



Programación Orientada a Objetos



GUI: Graphics User Interface



AWT: Abstract Windows Toolkit



Componentes

Button

TextField

Label

...

Contenedores

Frame

Panel

Dialog

...



Layouts

FlowLayout

Pone los componentes en fila

BorderLayout

Divide en Norte, Sur, Este, Oeste y Centro

Grid Layout

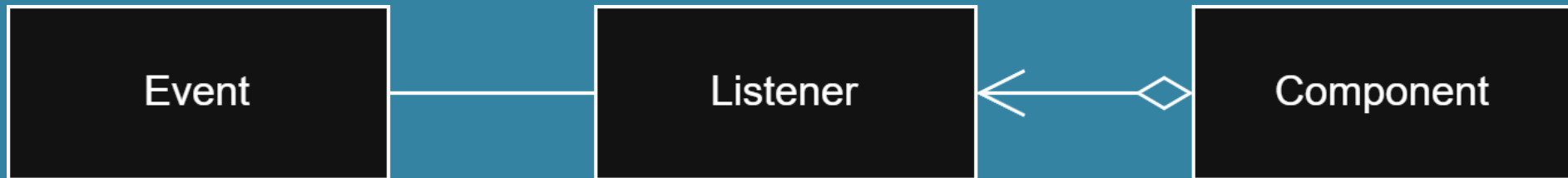
Crea una cuadrícula o tabla

...

...



Eventos





try-catch

```
try {  
    int resultado = 10 / 0;  
} catch (ArithmeticException e) {  
    System.out.println("Error: división por cero");  
}
```



Throw vs Throws

Throws

Indica que Excepciones o Errores puede tirar un método

Throw

Lanza la Excepción o Error Correspondiente



Archivos

```
File archivo = new File("datos.txt");
```




Manipular archivos

Clase	Tipo	Uso
InputStream	Byte (8bit)	Leer binarios
OutputStream	Byte (8bit)	Escribir binarios
Reader	Texto (16bit)	Leer archivos de texto
Writer	Texto (16bit)	Escribir archivos de texto
Buffered*	Texto (16bit)	Optimiza acceso al disco



JSON (JavaScript Object Notation)

```
{  
  «objeto»: {  
    «atributo» : < booleano | número | «texto» >,  
    «lista» : [ elemento1, elemento2, ...],  
    «objeto anidado»: {...}  
  }  
}
```



XML (eXtensible Markup Language)

```
<etiqueta 1>  
  <etiqueta 2> contenido </etiqueta 2>  
  <etiqueta 3>  
    <etiqueta 4> contenido </etiqueta 4>  
  </etiqueta 3>  
</etiqueta 1>
```



JSON vs XML

JSON

```
{  
  "juego": {  
    "nombre": "SuperJava Bros",  
    "ejecutable": "juego.exe",  
    "imagen": "portada.png",  
    "fullscreen": true  
  }  
}
```

XML

```
<juego>  
  <nombre>SuperJava Bros</nombre>  
  <ejecutable>juego.exe</ejecutable>  
  <imagen>portada.png</imagen>  
  <fullscreen>true</fullscreen>  
</juego>
```



