

## EXAMEN 3ª EVALUACIÓN – DAM Distancia

Nombre: \_\_\_\_\_ Fecha: 22/05/23

**Instrucciones generales:**

- Debes entregar una carpeta con tu nombre en la que guardes los proyectos comprimidos
- Debes exportar cada proyecto y guardarlos en una carpeta con tu nombre.
- Se valorará el código claro, estructurado y con nombres identificativos

**1. Ejercicio 1:** Vamos a crear una aplicación que cuenta con los siguientes requisitos

- Componente Ficha:

```
public class Ficha extends JButton {  
  
    int index;  
    Color colorFondo;  
  
    public Ficha() {  
  
    }  
    public Ficha(int index) {  
  
    }  
    public Ficha(int index, Color color) {  
  
    }  
    public int getIndex() {  
  
    }  
    public void setIndex(int index) {  
  
    }  
    public Color getColorFondo() {  
  
    }  
    public void setColorFondo(Color colorFondo) {  
  
    }  
    // Declaracion de los métodos de la clase Ficha  
    public void mostrarNumero() {  
  
    }  
    public void ocultarNumero() {  
  
    }  
  
    }
```

- Componente Botón.
- Clase App

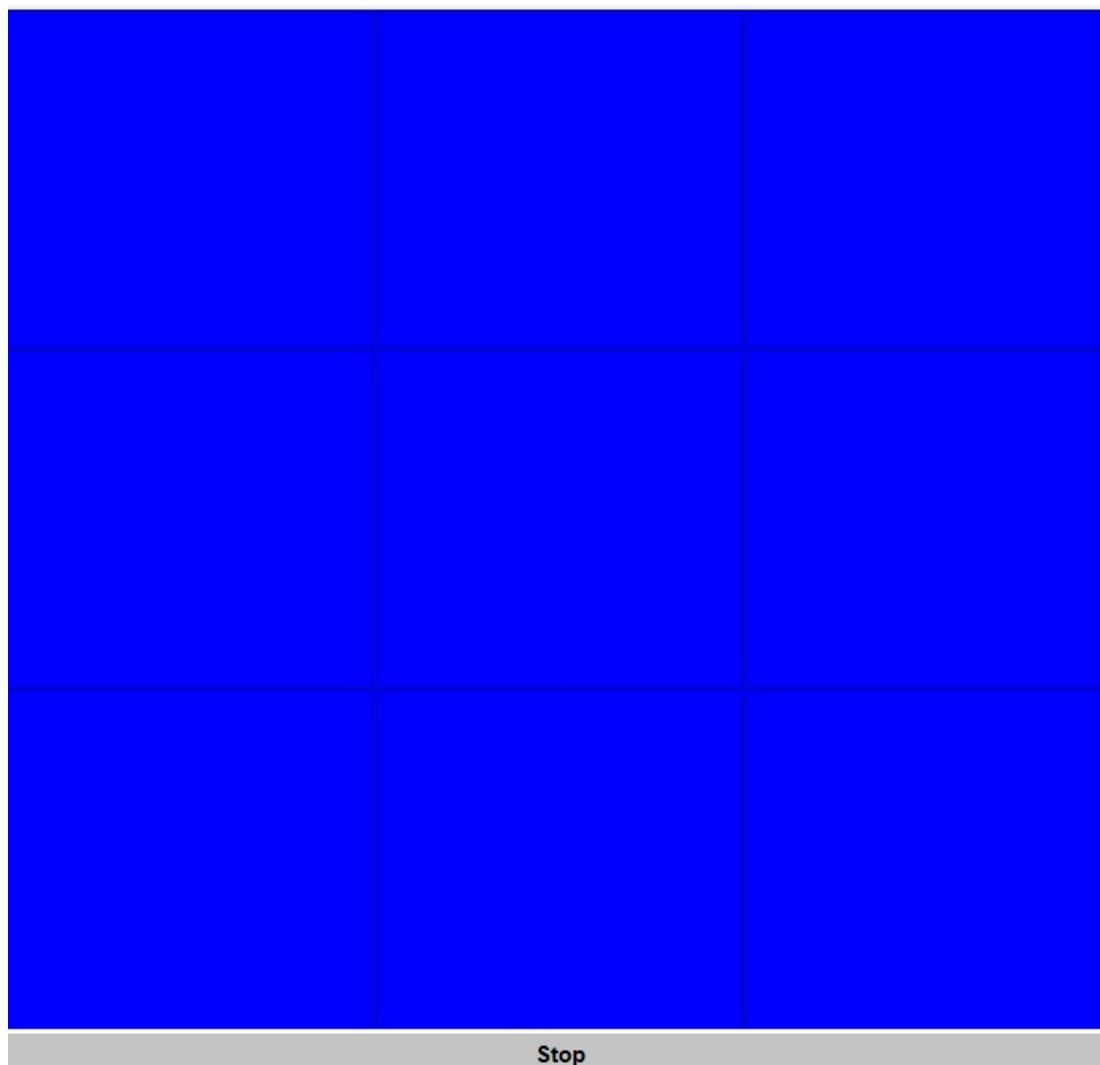
**Puntuación: (6 puntos)**

- **Crear la clase Ficha (1 punto)**
  - **Crear la clase Botón (1 punto)**
  - **Convertir la clase Ficha en un componente (0,25 puntos)**
  - **Convertir la clase Botón en un componente (0,25 puntos).**
  - **Diseño App (0,5 puntos)**
  - **Funcionalidad:**
    - **La funcionalidad se realiza accediendo a los métodos de cada componente.**
    - **Al pulsar el botón Start/Stop se modifica el aspecto y el texto que muestra. (1,5 puntos)**
      - **Start: Pone las fichas a la escucha del evento click. Al pulsar el botón Start aparece en cada ficha el valor de la propiedad index (el valor es un número correlativo, en función de la posición de cada ficha), y pone cada ficha a la escucha de que se produzca un evento click**
- Además modifica el aspecto del botón, cambiando el texto, el color de fondo y color de texto**



- **Stop:** Desaparece el texto de la ficha, vuelve a su color original y deja de estar a la escucha del evento click.

También cambia el aspecto del botón, tal y como se muestra en la imagen que aparece a continuación.

