	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

Puntuación:

Instrucciones:

Únicamente se puede utilizar como referencia los códigos incluidos en el anexo de este documento. No se podrá utilizar internet ni cualquier otro documento.

Completar este documento con capturas de pantalla, una vez finalizado el examen convertirlo a PDF y subirlo a la tarea correspondiente de classroom.

Subir en un mismo documento comprimido los cuatro proyectos sin comprimir por separado. Indica tu nombre y apellidos en el nombre del archivo comprimido.

Ejercicio 1 Table Model. 2 puntos .Realiza capturas de pantalla del código y de la ejecución.

Crea una aplicación que gestione las citas de la ITV de una estación de ITV.

- a) La clase Cita tendrá los siguientes parámetros: idCita, Matricula, Coche, Fecha. La clase LogicaNegocio creará 4 citas diferentes y las devolverá como un array de citas. **(0,5 puntos)**

Ejemplo de cita: "1", "7766KVG", "Renault Megane", "1-12-2022".

(Clase Cita)

```
public class Cita {

    // ATRIBUTOS
    private int idCita;
    private String matricula;
    private String coche;
    private String fecha;
```


(Clase LogicaNegocio)

```
public class LogicaNegocio {

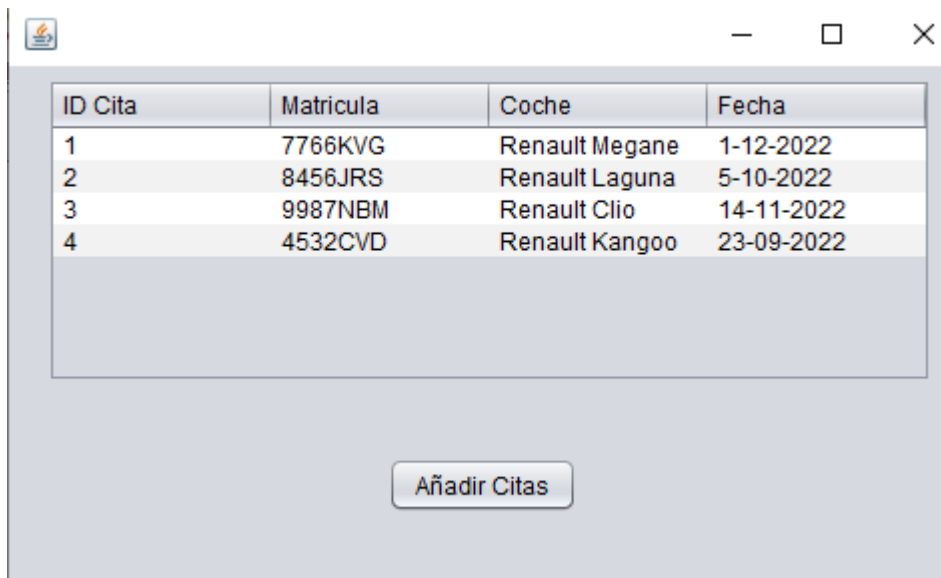
    private List<Cita> listaCitas = new ArrayList<>();

    public LogicaNegocio() {
        listaCitas.add(new Cita(1,"7766KVG","Renault Megane","1-12-2022"));
        listaCitas.add(new Cita(2,"8456JRS","Renault Laguna","5-10-2022"));
        listaCitas.add(new Cita(3,"9987NBM","Renault Clio","14-11-2022"));
        listaCitas.add(new Cita(4,"4532CVD","Renault Kangoo","23-09-2022"));
    }

    public List<Cita> getListaCitas() {
        return listaCitas;
    }
}
```

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]


(Ejemplo de ejecución)



ID Cita	Matricula	Coche	Fecha
1	7766KVG	Renault Megane	1-12-2022
2	8456JRS	Renault Laguna	5-10-2022
3	9987NBM	Renault Clio	14-11-2022
4	4532CVD	Renault Kangoo	23-09-2022

Añadir Citas

- b) Crea un table model propio que herede de la clase Abstracta AbstractTableModel generando los métodos abstractos correspondientes para mostrar las citas en una tabla. No olvides poner en los títulos de la tabla los nombres Matricula, Coche y Fecha haciendo override al método getColumnName. **Crea la ventana correspondiente para que se vea la tabla con los datos. . (0,75 puntos).**