JSON en Java

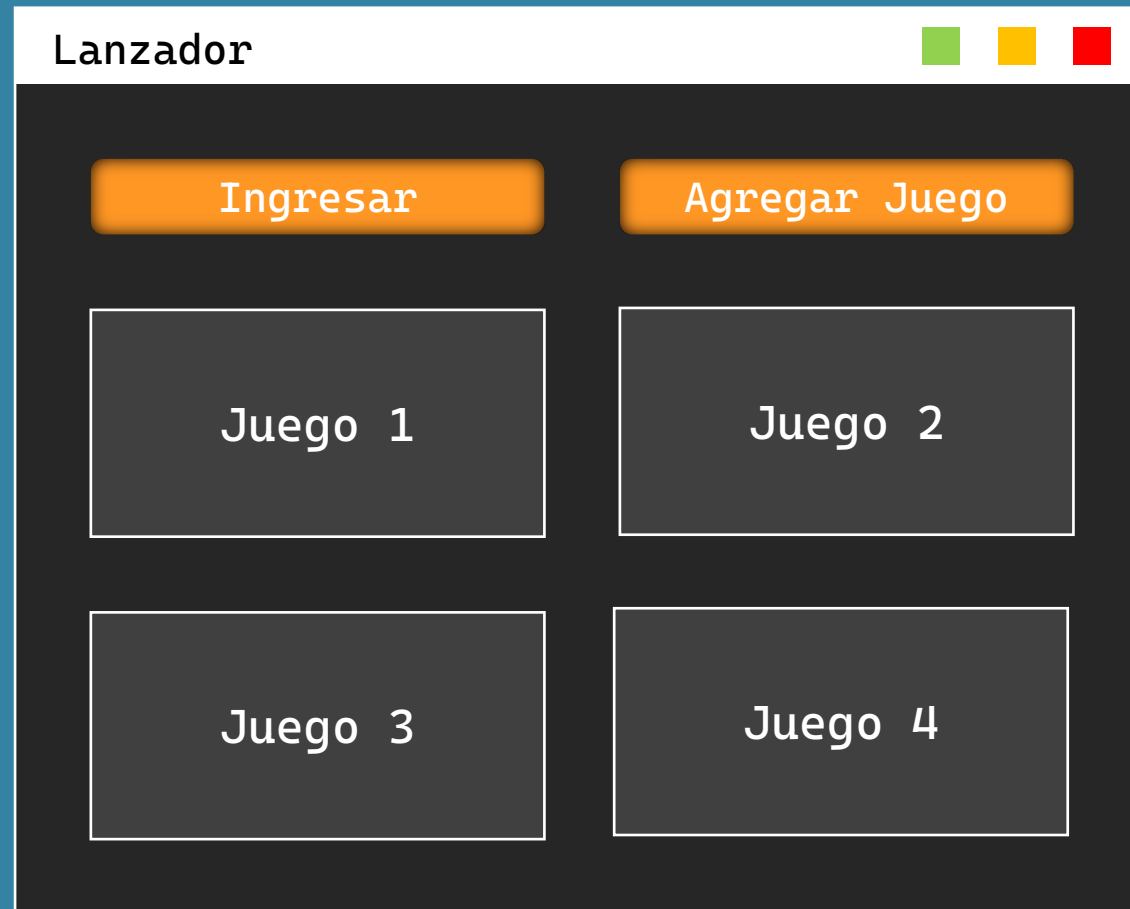
```
import org.json.JSONObject
```

```
...
```

```
JSONObject obj = new JSONObject();  
obj.put("nombre", campoNombre.getText());  
Files.writeString(Path.of("usuario.json"), obj.toString(2));
```

Tarea

1. Hacer una interfaz gráfica mínima que contenga al menos 2 botones.



Tarea

2. Hacer las siguientes ventanas emergentes.
 - a. Respetar los campos de la imagen.
 - b. Al presionar el botón guardar datos en formato JSON

Iniciar Sesión

Usuario

Contraseña

Ingresar

Agregar Juego

Juego

Ejecutable

Descripción

Guardar

Machete AWT

```
frame.setSize(<ancho>, <alto>);  
  
frame.setLayout(<Layout Manager>);  
  
TextField campo = new TextField(<ancho>);  
  
TextArea area = new TextArea(<alto>, <ancho>);  
  
String respuesta = campo.getText();  
  
Button boton = new Button(<texto>);  
  
boton.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
        <tu código>  
    }  
});
```


Machete JSON

```
import org.json.JSONObject;

JSONObject obj = new JSONObject();

obj.put(<clave>, <valor>);

Files.writeString(Path.of(<dirección archivo>), <texto>);

obj.toString(<indentado>);
```