**Programación orientada a objetos III**

Ingeniería en Desarrollo de Software

Unidad: 2. Hilos

Evidencia de aprendizaje. Programa con hilos

Francisco Manuel López García

Matrícula: ES1921010154

Profesor: Manuel Niño Navarrete

20 de noviembre 2021

Índice

[**Programa multiflujo** 3](#_Toc88342198)

[**Introducción:** 3](#_Toc88342199)

[**Desarrollo:** 3](#_Toc88342200)

[*Investigación:* 3](#_Toc88342201)

[*Capturas de pantalla:* 4](#_Toc88342202)

[**Conclusión:** 19](#_Toc88342203)

[**Bibliografía:** 19](#_Toc88342204)

# **Programa con hilos**

# **Introducción:**

El objetivo de la actividad es poner en práctica la manipulación de hilos a través del diseño y la programación de un proyecto informático en Java NetBeans que dé solución a la problemática planteada.  
Una Base de Datos es una serie de tablas que contienen información ordenada en alguna estructura que facilita el acceso a esas tablas, ordenarlas y seleccionar filas de las tablas según criterios específicos. Las bases de datos generalmente tienen índices asociados a alguna de sus columnas, de forma que el acceso sea lo más rápido posible.

Las Bases de Datos son, sin lugar a dudas, las estructuras más utilizadas en ordenadores; ya que son el corazón de sistemas tan complejos como el censo de una nación, la nómina de empleados de una empresa, el sistema de facturación de una multinacional, o el medio por el que nos expiden el billete para las próximas vacaciones.

# **Desarrollo:**

## *Investigación:*

**Bases de datos en java**

JDBC (Java DataBase Connectivity) es la tecnología Java que permite a las aplicaciones

interactuar directamente con motores de base de datos relacionales.

La API JDBC es una parte integral de la plataforma Java, por lo tanto no es necesario descargar ningún paquete adicional para usarla.

JDBC provee una interfase única, que independiza a las aplicaciones del motor de la base de datos.

Un driver JDBC es usado por la JVM para traducir las invocaciones JDBC en invocaciones que la base de datos entiende.

**Conexión a la Base de Datos**

La conexión se establece a través del driver, que se carga en ejecución mediante el método:

• Class.forName(String nombredelDriver)

Una vez cargado el driver, la conexión a la BD se realiza invocando a alguno de los siguientes métodos de la clase java.sql.DriverManager :

public static Connection getConnection(String url)

public static Connection getConnection(String url,java.util.Properties info)

public static Connection getConnection(String url,String usr, String pwd)  
  
Para realizar una consulta SQL a la BD se requiere de la creación de un objeto: Statement,

PreparedStatement o CallableStatement usando uno de los métodos de Connection.

Statement createStatement()

Statement createStatement(int resultSetType, int resultSetConcurrency)

PreparedStatement preparedStatement(String sql)

PreparedStatement preparedStatement(String sql, int resultSetType, int resultSetConcurrency)

CallableStatement prepareCall(String sql)

CallableStatement prepareCall(String sql, int resultSetType, int resultSetConcurrency)

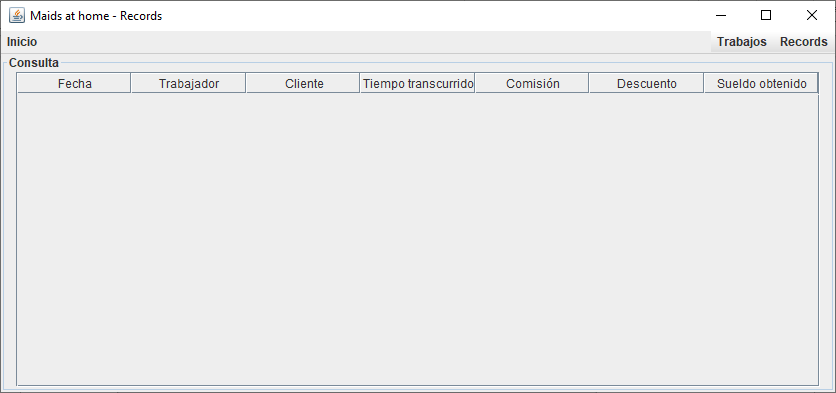
Es importante cerrar/liberar las conexiones una vez que ya no son usadas. De esta manera el objeto Connection será marcado para ser recolectado por el Garbage Collector y, además teniendo en cuenta que la cantidad de conexiones disponibles a una BD es limitada, es importante cerrar las conexiones que no son más usadas.

