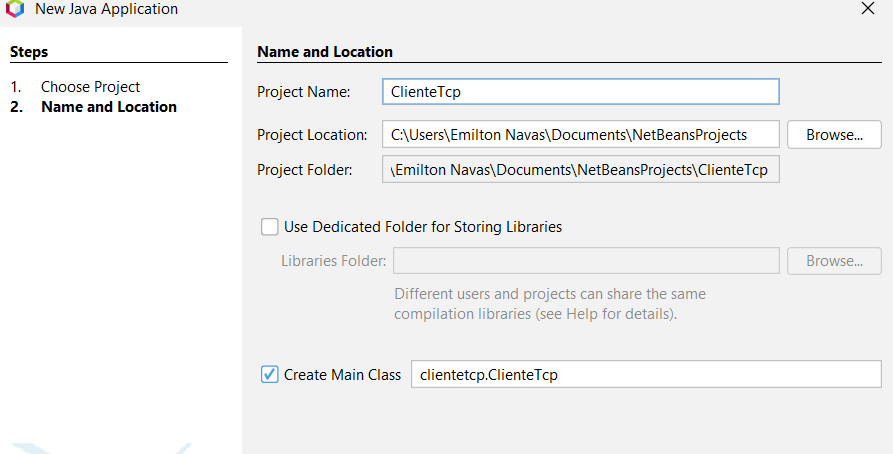
GUIA IMC TCP

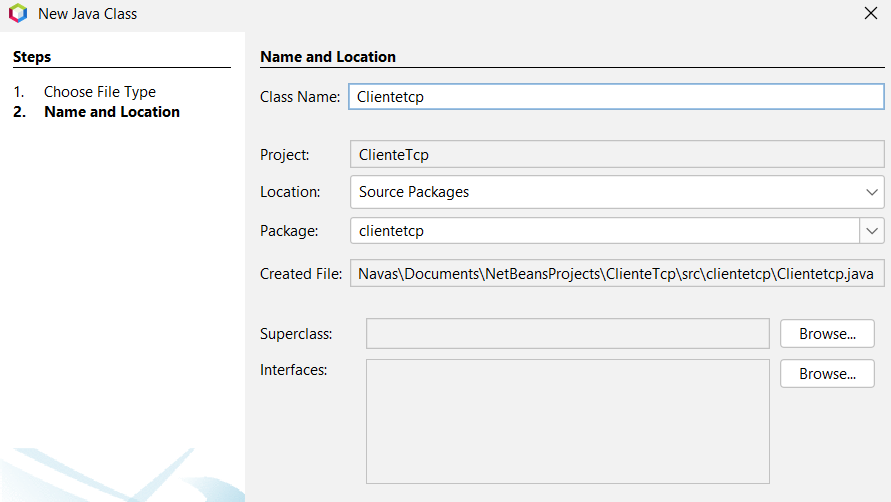
Emilton Navas

Cliente

Creamos el proyecto Cliente tcp



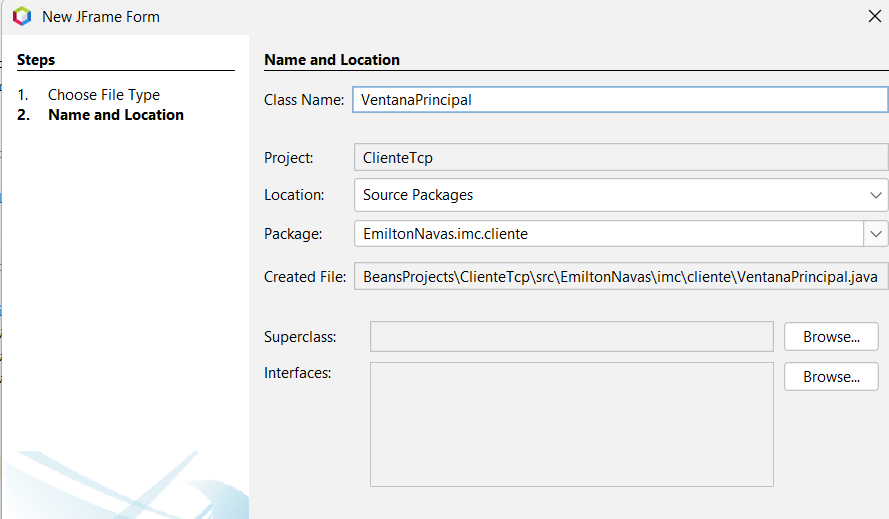
Se crea la clase



| package *clientetcp*;  import *EmiltonNavas.imc.cliente.VentanaPrincipal*;  */\*\**  *\**  *\* @author Emilton Navas*  *\*/*  *public* *class* ClienteTcp {  */\*\**  *\* @param args the command line arguments*  *\*/*  *public* *static* void main(String[] args) {  VentanaPrincipal v = new VentanaPrincipal();  v.setLocationRelativeTo(null);  v.setVisible(true);  }    } |
| --- |

