**Administrative**

**Today’s session**

Threads

Graphical user interface (GUI)

Swing API, part 1 of 2

Final exam review

**Session Topics**

**Threads**

● A **thread** is an independent execution unit managed by the operating system.

● A **process** consists of one or more threads.

● A Java application may start, manage, and end one or more simultaneous threads.

● There are two ways to initiate a thread from a Java application:

✓ Extend the Thread class

Class extended

public class SomeThreadClass1 extends Thread

{

…

public void run()

{

…

}

}

Class instantiated

SomeThreadClass1 t = new SomeThreadClass1();

t.start();

✓ Implement the Runnable interface **(preferred method)**

Interface implemented

public class SomeThreadClass2 implements Runnable

{

…

public void run()

{

…

}

}

Class instantiated

SomeThreadClass2 t = new SomeThreadClass2();

new Thread(t).start();

● See **Threads** sample application on Blackboard.

**Graphical user interface (GUI)**

● A **console application** uses only a screen and keyboard.

● A **GUI** uses a screen, keyboard, and mouse.

● A GUI is made up of one or more **controls** including:

✓ Labels

✓ Text boxes

✓ Buttons

✓ Radio buttons

✓ Check boxes

✓ List boxes

✓ Combo boxes

✓ Scroll bars

✓ Menus

✓ Icons and other graphics

✓ Windows

**Swing API, part 1 of 2**

● Java includes the Swing API for creating a GUI application.

● The Swing API contains several packages including:

javax.swing.text

javax.swing

javax.swing.text.html

javax.swing.border

javax.swing.text.html.parser

javax.swing.colorchooser

javax.swing.text.rtf

javax.swing.event

javax.swing.tree

javax.swing.filechooser

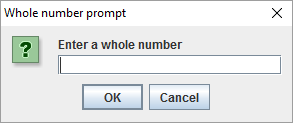
javax.swing.table

javax.swing.undo

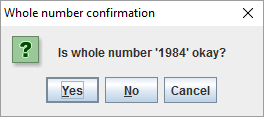
**Dialogs**

● The Swing API includes dialogs in **class javax.swing.JOptionPane**:

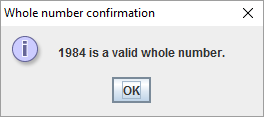
✓ **JOptionPane.showInputDialog** prompts the user for a value and returns it as a string:



✓ **JOptionPane.showConfirmDialog** asks the user a question and returns an integer code corresponding to the response:



✓ **JOptionPane.showMessageDialog** shows a message to the user:



● See **GUI dialog boxes** sample application on Blackboard.

**Controls**

● Package **javax.swing** includes the following controls for building a screen:

✓ **JFrame** is the container for the application. Controls are placed in the container with method **add**.

✓ **JPanel** is a container for controls. Controls are placed in the container with method **add**. One or more JPanels may be placed within a JFrame.

✓ **JLabel** is textual information to the user.

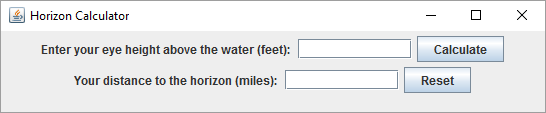
✓ **JTextField** provides an area for user entry.

✓ **JButton** triggers some action.

**JLabel / label**

**JButton / button**

**JTextField / text box**



**JPanel / gray box**

**JFrame / entire box**

**Listeners**

● The Swing API includes listeners for enabling an application to respond to a user action:

✓ Methods **addActionListener** and **actionPerformed** may be connected to:

→ A text box to respond to text entered into box.

→ A button to respond to a button click.

● See **GUI labels, text boxes, buttons** sample application on Blackboard.

**Final exam review topics**

● Assigned today.

● Available on Blackboard.

● Review in one week.