



**Camaleón**  
Human Computer Interaction  
Research Group

# Construyendo GUI- Imágenes

PROGRAMACIÓN INTERACTIVA

ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN



**Universidad del Valle**

## *Qué debes saber al finalizar la clase!!*

---

- ✓ *Usar varios JFrame (activarlos – desactivarlos).*
- ✓ *Usar clases BufferedImage, ImageIO, File*
- ✓ *Comprender el concepto de excepciones.*
- ✓ *Aplicar la personalización de la ventana (JFrame)*
- ✓ *Aplicar diferentes tipos de escuchas (ActionListener y MouseListener).*
- ✓ *Extender componentes propios de java (JButton)*
- ✓ *Aplicar técnica de reuso.*

# Construyendo GUI

✓ **Reto: Rompecocos**



# Construyendo GUI



## ✓ *Reto: Rompecocos*



# Construyendo GUI



## ✓ *Reto: Rompecocos – análisis (Tarjetas CRC)*

| Ficha<br><<JButton>>                  |  |
|---------------------------------------|--|
| Conocer posición que ocupa            |  |
| Conocer la imagen que contiene        |  |
| Conocer id de la imagen en orden      |  |
| Intercambiar contenido con otra ficha |  |
| Indicar si tiene o no imagen          |  |
| Mostrar la imagen                     |  |

| Rompecocos<br><<Jframe>>                                                |       |
|-------------------------------------------------------------------------|-------|
| Visualizar el rompecabezas con piezas revueltas                         | Ficha |
| Permitir al usuario mover el contenido de la pieza al espacio en blanco | Ayuda |
| Visualizar la ficha faltante una vez se ha armado el rompecabezas       |       |
| Permitir al usuario revolver las fichas del rompecabezas                |       |
| Permitir al usuario acceder a la ayuda                                  |       |
| Permitir al usuario cerrar la aplicación                                |       |



# Construyendo GUI



✓ **Reto: Rompecocos – análisis (Tarjetas CRC)**

| Ayuda<br><<Jframe>>                                                                                        |            |
|------------------------------------------------------------------------------------------------------------|------------|
| Mostrar la imagen a armar respetando las dimensiones que se manejan la imagen que se muestra en Rompecocos | Rompecocos |
| Inhabilitar la ventana Rompecocos, mientras Ayuda esté activa.                                             |            |

| PrincipalRompecocos |            |
|---------------------|------------|
| main                | Rompecocos |

# Construyendo GUI - Imagenes



## ✓ *BufferedImage*

Coordenadas (x,y) en pixeles  
(0,0)



- Representación de la Imagen en Memoria
- Permite manipulación de los datos de la imagen.  
(*getSubimage(..)*)

# Construyendo GUI - Imagenes



## ✓ ImagenIO

**Rompecocos.jpg**



- Acceder o leer un archivo de imagen (jpg, png, etc.)

**BufferedImage image = ImageIO.read(.....)**



# Construyendo GUI - Imagenes



## ✓ File



Rompecocos.jpg



Un conjunto de bits almacenados en un dispositivo y accesible a través de un *camino de acceso* que lo identifica

```
BufferedImage image = ImageIO.read(new File("src/imagenes/Rompecocos.jpg"))
```

Qué sucede si no se encuentra el archivo?



# Construyendo GUI - Imágenes

---



## ✓ Excepciones

- Una **Excepción** -> Situación anómala en un bloque de código
  - ✓ Error en ejecución
  - ✓ Cierre abrupto del Programa
- ¿Qué significa el manejo de excepciones?