Se crea la ventana principal

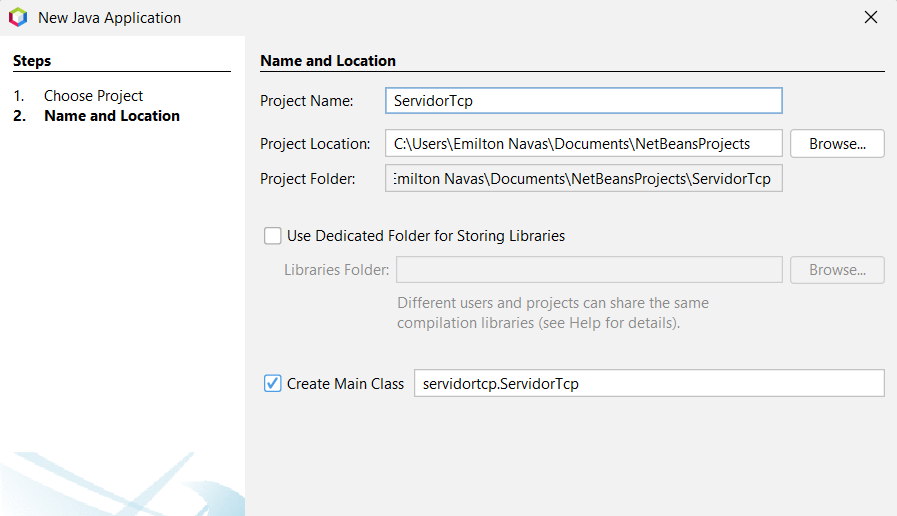




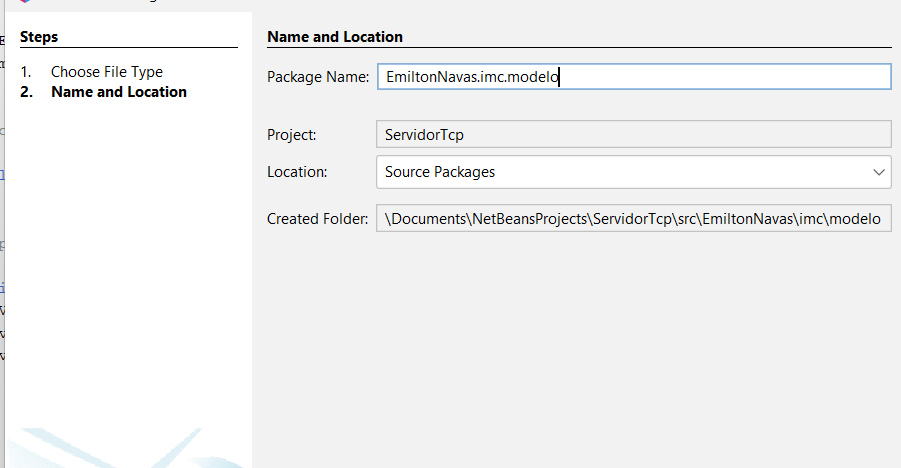
| /\*  \* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license  \* Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template  \*/ **package** EmiltonNavas.imc.cliente;  **import** java.awt.Color; **import** java.io.DataInputStream; **import** java.io.DataOutputStream; **import** java.io.IOException; **import** java.net.Socket; **import** javax.swing.JButton; **import** javax.swing.JLabel; **import** javax.swing.JOptionPane;  /\*\*  \*  \* **@author** Emilton Navas \*/ **public** **class** **VentanaPrincipal** **extends** **javax**.**swing**.**JFrame** {   /\*\*  \* Creates new form VentanaPrincipal  \*/  Socket servidor;  DataOutputStream out;  DataInputStream in;   **public** **VentanaPrincipal**() {  initComponents();  }   /\*\*  \* This method is called from within the constructor to initialize the form.  \* WARNING: Do NOT modify this code. The content of this method is always  \* regenerated by the Form Editor.  \*/  @SuppressWarnings("unchecked")  // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents  **private** **void** **initComponents**() {   jLabel1 = **new** javax.swing.JLabel();  jTabbedPane1 = **new** javax.swing.JTabbedPane();  jPanel1 = **new** javax.swing.JPanel();  jLabel2 = **new** javax.swing.JLabel();  campoPuertoServidor = **new** javax.swing.JTextField();  jLabel3 = **new** javax.swing.JLabel();  campoIPServidor = **new** javax.swing.JTextField();  btnIniciar = **new** javax.swing.JButton();  jLabel4 = **new** javax.swing.JLabel();  txtEstado = **new** javax.swing.JLabel();  jPanel2 = **new** javax.swing.JPanel();  jLabel5 = **new** javax.swing.JLabel();  campoPeso = **new** javax.swing.JTextField();  jLabel6 = **new** javax.swing.JLabel();  campoAltura = **new** javax.swing.JTextField();  jLabel7 = **new** javax.swing.JLabel();  txtResultado = **new** javax.swing.JLabel();  txtMensaje = **new** javax.swing.JLabel();  btnIniciar1 = **new** javax.swing.JButton();   setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);   jLabel1.setFont(**new** java.awt.Font("Segoe Print", 1, 36)); // NOI18N  jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  jLabel1.setText("CLIENTE IMC");   jLabel2.setText("DIRECCION IP:");   campoPuertoServidor.setText("9007");   jLabel3.setText("PUERTO DE RED:");   campoIPServidor.setText("localhost");  campoIPServidor.addActionListener(**new** java.awt.event.ActionListener() {  **public** **void** **actionPerformed**(java.awt.event.ActionEvent evt) {  campoIPServidorActionPerformed(evt);  }  });   btnIniciar.setFont(**new** java.awt.Font("Tahoma", 1, 18)); // NOI18N  btnIniciar.setForeground(**new** java.awt.Color(0, 204, 0));  btnIniciar.setText("CONECTAR");  btnIniciar.addActionListener(**new** java.awt.event.ActionListener() {  **public** **void** **actionPerformed**(java.awt.event.ActionEvent evt) {  btnIniciarActionPerformed(evt);  }  });   jLabel4.setText("ESTADO:");   txtEstado.setText("DESCONECTADO");   javax.swing.GroupLayout jPanel1Layout = **new** javax.swing.GroupLayout(jPanel1);  jPanel1.setLayout(jPanel1Layout);  jPanel1Layout.setHorizontalGroup(  jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel1Layout.createSequentialGroup()  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel1Layout.createSequentialGroup()  .addGap(30, 30, 30)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel1Layout.createSequentialGroup()  .addComponent(jLabel4)  .addGap(18, 18, 18)  .addComponent(txtEstado))  .addGroup(jPanel1Layout.createSequentialGroup()  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addComponent(jLabel2)  .addComponent(jLabel3))  .addGap(35, 35, 35)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, **false**)  .addComponent(campoIPServidor)  .addComponent(campoPuertoServidor, javax.swing.GroupLayout.DEFAULT\_SIZE, 167, Short.MAX\_VALUE)))))  .addGroup(jPanel1Layout.createSequentialGroup()  .addGap(167, 167, 167)  .addComponent(btnIniciar, javax.swing.GroupLayout.PREFERRED\_SIZE, 197, javax.swing.GroupLayout.PREFERRED\_SIZE)))  .addContainerGap(167, Short.MAX\_VALUE))  );  jPanel1Layout.setVerticalGroup(  jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel1Layout.createSequentialGroup()  .addGap(23, 23, 23)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel2)  .addComponent(campoIPServidor, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGap(29, 29, 29)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel3)  .addComponent(campoPuertoServidor, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGap(28, 28, 28)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel4)  .addComponent(txtEstado))  .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 50, Short.MAX\_VALUE)  .addComponent(btnIniciar, javax.swing.GroupLayout.PREFERRED\_SIZE, 32, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addGap(29, 29, 29))  );   jTabbedPane1.addTab("CONEXION", jPanel1);   jLabel5.setText("PESO:");   jLabel6.setText("ALTURA:");   jLabel7.setText("IMC:");   txtResultado.setFont(**new** java.awt.Font("Tahoma", 1, 14)); // NOI18N  txtResultado.setForeground(**new** java.awt.Color(255, 0, 0));  txtResultado.setText("0.0");   txtMensaje.setBorder(javax.swing.BorderFactory.createEtchedBorder());   btnIniciar1.setFont(**new** java.awt.Font("Tahoma", 1, 18)); // NOI18N  btnIniciar1.setForeground(**new** java.awt.Color(0, 153, 0));  btnIniciar1.setText("CALCULAR");  btnIniciar1.addActionListener(**new** java.awt.event.ActionListener() {  **public** **void** **actionPerformed**(java.awt.event.ActionEvent evt) {  btnIniciar1ActionPerformed(evt);  }  });   javax.swing.GroupLayout jPanel2Layout = **new** javax.swing.GroupLayout(jPanel2);  jPanel2.setLayout(jPanel2Layout);  jPanel2Layout.setHorizontalGroup(  jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel2Layout.createSequentialGroup()  .addGap(34, 34, 34)  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel2Layout.createSequentialGroup()  .addComponent(jLabel7, javax.swing.GroupLayout.PREFERRED\_SIZE, 37, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addGap(18, 18, 18)  .addComponent(txtResultado, javax.swing.GroupLayout.PREFERRED\_SIZE, 37, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 40, Short.MAX\_VALUE)  .addComponent(txtMensaje, javax.swing.GroupLayout.PREFERRED\_SIZE, 359, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addContainerGap())  .addGroup(jPanel2Layout.createSequentialGroup()  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addComponent(jLabel6, javax.swing.GroupLayout.PREFERRED\_SIZE, 58, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGap(27, 27, 27)  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addComponent(campoAltura, javax.swing.GroupLayout.PREFERRED\_SIZE, 169, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addComponent(campoPeso, javax.swing.GroupLayout.PREFERRED\_SIZE, 169, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  .addComponent(btnIniciar1, javax.swing.GroupLayout.PREFERRED\_SIZE, 161, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addGap(34, 34, 34))))  );  jPanel2Layout.setVerticalGroup(  jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel2Layout.createSequentialGroup()  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel2Layout.createSequentialGroup()  .addGap(36, 36, 36)  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel5)  .addComponent(campoPeso, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGap(33, 33, 33)  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel6)  .addComponent(campoAltura, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGap(44, 44, 44))  .