Nick Hippen

Intro to Game Programming

Program #1

**edu.unomaha.nhippen.paint**

This package contains all of the base classes as well as input/button classes.

Paint.java

The starting point of the application; contains the main method.

CustomButton.java

Abstract class that defines a button. Used to create the buttons on the toolbar for both tools and colors. The performAction method is implemented inside PaintApp.java.

KeyboardInput.java

Unchanged book code

PaintApp.java

This is the heart of the application. Most of the code here is book code, but there are a few changes inside. The main changes are the implementation of the buttons, the input redirection (most of the logic actual is handled outside of this class), and render redirection (except for cursor/toolbar).

RelativeMouseInput.java

Unchanged book code

ToolButton.java

An extension of CustomButton.java. The only difference with this class is that it also holds a tool object reference.

**edu.unomaha.nhippen.paint.shapes**

This package contains the shapes that can be drawn.

Shape.java

An abstract class which describes a shape to be drawn on the canvas. It stores color and previewing information. This class is the super-class of everything in this package besides the Drawable interface.

Drawable.java

An interface which describes an object as drawable with a graphics object.

FreeLine.java

A shape which represents a free-drawn line. Contains a list of points to render lines.

Line.java

A shape which simply represents a line. Contains two points which describe the line.

PolyLine.java

A shape which represents a poly line. Contains a list of points to draw lines between.

Rectangle.java

A shape which represents a rectangle. Contains two points which describe the rectangle.

**edu.unomaha.nhippen.paint.tools**

This package contains the tools that can be used to draw shapes.

FreeLineTool.java

A tool which is used to draw a FreeLine shape.

LineTool .java

A tool which is used to draw a Line shape.

PolyLineTool.java

A tool which is used to draw a PolyLine shape.

RectangleTool.java

A tool which is used to draw a Rectangle shape.

Tool.java

An abstract class which describes a tool that will be used to create shapes. It contains two abstract methods. The method “processInput()” is used to handle input (the logic needed to generate a given shape). The “reset()” method is used to reset the tool back to its start state. It statically contains references to the “sub-tools”. Also contains an exception class for handling tool exceptions.

ToolClick.java

A simple object that holds various useful properties for a tool click event. This object is passed through the “processInput()” method of tools.