• El método destroy() es el apropiado para llevar a cabo esta acción.

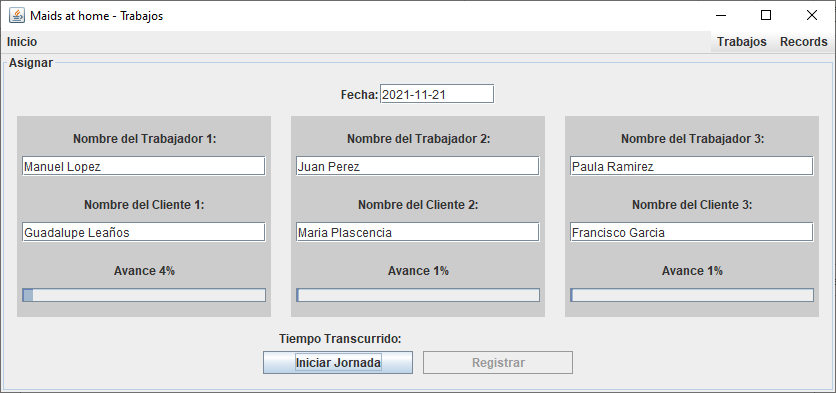
## *Capturas de pantalla:*

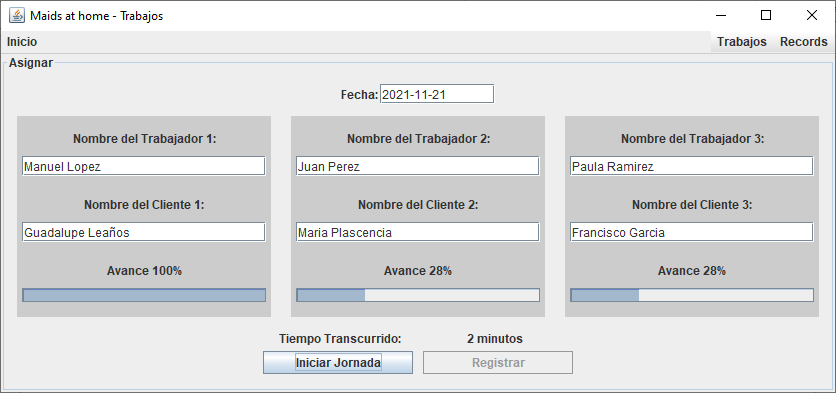
Inicio de la aplicación

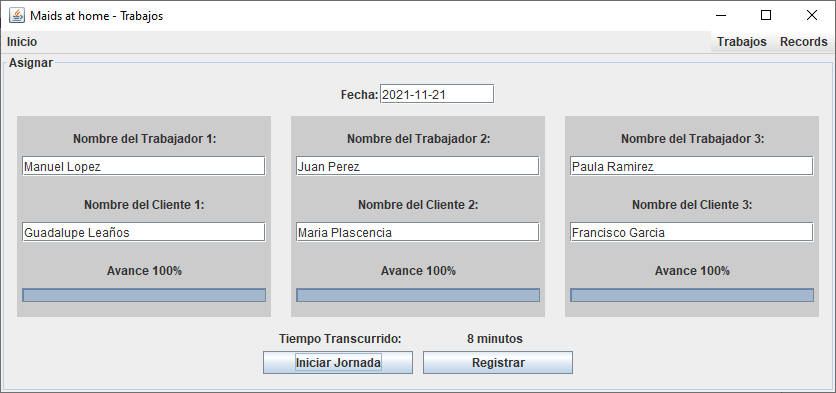


Reporte de Records vacio  


Llenado del apartado asignar trabajos  

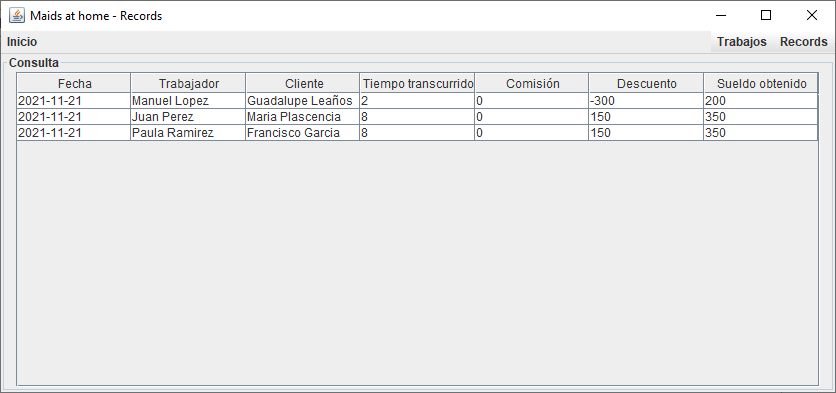

Inicio de los tres hilos simultaneos  




Finalización del avance  


Se registran en los récords  


Se muestran los récords



*Código fuente:*

***Clase principal Main***

*package maids;*

*public class MAIDS {*

*public static void main(String[] args) {*

*framePrincipal fp = new framePrincipal();*

*fp.aparienciaTotal();*

*}*

*}*

***Clase framePrincipal***

*package maids;*

*import java.awt.BorderLayout;*

*import java.awt.event.ActionEvent;*

*import java.awt.event.WindowAdapter;*

*import java.awt.event.WindowEvent;*

*import javax.swing.\*;*

*public class framePrincipal {*

*static JFrame frame = new JFrame("Maids at home");* ***//Creación del frame principal***

*static Trabajos t= new Trabajos(); /****/Creación del objeto de tipo Trabajos***

*static Records r= new Records();* ***//Creación del objeto de tipo Records***

*JPanel pnlInicio = new javax.swing.JPanel();* ***//Se crea el panel de Inicio***