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

(Clase CitaTableModel)

```

public class CitaTableModel extends AbstractTableModel{

    private String[] columnas = {"ID Cita", "Matricula", "Coche", "Fecha"};
    private List<Cita> listaCitas;

}

public CitaTableModel(List<Cita> listaCitas) {
    this.listaCitas = listaCitas;
}


@Override
public int getRowCount() {
    return listaCitas.size();
}

@Override
public int getColumnCount() {
    return 4;
}

@Override
public Object getValueAt(int rowIndex, int columnIndex) {
    switch(columnIndex)
    {
        case 0:
            return listaCitas.get(rowIndex).getIdCita();
        case 1:
            return listaCitas.get(rowIndex).getMatricula();
        case 2:
            return listaCitas.get(rowIndex).getCoche();
        case 3:
            return listaCitas.get(rowIndex).getFecha();
    }
    return null;
}

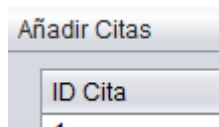
@Override
public String getColumnName(int column) {
    return columnas[column];
}

```

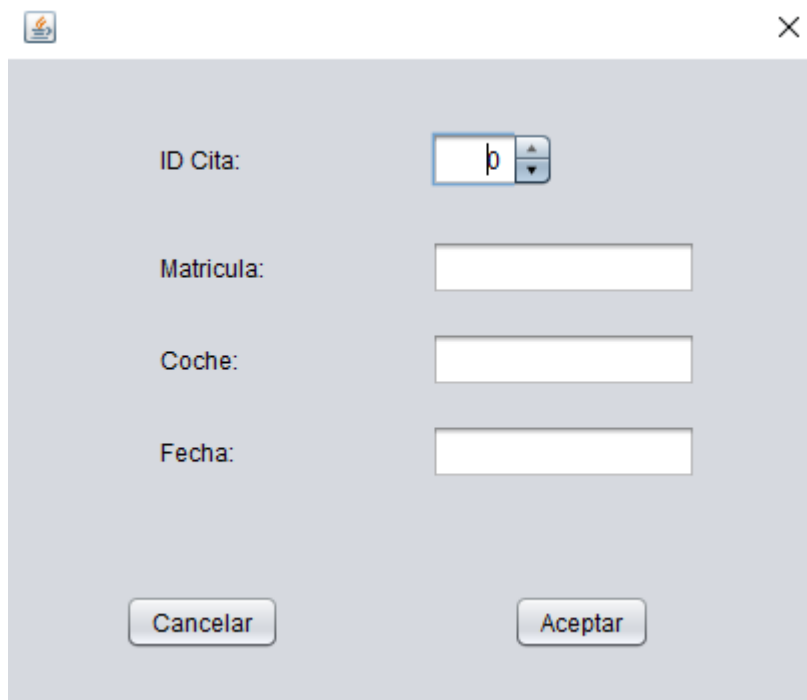
	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

- c) Crea en una nueva ventana, que se abrirá desde una barra superior de menú, un formulario para añadir nuevas citas y que se actualicen en la tabla **(0,75 puntos)**

(Botón para añadir citas)



(Ventana para añadir citas)



```
private void jButtonAceptarActionPerformed(java.awt.event.ActionEvent evt) {
    int id = (Integer) this.jSpinnerID.getValue();
    String matricula = this.jTextFieldMatricula.getText();
    String coche = this.jTextFieldCoche.getText();
    String fecha = this.jTextFieldFecha.getText();


    Cita cita = new Cita(id, matricula, coche, fecha);

    padre.anhadirCitasTabla(cita);
    dispose();
}
```

(Nueva cita añadida)

5	4345JKL	Peugeot 206	21-04-2023
---	---------	-------------	------------

Ejercicio 2 Extensión de Componentes. 2,5 puntos. Completar con capturas de pantalla de código y de

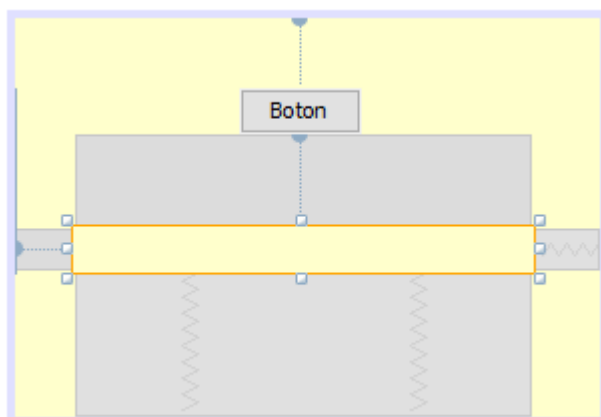
	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

ejecución.


