FLYING NUMBERS

FLYNUMBERSPANEL

X,Y,SCORE,VX,VY,FONT(NAME,) COUNTER=60

PAINTCOMPENT(G)

SUPER.PAINTCOMPONENT()

G.SETFONT

G.SETCOLOR

IFCOUNTER>0{

G.DRAWSTRING()}

UPDATE(){

SCORE++

X+=VX

Y+=VY

TASK

IF(COUNTER>0)

COUNTER--}

package mahjong\_solitaire.events;

import java.awt.event.KeyAdapter;

import java.awt.event.KeyEvent;

import static mahjong\_solitaire.ZombieCrushConstants.GAME\_SCREEN\_STATE;

import mahjong\_solitaire.data.ZombieCrushDataModel;

import mahjong\_solitaire.data.ZombieCrushMove;

import mahjong\_solitaire.ui.ZombieCrushMiniGame;

/\*\*

\* This event handler lets us provide additional custom responses

\* to key presses while Mahjong is running.

\*

\* @author Richard McKenna and jing

\*/

public class MahjongKeyHandler extends KeyAdapter

{

// THE MAHJONG GAME ON WHICH WE'LL RESPOND

private ZombieCrushMiniGame game;

/\*\*

\* This constructor simply inits the object by

\* keeping the game for later.

\*

\* @param initGame The Mahjong game that contains

\* the back button.

\*/

public MahjongKeyHandler(ZombieCrushMiniGame initGame)

{

game = initGame;

}

/\*\*

\* This method provides a custom game response to when the user

\* presses a keyboard key.

\*

\* @param ke Event object containing information about the event,

\* like which key was pressed.

\*/

@Override

public void keyPressed(KeyEvent ke)

{

int i=0;

// CHEAT BY ONE MOVE. NOTE THAT IF WE HOLD THE C

// KEY DOWN IT WILL CONTINUALLY CHEAT

if (ke.getKeyCode() == KeyEvent.VK\_C)

{

ZombieCrushDataModel data = (ZombieCrushDataModel)game.getDataModel();

// FIND A MOVE IF THERE IS ONE

ZombieCrushMove move = data.findMove();

if (move != null){

i++;

data.processMove(move);

}

//else if(i==0&&move==null)

//data.endGameAsLoss();

}

else if(ke.getKeyCode() == KeyEvent.VK\_U){

ZombieCrushDataModel data = (ZombieCrushDataModel)game.getDataModel();

data.undoLastMove();

}

}

}

if (((ZombieCrushMiniGame)game).isCurrentScreenState(GAME\_SCREEN\_STATE)){

if(data.getTemScore()!=0){

String score=""+data.getTemScore();

if(counter>0){

g.setColor(DEBUG\_TEXT\_COLOR);

g.setFont(TEXT\_DISPLAY\_FONT);

g.drawString(score, x, y);

}

update(g);

}

}