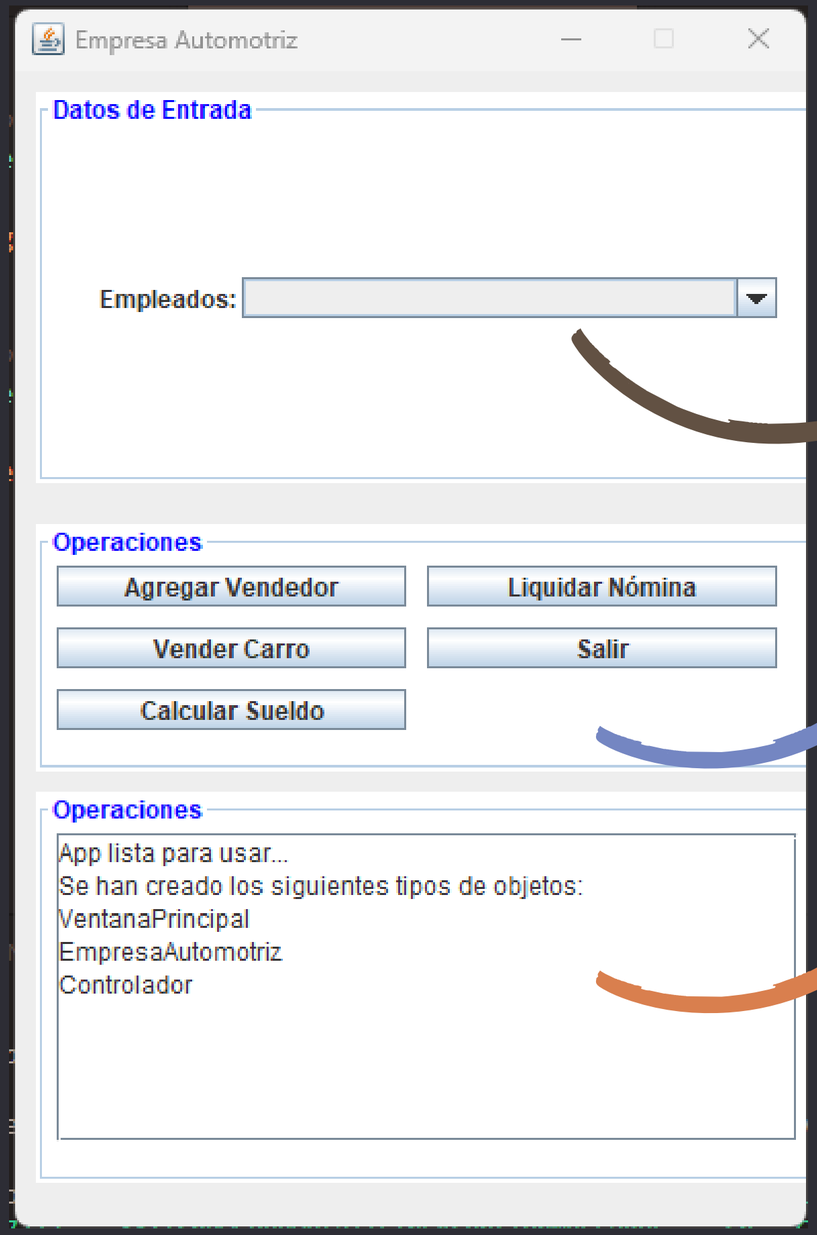
<!--Programación Orientada a Objetos-->

Empresa Automotriz

<Por="JAIRO ARMANDO CARDOZO MENDOZA"