***/\****

***Funcion para limpiar el frame***

***\*/***

*public void limpiarFrame(){*

*frame.remove(t.pnlAsignar);*

*frame.remove(r.pnlConsulta);*

*frame.remove(pnlInicio);*

*r.modelo.setNumRows(0);*

*}*

***/\****

***Funcion para crear todos los componentes de la apariencia***

***\*/***

*public void aparienciaTotal(){*

***//Se pone titulo al Panel***

*pnlInicio.setBorder(javax.swing.BorderFactory.createTitledBorder("Inicio"));*

*t.aparienciaAsignar();*

*r.aparienciaConsulta();*

***//Se crea una etiqueta para mostrar el texto Bienvenido***

*JLabel ini = new javax.swing.JLabel();*

*ini.setText("Bienvenidos a Maids at home");*

***//Se añade al panel***

*pnlInicio.add(ini);*

***//Se ingresan los atributos del frame***

*frame.setLocation(250,200);*

*frame.setVisible(true);*

*frame.setSize(850,400);*

***//Se crea un evento para salir de la aplicación al pulsar la equis***

*frame.addWindowListener(new WindowAdapter() {*

*@Override*

*public void windowClosing(WindowEvent e){*

*System.exit(0);*

*}*

*});*

***//Se crea un menu***

*JMenuBar menu = new javax.swing.JMenuBar();*

***//Se crea el item inicio***

*JMenuItem inicio = new javax.swing.JMenuItem("Inicio");*

*inicio.addActionListener((ActionEvent e) -> {*

*if(e.getSource()==inicio){*

*frame.setTitle("Maids at home");*

*limpiarFrame();*

*frame.add(pnlInicio,BorderLayout.CENTER);*

*frame.repaint();*

*frame.validate();*

*}*

*});*

***//Se crea el item trabajo***

*JMenu trabajos = new javax.swing.JMenu("Trabajos");*

*JMenuItem asignar = new javax.swing.JMenuItem("Asignar");*

*asignar.addActionListener((ActionEvent e) -> {*

*if(e.getSource()==asignar){*

*frame.setTitle("Maids at home - Trabajos");*

*limpiarFrame();*

*frame.add(t.pnlAsignar,BorderLayout.CENTER);*

*frame.repaint();*

*frame.validate();*

*}*

*});*

*JMenu records = new javax.swing.JMenu("Records");*

*JMenuItem reportes = new javax.swing.JMenuItem("Reportes");*

*reportes.addActionListener((ActionEvent e) -> {*

*if(e.getSource()==reportes){*

*frame.setTitle("Maids at home - Records");*

*limpiarFrame();*

*r.BuscarConsulta();*

*frame.add(r.pnlConsulta,BorderLayout.CENTER);*

*frame.repaint();*

*frame.validate();*

*}*

*});*

***//Se añaden los items al menu***

*menu.add(inicio);*

*menu.add(trabajos);*

*trabajos.add(asignar);*

*menu.add(records);*

*records.add(reportes);*

*frame.add(menu, BorderLayout.PAGE\_START);*

*frame.add(pnlInicio,BorderLayout.CENTER);*

*frame.repaint();*

*frame.validate();*

*}*

*}*

***Clase Conexión***

*package maids;*

***/\*\****

***\* Clase encargada de las funciones SQL***

***\*/***

*import java.sql.Connection;*

*import java.sql.DriverManager;*

*import java.sql.SQLException;*

*import java.sql.Statement;*

*import javax.swing.JOptionPane;*

*public class Conexion {*

***//Funcion encargada de la inserción en la tabla comisiones***

*public void insertarRegistro(String fecha,String trabajador, String cliente, int tiempo, int comision,int descuento,int sueldo){*

*try{*

*Class.forName("com.mysql.cj.jdbc.Driver");*

*Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/bd\_ea","manne","Fco-1402");*

*Statement st = conexion.createStatement();*

*String insercion ="INSERT INTO tbl\_comisiones VALUES (0,'"+fecha+"','"+trabajador+"','"+cliente+"',"+tiempo+","+comision+","+descuento+","+sueldo+");";*

*st.executeUpdate(insercion);*

*st.close();*

*conexion.close();*

*}*

*catch(ClassNotFoundException | SQLException e){*

*JOptionPane.showMessageDialog(null,"Error al conectar con la Base de Datos","Error", JOptionPane.ERROR\_MESSAGE);*

