JTextField velocidad;
17     private JLabel cedula;
18     private JTextField anio;
```

```
Datos.java x registro.java x registro.form x
40
41 3 usages new *
42 public registro() {
43     listaDatos = new ArrayList<>();
44     new *
45     regBt.addActionListener(new ActionListener() {
46         new *
47         @Override
48         public void actionPerformed(ActionEvent e) {
49             String nombreIngreso = nombre.getText();
50             double codigoIngreso = Double.parseDouble(codigo.getText());
51             int cedulaIngreso = Integer.parseInt(cedula.getText());
52             String apellidoIngreso = apellido.getText();
53
54             Datos ingreso = new Datos(nombreIngreso, codigoIngreso, cedulaIngreso, apellidoIngreso);
55             listaDatos.add(ingreso);
56
57             for(int i = 0; i < listaDatos.size(); i++){
58                 String filePath = i + "binarydata.dat";
59                 try{
60                     FileOutputStream fileOut = new FileOutputStream(filePath);
61                     ObjectOutputStream objOut = new ObjectOutputStream(fileOut);
62                 }{
63                     objOut.writeObject(ingreso);
64                     System.out.println("Archivo guardado correctamente");
65                 }
66             }
67         }
68     });
69 }
```

```

89      new *
      atrBt.addActionListener(new ActionListener() {
90          new *
          @Override
91          public void actionPerformed(ActionEvent e) {
92              ind--;
93              if(ind > 0){
94                  Datos datos = listaDatos.get(ind);
95                  marca.setText(datos.getNombre());
96                  codigo.setText(String.valueOf(datos.getCodigo()));
97                  cedula.setText(String.valueOf(datos.getCedula()));
98                  apellido.setText(datos.getApellido());
99              }
100          }
101      });
102  }
103
104  new *
105  public static void main(String[] args) {
106      JFrame frame = new JFrame( title: "Registro Vehiculos");
107      frame.setContentPane(new registro().registro);
108      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
109      frame.pack();
110      frame.setVisible(true);
111  }

```

Component Tree

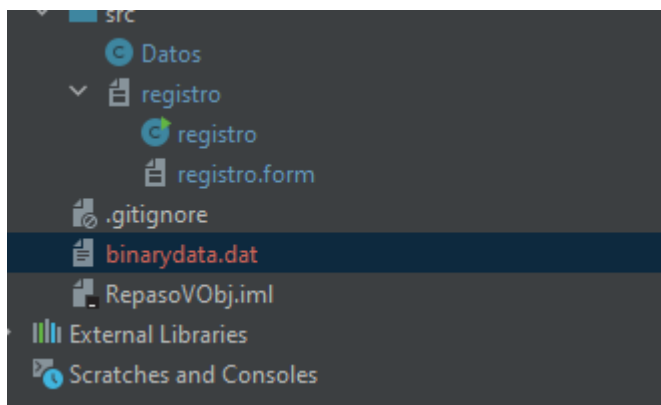
- Form (registro)
 - JPanel
 - registro : JPanel

Property Value

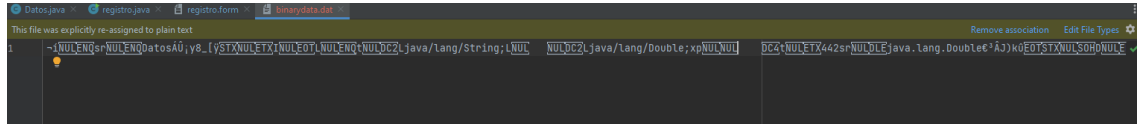
Property	Value
field name	a
Custom Create	
Layout Manager	GridLayoutManager (IntelliJ)
border	None
margins	[0, 0, 0, 0]
Horizontal Gap	-1
Vertical Gap	-1
Same Size Horizontally	
Same Size Vertically	
Client Properties	
background	[60, 63, 65]
enabled	<input checked="" type="checkbox"/>
font	<default>

Form Preview:

Vehiculos				
Codigo	<input type="text"/>	Fecha de nacimiento	año <input type="text"/>	mes <input type="text"/>
Cédula	<input type="text"/>	color favorito	<input type="checkbox"/> Rojo	<input type="checkbox"/> Verde
Nombres	<input type="text"/>	Casado	<input type="radio"/> Si	<input type="radio"/> No
Apellido	<input type="text"/>			
Signo	Acuario <input type="text"/>			
Cargar datos desde el disco		Guardar datos en disco		Ver siguiente registro
Ver anterior registro				



Binarydata.dat



EJECUCION:

A screenshot of a Java application window titled 'Registro Vehículos'. The window contains a form with the following fields and controls:

- Codigo:** Text input field with value '135'.
- Cédula:** Text input field with value '1424'.
- Nombres:** Text input field with value 'Andrew'.
- Apellido:** Text input field with value 'Vilcacundo'.
- Signo:** Dropdown menu with 'Escorpio' selected.
- Fecha de nacimiento:** Three dropdown menus for year (2003), month (noviembre), and day (15).
- color favorito:** Three checkboxes: 'Rojo' (checked), 'Verde' (unchecked), and 'Ninguno' (unchecked).
- Casado:** Two radio buttons: 'Si' (selected) and 'No' (unchecked).

At the bottom of the form, there are four buttons: 'Cargar datos desde el disco', 'Guardar datos en disco', 'Ver siguiente registro', and 'Ver anterior registro'.