}

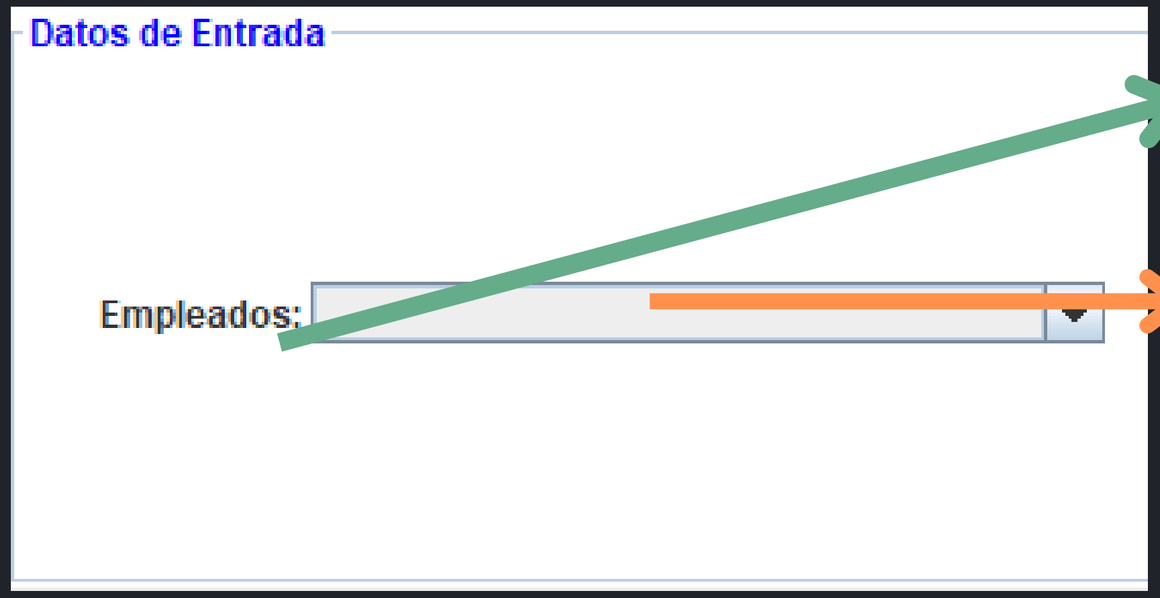
Ventana Principal{




```java
//Constructor
public VentanaPrincipal()
{
    //Definición del layout de la ventana
    this.setLayout(manager: null);

    //Creación y adición del PanelEntradaDatos
    miPanelEntradaDatos = new PanelEntradaDatos();
    miPanelEntradaDatos.setBounds(x: 10,y: 10,width: 380,height: 190);
    this.add(miPanelEntradaDatos);

    //Creación y adición del PanelOperaciones
    miPanelOperaciones = new PanelOperaciones();
    miPanelOperaciones.setBounds(x: 10,y: 220,width: 380,height: 120);
    this.add(miPanelOperaciones);

    //Creación y adición del PanelResultados
    miPanelResultados = new PanelResultados();
    miPanelResultados.setBounds(x: 10,y: 350,width: 380,height: 190);
    this.add(miPanelResultados);

    //Caracteristicas de la ventana
    this.setTitle(title: "Empresa Automotriz");
    this.setSize(width: 400,height: 600);
    this.setLocationRelativeTo(c: null);
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);
    this.setResizable(resizable: false);
    this.setVisible(b: true);
}
```