*}*

*}*

*}*

***Clase Trabajos***

*package maids;*

*import java.awt.GridBagConstraints;*

*import java.awt.GridBagLayout;*

*import java.awt.event.MouseAdapter;*

*import java.awt.event.MouseEvent;*

*import java.util.logging.Level;*

*import java.util.logging.Logger;*

*import javax.swing.JButton;*

*import javax.swing.JLabel;*

*import javax.swing.JOptionPane;*

*import javax.swing.JPanel;*

*import javax.swing.JProgressBar;*

*import javax.swing.JTextField;*

*public class Trabajos {*

*Thread trabajador1 = new Thread();*

*Thread trabajador2 = new Thread();*

*Thread trabajador3 = new Thread();*

*int sueldo, comision, descuento, tiempo, random1,random2,random3;*

***//Se crea objeto de la clase Conexion***

*Conexion BD = new Conexion();*

***//Componentes graficos Trabajos***

*JPanel pnlAsignar = new javax.swing.JPanel();*

*JPanel pnlFecha = new javax.swing.JPanel();*

*JPanel pnlTrabajador1 = new javax.swing.JPanel();*

*JPanel pnlTrabajador2 = new javax.swing.JPanel();*

*JPanel pnlTrabajador3 = new javax.swing.JPanel();*

*JPanel pnlBotones = new javax.swing.JPanel();*

*JLabel lblFecha= new javax.swing.JLabel();*

*JTextField txtFecha = new javax.swing.JTextField();*

*JLabel lblTrabajador1 = new javax.swing.JLabel();*

*JTextField txtTrabajador1 = new javax.swing.JTextField();*

*JLabel lblCliente1 = new javax.swing.JLabel();*

*JTextField txtCliente1= new javax.swing.JTextField();*

*JLabel lblAvance1 = new javax.swing.JLabel();*

*JProgressBar pbTrabajador1 = new javax.swing.JProgressBar();*

*JLabel lblTrabajador2 = new javax.swing.JLabel();*

*JTextField txtTrabajador2 = new javax.swing.JTextField();*

*JLabel lblCliente2 = new javax.swing.JLabel();*

*JTextField txtCliente2= new javax.swing.JTextField();*

*JLabel lblAvance2 = new javax.swing.JLabel();*

*JProgressBar pbTrabajador2 = new javax.swing.JProgressBar();*

*JLabel lblTrabajador3 = new javax.swing.JLabel();*

*JTextField txtTrabajador3 = new javax.swing.JTextField();*

*JLabel lblCliente3 = new javax.swing.JLabel();*

*JTextField txtCliente3= new javax.swing.JTextField();*

*JLabel lblAvance3 = new javax.swing.JLabel();*

*JProgressBar pbTrabajador3 = new javax.swing.JProgressBar();*

*JLabel lblTiempo = new javax.swing.JLabel();*

*JLabel lblTiempoTrans = new javax.swing.JLabel();*

*JButton btnIniciar = new javax.swing.JButton();*

*JButton btnRegistrar = new javax.swing.JButton();*

*public void aparienciaAsignar(){*

*pnlAsignar.setBorder(javax.swing.BorderFactory.createTitledBorder("Asignar"));*

*lblFecha.setText("Fecha:");*

*lblTiempoTrans.setText("Tiempo Transcurrido:");*

*lblTiempo.setText("");*

*lblTrabajador1.setText("Nombre del Trabajador 1:");*

*lblCliente1.setText("Nombre del Cliente 1:");*

*lblAvance1.setText("Avance 0%");*

*lblTrabajador2.setText("Nombre del Trabajador 2:");*

*lblCliente2.setText("Nombre del Cliente 2:");*

*lblAvance2.setText("Avance 0%");*

*lblTrabajador3.setText("Nombre del Trabajador 3:");*

*lblCliente3.setText("Nombre del Cliente 3:");*

*lblAvance3.setText("Avance 0%");*

*btnIniciar.setText("Iniciar Jornada");*

*btnIniciar.addMouseListener(new MouseAdapter(){*

*@Override*

*public void mouseClicked(MouseEvent e){*

*try {*

*iniciar();*

*} catch (InterruptedException ex) {*

*Logger.getLogger(Trabajos.class.getName()).log(Level.SEVERE, null, ex);*

*}*

*}*