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()  .addContainerGap()  .addComponent(btnIniciar1, javax.swing.GroupLayout.PREFERRED\_SIZE, 99, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addGap(30, 30, 30)))  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel7)  .addComponent(txtResultado))  .addComponent(txtMensaje, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addContainerGap(69, Short.MAX\_VALUE))  );   jTabbedPane1.addTab("CALCULAR IMC", jPanel2);   javax.swing.GroupLayout layout = **new** javax.swing.GroupLayout(getContentPane());  getContentPane().setLayout(layout);  layout.setHorizontalGroup(  layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(layout.createSequentialGroup()  .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(layout.createSequentialGroup()  .addGap(154, 154, 154)  .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 268, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGroup(layout.createSequentialGroup()  .addGap(24, 24, 24)  .addComponent(jTabbedPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 531, javax.swing.GroupLayout.PREFERRED\_SIZE)))  .addContainerGap(21, Short.MAX\_VALUE))  );  layout.setVerticalGroup(  layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(layout.createSequentialGroup()  .addGap(23, 23, 23)  .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 65, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  .addComponent(jTabbedPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 286, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addContainerGap(44, Short.MAX\_VALUE))  );   pack();  }// </editor-fold>//GEN-END:initComponents   **private** **void** **campoIPServidorActionPerformed**(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_campoIPServidorActionPerformed  // TODO add your handling code here:  }//GEN-LAST:event\_campoIPServidorActionPerformed   **private** **void** **btnIniciar1ActionPerformed**(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_btnIniciar1ActionPerformed  // TODO add your handling code here:  **if** (!servidor.isConnected()) {  JOptionPane.showConfirmDialog(**this**, "Cliente OffLine, Conecte con el Servidor");  **return**;  }  **float** peso = Float.parseFloat(campoPeso.getText());  **float** altura = Float.parseFloat(campoAltura.getText());  Thread hilo = **new** Thread() {  @Override  **public** **void** **run**() {  **try** {  System.out.println("Peso: " + peso);  System.out.println("Altura: " + altura);  out.writeFloat(peso);  out.writeFloat(altura);  out.flush();  System.out.println("Enviados los datos\nEsperando respuesta");  **float** imc = in.readFloat();  String msj = in.readUTF();  System.out.println("IMC: " + imc + "\nMensaje: " + msj);  txtResultado.setText(imc + "");  txtMensaje.setText(msj);  } **catch** (IOException ex) {  JOptionPane.showMessageDialog(VentanaPrincipal.**this**, "ERROR con el cliente " + ex.getMessage());  System.out.println("ERROR con el cliente " + ex.getMessage());  ex.printStackTrace();  }  }  };  hilo.start();  }//GEN-LAST:event\_btnIniciar1ActionPerformed   **private** **void** **btnIniciarActionPerformed**(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_btnIniciarActionPerformed  // TODO add your handling code here:  **int** puerto = Integer.parseInt(campoPuertoServidor.getText());  String ip = campoIPServidor.getText();  **try** {  **if** (btnIniciar.getText().equalsIgnoreCase("Conectar")) {  servidor = **new** Socket(ip, puerto);  out = **new** DataOutputStream(servidor.getOutputStream());  in = **new** DataInputStream(servidor.getInputStream());  btnIniciar.setText("Desconectar");  btnIniciar.setForeground(Color.RED);  txtEstado.setText("Conectado");  txtEstado.setForeground(Color.GREEN);  } **else** **if** (btnIniciar.getText().equalsIgnoreCase("Desconectar")) {  **if** (servidor.isConnected()) {  servidor.close();  }  btnIniciar.setText("Conectar");  txtEstado.setText("Desconectado");  btnIniciar.setForeground(Color.GREEN);  txtEstado.setForeground(Color.RED);  }  } **catch** (IOException ex) {  System.out.println("ERROR AL CONECTAR");  ex.printStackTrace();  }  }//GEN-LAST:event\_btnIniciarActionPerformed   **public** JLabel **getTxtEstado**() {  **return** txtEstado;  }   **public** JButton **getBtnIniciar**() {  **return** btnIniciar;  }    /\*\*  \* **@param** args the command line arguments  \*/  **public** **static** **void** **main**(String args[]) {  /\* Set the Nimbus look and feel \*/  //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">  /\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.  \* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html   \*/  **try** {  **for** (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {  **if** ("Nimbus".equals(info.getName())) {  javax.swing.UIManager.setLookAndFeel(info.getClassName());  **break**;  }  }  } **catch** (ClassNotFoundException ex) {  java.util.logging.Logger.getLogger(VentanaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, **null**, ex);  } **catch** (InstantiationException ex) {  java.util.logging.Logger.getLogger(VentanaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, **null**, ex);  } **catch** (IllegalAccessException ex) {  java.util.logging.Logger.getLogger(VentanaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, **null**, ex);  } **catch** (javax.swing.UnsupportedLookAndFeelException ex) {  java.util.logging.Logger.getLogger(VentanaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, **null**, ex);  }  //</editor-fold>  //</editor-fold>   /\* Create and display the form \*/  java.awt.EventQueue.invokeLater(**new** Runnable() {  **public** **void** **run**() {  **new** VentanaPrincipal().setVisible(**true**);  }  });  }   // Variables declaration - do not modify//GEN-BEGIN:variables  **private** javax.swing.JButton btnIniciar;  **private** javax.swing.JButton btnIniciar1;  **private** javax.swing.JTextField campoAltura;  **private** javax.swing.JTextField campoIPServidor;  **private** javax.swing.JTextField campoPeso;  **private** javax.swing.JTextField campoPuertoServidor;  **private** javax.swing.JLabel jLabel1;  **private** javax.swing.JLabel jLabel2;  **private** javax.swing.JLabel jLabel3;  **private** javax.swing.JLabel jLabel4;  **private** javax.swing.JLabel jLabel5;  **private** javax.swing.JLabel jLabel6;  **private** javax.swing.JLabel jLabel7;  **private** javax.swing.JPanel jPanel1;  **private** javax.swing.JPanel jPanel2;  **private** javax.swing.JTabbedPane jTabbedPane1;  **private** javax.swing.JLabel txtEstado;  **private** javax.swing.JLabel txtMensaje;  **private** javax.swing.JLabel txtResultado;  // End of variables declaration//GEN-END:variables } |
| --- |

