11. Java Swing
Individual Project 1

Minhaeng Lee
Thread

• Use multi flow of probgam
• When?
  – To utilize 100% of multi-core cpu
  – To make interactive program
  – Ex) countdown while waiting key press

• Lets see some examples
  – Simple thread
  – Thread with detail (number)
  – Thread join
  – Thread by using runnable interface
Drawing ball and key input

• By implement **KeyListener**
• Once implement KeyListener, you must implement related methods.
  – `keyTyped`, `keyPressed`, and `keyReleased`
• Using **KeyPressed** Methods
  – Cannot be called directly
  – ONLY called via KeyListener interface
  – Using If statement to each key input
public void keyPressed(KeyEvent e) {
    int keyCode = e.getKeyCode();
    System.out.println("pressed: "+keyCode);
    synchronized (keyLock) {
        keysDown.add(e.getKeyCode());
    }
    if (e.getKeyCode() == KeyEvent.VK_RIGHT) {
        // Right arrow key code
        rx += 0.1;
    } else if (e.getKeyCode() == KeyEvent.VK_LEFT) {
        // Left arrow key code
        rx -= 0.1;
    } else if (e.getKeyCode() == KeyEvent.VK_UP) {
        // Up arrow key code
        ry += 0.1;
    } else if (e.getKeyCode() == KeyEvent.VK_DOWN) {
        // Down arrow key code
        ry -= 0.1;
    }
}
Drawing ball and mouse input

• Similar to keyinput

• Using MouseListener
  – For mouse click/press/unpressed
  – `mouseClicked`, `mouseEntered`, `mouseExited`, `mousePressed`, `mouseReleased`, and `mouseDragged`

• Using MouseMotionListener
  – For mouse movement
  – `mouseDragged` and `mouseMoved`

• Let’s apply mouse input event on drawing ball
  – Make object follow mouse pointer!
N-Dimensional Array

- We know how to allocate 1-dim array
  - `int[] a = new int[10];`

- What if dim is greater than 1 like 2 or more?
- Attach more bracket to imply multi dimension
  - `int[][] a2 = new int[10][10];`
  - `int[][][] a3 = new int[10][10][10];`
Sleep!

- Need to make time based event
- Use `Thread.sleep(<time_val>)` method with try-catch statement
Sleep example

```java
import java.util.Date;
public class SleepTest {

    public static void main(String[] args) {
        Date date = new Date(); // To measure current time
        System.out.println("before..");
        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        date = new Date(); // To measure current time
        System.out.println("End! "+date.getTime());
    }
}
```
Now, ready to make your program

• Design GUI using scratch paper (or ppt)
• Create image file (or download)
• Create game map using array
• Next?
Java Swing?

- Java GUI building library
- Not very famous for standalone application
  - Slow (because of JVM)
  - Current trends move to web
- Still useful and worth to try
- Logics are same
- http://docs.oracle.com/javase/tutorial/uiswing/
Necessary Components

- JK, JW
  - Drawing
- JM
  - CoverFlow
- Aiden
  - Miglayout
- Sien
  - Flipping animation?
Why others’ source?

- Difficult and inefficient to make from scratch
Basic JComponents

• JFrame
  – Most Basic Component to make window
  – One per each window
  – Window as is

• JPanel
  – Common component inside of a window
  – Multiple in each window

• JButton, JLabel, JTextArea ... etc.
Components
Your first Swing Example

Run **HelloWorldSwing.java** in ExampleSwing project

Does your own look like this?

```java
import javax.swing.*;

public class HelloWorldSwing {
    /**
     * Create the GUI and show it. For thread safety,
     * this method should be invoked from the
     * event-dispatching thread.
     */
    private static void createAndShowGUI() {
        // Create and set up the window.
        JFrame frame = new JFrame("HelloWorldSwing");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // Add the ubiquitous "Hello World" label.
        JLabel label = new JLabel("Hello World");
        frame.getContentPane().add(label);

        // Display the window.
        frame.pack();
        frame.setVisible(true);
    }

    public static void main(String[] args) {
        // Schedule a job for the event-dispatching thread:
        // creating and showing this application's GUI.
        javax.swing.SwingUtilities.invokeLater(new Runnable() {
            public void run() {
                createAndShowGUI();
            }
        });
    }
}
```
File Chooser

Run FileChooserDemo.java in ExampleSwing project
Run *BorderDemo.java*
   in ExampleSwing project
Text, Password, ....more and more

Run `TextDemo.java`
   in ExampleSwing project

Run `PasswordDemo.java`
   in ExampleSwing project

What do you need?
Check here great examples:
http://docs.oracle.com/javase/tutorial/uiswing/examples/components/index.html
LookAndFeel

- Skin for java GUI
- Easy to apply
- A number of resources : ex)
  http://www.jyloo.com/synthetica/themes/

```java
try {
    UIManager.setLookAndFeel(new SubstanceBusinessBlueSteelLookAndFeel());
} catch (final UnsupportedLookAndFeelException e1) {
}
```
Source management tools

• Famous tools
  – CVS
  – SubVersion
  – GIT
  – Etc.

• Necessary for source management
  – Version control
  – Cooperation
  – Useful to show your work to public!
Using External Library for Eclipse

• Adding Library path to build path
• When library import problem happened!

```java
import javax.swing.*;
import javax.swing.filechooser.*;
import org.pushingpixels.substance.api.skin.SubstanceBusinessBlueSteelLookAndFeel;
```

Example Swing tree view:
- `src`
  - `components`
  - `components.images`
  - `JRE System Library [JavaSE-1.8]`
- `lib`
  - `substance-6.1.jar`
  - `trident.jar`

Java Problems (55 items):
- SubstanceBusinessBlueSteelLookAndFeel cannot be resolved to a type
- The import org.pushingpixels cannot be resolved
- The import org.pushingpixels cannot be resolved
Using External Library for Eclipse (1/3)

Right click on the project
-> Properties

Click “Add JARs”
Using External Library for Eclipse (2/3)

Select Required “jar”s
Then click ok

Check the jar list and then Click ok
Using External Library for Eclipse (3/3)

Successfully add and error fixed!
How can I get required library?

- Mostly on the web
- Use Google
- Read publisher’s document carefully
- Search using package name

- In this case
  - Keyword: “org. pushingpixels.substance jar”
Individual Project : Tetris

• **Basic**
  – Get Block (and other) images
  – Generation 2D Map
  – Key input
  – **Thread based Timing Control**
    • Frame control
  – Game, Score, Next zone

• **Advanced**
  – Continue from previous
  – Item
  – Auto Play
Individual Project : Sneak Game

**Basic**
- Get Sneak Parts images
- Generate 2D Map
- Key input
- **Thread based Time Control**
  - Frame control
  - Game, Score zone

**Advanced**
- Continue from previous
- Item
- Auto Play
Individual Project : Music Player

• Basic
  – Get required button images (play, stop ... etc.)
  – CoverFlow (template provided)
    • ExampleGUI – CoverFlowDemo.java
  – File Scanning
  – File load/save
  – Play List Management
  – Music Play/Stop

• Advanced
  – Music Equalize
Individual Project : Calendar

• Basic
  – Date control
  – JComponents
  – Layout using MigLayout (Example Provided)
    • ExampleGUI – DashboardDemo.java, SwingDemo.java
  – Event Add/Delete/Edit

• Advanced
  – Sync to the web
Homework

• Record Your Progress during week
  – What Problem you have
  – What have you done
  – No progress, nothing to learn

• Read Layout Source
  – ExampleGUI – DashboardDemo.java
References

• http://docs.oracle.com/javase/tutorial/uiswing/