*});*

*btnRegistrar.setText("Registrar");*

*btnRegistrar.setEnabled(false);*

*btnRegistrar.addMouseListener(new MouseAdapter(){*

*@Override*

*public void mouseClicked(MouseEvent e){*

*registrar();*

*}*

*});*

*pnlTrabajador1.setBackground(new java.awt.Color(204, 204, 204));*

*pnlTrabajador2.setBackground(new java.awt.Color(204, 204, 204));*

*pnlTrabajador3.setBackground(new java.awt.Color(204, 204, 204));*

*GridBagLayout gbl = new GridBagLayout();*

*GridBagConstraints gridBagConstraints = new GridBagConstraints();*

*pnlAsignar.setLayout(gbl);*

*pnlFecha.setPreferredSize(new java.awt.Dimension(100, 24));*

*javax.swing.GroupLayout pnlFechaLayout = new javax.swing.GroupLayout(pnlFecha);*

*pnlFecha.setLayout(pnlFechaLayout);*

*pnlFechaLayout.setHorizontalGroup(*

*pnlFechaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGap(0, 0, Short.MAX\_VALUE)*

*);*

*pnlFechaLayout.setVerticalGroup(*

*pnlFechaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)*

*.addGap(0, 0, Short.MAX\_VALUE)*

*);*

*java.awt.FlowLayout flowLayout1 = new java.awt.FlowLayout(java.awt.FlowLayout.CENTER, 2, 2);*

*pnlFecha.setLayout(flowLayout1);*

*pnlFecha.add(lblFecha);*

*txtFecha.setPreferredSize(new java.awt.Dimension(115, 20));*

*pnlFecha.add(txtFecha);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 0;*

*gridBagConstraints.gridwidth = 3;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(10, 10, 0, 10);*

*pnlAsignar.add(pnlFecha, gridBagConstraints);*

*pnlTrabajador1.setPreferredSize(new java.awt.Dimension(100, 100));*

*pnlTrabajador1.setLayout(new java.awt.GridBagLayout());*

*lblTrabajador1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*lblTrabajador1.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 0;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador1.add(lblTrabajador1, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 1;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 15, 5);*

*pnlTrabajador1.add(txtTrabajador1, gridBagConstraints);*

*lblCliente1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 2;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador1.add(lblCliente1, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 3;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 15, 5);*

*pnlTrabajador1.add(txtCliente1, gridBagConstraints);*

*lblAvance1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 4;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador1.add(lblAvance1, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 5;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador1.add(pbTrabajador1, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 1;*

*gridBagConstraints.gridheight = 4;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.weighty = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(10, 10, 10, 10);*

*pnlAsignar.add(pnlTrabajador1, gridBagConstraints);*

*pnlTrabajador2.setPreferredSize(new java.awt.Dimension(100, 100));*

*pnlTrabajador2.setLayout(new java.awt.GridBagLayout());*

*lblTrabajador2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*lblTrabajador2.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 0;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador2.add(lblTrabajador2, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 1;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 15, 5);*