}

Panel Entrada Datos {

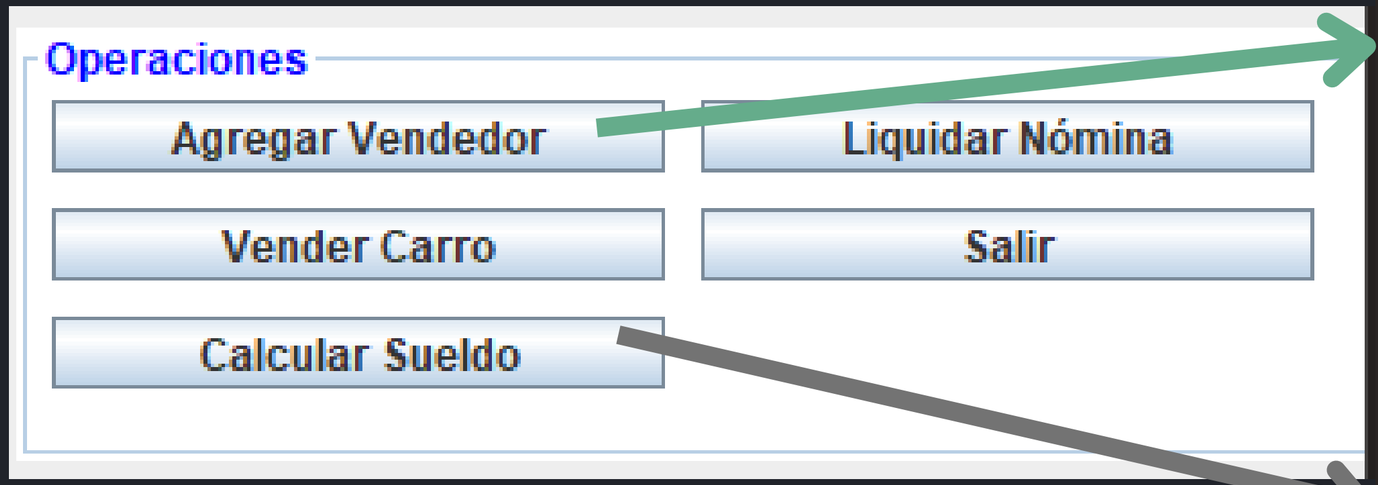



```java
//Definición del contenedor del panel
this.setLayout(mgr: null);
this.setBackground(Color.WHITE);

//Crear y agregar etiqueta empleados
lbEmpleados = new JLabel(text: "Empleados: ", JLabel.RIGHT);
lbEmpleados.setBounds(x: 0,y: 90,width: 100,height: 20);
this.add(lbEmpleados);

//Crear y agregar combo lista Empleados
cbEmpleados = new JComboBox();
cbEmpleados.setBounds(x: 100,y: 90,width: 260,height: 20);
this.add(cbEmpleados);

//Borde y titulo del panel
TitledBorder borde = BorderFactory.createTitledBorder(title: "Datos de Entrada");
borde.setTitleColor(Color.BLUE);
this.setBorder(borde);
```

Para el nombre "Empleados: " implementamos el método JLabel, y para la lista utilizamos el método JComboBox

}

Panel Operaciones {

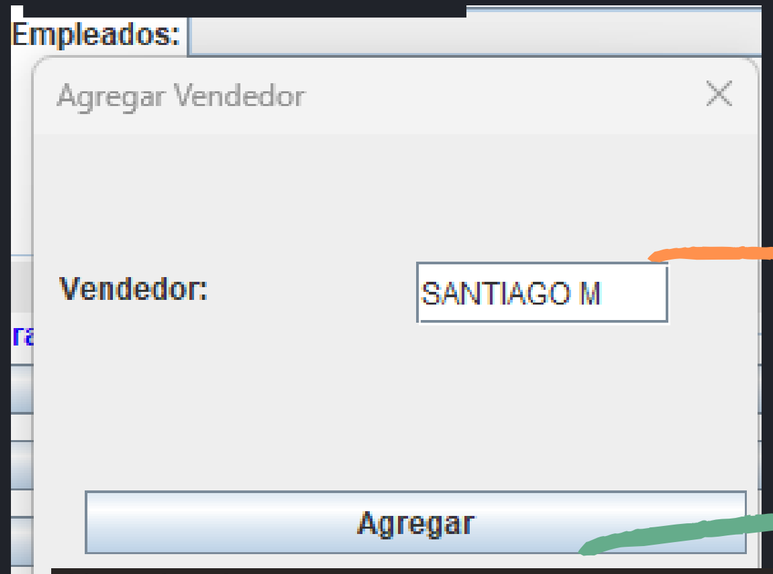



```java
//Definición del contenedor del panel
this.setLayout(mgr: null);
this.setBackground(Color.WHITE);

//Crear y agregar boton AgregarVendedor
btAgregarVendedor = new JButton(text: "Agregar Vendedor");
btAgregarVendedor.setBounds(x: 10,y: 20,width: 170,height: 20);
btAgregarVendedor.setActionCommand(actionCommand: "agregarVendedor");
this.add(btAgregarVendedor);

//Crear y agregar boton VenderCarro
btVenderCarro = new JButton(text: "Vender Carro");
btVenderCarro.setBounds(x: 10,y: 50,width: 170,height: 20);
btVenderCarro.setActionCommand(actionCommand: "venderCarro");
this.add(btVenderCarro);

//Crear y agregar boton CalcularSueldo
btCalcularSueldo = new JButton(text: "Calcular Sueldo");
btCalcularSueldo.setBounds(x: 10,y: 80,width: 170,height: 20);
btCalcularSueldo.setActionCommand(actionCommand: "calcularSueldo");
this.add(btCalcularSueldo);

//Crear y agregar boton LiquidarNomina
btLiquidarNomina = new JButton(text: "Liquidar Nómina");
btLiquidarNomina.setBounds(x: 190,y: 20,width: 170,height: 20);
btLiquidarNomina.setActionCommand(actionCommand: "liquidarNomina");
this.add(btLiquidarNomina);

//Crear y agregar boton Salir
btSalir = new JButton(text: "Salir");
btSalir.setBounds(x: 190,y: 50,width: 170,height: 20);
btSalir.setActionCommand(actionCommand: "salir");
this.add(btSalir);
```