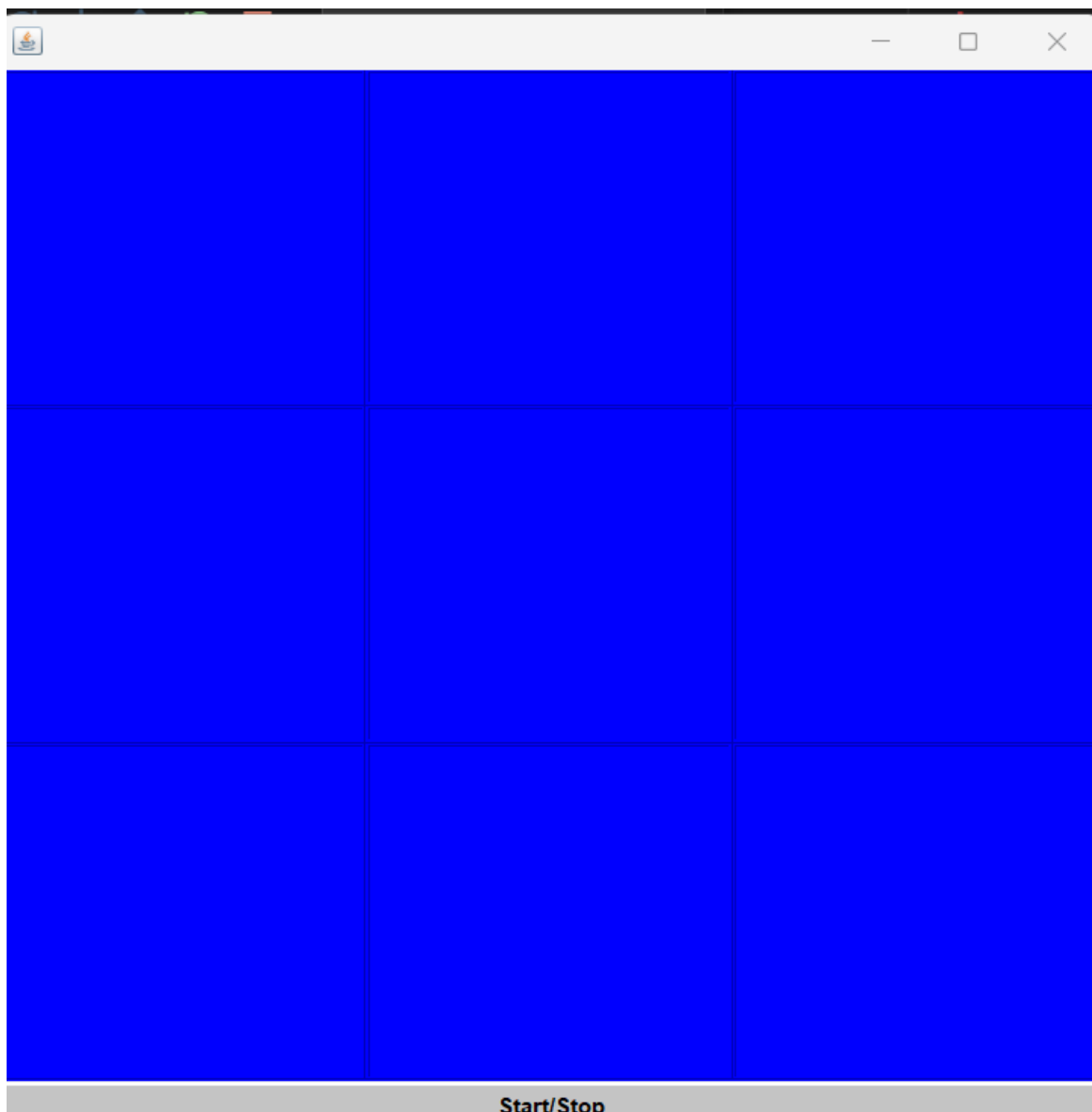


- **Funcionalidad de la ficha (1,5 puntos)**
  - Al hacer clic en la ficha, se modifica el valor de la propiedad index, cambiando el valor inicial con un valor aleatorio.  
Además se modifica el color de fondo de dicha fichas. El nuevo color de fondo va a estar asociado al nuevo valor de la propiedad index de la ficha. (Nota: Debes decidir que estructura de datos es la más idónea para almacenar los distintos colores)
  - Math.random()\*10:** Permite generar números aleatorios del 0 al 9.



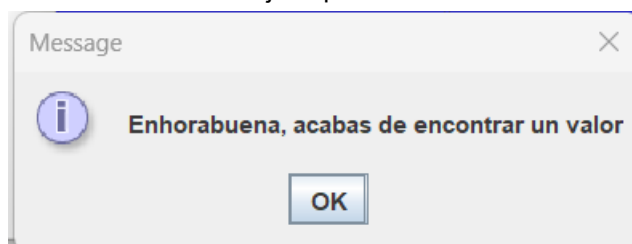
**2. Ejercicio 2:** Partiendo del ejercicio1 vamos a modificar la funcionalidad de la clase Ficha. En este caso dicha funcionalidad se va a resolver creando una interfaz y que el componente pueda reconecer el evento. Para que el componente pueda reconocer el evento y responder ante el tendremos que:

- Crear una clase para los eventos que se lancen.
- Definir una interfaz que represente el oyente (listener) asociado al evento. Debe incluir una operación para el procesamiento del evento.
- Por último podemos crear una clase manejador para dar funcionalidad al evento o implementar la interfaz creada en la clase Fichas.