*pnlTrabajador2.add(txtTrabajador2, gridBagConstraints);*

*lblCliente2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 2;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador2.add(lblCliente2, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 3;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 15, 5);*

*pnlTrabajador2.add(txtCliente2, gridBagConstraints);*

*lblAvance2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 4;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador2.add(lblAvance2, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 5;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador2.add(pbTrabajador2, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 1;*

*gridBagConstraints.gridy = 1;*

*gridBagConstraints.gridheight = 4;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.weighty = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(10, 10, 10, 10);*

*pnlAsignar.add(pnlTrabajador2, gridBagConstraints);*

*pnlTrabajador3.setPreferredSize(new java.awt.Dimension(100, 100));*

*pnlTrabajador3.setLayout(new java.awt.GridBagLayout());*

*lblTrabajador3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*lblTrabajador3.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 0;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador3.add(lblTrabajador3, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 1;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 15, 5);*

*pnlTrabajador3.add(txtTrabajador3, gridBagConstraints);*

*lblCliente3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 2;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador3.add(lblCliente3, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 3;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 15, 5);*

*pnlTrabajador3.add(txtCliente3, gridBagConstraints);*

*lblAvance3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 4;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador3.add(lblAvance3, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 5;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 5, 5);*

*pnlTrabajador3.add(pbTrabajador3, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 2;*

*gridBagConstraints.gridy = 1;*

*gridBagConstraints.gridheight = 4;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;*

*gridBagConstraints.weightx = 1.0;*

*gridBagConstraints.weighty = 1.0;*

*gridBagConstraints.insets = new java.awt.Insets(10, 10, 10, 10);*

*pnlAsignar.add(pnlTrabajador3, gridBagConstraints);*

*pnlBotones.setMinimumSize(new java.awt.Dimension(100, 100));*

*pnlBotones.setPreferredSize(new java.awt.Dimension(100, 50));*

*pnlBotones.setLayout(new java.awt.GridBagLayout());*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 2;*

*gridBagConstraints.gridy = 0;*

*pnlBotones.add(lblTiempo, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 1;*

*gridBagConstraints.gridy = 0;*

*pnlBotones.add(lblTiempoTrans, gridBagConstraints);*

*btnIniciar.setPreferredSize(new java.awt.Dimension(150, 23));*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 1;*

*gridBagConstraints.gridy = 1;*

*gridBagConstraints.insets = new java.awt.Insets(5, 0, 0, 5);*

*pnlBotones.add(btnIniciar, gridBagConstraints);*

*btnRegistrar.setPreferredSize(new java.awt.Dimension(150, 23));*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 2;*