}

En el Panel Operaciones encontramos cada uno de los botones que creamos para este trabajo cada uno utiliza el método **"JButton"** cada botón tiene sus medidas necesarias.

Dialogo Agregar Vendedor{



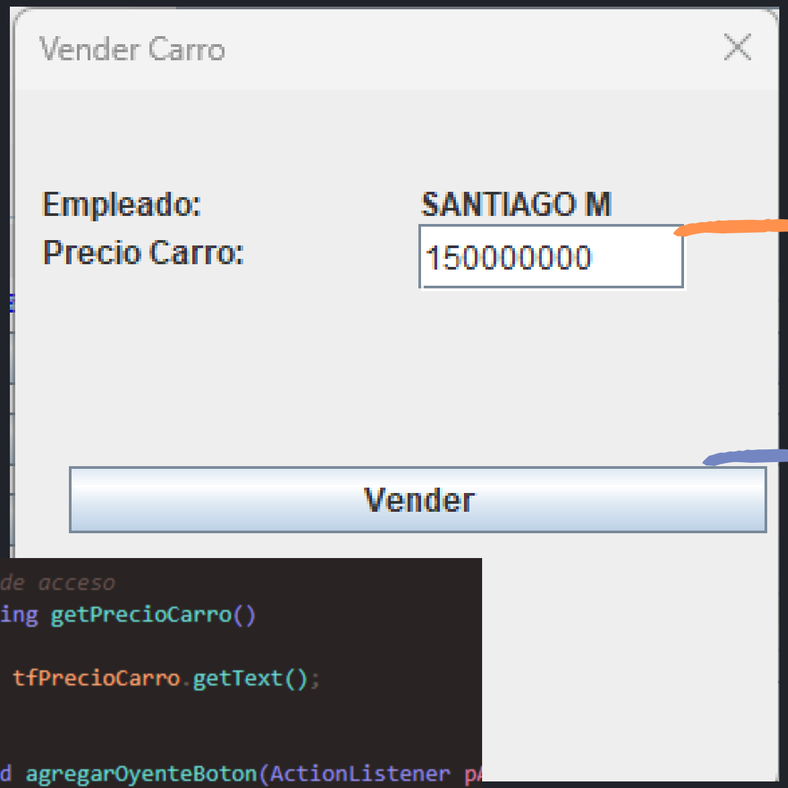

```java
//Metodos de acceso
public String getNombreVendedor()
{
    return tfNombreVendedor.getText();
}

public void agregarOyenteBoton(ActionListener pAL)
{
    btAgregarVendedor.addActionListener(pAL);
}

public void cerrarDialogoAgregarVendedor()
{
    this.dispose();
}
```

```java
//Definición del layout del Dialogo
this.setLayout(manager: null);

//Crear y agregar elementos
lbNombreVendedor = new JLabel(text: "Vendedor: ");
lbNombreVendedor.setBounds(x: 10,y: 50,width: 140,height: 20);
this.add(lbNombreVendedor);

tfNombreVendedor = new JTextField();
tfNombreVendedor.setBounds(x: 150,y: 50,width: 100, height: 25);
this.add(tfNombreVendedor);

btAgregarVendedor = new JButton(text: "Agregar");
btAgregarVendedor.setBounds(x: 20,y: 140,width: 260,height: 25);
btAgregarVendedor.setActionCommand(actionCommand: "agregar");
this.add(btAgregarVendedor);

//Caracteristicas de la ventana
this.setTitle(title: "Agregar Vendedor");
this.setSize(width: 300,height: 300);
this.setLocationRelativeTo(c: null);
this.setResizable(resizable: false);
this.setVisible(b: true);
```