- a) Crea un componente nuevo en un nuevo proyecto llamado PanelExamen que extienda de JPanel a partir de un JPanel Form con el nombre PanelExamen que contenga un JLabel, un botón y una variable llamada aleatorio:

Se debe inicializar el botón con un color de fondo rojo, y en aleatorio se debe guardar un nº aleatorio, en el label debe aparecer: Examen de “tu nombre” + su nº aleatorio. Añade el componente a la paleta de componentes(haz capturas de pantalla)(1 punto)

(Panel)



(Funcionamiento del aleatorio)

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

```

public PanelExamen() {
    initComponents();
    this.jButton1.setBackground(Color.red);
}

public int generarAleatorio() {
    return (int) (Math.random() * 100);
}

/**
 * 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.
 */
@Override
@SuppressWarnings("unchecked")
Generated Code

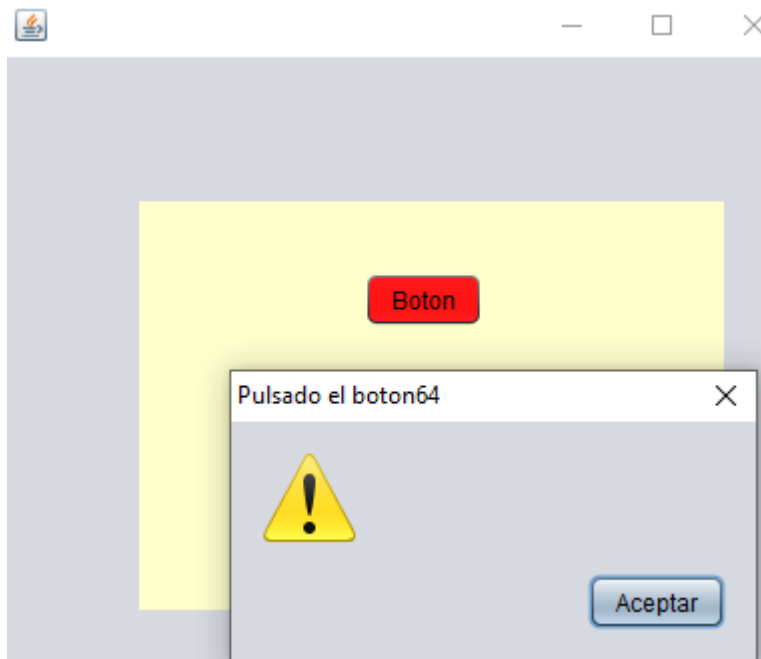
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    this.jLabelAleatorio.setText("Examen de Javier Losa: "+generarAleatorio());
}


```

- b) Utiliza el componente en otro proyecto de la siguiente forma:

Al pulsar el botón se generará un JOptionPane con el icono Warning y con el texto “Pulsado el botón nº ” + el número aleatorio que tenía el botón. **(0,5 puntos)**

(El componente utilizado en otro proyecto)



	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: XXXXXXXXXX

- c) En el proyecto del apartado b crea un MouseListener propio y añadelo al botón anterior que haga::
(1 punto)
- Al pasar el ratón por encima, el botón se ponga rojo.
 - Al salir, el botón se ponga amarillo.
 - Mientras se pulse el botón se ponga verde.

(Parte de la clase CustomMouseListener y ejemplos de ejecución)

```

@Override
public void mousePressed(MouseEvent e) {
    JButton boton = (JButton) e.getComponent();
    boton.setBackground(Color.green);
}

@Override
public void mouseReleased(MouseEvent e) {


}

@Override
public void mouseEntered(MouseEvent e) {
    JButton boton = (JButton) e.getComponent();
    boton.setBackground(Color.red);
}

@Override
public void mouseExited(MouseEvent e) {
    JButton boton = (JButton) e.getComponent();
    boton.setBackground(Color.yellow);
}

```



	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

Ejercicio 3. Diseño 2,5 puntos. Completar con capturas de pantalla de código y de ejecución.

Crea un proyecto Ejercicio3Examen que cumpla con las siguientes características:

- a) Una clase **Juego** con los siguientes parámetros: Nombre(String), Plataforma(String), Precio(float)
 Una clase **LogicaNegocio** donde tengamos una lista de Juegos (iniciar con 4 Juegos). El identificador será una variable que empezará en 0 y se incrementará de 1 en 1 cada vez que se cree un Juego. (1 punto)

(Clase Juego)

```
// ATRIBUTOS
private static int contadorId = 0;

private int id;
private String nombre;
private String plataforma;
private float precio;

// CONSTRUCTORES
public Juego() {
    contadorId++;


    this.id = contadorId;
    this.nombre = "";
    this.plataforma = "";
    this.precio = 0;
}

public Juego(String nombre, String plataforma, float precio) {
    contadorId++;

    this.id = contadorId;
    this.nombre = nombre;
    this.plataforma = plataforma;
    this.precio = precio;
}

// GETTERS Y SETTERS
```

