

Java para Cibernética y Computación

Colegio de Ciencias y Humanidades – Plantel Oriente



Instructores:

!!!**TODOS LOS PROFESORES INSCRITOS EN EL CURSO!!!**

Día 4



Herencia - Super y This

- › La palabra **super** se utiliza para hacer referencia al constructor de la clase padre. En Java no existe la “herencia múltiple”, por lo tanto, para cualquier clase siempre se tendrá un único padre. Además, se debe tener muy en cuenta que los constructores no se heredan.
- › La palabra **this** se utiliza para hacer referencia a los constructores de la misma clase. Se pueden invocar tanto atributos como métodos.



Manejo de excepciones

- › El control de excepciones sirve para controlar los errores que se pueden presentar en tiempo de ejecución de un programa. Las sentencias que se utilizan son las siguientes:
- › Try
- › Catch
- › Finally



Manejo de excepciones - Sintaxis

```
try {  
    System.out.println("bloque de código donde pudiera  
    encontrarse un error");  
}  
catch (Exception e) {  
    System.out.println("bloque de código donde se trata  
    el problema");  
}  
finally {  
    System.out.println("bloque de código ejecutado  
    siempre");  
}
```



Jlabel - Etiquetas

› El JLabel sólo se usa para mostrar texto en una línea dentro de un contenedor.

› Instanciar un JLabel

```
JLabel label = new JLabel();
```

› Poner texto en el JLabel

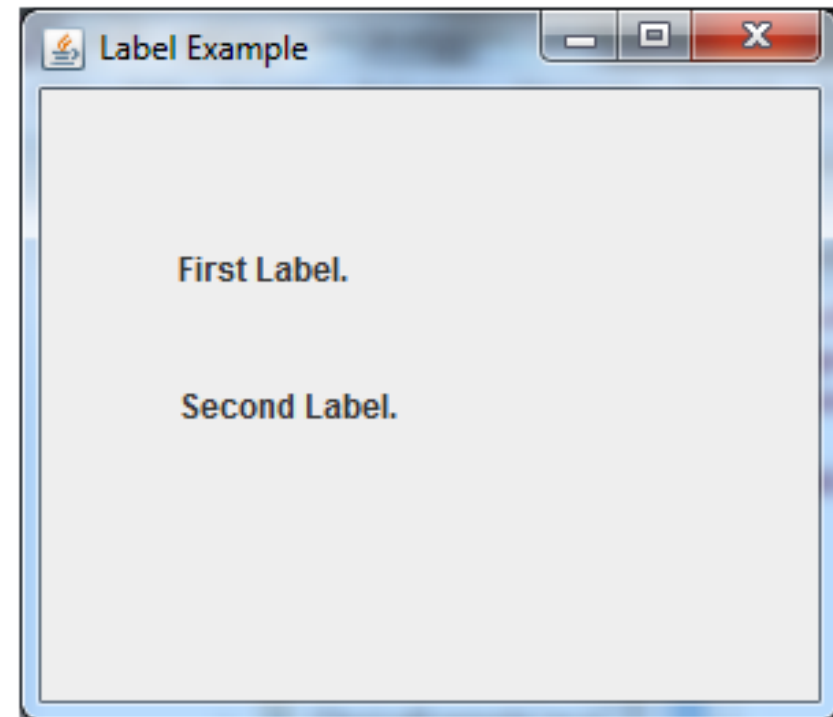
```
label.setText("Hola");
```

› Leer texto en el JLabel

```
String texto = label.getText();
```

› Limpiar el JLabel

```
label.setText("");
```





JTextField – Cajas de texto

› Instanciar un JTextField

```
JTextField textField = new JTextField();
```

› Poner texto en el *JTextField*

```
textField.setText("Hola");
```

› Leer texto en el *JTextField*

```
String texto = textField.getText();
```

› Limpiar el JTextField

```
textField.setText("");
```

JTextField



JTextField – Cajas de texto

JTextField

- › El JTextField sólo admite y devuelve String, por lo que si queremos ingresar u obtener números, debemos hacer una conversión.

- › Poner números en el *JTextField*

```
int valor = 33;  
textField.setText(Integer.toString(valor));
```

- › Leer números en el *JTextField*

```
int valor;  
String texto = textField.getText();  
valor = Integer.parseInt(texto)
```



JButton - Botones



- › Un objeto de control JButton permite dibujar en el formulario un objeto que contiene un proceso a ejecutar.

- › Propiedades más usadas:
 - **Text**: Contiene el valor o dato introducido en el cuadro de texto.
 - **Font**: Permite establecer el tipo de letra del texto en la caja.
 - **Enabled**: Para habilitar o inhabilitar el uso del objeto de control.

- › Evento más usado:
 - **ActionPerformed**: Este evento se lleva a cabo cuando el usuario da click sobre el objeto de control JButton.