Agregamos una clase llamada "Dialogo Agregar Vendedor" para crear otra pantalla adicional donde podamos agregar el nombre del empleado (JTextField) y el boton agregar vendedor (JButton)

}

Dialogo Vender Carro{



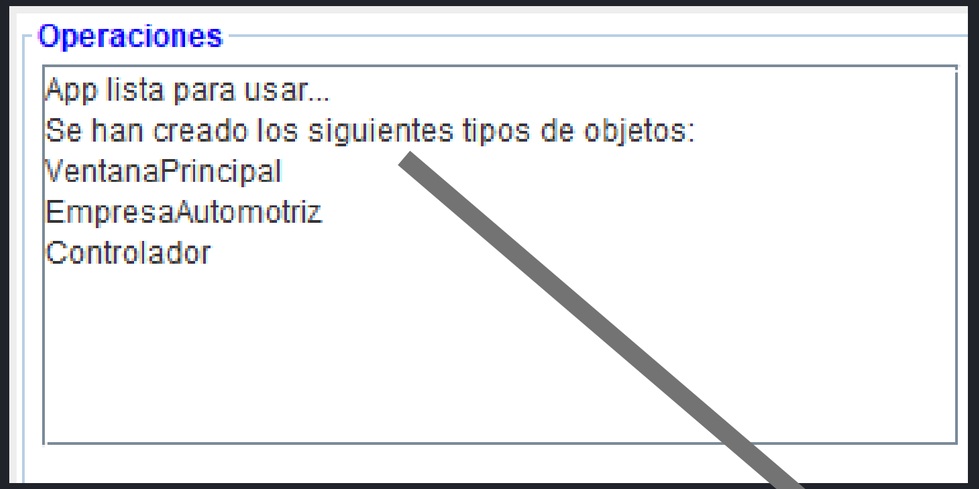

```java
//Crear y agregar elementos
lbPrecioCarro = new JLabel(text: "Precio Carro: ");
lbPrecioCarro.setBounds(x: 10,y: 50,width: 140,height: 20);
this.add(lbPrecioCarro);

lbNombre = new JLabel();
lbNombre.setBounds(x: 150,y: 30,width: 100,height: 25);
this.add(lbNombre);

lbEmpleado = new JLabel(text: "Empleado: ");
lbEmpleado.setBounds(x: 10,y: 30,width: 100,height: 25);
this.add(lbEmpleado);

tfPrecioCarro = new JTextField();
tfPrecioCarro.setBounds(x: 150,y: 50,width: 100, height: 25);
this.add(tfPrecioCarro);

btVenderCarro = new JButton(text: "Vender");
btVenderCarro.setBounds(x: 20,y: 140,width: 260,height: 25);
btVenderCarro.setActionCommand(actionCommand: "vender");
this.add(btVenderCarro);
```

```java
//Metodos de acceso
public String getPrecioCarro()
{
    return tfPrecioCarro.getText();
}

public void agregarOyenteBoton(ActionListener pA
{
    btVenderCarro.addActionListener(pAL);
}

public void cerrarDialogoVenderCarro()
{
    this.dispose();
}

public void mostrarResultado(String msj)
{
    //taResultado.append(msj + "\n");
    lbNombre.setText(msj);
}
```

Agregamos una clase llamada "Dialogo Vender Carro" para crear otra pantalla adicional donde podamos visualizar el nombre del empleado (JLabel) el precio del carro que vamos a digitar (JTextField) y el boton de vender o agregar esa venta (JButton)