SERVIDOR

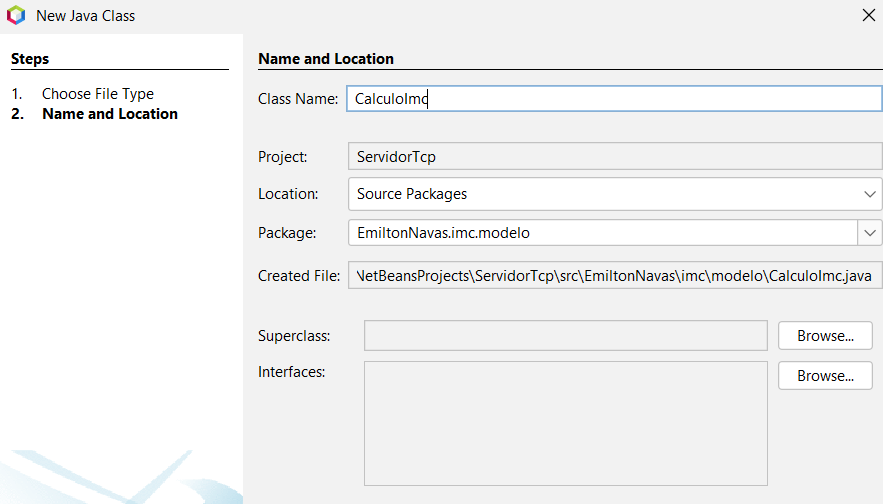
creamos el proyecto servidor tcp



creamos el paquete del modelo para calcular el Imc

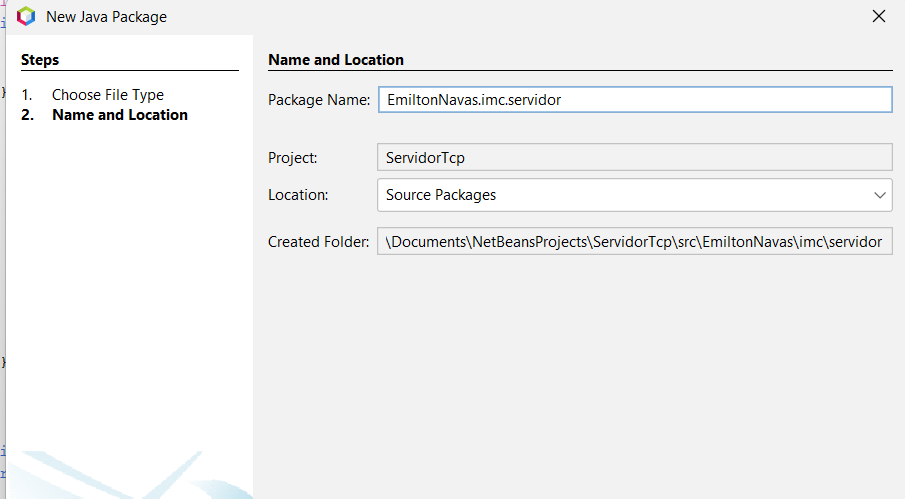


Creamos la clase para calcular el imc

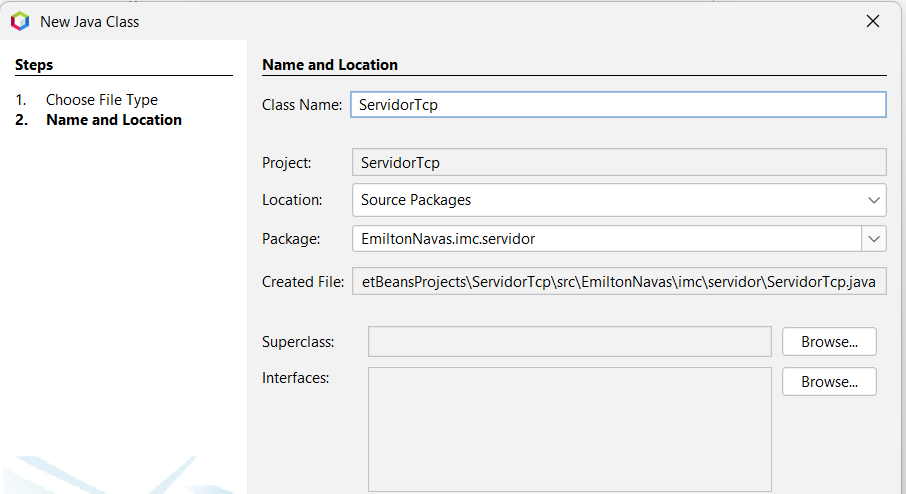


| **package** EmiltonNavas.imc.modelo;  **import** java.io.Serializable;  /\*\*  \* **@author** Emilton Navas  \*/ **public** **class** **CalculoImc** **implements** **Serializable** {      **private** **float** peso;  **private** **float** altura;   **public** **static** **class** **Imc** {   **public** **float** resultado;  **public** String mensaje;  }   **private** Imc imc;   **public** **CalculoImc**() { }   **public** **CalculoImc**(**float** peso, **float** altura) {  **this**.peso = peso;  **this**.altura = altura;  }   **public** Imc **getImc**() {  imc = **new** Imc();  **if** (peso <= 0 || altura <= 0) {  imc.mensaje = "ERROR: El peso y la altura deben ser mayores que 0";  **return** imc;  } **else** {  imc.resultado = peso / (altura \* altura);  **if** (imc.resultado < 18.5) {  imc.mensaje = "Debes consultar un Medico, tu peso es muy bajo";  } **else** **if** (imc.resultado >= 18.5 && imc.resultado <= 24.9) {  imc.mensaje = "Estas bien de peso";  } **else** **if** (imc.resultado > 24.9 && imc.resultado <= 29.9) {  imc.mensaje = "Debes bajar un poco de peso";  } **else** {  imc.mensaje = "Debes consultar un Medico, tu peso es muy alto";  }  **return** imc;  }   }   **public** **float** **getPeso**() {  **return** peso;  }   **public** **void** **setPeso**(**float** peso) {  **this**.peso = peso;  }   **public** **float** **getAltura**() {  **return** altura;  }   **public** **void** **setAltura**(**float** altura) {  **this**.altura = altura;  }  } |
| --- |

Se cre el paquete EmiltonNavas.Imc.servidor

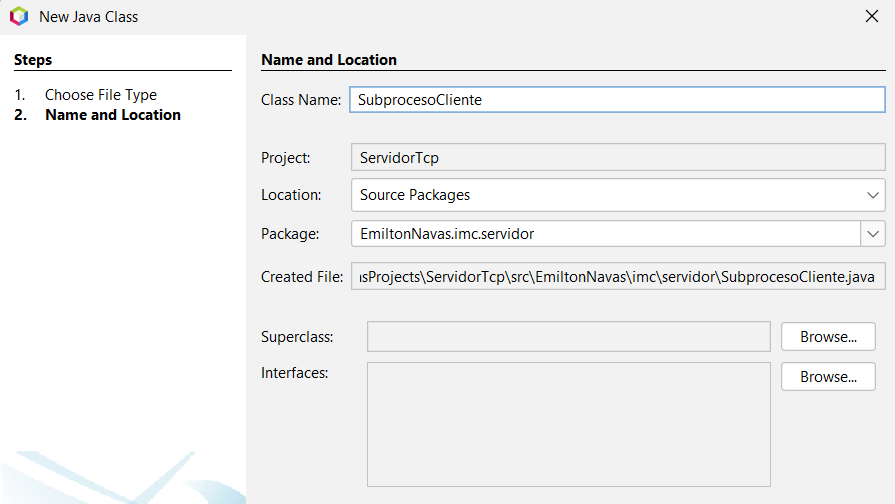


se crea la clase



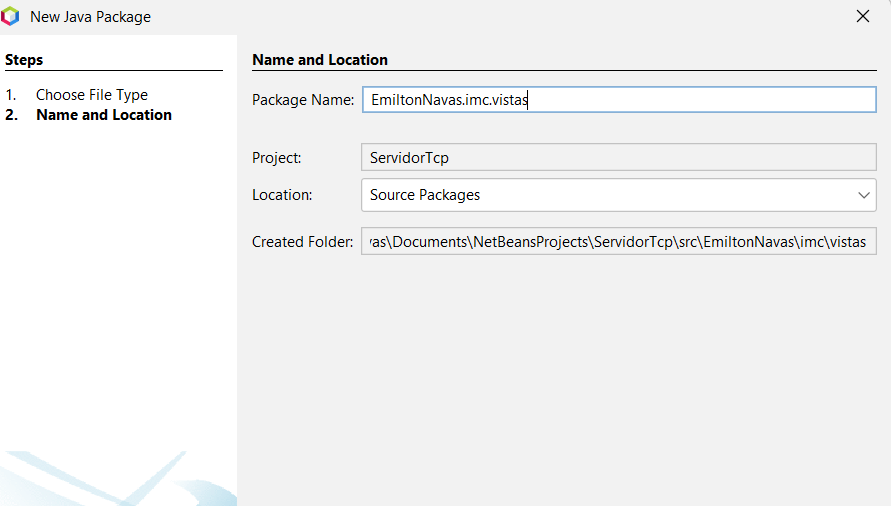
| **package** EmiltonNavas.imc.servidor;  /\*  \* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license  \* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template  \*/  **import** java.awt.Color; **import** java.io.IOException; **import** java.net.ServerSocket; **import** java.net.Socket; **import** java.text.SimpleDateFormat; **import** java.util.Date; **import** java.util.HashMap; **import** java.util.Map; **import** java.util.function.Function; **import** EmiltonNavas.imc.vistas.VentanaPrincipal; /\*\*  \*  \* **@author** Emilton Navas  \*/ **public** **class** **ServidorTcp** **extends** **Thread**{  **private** Boolean estado;  **public** **static** Map<String, SubProcesoCliente> listaDeClientes;  **private** Integer puerto = 9007;  **private** ServerSocket servicio;  **private** VentanaPrincipal ventana;   **public** **ServidorTcp**(Integer puerto, VentanaPrincipal v) {  **if** (puerto != **null** || puerto != 0) {  **this**.puerto = puerto;  }  ventana = v;  listaDeClientes = **new** HashMap<>();  }   @Override  **public** **void** **run**() {  **super**.run(); //To change body of generated methods, choose Tools | Templates.  