Josh Kornfeld, Anton Gorshkov, Andrei Martins May 28, 2019

Programming 2 Monitored Assignment 1 – Prof Kratzer

Idea

Create a GridPane with 16 squares and a gap of 1 pixel between them which colour can change with mouse events. Rows & columns of the GridPane will be used to identify each square.

Creating the squares

Squares will be created with the class Rectangle. Height & width of rectangle will be set to the same value, making it a square. Colour of square will be set to white initially so when the program starts, they are invisible.

The GridPane

The GridPane will be filled with 16x16 squares created by the Rectangle class. The columns & rows of the GridPane will be used to assign the location of each square. A nested for loop will be used to add a square at each coordinate of the GridPane. Additionally, a vertical & horizontal gap of 1 pixel will separate each square.

Handling mouse events

In total 3 mouse events will need to be handled: right & left click as well as mouse dragged while left click is pressed down. A left click will change a square's colour from white to black. A sequence of squares' colour will change when the mouse is dragged over them while the left button is held down. Finally, a right click will reset the grid to all white squares.

To determine the column of a square which has been clicked and colour needs changing, the following calculation will be performed: horizontal location of cursor / (height of square + 1). This result will then need to be casted to int so the method add(Node, column, row) can use it to add a new black square. Similarly, to determine the row, the following calculation will be performed: vertical location of cursor / (width of square + 1). Likewise, the result will be cast to int to be use by the add method. Adding +1 to the width/height of the square compensates for the 1 pixel used to create the gap between each square.

To reset the grid to all white squares. The nested for loop used to create the initial grid will be reused to make all the squares white.

References

https://docs.oracle.com/javase/8/javafx/api/javafx/scene/input/MouseEvent.html

https://docs.oracle.com/javase/8/javafx/api/javafx/scene/layout/GridPane.html

https://docs.oracle.com/javase/8/javafx/api/javafx/scene/shape/Rectangle.html