}

Panel Resultados {



En el Panel Resultados encontramos el código **"public void mostarResultado"** donde agregamos el mensaje que aparece en el panel a través del método **"JTextArea"**

```java
public PanelResultados()
{
    //Definición del contenedor del panel
    this.setLayout(mgr: null);
    this.setBackground(Color.WHITE);

    //Crear y agregar area de texto Resultados
    taResultado = new JTextArea();
    spResultado = new JScrollPane(taResultado);
    spResultado.setBounds(x: 10,y: 20,width: 360,height: 150);
    this.add(spResultado);

    //Borde y titulo del panel
    TitledBorder borde = BorderFactory.createTitledBorder(title: "Operaciones");
    borde.setTitleColor(Color.BLUE);
    this.setBorder(borde);
}

//Metodos de acceso
public void mostrarResultado(String msj)
{
    //taResultado.append(msj + "\n");
    taResultado.setText(msj);
}
```

Implementamos metodo JTextArea

}

Modelo Carro{

```java
package modelo;

public class Carro
{
    //----------------------
    // Atributos
    //----------------------
    private double precio;

    //----------------------
    // Metodos
    //----------------------

    public Carro(double pPrecio)
    {
        this.precio = pPrecio;
    }

    public double getPrecio()
    {
        return precio;
    }
}
```

En el paquete modelo del trabajo encontramos la clase carro, donde encontramos el atributo **"double precio"** donde implementamos el método **bouble** para guardar los números o valores del carro y se utiliza el método **return** para devolver dicho término.

Modelo Empleado{

```java
//---------------------
// Constantes
//---------------------
public final static double SALARIO_MINIMO = 1000000;

//---------------------
// Atributos
//---------------------
private String nombre;
private ArrayList ventas;
private double sueldo;

//---------------------
// Metodos
//---------------------
public Empleado(String pNombre)
{
    this.nombre = pNombre;
    this.ventas = new ArrayList();
}

public void venderCarro(Carro carroVendido)
{
    ventas.add(carroVendido);
}
```

Otra de las clases que utilizamos es Empleado, donde encontramos un valor constante que es el SALARIO_MINIMO, y se crea una ArrayList llamada ventas donde se pueda guardar la información que se suministre.

Modelo Empleado{

```java
public void calcularSueldo()
{
    if(ventas.isEmpty())
    {
        sueldo = SALARIO_MINIMO;
    }
    else
    {
        //Numero carros vendidos
        int numCarrosVendidos = ventas.size();

        sueldo = SALARIO_MINIMO + 100000*numCarrosVendidos;

        double totalVentas = 0;
        for(int i=0; i<numCarrosVendidos;i++)
        {
            Carro carro = (Carro)ventas.get(i);
            totalVentas = totalVentas + carro.getPrecio();
        }

        sueldo = sueldo + 0.02*totalVentas;
    }
}
```

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

public double getSueldo()
{
    return sueldo;
}
```

Tambien econtramos la forma en la que vamos a calcular el sueldo del empleado y el numero de los carros vendidos agregando el 2% de comisión por cada venta, y encontramos el método return para devolver dicho término.

Modelo Empresa Automotriz{

```java
public EmpresaAutomotriz()
{
    //empleados = new Empleado[NUMERO_EMPLEADOS];
    empleados = new ArrayList();
}

public void agregarEmpleado(Empleado emp)
{
    empleados.add(emp);
}


public double calcularNomina()
{
    double totalnomina = 0;
    for(int i=0; i<empleados.size();i++)
    {
        Empleado temp = (Empleado) empleados.get(i);
        totalnomina = totalnomina + temp.getSueldo();
    }
    return totalnomina;
}
```

```java
public Empleado getEmpleado(int i)
{
    return (Empleado) empleados.get(i);
}

public int getNumeroEmpleados()
{
    return empleados.size();
}
```

Otra de las clases que encontramos es Empresa Automotriz donde por medio de el array list de la clase empleado, donde podemos seleccionar el numero de empleados que necesitamos, podemos calcular la momina.

Controlador{

```java
//Constructor
public Controlador(VentanaPrincipal pVenPrin, EmpresaAutomotriz pEmpresa)
{
    this.venPrin = pVenPrin;
    this.empresa = pEmpresa;
    this.venPrin.miPanelOperaciones.agregarOyentesBotones(this);
    this.venPrin.miPanelResultados.mostrarResultado(msj: "App lista para usar... \nSe han creado los
    siguientes tipos de objetos: \nVentanaPrincipal\nEmpresaAutomotriz\nControlador");
}
```

```java
@Override
public void actionPerformed(ActionEvent ae)
{
    String comando = ae.getActionCommand();