iniciarServicio();  }   **public** **void** **iniciarServicio**() {  **try** {  servicio = **new** ServerSocket(puerto);  estado = **true**;  ventana.getBtnIniciar().setText("DETENER");  ventana.getTxtEstado().setText("ONLINE");  ventana.getTxtEstado().setForeground(Color.green);  ventana.getBtnIniciar().setForeground(Color.RED);    String msg = log() + "Servidor disponible en el Puerto " + puerto;  System.out.println(msg);  ventana.getCajaLog().append(msg + "\n");  **while** (estado) {  Socket cliente = servicio.accept();  String ip = cliente.getInetAddress().getHostAddress();  msg =log()+"Cliente " + ip + " conectado";  System.out.println(msg + "\n");  ventana.getCajaLog().append(msg + "\n");  SubProcesoCliente atencion = **new** SubProcesoCliente(cliente, ventana);  ServidorTcp.listaDeClientes.put(ip, atencion);  atencion.start();  }  } **catch** (IOException ex) {  String msg = log()+"ERROR al abrir el puerto " + puerto;  System.out.println(msg);  ventana.getCajaLog().append(msg + "\n");  ventana.getBtnIniciar().setText("INICIAR");  ventana.getTxtEstado().setText("OFF LINE");  }  }   **public** **void** **detenerServicio**() {  **if** (estado) {  estado = **false**;  ventana.getBtnIniciar().setText("INICIAR");  ventana.getBtnIniciar().setForeground(Color.GREEN);  ventana.getTxtEstado().setText("OFF LINE");  ventana.getTxtEstado().setForeground(Color.RED);  ServidorTcp.listaDeClientes.entrySet().stream().map(**new** Function<Map.Entry<String, SubProcesoCliente>, String>() {  @Override  **public** String **apply**(Map.Entry<String, SubProcesoCliente> elemento) {  String ip = elemento.getKey();  SubProcesoCliente cliente = elemento.getValue();  String msg = log()+"Desconectando cliente " + ip;  System.out.println(msg);  ventana.getCajaLog().append(msg + "\n");  **try** {  cliente.getCliente().close();  cliente = **null**;  ServidorTcp.listaDeClientes.remove(elemento);  msg = log()+"Cliente desconectado" + ip;  System.out.println(msg);  ventana.getCajaLog().append(msg + "\n");  } **catch** (IOException ex) {  cliente = **null**;  ServidorTcp.listaDeClientes.remove(elemento);  msg = log()+"Cliente desconectado" + ip;  System.out.println(msg);  ventana.getCajaLog().append(msg + "\n");  }  **return** ip;  }  }).forEachOrdered(ip -> {  System.out.println("cliente " + ip + " Desconectado");  });  **try** {  servicio.close();  } **catch** (IOException ex) {  System.out.println("ERRRO no se puede cerrar el Puerto " + puerto);  String msg = log()+"ERRRO no se puede cerrar el Puerto " + puerto;  System.out.println(msg);  ventana.getCajaLog().append(msg + "\n");  }  }  }   **public** String **log**() {  SimpleDateFormat f = **new** SimpleDateFormat("dd-MM-yyyy hh:mm:ss a");  **return** f.format(**new** Date()) + " - ";  } } |
| --- |

