

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL  
FACULTAD DE INGENIERIA EN ELECTRICIDAD Y COMPUTACIÓN  
FUNDAMENTOS DE JAVA  
TERCERA EVALUACIÓN (I TÉRMINO 2012-2013)

NOMBRE: \_\_\_\_\_

MATRICULA: \_\_\_\_\_

**TEMA 1 (70 PUNTOS)**

Dado el siguiente formulario para el ingreso de datos de un empleado usted debe implementar lo siguiente:

1.- La clase **Empleado** con sus respectivos atributos (String cedula, String nombres, String apellidos, String direccion, String genero), métodos (GET, SET) y constructor que inicializa todos sus atributos. **(20 PUNTOS)**

2.- El método **private void ingresarEmpleado()** que es invocado al momento que el usuario presiona el botón **INGRESAR**. **(20 PUNTOS)**

3.- El método **private boolean formularioValido()** que verifica que todos los campos del formulario (cedula, nombres, apellidos y dirección) no estén vacíos. **(20 PUNTOS)**

4.- El método **private void limpiarFormulario()** que deja en blanco el valor de los JTextField. **(10 PUNTOS)**



**NOTA:** Tome en cuenta que existe la clase **Conexion** que tiene además de los métodos utilizados en las prácticas uno que es **public boolean insertarRegistroEmpleado(Empleado e)**.

Para la presentación de mensajes tenga en cuenta los siguientes ejemplos:

```
JOptionPane.showMessageDialog(rootPane, "ingreso exitoso", "EXITO", JOptionPane.INFORMATION_MESSAGE);  
JOptionPane.showMessageDialog(rootPane, "error en el ingreso", "ERROR", JOptionPane.ERROR_MESSAGE);
```

## TEMA 2 (20 PUNTOS)

Analizar el siguiente código para la consulta de personas e indique los posibles errores y cual seria su forma correcta.

```
if(busquedaValida()){
    String tipo          = this.cbTipo.getSelectedItem().toString();
    String descripcion   = this.txtDescripcion.setText();

    Conexion con = Conexion;
    try{
        con.conectar();
        ArrayList personas = con.consultarPersonas(descripcion, tipo);

        Object [][] resultado = new Object[personas.size()][7];

        for(int i=0;i<personas.size();i++){
            Persona p = (Persona)personas.get(i);
            resultado[i][0] = p.setCedula();
            resultado[i][1] = p.setNombres();
            resultado[i][2] = p.setApellidos();
            resultado[i][3] = p.setEdad();
            resultado[i][4] = p.setDireccion();
            resultado[i][5] = p.setTelefono();
            resultado[i][6] = p.setGenero();
        }

        tbResultado.setModel(new javax.swing.table.DefaultTableModel(
            resultado,
            new String [] {
                "CEDULA", "NOMBRES", "APELLIDOS", "EDAD", "DIRECCION", "TELEFONO", "GENERO"
            }
        ));

    }catch(Exception e){
        JOptionPane.showMessageDialog(rootPane,"OCURRIO UN ERROR EN LA CONEXION A LA BASE", "ERROR", JOptionPane.ERROR_MESSAGE);
    }
}
```

## TEMA 3 (10 PUNTOS)

Analizar el siguiente código para la eliminación del registro de una Persona y complete los espacios indicados.

```
int fila_seleccionada = this.tbResultado.getSelectedRow();

if(fila_seleccionada!=-1){

    String cedula      = this.tbResultado.getValueAt(fila_seleccionada, 0).toString();
    Persona p          = new Persona(cedula);

    try{
        Conexion c = .....;
        c.....();

        if(c.eliminarPersona(...)){
            JOptionPane.showMessageDialog(rootPane, "SE HA ELIMINADO EXITOSAMENTE EL REGISTRO", "EXITO", JOptionPane.INFORMATION_MESSAGE);
        }else{
            JOptionPane.showMessageDialog(rootPane, "HA OCURRIDO UN ERROR EN LA ELIMINACION", "ERROR", JOptionPane.ERROR_MESSAGE);
        }

    }catch(.....){
        JOptionPane.showMessageDialog(rootPane, "HA OCURRIDO UN ERROR EN LA CONEXION", "ERROR", JOptionPane.ERROR_MESSAGE);
    }

    consultarFormulario();

}else{
    JOptionPane.showMessageDialog(rootPane, "SELECCIONE POR LO MENOS UN REGISTRO", "ERROR", JOptionPane.ERROR_MESSAGE);
}
```