# *Construyendo GUI - Imagenes*

---



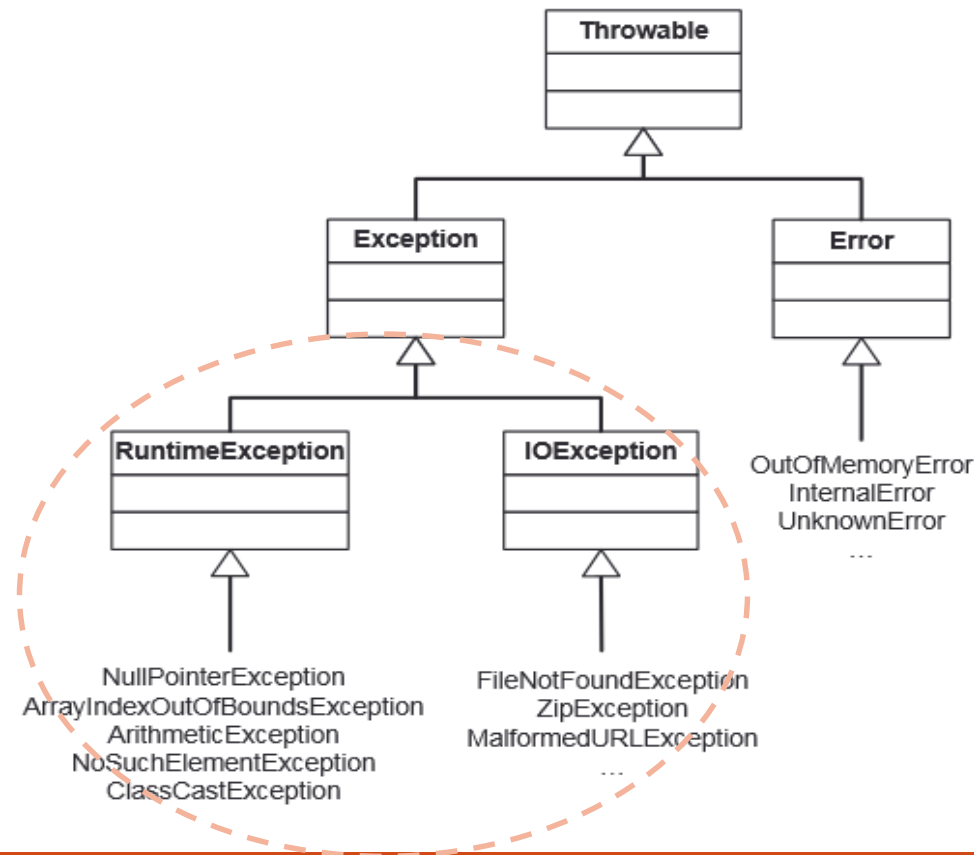
## ✓ *Excepciones – Errores comunes*

- *Índices fuera de rango*
- *Desbordamiento aritmético*
- *División entre cero*
- *Acceso a objetos no creados*
- *Interrupción de un subprocesso*
- *Asignación inválida de memoria (falla en memoria)*
- *Acceso a archivos que no existen en el camino indicado*

# Construyendo GUI - Imágenes



## ✓ Excepciones – jerarquía Java



# Construyendo GUI - Imágenes



## ✓ Excepciones

| Tipo Excepción                 | Descripción                                                                        |
|--------------------------------|------------------------------------------------------------------------------------|
| ArithmeticException            | Errores Matemáticos, como división por cero                                        |
| ArrayIndexOutOfBoundsException | Un programa tratando de almacenar un dato en una posición no válidos de un arreglo |
| FileNotFoundException          | Un intento de acceder a un archivo que no existe                                   |
| IOException                    | Fallas de entrada/salida, tal como la inhabilidad de leer desde un archivo         |
| NullPointerException           | Referencia a un objeto NULL                                                        |
| NumberFormatException          | Una conversión fallida entre String y números                                      |



# Construyendo GUI - Imagenes



## ✓ Excepciones – Verificadas vs No Verificadas

### Checked

Excepciones que se deben manejar durante el proceso de compilación. Si no se manejan sale un error de compilación:

- IOException
- SQLException
- DataAccessException
- ClassNotFoundException

### Unchecked

Excepciones que no se verifican en el proceso de compilación. Generalmente, son errores del programador y se lanzan en tiempo de ejecución

