

Transformations Application

Packages

Transformations

The top-level transformations package contains the entry point to the application as well as the sub-packages that make up the bulk of the application.

Logic

The logic package contains the classes responsible for managing the state of the application. For this simple example, the single State class is needed.

Render

The render package contains the classes responsible for rendering output in the main canvas. It consists of the WorldCanvas class that performs the buffering/rendering as well as a couple of helper classes that help represent render-able objects.

Util

The util package contains helper classes and common utilities used to simplify or abstract away complex portions of the application. This includes vector/matrix representation and listening for user input.

Classes

Transformations

- The Transformations class serves as the entry point to the application. It is a JFrame that wires together the application window and canvas and then creates a separate thread in which to execute the game loop and rendering.

Logic

- The State class manages the state of the application; a singleton pattern was utilized to allow the game state to be accessed from any class at any time. For this application, the game state consists of 3 entities whose state is maintained by this class. All entities are updated when processInput is called.

Render

- The WorldCanvas class provides a buffered, draw-able surface on which to render the game. The renderFrame method is responsible for getting the graphics context for the current drawing buffer and passing it to the internal render method. Checks are in place to redo the drawing if the buffered image is lost before being shown. The render method overrides the default cursor by replacing it with a blank image and delegates rendering to the vector object instances' render methods.
- The Drawable interface is used to represent a generic entity that has the ability to update its own internal state and render itself using a provided graphics instance. This will allow us to interact with entities using multiple different graphics strategies using the common interface.

- The VectorObject class represents an entity that is rendered using vector graphics. The entity is stored as a collection of vectors with a location, color, x-scale, y-scale, and rotation. The updateWorld method is responsible for generating a worldMatrix from these attributes that is used to transform each of the vectors to place it correctly in relation to the coordinate grid. The render method iterates over the transformed vectors, drawing a line between each adjacent pair.

Util

- The KeyboardInput class is a KeyAdapter that listens for KeyEvents and stores which keys are currently depressed. When processing input to update the game state, the poll method can be called to store a snapshot of the pressed keys. This snapshot can be used to determine if a key is currently pressed and if the key has been pressed for multiple frames.
- The MouseInput class is a MouseAdapter that listens for MouseEvents and stores the mouse location, scroll wheel notches, and which buttons are pressed. When processing input to update the game state, the poll method can be called to store a snapshot of these properties. This snapshot can be used to determine the mouse location, the state of the scroll wheel, if a button is currently pressed, and if a button has been pressed for multiple frames.
- The Matrix3x3f class is used to represent a 3x3 matrix of floating point numbers and perform several operations on them. Methods are provided to add, subtract, and multiple matrices together. Methods are provided to generate the zero, the identity, transformation, scaling, rotation, and shearing matrices. Finally, a matrix may also be multiplied by a vector to transform it using the matrix.
- The Vector2f class represents a vector in 2D space. It is made up of x and y coordinates along with an additional w field that determines whether the vector may be transformed by matrix multiplication via 3x3 transformation matrices generated using the Matrix3x3f class. This class also provides a few methods to transform the vector.