Se crea la clase subproceso cliente

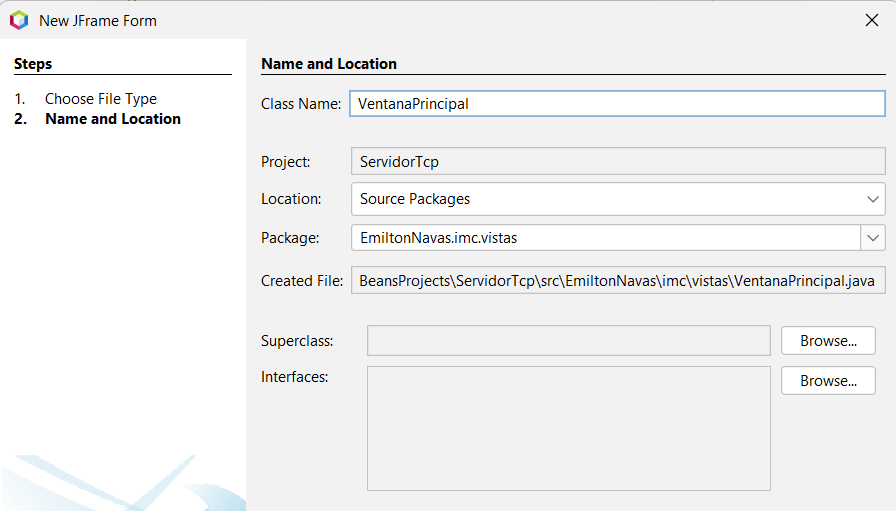


| **package** EmiltonNavas.imc.servidor;  **import** java.io.DataInputStream; **import** java.io.DataOutputStream; **import** java.io.IOException; **import** java.net.Socket; **import** java.text.SimpleDateFormat; **import** java.util.Date; **import** EmiltonNavas.imc.modelo.CalculoImc; **import** EmiltonNavas.imc.vistas.VentanaPrincipal;   **public** **class** **SubProcesoCliente** **extends** **Thread** {   **private** Socket cliente;  **private** String ip;  **private** VentanaPrincipal ventana;     **public** **SubProcesoCliente**(Socket cliente, VentanaPrincipal v) {  **this**.cliente = cliente;  ip = cliente.getInetAddress().getHostAddress();  ventana = v;  }   @Override  **public** **void** **run**() {  **try** {  CalculoImc.Imc imc = calcularImc();  enviarRespuesta(imc);  } **catch** (Exception ex) {  System.out.println(log()+ex.getMessage());  ventana.getCajaLog().append(log()+ex.getMessage() + "\n");  **try** {  cliente.close();  } **catch** (IOException ex1) {  ServidorTcp.listaDeClientes.remove(ip);  } **finally** {  ServidorTcp.listaDeClientes.remove(ip);  }  }  }   **public** CalculoImc.Imc **calcularImc**() **throws** Exception {  DataInputStream input = **null**;  **try** {  input = **new** DataInputStream(cliente.getInputStream());  String msg = "Esperando el PESO: ";  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n" + "\n");  **float** peso = input.readFloat();  msg = "PESO: " + peso;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  msg = "Esperando La Altura: ";  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  **float** altura = input.readFloat();  msg = "ALTURA: " + altura;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  CalculoImc datosImc = **new** CalculoImc(peso, altura);  System.out.println(log()+"IMC: " + datosImc.getImc().resultado);  msg = "IMC: " + datosImc.getImc().resultado;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  System.out.println(log()+"MENSAJE: " + datosImc.getImc().mensaje);  msg = "MENSAJE: " + datosImc.getImc().mensaje;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  **return** datosImc.getImc();  } **catch** (IOException ex) {  String msg = "Error al caputurar datos del cliente " + ip;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  **throw** **new** Exception("Error al caputurar datos del cliente " + ip);   }   }   **public** **void** **enviarRespuesta**(CalculoImc.Imc imc) {   Thread hiloResponde = **new** Thread() {  @Override  **public** **void** **run**() {  DataOutputStream output = **null**;  **try** {  output = **new** DataOutputStream(cliente.getOutputStream());  output.writeFloat(imc.resultado);  output.writeUTF(imc.mensaje);  String msg = "IMC: " + imc.resultado;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  msg = "MENSAJE: " + imc.mensaje;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  output.flush();  enviarRespuesta(calcularImc());  } **catch** (IOException ex) {  String msg = "Error al enviar datos al cliente " + ip;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  ServidorTcp.listaDeClientes.remove(ip);  } **catch** (Exception ex) {  String msg = "Error al leer datos del cliente " + ip;  System.out.println(log()+msg);  ventana.getCajaLog().append(log()+msg + "\n");  **try** {  cliente.close();  } **catch** (IOException ex1) {  ServidorTcp.listaDeClientes.remove(ip);  } **finally** {  ServidorTcp.listaDeClientes.remove(ip);  }  }  }  };  hiloResponde.start();  }   **public** Socket **getCliente**() {  **return** cliente;  }   **public** **void** **setCliente**(Socket cliente) {  **this**.cliente = cliente;  }   **public** String **log**() {  SimpleDateFormat f = **new** SimpleDateFormat("dd-MM-yyyy hh:mm:ss a");  **return** ip + " -> " + f.format(**new** Date()) + " - ";  } } |
| --- |