*gridBagConstraints.gridy = 1;*

*gridBagConstraints.insets = new java.awt.Insets(5, 5, 0, 0);*

*pnlBotones.add(btnRegistrar, gridBagConstraints);*

*gridBagConstraints = new java.awt.GridBagConstraints();*

*gridBagConstraints.gridx = 0;*

*gridBagConstraints.gridy = 5;*

*gridBagConstraints.gridwidth = 3;*

*gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;*

*gridBagConstraints.insets = new java.awt.Insets(0, 10, 10, 10);*

*pnlAsignar.add(pnlBotones, gridBagConstraints);*

*}*

*public void iniciar() throws InterruptedException{*

*trabajador1 = new Thread(){*

*@Override*

*public void run(){*

*try {*

*long velocidad = ((random1+5)\*60000)/100;*

*for(int seg=1; seg<=100;seg++){*

*lblAvance1.setText("Avance "+seg+"%");*

*pbTrabajador1.setValue(seg);*

*trabajador1.sleep(velocidad);*

*if(random1>=random2 && random1 >=random3)*

*if(((velocidad\*seg)/60000)>=1&&((velocidad\*seg)/60000)<2)*

*lblTiempo.setText(((velocidad\*seg)/60000)+" minuto");*

*else*

*lblTiempo.setText(((velocidad\*seg)/60000)+" minutos");*

*}*

*if(random1>=random2 && random1 >=random3)*

*btnRegistrar.setEnabled(true);*

*}catch (Exception e){*

*}*

*}*

*};*

*trabajador2 = new Thread(){*

*@Override*

*public void run(){*

*try {*

*long velocidad = ((random2+5)\*60000)/100;*

*for(int seg=1; seg<=100;seg++){*

*lblAvance2.setText("Avance "+seg+"%");*

*pbTrabajador2.setValue(seg);*

*trabajador2.sleep(velocidad);*

*if(random2>=random1 && random2 >=random3)*

*if(((velocidad\*seg)/60000)>=1&&((velocidad\*seg)/60000)<2)*

*lblTiempo.setText(((velocidad\*seg)/60000)+" minuto");*

*else*

*lblTiempo.setText(((velocidad\*seg)/60000)+" minutos");*

*}*

*if(random2>=random1 && random2 >=random3)*

*btnRegistrar.setEnabled(true);*

*}catch (Exception e){*

*}*

*}*

*};*

*trabajador3 = new Thread(){*

*@Override*

*public void run(){*

*try {*

*long velocidad = ((random3+5)\*60000)/100;*

*for(int seg=1; seg<=100;seg++){*

*lblAvance3.setText("Avance "+seg+"%");*

*pbTrabajador3.setValue(seg);*

*trabajador3.sleep(velocidad);*

*if(random3>=random2 && random3 >=random1)*

*if(((velocidad\*seg)/60000)>=1&&((velocidad\*seg)/60000)<2)*

*lblTiempo.setText(((velocidad\*seg)/60000)+" minuto");*

*else*

*lblTiempo.setText(((velocidad\*seg)/60000)+" minutos");*

*}*

*if(random3>=random2 && random3 >=random1)*

*btnRegistrar.setEnabled(true);*

*}catch (Exception e){*

*}*

*}*

*};*

*if("".equals(txtFecha.getText()) || "".equals(txtTrabajador1.getText()) || "".equals(txtCliente1.getText()) || "".equals(txtTrabajador2.getText()) || "".equals(txtCliente2.getText()) || "".equals(txtTrabajador3.getText()) || "".equals(txtCliente3.getText()) ){*

*JOptionPane.showMessageDialog(null,"Algunos campos se encuentran vacios","Error", JOptionPane.ERROR\_MESSAGE);*

*}else{*

*random1=(int) (Math.floor(Math.random()\*(3-(-3)+1)+(-3)));*

*trabajador1.start();*

*random2=(int) (Math.floor(Math.random()\*(3-(-3)+1)+(-3)));*

*trabajador2.start();*

*random3=(int) (Math.floor(Math.random()\*(3-(-3)+1)+(-3)));*

*trabajador3.start();*

*}*

*}*

*public int[] calcularSueldo(int aleatorio){*

*if (aleatorio == 0){*

*comision=200;*

*tiempo = 5+aleatorio;*

*descuento=0;*

*sueldo = 500+comision-descuento;*

*}else{*

*if(aleatorio >0){*

*comision=0;*

*tiempo = 5+aleatorio;*

*descuento=aleatorio\*50;*

*sueldo = 500+comision-descuento;*

*}else{*

*if(aleatorio <0){*

*comision=0;*

*tiempo = 5+aleatorio;*

*descuento=aleatorio\*100;*

*sueldo = 500+comision+descuento;*

*}*

*}*

*}*

*int sueldos [] = new int []{comision,tiempo,descuento,sueldo};*

*return sueldos;*

*}*

*public void registrar(){*

*int sueldos [];*

*if(trabajador1.isAlive()|| trabajador2.isAlive() || trabajador3.isAlive()){*

*JOptionPane.showMessageDialog(null,"Aun no terminan los trabajadores","Error", JOptionPane.ERROR\_MESSAGE);*

*}else{*

*sueldos=calcularSueldo(random1);*

*BD.insertarRegistro(txtFecha.getText(), txtTrabajador1.getText(), txtCliente1.getText(), sueldos[1], sueldos[0], sueldos[2], sueldos[3]);*

*sueldos=calcularSueldo(random2);*

*BD.insertarRegistro(txtFecha.getText(), txtTrabajador2.getText(), txtCliente2.getText(), sueldos[1], sueldos[0], sueldos[2], sueldos[3]);*

*sueldos=calcularSueldo(random3);*

*BD.insertarRegistro(txtFecha.getText(), txtTrabajador3.getText(), txtCliente3.getText(), sueldos[1], sueldos[0], sueldos[2], sueldos[3]);*

*JOptionPane.showMessageDialog(null,"Registro Completado","Registro", JOptionPane.PLAIN\_MESSAGE);*

