Table of Contents

[1)METHODS 2](#__RefHeading__107_1522893208)

[1.1)ai\_v\_ai\_butMouseClicked(MouseEvent evt) 2](#__RefHeading__109_1522893208)

[1.2)ai\_v\_human\_butMouseClicked(MouseEvent evt) 2](#__RefHeading__111_1522893208)

[1.3)human\_v\_ai\_butMouseClicked(MouseEvent evt) 2](#__RefHeading__378_421056963)

[1.4)human\_v\_humanbutMouseClicked(MouseEvent evt) 2](#__RefHeading__380_421056963)

[1.5)game\_board\_panelMouseClicked(MouseEvent evt) 2](#__RefHeading__113_1522893208)

[1.6)quit\_butMouseClicked(MouseEvent evt) 2](#__RefHeading__119_1522893208)

[1.7)play\_again\_butMouseClicked(MouseEvent evt) 2](#__RefHeading__121_1522893208)

[1.8)paint\_board(Graphics g) 2](#__RefHeading__123_1522893208)

[1.9)set\_all\_focusable() 3](#__RefHeading__125_1522893208)

[1.10)set\_all\_non\_focusable() 3](#__RefHeading__127_1522893208)

[2)VARIABLES 3](#__RefHeading__129_1522893208)

[2.1)char player 3](#__RefHeading__382_421056963)

[2.2)legal\_move[] 3](#__RefHeading__384_421056963)

[2.3)legal\_moves[][] 3](#__RefHeading__386_421056963)

[2.4)ai\_progress\_bar:JprogressBar 3](#__RefHeading__388_421056963)

[2.5)game\_board\_panel:Jpanel 3](#__RefHeading__390_421056963)

[2.6)game\_state\_display:JtextArea 3](#__RefHeading__392_421056963)

Stuff we may need to access from C programs

# METHODS

## ai\_v\_ai\_butMouseClicked(MouseEvent evt)

Not implemented yet: Will set Player\_X to ai, and Player\_O to ai

## ai\_v\_human\_butMouseClicked(MouseEvent evt)

Not implemented yet: will set player\_X to ai (ai gets first move) and player\_O to user

## human\_v\_ai\_butMouseClicked(MouseEvent evt)

Not implemented yet: will set player\_X human (user gets first move) and player\_O to ai

## human\_v\_humanbutMouseClicked(MouseEvent evt)

The current default: user moves for X and O respectively

## game\_board\_panelMouseClicked(MouseEvent evt)

Gets current mouse click location and compares it to all legal moves

Condition: if move is legal will paint the respective pixels either X or O based on who the current player is.

Condition: Win state reached, displayer either player X won, or player O won and activate play\_again\_but (Not implemented yet)

Condition: Move was illegal, player does not switch, game state display tells user not a legal move

Condition: Game board is full, and no win state, display game over graphic

## quit\_butMouseClicked(MouseEvent evt)

System.exit(0);

## play\_again\_butMouseClicked(MouseEvent evt)

Resets game board and enables all buttons if disabled else throws java.lang.NullPointerException

## paint\_board(Graphics g)

Repaints the board to fresh, resets legal\_moves[i][2] to zero, used by play again button

## set\_all\_focusable()

Switches all buttons to enabled and turns on all mouse listeners for buttons

## set\_all\_non\_focusable()

Switches all buttons to disabled and disables all mouse listeners on buttons except the “quit” button

# VARIABLES

## char player

Will either be X or O

## legal\_move[]

These are singe int arrays of size 3 int[0,1,2] containing where:

[0] = x coordinate

[1] = y coordinate

[2] = 0 if the location is available to be moved on and 1 if the location is already moved on.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a1 | b1 | c1 | d1 | e1 | f1 | g1 | h1 | i1 | j1 | k1 | l1 |
| a2 | b2 | c2 | d2 | e2 | f2 | g2 | h2 | i2 | j2 | k2 | l2 |
| a3 | b3 | c3 | d3 | e3 | f3 | g3 | h3 | i3 | j3 | k3 | l3 |
| a4 | b4 | c4 | d4 | e4 | f4 | g4 | h4 | i4 | j4 | k4 | l4 |

## legal\_moves[][]

legal\_moves:int[][] #holds a1,a2...k4,l1,l2,l3,l4 see above

## ai\_progress\_bar:JprogressBar

access to the progress bar to track ai

## game\_board\_panel:Jpanel

access to the display area

## game\_state\_display:JtextArea

access to the game state text field