- NullPointerException
- ArrayIndexOutOfBoundsException
- ArithmeticException

# Construyendo GUI - Imagenes



## ✓ Excepciones – Ejemplo

```
public static double division(int x, int y){  
    return x/y;  
  
}
```



```
Numerador: 5  
Denominador: 0  
Exception in thread "main" java.lang.ArithmeticException: / by zero  
    at lab1pi.ExceptionManeger.division(ExceptionManeger.java:18)  
    at lab1pi.ExceptionManeger.main(ExceptionManeger.java:29)  
Java Result: 1
```

# Construyendo GUI - Imágenes



## ✓ Excepciones – Ejemplo - Manejo

```
try {  
  
    // bloque de código en el que gestionamos la excepción  
}  
  
catch ( tipo_excepción1 objeto ) {  
  
}  
  
catch ( tipo_excepción2 objeto ) {  
  
}
```

```
try{
    System.out.print( "Numerador: " );
    int numerador= scanner.nextInt();

    System.out.print( "Denominador: " );
    int denominador= scanner.nextInt();
    double divi= division(numerador, denominador);
    System.out.println("Division: " + divi);

}

catch(InputMismatchException ep){

    System.out.println("Usted debe ingresar números enteros.\n");
} //end catch

catch ( ArithmeticException arithmeticException ){

    System.err.printf( "\nException: %s\n", arithmeticException );
    System.out.println(
        "No se puede dividir por cero.\n" );
} // end catch
```



```
continueLoop=true;

do{
    try{
        System.out.print( "Numerador: " );
        int numerador= scanner.nextInt();

        System.out.print( "Denominador: " );
        int denominador= scanner.nextInt();
        double divi= division(numerador, denominador);

        continueLoop = false;
    }

    catch ( ArithmeticException arithmeticException ){

        System.err.printf( "\nException: %s\n", arithmeticException );
        System.out.println(
            "No se puede dividir por cero.\n" );
    } // end catch
}while (continueLoop );
```





# Construyendo GUI - Imágenes

---



## ✓ Excepciones – Métodos que lanzan excepciones

```
public static double division(int x, int y) throws ArithmeticException{  
  
    if (y==0){  
        throw new ArithmeticException("Division por cero");  
    }  
  
    return x/y;  
  
}
```

---

# Construyendo GUI



## ✓ Reto: Rompecocos – análisis (Tarjetas CRC)

| Ficha<br><<JButton>>                  |  |
|---------------------------------------|--|
| Conocer posición que ocupa            |  |
| Conocer la imagen que contiene        |  |
| Conocer id de la imagen en orden      |  |
| Intercambiar contenido con otra ficha |  |
| Indicar si tiene o no imagen          |  |
| Mostrar la imagen                     |  |

| Rompecocos<br><<Jframe>>                                                |       |
|-------------------------------------------------------------------------|-------|
| Visualizar el rompecabezas con piezas revueltas                         | Ficha |
| Permitir al usuario mover el contenido de la pieza al espacio en blanco | Ayuda |
| Visualizar la ficha faltante una vez se ha armado el rompecabezas       |       |
| Permitir al usuario revolver las fichas del rompecabezas                |       |
| Permitir al usuario acceder a la ayuda                                  |       |
| Permitir al usuario cerrar la aplicación                                |       |

# Construyendo GUI



✓ **Reto: Rompecocos – análisis (Tarjetas CRC)**

| Ayuda<br><<Jframe>>                                                                                        |            |
|------------------------------------------------------------------------------------------------------------|------------|
| Mostrar la imagen a armar respetando las dimensiones que se manejan la imagen que se muestra en Rompecocos | Rompecocos |
| Inhabilitar la ventana Rompecocos, mientras Ayuda esté activa.                                             |            |

| PrincipalRompecocos |            |
|---------------------|------------|
| main                | Rompecocos |