*limpiar();*

*}*

*}*

*public void limpiar(){*

*pbTrabajador1.setValue(0);*

*pbTrabajador2.setValue(0);*

*pbTrabajador3.setValue(0);*

*lblAvance1.setText("Avance 0%");*

*lblAvance2.setText("Avance 0%");*

*lblAvance3.setText("Avance 0%");*

*txtTrabajador1.setText("");*

*txtCliente1.setText("");*

*txtTrabajador2.setText("");*

*txtCliente2.setText("");*

*txtTrabajador3.setText("");*

*txtCliente3.setText("");*

*txtFecha.setText("");*

*lblTiempo.setText("");*

*btnRegistrar.setEnabled(false);*

*}*

*}*

***Clase Records***

*package maids;*

*import java.awt.GridBagConstraints;*

*import java.awt.GridBagLayout;*

*import java.awt.Insets;*

*import java.sql.Connection;*

*import java.sql.DriverManager;*

*import java.sql.ResultSet;*

*import java.sql.SQLException;*

*import java.sql.Statement;*

*import javax.swing.JOptionPane;*

*import javax.swing.JPanel;*

*import javax.swing.JScrollPane;*

*import javax.swing.JTable;*

*import javax.swing.table.DefaultTableModel;*

*public class Records {*

*JPanel pnlConsulta = new javax.swing.JPanel();*

*DefaultTableModel modelo = new DefaultTableModel();*

*JScrollPane scroll = new JScrollPane();*

*JTable tabla = new JTable();*

*public void aparienciaConsulta(){*

*pnlConsulta.setBorder(javax.swing.BorderFactory.createTitledBorder("Consulta"));*

*modelo.addColumn("Fecha");*

*modelo.addColumn("Trabajador");*

*modelo.addColumn("Cliente");*

*modelo.addColumn("Tiempo transcurrido");*

*modelo.addColumn("Comisión");*

*modelo.addColumn("Descuento");*

*modelo.addColumn("Sueldo obtenido");*

*tabla.setModel(modelo);*

*tabla.setEnabled(false);*

*scroll.setViewportView(tabla);*

*GridBagLayout gbl = new GridBagLayout();*

*GridBagConstraints gbc = new GridBagConstraints();*

*pnlConsulta.setLayout(gbl);*

*gbc.weightx = 1;*

*gbc.weighty = 1;*

*gbc.fill = GridBagConstraints.BOTH;*

*gbc.anchor = GridBagConstraints.CENTER;*

*gbc.gridwidth = 0;*

*gbc.insets = new Insets(0,10,0,10);*

*pnlConsulta.add(scroll,gbc);*

*gbc.fill = GridBagConstraints.HORIZONTAL;*

*}*

*public void BuscarConsulta(){*

*try{*

*Class.forName("com.mysql.cj.jdbc.Driver");*

*Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/bd\_ea","manne","Fco-1402");*

*Statement st = conexion.createStatement();*

*String consulta = "SELECT \* FROM bd\_ea.tbl\_comisiones;";*

*ResultSet rs=null;*

*rs = st.executeQuery(consulta);*

*while(rs.next()){*

*Object[] data ={rs.getString(2),rs.getString(3),rs.getString(4),rs.getString(5),rs.getString(6),rs.getString(7),rs.getString(8)};*

*modelo.addRow(data);*

*}*

*rs.close();*

*st.close();*

*conexion.close();*

*}*

*catch(ClassNotFoundException | SQLException e){*

*JOptionPane.showMessageDialog(null,"Error al conectar con la Base de Datos","Error", JOptionPane.ERROR\_MESSAGE);*

*}*

*}*

*}*

# **Conclusión:**

Los Hilos o los “Threads” en Java, son básicamente una forma de poder ejecutar varios procesos simultáneamente en nuestros programas en Java.   
La programación multihilo sin duda tiene mucho campo de aplicación, desde los sistemas operativos hasta en la tecnología que usamos cotidianamente como los celulares, cajeros etc.

En la elaboración de este trabajo obtuvimos conceptos que nos darán la fluidez para desarrollar aplicaciones javas utilizando multihilo, con la finalidad de poder realizar más de una tarea a la vez y administrando correctamente los recursos del ordenador.

# **Bibliografía:**

* Froufe, A. (2009) Java 2 Manual de usuario y tutorial. (5ta Ed.). México: Alfaomega.
* García, J. (2000) Aprenda Java como si estuviera en primero. España: Tecnun.
* Joyanes, L. (2002) Java 2 Manual de programación. México: MC Graw Hill
* Martín, A. (2010) Programador certificado Java 2 curso práctico 3ra edición. México: Alfaomega.