### Funcionamiento:

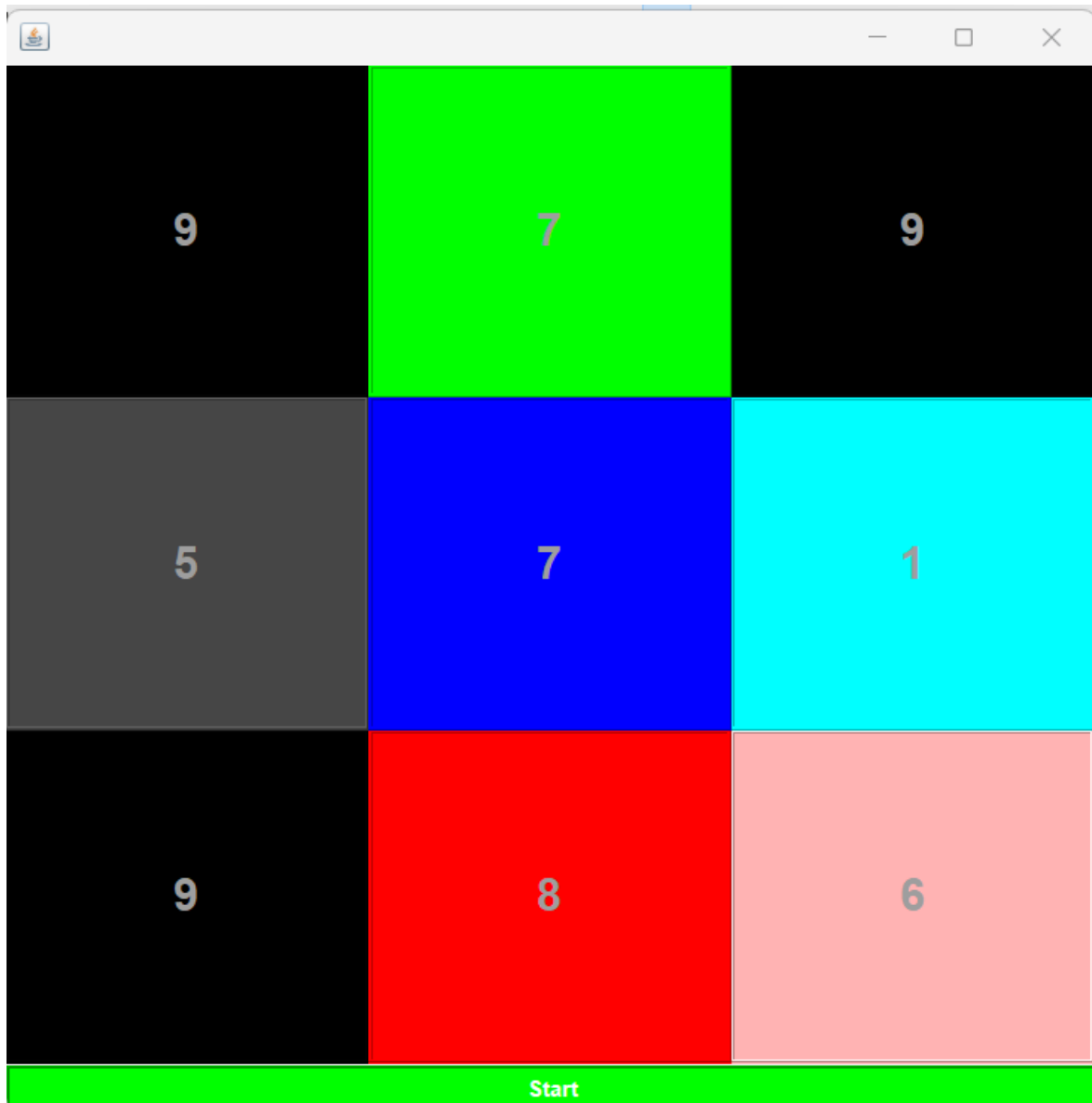
- Al pulsar el botón Start/Stop, se modifican las características del botón (aspecto externo) y se modifica el color de fondo de cada ficha y el valor de la propiedad index.
  - La funcionalidad se asocia al método **setIndex** de la clase Ficha, de tal forma que al pulsar el botón Start se genera un valor aleatorio  $\text{Math.random()} * 10$ . Este número es el que se le va a asignar a la propiedad index de la ficha. Si este valor es igual a la propiedad index que ya tenía la ficha se llama al método **Parar** y si es distinta se llama al método **Cambiar**.
  - **Método Parar:** Muestra un mensaje JOptionPane.



- **Método Cambiar:** Se modifica el valor de la propiedad index y el color de fondo asociado al índice.



- **Implementar la funcionalidad**
  - `public interface FichaEventListener`
  - `public class ManejadorFichaEventListener implements FichaEventListener{}`
  - `public class FichaEventObject extends EventObject{}`



- **Puntuación: (4 puntos)**
  - **Funcionamiento :2 puntos**
  - **Implementar la funcionalidad : 2 puntos**