    //Abrir ventana Agregar vendedor
    if(comando.equals(anObject: "agregarVendedor"))
    {
        venPrin.crearDialogoAgregarVendedor();
        this.venPrin.miDialogoAgregarVendedor.agregarOyenteBoton(this);
    }

    //Agregar vendedor
    if(comando.equals(anObject: "agregar"))
    {
        String nombre = venPrin.miDialogoAgregarVendedor.getNombreVendedor();
        empresa.agregarEmpleado(new Empleado(nombre));
        venPrin.miPanelEntradaDatos.setEmpleado(nombre);
        venPrin.miPanelResultados.mostrarResultado("Se ha agreado un nuevo empleado. \nNombre: " + nombre);
        venPrin.miDialogoAgregarVendedor.cerrarDialogoAgregarVendedor();
    }
```

En el controlador ya empezamos con los comandos para que el programa corra, donde invocamos cada uno de los metodos que necesitamos para Agregar un vendedor con el metodo **getActionCommand.**

Controlador{

```java
//Abrir ventana Vender un carro
if(comando.equals(anObject: "venderCarro"))
{
    venPrin.crearDialogoVenderCarro();
    String Nombre = (String)this.venPrin.miPanelEntradaDatos.cbEmpleados.getSelectedItem();
    venPrin.miDialogoVenderCarro.mostrarResultado(Nombre);
    this.venPrin.miDialogoVenderCarro.agregarOyenteBoton(this);
}

//Vender carro
if(comando.equals(anObject: "vender"))
{
    int indexVendedor = venPrin.miPanelEntradaDatos.getIndexEmpleado();

    double precio = Double.parseDouble(venPrin.miDialogoVenderCarro.getPrecioCarro());

    Empleado emp = empresa.getEmpleado(indexVendedor);
    emp.venderCarro(new Carro(precio));

    venPrin.miPanelResultados.mostrarResultado("El empleado: " + emp.getNombre() + " ha vendido un
    carro\nValor: " + precio);
    venPrin.miDialogoVenderCarro.cerrarDialogoVenderCarro();
}
```

Encontramos el comando para abrir la ventana para visualizar el panel de Vender el carro, utilizando el comando equals, donde ingresamos el crearDialogoVenderCarro

Encontramos los atributos venPrin donde agregamos cada panel para poder empezar a trabajar

Controlador{

```
//Calcular el sueldo de un vendedor
if(comando.equals(anObject: "calcularSueldo"))
{
    int indexVendedor = venPrin.miPanelEntradaDatos.getIndexEmpleado();
    Empleado emple = empresa.getEmpleado(indexVendedor);
    emple.calcularSueldo();
    venPrin.miPanelResultados.mostrarResultado("El sueldo del empleado: " + emple.getNombre() + " es " +
    emple.getSueldo());
}

//Liquidar nomina total empresa
if(comando.equals(anObject: "liquidarNomina"))
{
    double valorNomina = empresa.calcularNomina();
    String listaEmpleados = "";
    for(int i=0; i<empresa.getNumeroEmpleados(); i++)
    {
        listaEmpleados = listaEmpleados + empresa.getEmpleado(i).getNombre() + ": " + empresa.getEmpleado
        (i).getSueldo() + "\n";
    }
    venPrin.miPanelResultados.mostrarResultado("El valor total de la nómina es." + valorNomina + "\n" +
    listaEmpleados);
}
```

Para calcular el sueldo del empleado abrimos el comando de mi ventana para calcular el suelddo y liquidar la nomina

Ejecutable (Test) {

En el ejecutable encontramos la clase Test, donde invocamos o importamos todas las clases que necesitamos para que poder observar el programa o el funcionamiento.

```java
package ejecutable;

import controlador.Controlador;
import modelo.Carro;
import modelo.Empleado;
import modelo.EmpresaAutomotriz;
import vista.VentanaPrincipal;

public class Test
{
    Run | Debug
    public static void main(String[] args)
    {

        VentanaPrincipal miVentana = new VentanaPrincipal();
        EmpresaAutomotriz miEmpresa = new EmpresaAutomotriz();
        Controlador miControlador = new Controlador(miVentana, miEmpresa);
    }
}
```