Se creel paquete para alojar las vistas



creamos un Jframe



/\*

| \* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license  \* Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template  \*/ **package** EmiltonNavas.imc.vistas;  //import Emiltonnavas.imc.servidor.ServidorTcp; **import** java.awt.Color; **import** java.net.InetAddress; **import** java.net.UnknownHostException; **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.JTextArea; **import** EmiltonNavas.imc.servidor.ServidorTcp;  /\*\*  \*  \* **@author** emanu  \*/ **public** **class** **VentanaPrincipal** **extends** **javax**.**swing**.**JFrame** {   /\*\*  \* Creates new form VentanaPrincipal  \*/  **public** **VentanaPrincipal**() {  initComponents();  }   /\*\*  \* This method is called from within the constructor to initialize the form.  \* WARNING: Do NOT modify this code. The content of this method is always  \* regenerated by the Form Editor.  \*/  @SuppressWarnings("unchecked")  // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents  **private** **void** **initComponents**() {   jLabel1 = **new** javax.swing.JLabel();  jTabbedPane1 = **new** javax.swing.JTabbedPane();  jPanel1 = **new** javax.swing.JPanel();  jLabel2 = **new** javax.swing.JLabel();  campoPuerto = **new** javax.swing.JTextField();  jLabel3 = **new** javax.swing.JLabel();  campoIP = **new** javax.swing.JTextField();  btnIniciar = **new** javax.swing.JButton();  jLabel4 = **new** javax.swing.JLabel();  txtEstado = **new** javax.swing.JLabel();  jPanel2 = **new** javax.swing.JPanel();  jScrollPane1 = **new** javax.swing.JScrollPane();  cajaLog = **new** javax.swing.JTextArea();  btnLimpiar = **new** javax.swing.JButton();   setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);   jLabel1.setFont(**new** java.awt.Font("Segoe Print", 1, 24)); // NOI18N  jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  jLabel1.setText("SERVIDOR IMC");  jLabel1.setToolTipText("");   jTabbedPane1.setBorder(javax.swing.BorderFactory.createEtchedBorder());   jLabel2.setText("DIRECCION IP:");   campoPuerto.setText("9007");  campoPuerto.addActionListener(**new** java.awt.event.ActionListener() {  **public** **void** **actionPerformed**(java.awt.event.ActionEvent evt) {  campoPuertoActionPerformed(evt);  }  });   jLabel3.setText("PUERTO DE RED:");   campoIP.addActionListener(**new** java.awt.event.ActionListener() {  **public** **void** **actionPerformed**(java.awt.event.ActionEvent evt) {  campoIPActionPerformed(evt);  }  });   btnIniciar.setFont(**new** java.awt.Font("Serif", 1, 18)); // NOI18N  btnIniciar.setForeground(**new** java.awt.Color(0, 153, 51));  btnIniciar.setText("INICIAR");  btnIniciar.addActionListener(**new** java.awt.event.ActionListener() {  **public** **void** **actionPerformed**(java.awt.event.ActionEvent evt) {  btnIniciarActionPerformed(evt);  }  });   jLabel4.setText("ESTADO:");   txtEstado.setForeground(**new** java.awt.Color(255, 0, 51));  txtEstado.setText("DETENIDO");   javax.swing.GroupLayout jPanel1Layout = **new** javax.swing.GroupLayout(jPanel1);  jPanel1.setLayout(jPanel1Layout);  jPanel1Layout.setHorizontalGroup(  jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel1Layout.createSequentialGroup()  .addGap(20, 20, 20)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel1Layout.createSequentialGroup()  .addComponent(jLabel4)  .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  .addComponent(txtEstado))  .addGroup(jPanel1Layout.createSequentialGroup()  .addComponent(jLabel3)  .addGap(18, 18, 18)  .addComponent(campoPuerto, javax.swing.GroupLayout.PREFERRED\_SIZE, 189, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGroup(jPanel1Layout.createSequentialGroup()  .addComponent(jLabel2)  .addGap(31, 31, 31)  .addComponent(campoIP, javax.swing.GroupLayout.PREFERRED\_SIZE, 189, javax.swing.GroupLayout.PREFERRED\_SIZE)))  .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))  .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()  .addContainerGap(179, Short.MAX\_VALUE)  .addComponent(btnIniciar, javax.swing.GroupLayout.PREFERRED\_SIZE, 160, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addGap(175, 175, 175))  );  jPanel1Layout.setVerticalGroup(  jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(jPanel1Layout.createSequentialGroup()  .addGap(29, 29, 29)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel2)  .addComponent(campoIP, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGap(23, 23, 23)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel3)  .addComponent(campoPuerto, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGap(26, 26, 26)  .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  .addComponent(jLabel4)  .addComponent(txtEstado))  .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 35, Short.MAX\_VALUE)  .addComponent(btnIniciar, javax.swing.GroupLayout.PREFERRED\_SIZE, 32, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addGap(17, 17, 17))  );   jTabbedPane1.addTab("CONEXION", jPanel1);   cajaLog.setColumns(20);  cajaLog.setRows(5);  jScrollPane1.setViewportView(cajaLog);   btnLimpiar.setText("LIMPIAR");  btnLimpiar.addActionListener(**new** java.awt.event.ActionListener() {  **public** **void** **actionPerformed**(java.awt.event.ActionEvent evt) {  btnLimpiarActionPerformed(evt);  }  });   javax.swing.GroupLayout jPanel2Layout = **new** javax.swing.GroupLayout(jPanel2);  jPanel2.setLayout(jPanel2Layout);  jPanel2Layout.setHorizontalGroup(  jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()  .addContainerGap(391, Short.MAX\_VALUE)  .addComponent(btnLimpiar, javax.swing.GroupLayout.PREFERRED\_SIZE, 102, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addGap(21, 21, 21))  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()  .addContainerGap(15, Short.MAX\_VALUE)  .