(Clase LogicaNegocio)

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

```

public class LogicaNegocio {

    private List<Juego> listaJuegos = new ArrayList<>();

    public LogicaNegocio() {
        listaJuegos.add(new Juego("Juego1", "Plataforma1", 3f));
        listaJuegos.add(new Juego("Juego2", "Plataforma5", 2f));
        listaJuegos.add(new Juego("Juego3", "Plataforma7", 4f));
        listaJuegos.add(new Juego("Juego4", "Plataforma9", 6f));
    }


    public List<Juego> getListaJuegos() {
        return listaJuegos;
    }
}

```

- b) Un JFrame llamado **ventanaPrincipal** en el que aparezca nombre y Plataforma primer juego dentro de JTextFields y el identificador en un JLabel.,
Tendremos los botones: Guardar Cambios, siguiente, anterior.
Al pulsar Guardar Cambios, se guardarán los datos que estén en los campos en el juego correspondiente. Al pulsar siguiente se mostrará el siguiente y al pulsar anterior se mostrará el anterior (1 punto)
(Ejemplo de ejecución (se muestran los valores por defecto, que son los del primer elemento de la lista))



- c) Si se pulsa anterior y no hay anterior, se mostrará un JOptionPane indicando que no hay anterior, si

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

se pulsa siguiente y no hay siguiente, se creará un nuevo Juego con los datos por defecto: Nombre, Plataforma e Identificador que corresponda.(0,5 puntos)

(Métodos de siguiente y anterior, que no me funcionan pero no sé por qué, pero esa era mi idea)

```
private void jButtonSiguienteActionPerformed(java.awt.event.ActionEvent evt) {
    if(Integer.parseInt(this.jLabelID.getText())+1 <= listaJuegos.size())
    {
        inicializarValores(Integer.parseInt(this.jLabelID.getText())+1);
    }
    else
    {
        Juego juego = new Juego();
        listaJuegos.add(juego);
    }
}

private void jButtonAnteriorActionPerformed(java.awt.event.ActionEvent evt) {
    if(Integer.parseInt(this.jLabelID.getText())-1 >= 0)
    {
        inicializarValores(Integer.parseInt(this.jLabelID.getText())-1);
    }
    else
    {
        JOptionPane.showMessageDialog(this, "ERROR ", "Error, no hay anterior", JOptionPane.WARNING_MESSAGE);
    }
}
```


Ejercicio 4. Crea un nuevo proyecto llamado Ejercicio4Examen (3 puntos)

Crea el juego de las 3 en raya.

- En la ventana principal se mostrarán 9 botones a modo de tablero. **0,25 puntos**
- Añade un icono de ficha azul para el jugador, y ficha roja para la CPU y cambia los colores de la aplicación para que sea más llamativo. **0,25 puntos**

(Llamada al método en el CustomMouseListener)

```
@Override
public void mouseClicked(MouseEvent e) {
    if(!((BotonCustom)e.getComponent()).isEstaOcupada())
    {
        ((BotonCustom)e.getComponent()).setEstaOcupada(true);
        padre.turno++;
        padre.ponerImagen((BotonCustom)e.getComponent(), padre.turno);
    }
}
```

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

(Método para poner las imágenes en los botones)

```
public void ponerImagen(BotonCustom boton, int turno) {
    String ruta = "";
    if(turno %2 == 0)
    {
        ruta = "azul.PNG";
    }
    else
    {
        ruta = "rojo.PNG";
    }

    ImageIcon imageIcon = new ImageIcon(getClass().getResource(ruta));
    Image image = imageIcon.getImage();
    Image newImg = image.getScaledInstance(100, 100, Image.SCALE_SMOOTH);
    imageIcon = new ImageIcon(newImg);


    if(imageIcon != null)
    {
        boton.setIcon(imageIcon);
    }
}
```

- c) La CPU elegirá la casilla de manera aleatoria entre las casillas que queden libres. **0,25 puntos**

(Elige una casilla libre y pone la imagen correspondiente)

```
public void elegirCasillaCPU() {
    boolean valido = false;
    int casilla = 0;
    if(turno %2 == 1)
    {
        do
        {
            casilla = (int) (Math.random()*9);
            if(!listaBotones.get(casilla).isEstaOcupada()) // si la casilla NO está ocupada...
            {
                valido = true;
            }
        }while(!valido);

        turno++;
        this.listaBotones.get(casilla).setEstaOcupada(true);
        ponerImagen(this.listaBotones.get(casilla), turno);
    }
}
```

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

- d) Existirá un contador de partidas ganadas por la CPU y otro para el Usuario que se reiniciará cada vez que se abra el programa. **0,25**

(Contenido del documento con el registro de las partidas)

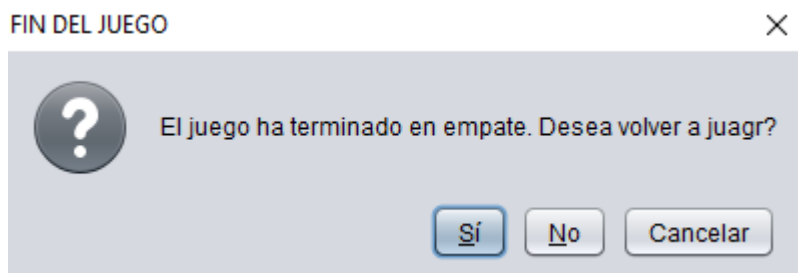
```
Partidas ganadas usuario:0
Partidas ganadas CPU:0|
```

- e) El juego empieza cuando el jugador pulsa un botón. Luego le toca a la CPU, luego al jugador.. el juego acaba en empate si se han pulsado los 9 botones, o bien si el jugador o la cpu han conseguido 3 en raya. Saldrá un JOptionPane preguntando si el usuario quiere volver a jugar. **2 puntos** (Se evalúa si el juego funciona correctamente)

(Final del método de poner las imágenes)


```
if(turno == 9)
{
    JOptionPane.showMessageDialog(this, "El juego ha terminado en empate", "FIN DEL JUEGO", JOptionPane.INFORMATION_MESSAGE);
}
```

(JOptionPane para indicar que el juego ha terminado en empate)



ANEXOS

Ejemplos para la Práctica 1

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

```

@Override
public Object getValueAt(int rowIndex, int columnIndex) {

    switch (columnIndex){
        case 0:
            return listAlumno.get(rowIndex).getNombre();
        case 1:
            return listAlumno.get(rowIndex).getCurso();
    }
    return null;
}

```

```

public List<Alumno> listaAlumnos = new ArrayList<>();

```

```


public List<Alumno> getListAlumnos() {
    return listaAlumnos;
}

```

```

public PantallaPrincipal() {
    initComponents();
    this.jTable1.setModel(new AlumnosTableModel(logicaNegocio.getListAlumnos()));
}

```


	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

En dialogo alta en la acción del botón:

Pasar el cliente pulsando el botón.

```
private void jButtonAltaActionPerformed(java.awt.event.ActionEvent evt) {
    String nombre= this.jTextFieldNombre.getText();
    String apellidos= this.jTextFieldApellidos.getText();
    Date fechaAlta=(Date)this.jSpinnerFechaAlta.getValue();
    String provincia= this.jComboBox1.getSelectedItem().toString();
    Cliente cliente=new Cliente(nombre,apellidos,fechaAlta,provincia);
    principal.aniadirCliente(cliente);

    dispose();      // TODO add your handling code here:
}
```

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]


Ejemplos práctica 2:

Cambiar color a un botón:

```
this.setBackground(Color.red);
```

Ejemplo JOptionPane

<pre>private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { // TODO add your handling code here: JOptionPane.showMessageDialog(this, "Un mensaje", "titulo", JOptionPane.WARNING_MESSAGE); }</pre>	Tipo de icono
Mensaje de info	

	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: [REDACTED]

Ejemplos práctica 3:

```

public class ventanaPrincipal extends javax.swing.JFrame {
    int actual=0;
    List<Alumno> lista;

    public class Alumno {

        String nombre;
        String curso;
        static int contador;
        int identificador;
    }
}

```

Ejemplos práctica 4:


```

]    public VentanaPrincipal() {
        initComponents();

        ImageIcon imageIcon = new ImageIcon(getClass().getResource("/imagenmina/minal.png")); //
        Image image = imageIcon.getImage(); // transform it
        Image newimg = image.getScaledInstance(50, 50, java.awt.Image.SCALE_SMOOTH); // scale
        imageIcon = new ImageIcon(newimg); // transform it back

        // Icon icon= new ImageIcon(getClass().getResource("/imagenmina/minal.png"));
        if(imageIcon !=null){
            jButton1.setIcon(imageIcon);
        }
    }
}

```


	CICLO FORMATIVO DE GRADO 2º DAM
	Módulo: Desarrollo de Interfaces
	Profesor: José Jarones Bueno
	Primer Trimestre
Fecha: 1-12-2022	Nombre: XXXXXXXXXX