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 481, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addContainerGap(18, Short.MAX\_VALUE)))  );  jPanel2Layout.setVerticalGroup(  jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()  .addContainerGap(185, Short.MAX\_VALUE)  .addComponent(btnLimpiar, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addContainerGap())  .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()  .addContainerGap(18, Short.MAX\_VALUE)  .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 158, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addContainerGap(46, Short.MAX\_VALUE)))  );   jTabbedPane1.addTab("LOG DE CONEXIONES", jPanel2);   javax.swing.GroupLayout layout = **new** javax.swing.GroupLayout(getContentPane());  getContentPane().setLayout(layout);  layout.setHorizontalGroup(  layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(layout.createSequentialGroup()  .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(layout.createSequentialGroup()  .addGap(16, 16, 16)  .addComponent(jTabbedPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 518, javax.swing.GroupLayout.PREFERRED\_SIZE))  .addGroup(layout.createSequentialGroup()  .addGap(175, 175, 175)  .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 190, javax.swing.GroupLayout.PREFERRED\_SIZE)))  .addContainerGap(17, Short.MAX\_VALUE))  );  layout.setVerticalGroup(  layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  .addGroup(layout.createSequentialGroup()  .addGap(23, 23, 23)  .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 55, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addGap(18, 18, 18)  .addComponent(jTabbedPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  .addContainerGap(54, Short.MAX\_VALUE))  );   pack();  }// </editor-fold>//GEN-END:initComponents   **private** **void** **campoPuertoActionPerformed**(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_campoPuertoActionPerformed  // TODO add your handling code here:  }//GEN-LAST:event\_campoPuertoActionPerformed   **private** **void** **campoIPActionPerformed**(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_campoIPActionPerformed  // TODO add your handling code here:  }//GEN-LAST:event\_campoIPActionPerformed  ServidorTcp s;  **private** **void** **btnIniciarActionPerformed**(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_btnIniciarActionPerformed  // TODO add your handling code here:  **if** (btnIniciar.getText().equalsIgnoreCase("INICIAR")) {  **int** puerto = Integer.parseInt(campoPuerto.getText());  s = **new** ServidorTcp(puerto, **this**);  s.start();  btnIniciar.setText("DETENER");  txtEstado.setText("ONLINE");  txtEstado.setForeground(Color.GREEN);  } **else** **if** (btnIniciar.getText().equalsIgnoreCase("DETENER")) {  s.detenerServicio();  btnIniciar.setText("INICIAR");  txtEstado.setText("OFF LINE");  txtEstado.setForeground(Color.RED);  }  }//GEN-LAST:event\_btnIniciarActionPerformed   **private** **void** **formWindowOpened**(java.awt.event.WindowEvent evt) {   // TODO add your handling code here:  String ip;  **try** {  campoIP.setEditable(**false**);  ip = InetAddress.getLocalHost().getHostAddress();  campoIP.setText(ip);  } **catch** (UnknownHostException ex) {  JOptionPane.showMessageDialog(**this**,"Falla en la conexion");  }  }    **public** JLabel **getTxtEstado**() {  **return** txtEstado;  }   **public** JTextArea **getCajaLog**() {  **return** cajaLog;  }   **public** JButton **getBtnIniciar**() {  **return** btnIniciar;  }    **private** **void** **btnLimpiarActionPerformed**(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_btnLimpiarActionPerformed  // TODO add your handling code here:  }//GEN-LAST:event\_btnLimpiarActionPerformed   /\*\*  \* **@param** args the command line arguments  \*/  **public** **static** **void** **main**(String args[]) {  /\* Set the Nimbus look and feel \*/  //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">  /\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.  \* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html   \*/  **try** {  **for** (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {  **if** ("Nimbus".equals(info.getName())) {  javax.swing.UIManager.setLookAndFeel(info.getClassName());  **break**;  }  }  } **catch** (ClassNotFoundException ex) {  java.util.logging.Logger.getLogger(VentanaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, **null**, ex);  } **catch** (InstantiationException ex) {  java.util.logging.Logger.getLogger(VentanaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, **null**, ex);  } **catch** (IllegalAccessException ex) {  java.util.logging.Logger.getLogger(VentanaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, **null**, ex);  } **catch** (javax.swing.UnsupportedLookAndFeelException ex) {  java.util.logging.Logger.getLogger(VentanaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, **null**, ex);  }  //</editor-fold>  //</editor-fold>   /\* Create and display the form \*/  java.awt.EventQueue.invokeLater(**new** Runnable() {  **public** **void** **run**() {  **new** VentanaPrincipal().setVisible(**true**);  }  });  }   // Variables declaration - do not modify//GEN-BEGIN:variables  **private** javax.swing.JButton btnIniciar;  **private** javax.swing.JButton btnLimpiar;  **private** javax.swing.JTextArea cajaLog;  **private** javax.swing.JTextField campoIP;  **private** javax.swing.JTextField campoPuerto;  **private** javax.swing.JLabel jLabel1;  **private** javax.swing.JLabel jLabel2;  **private** javax.swing.JLabel jLabel3;  **private** javax.swing.JLabel jLabel4;  **private** javax.swing.JPanel jPanel1;  **private** javax.swing.JPanel jPanel2;  **private** javax.swing.JScrollPane jScrollPane1;  **private** javax.swing.JTabbedPane jTabbedPane1;  **private** javax.swing.JLabel txtEstado;  // End of variables declaration//GEN-END:variables } |
| --- |



