**Coding**

import javax.swing.\*;

import javax.swing.event.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.io.\*;

import java.net.\*;

import java.applet.AudioClip;

import sun.audio.\*;

public class GAME extends JFrame

{

URL url1=null,url2=null,url3=null,url4=null,url5=null,url6=null,url7=null,url8=null,urli=null;

Image img1,img2,img3,img4,img5,img6,img7,img8,img9,img10,img11;

URL URLt=null;

//Image title=null;

Container cont;

final int WIDTH=900,HEIGHT=650;

boolean winnerChosen=false;

int p1Laps=0,p2Laps=0;

JLabel l = new JLabel("p1LAPS: " +p1Laps);

JLabel r = new JLabel("p2LAPS: " +p2Laps);

//static String[] names=new String[10];

static String[] pnames1=new String[10];

static String[] pnames2=new String[10];

static int set=1;

//String[] pnames=new String[10];

double p1Speed=0.5,p2Speed=0.5;

static int i;

final int UP=0,RIGHT=1,DOWN=2,LEFT=3;

int p1Direction=UP,p2Direction=UP;

Rectangle left=new Rectangle(0,0,WIDTH/9,HEIGHT);

Rectangle right=new Rectangle((WIDTH/9)\*9,0,(WIDTH/9),HEIGHT);

Rectangle top=new Rectangle(0,0,WIDTH,HEIGHT/9);

Rectangle bottom=new Rectangle(0,(HEIGHT/9)\*9,WIDTH,HEIGHT/9);

Rectangle center=new Rectangle((int)((WIDTH/9)\*2.5),(int)((HEIGHT/9)\*2.5),(int)((WIDTH/9)\*5),(HEIGHT/9)\*4);

Rectangle l1=new Rectangle(0,0,(WIDTH/9),(HEIGHT/5));

Rectangle obstacle=new Rectangle(WIDTH/2,(int)((HEIGHT/9)\*5),WIDTH/20,(int)(HEIGHT/1));

//Rectangle obstacle2=new Rectangle(WIDTH/3,(int)((HEIGHT/9)\*5),WIDTH/10,HEIGHT/4);

//Rectangle obstacle3=new Rectangle(2\*(WIDTH/3),(int)((HEIGHT/9)\*5),WIDTH/10,HEIGHT/4);

//Rectangle obstacle4=new Rectangle(WIDTH/3,HEIGHT/9,WIDTH/30,HEIGHT/9);

//Rectangle obstacle5=new Rectangle(WIDTH/2,(int)((HEIGHT/9)\*1.5),WIDTH/30,HEIGHT/4);

Rectangle finish=new Rectangle(WIDTH/9,(HEIGHT/2)-(HEIGHT/9),(int)((WIDTH/9)\*1.5),HEIGHT/70);

Rectangle p1=new Rectangle(WIDTH/9,HEIGHT/2,WIDTH/30,WIDTH/30);

Rectangle p2=new Rectangle((WIDTH/9)+(int)(((WIDTH/9)\*2)/2),(HEIGHT/2)+(HEIGHT/10),WIDTH/30,WIDTH/30);

public GAME()

{

super("RACE N CHASE\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(C) GROUP 4");

setVisible(true);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(WIDTH,HEIGHT);

cont = getContentPane();

cont.setLayout(null);

cont.setBackground(Color.BLACK);

cont.add(l);

l.setBounds(50,50,100,30);

l.setForeground(Color.WHITE);

cont.add(r);

r.setBounds(WIDTH-150,50,100,30);

r.setForeground(Color.WHITE);

cont.setLayout(null);

//loading the URLS

try

{

// url1=this.getClass().getResource("G1P1Img1.jpg");

//url2=this.getClass().getResource("G1P1Img2.jpg");

//url3=this.getClass().getResource("G1P1Img3.jpg");

//url4=this.getClass().getResource("G1P1Img4.jpg");

url5=this.getClass().getResource("blue\_car.gif");

url6=this.getClass().getResource("G1P1Img6.jpg");

url7=this.getClass().getResource("G1P1Img7.jpg");

url8=this.getClass().getResource("G1P1Img8.jpg");

//URLt=this.getClass().getResource("title.png");

}

catch(Exception e){}

img1=Toolkit.getDefaultToolkit().getImage("red\_car01.png");

img2=Toolkit.getDefaultToolkit().getImage("red\_car02.png");

img3=Toolkit.getDefaultToolkit().getImage("red\_car03.png");

img4=Toolkit.getDefaultToolkit().getImage("red\_car04.png");

img5=Toolkit.getDefaultToolkit().getImage("car.png");

img6=Toolkit.getDefaultToolkit().getImage("car3.png");

img7=Toolkit.getDefaultToolkit().getImage("car2.png");

img8=Toolkit.getDefaultToolkit().getImage("car4.png");

img9=Toolkit.getDefaultToolkit().getImage("racecar.png");

img10=Toolkit.getDefaultToolkit().getImage("flag.png");

img11=Toolkit.getDefaultToolkit().getImage("fb.png");

//title=Toolkit.getDefaultToolkit().getImage(URLt);

JOptionPane.showMessageDialog(null,"WELCOME TO RACE N CHASE v1\n\n"+"Game: 2 player game\n\n"+"GOAL :COMPLETE 3 LAPS BEFORE THE OPPONENT\n\n"+"CONTROLS:\n"+"PLAYER1:\n"+"(BLUE CAR) WASD directional,speed auto\n\n"+"keep off the grass or else speed will come down :("+"and may spin your car :)\n\n\n"+"ok BEST OF LUCK and CLICK READY TO START");

title();

pnames1[i]=JOptionPane.showInputDialog("Enter a name");

pnames2[i]=JOptionPane.showInputDialog("Enter a name");

//getData();

Move1 m1=new Move1();

Move2 m2=new Move2();

m1.start();

setContentPane(cont);

m2.start();

//printValues();

try

{URL eng=this.getClass().getResource("engine.wav");

AudioClip snd=JApplet.newAudioClip(eng);

snd.loop();

}

catch(Exception e){}

}

/\*public void printValues()

{

for(int i=0; i < names.length && i< pnames.length; i++)

{if(set==1)

System.out.println(names[i]);

else

System.out.println(pnames[i]);

}

}\*/

/\*public void getData(){

for(int i=0; i < names.length && i <pnames.length; i++){

names[i]=pnames1[i];

pnames[i]=pnames2[i];

}

}\*/

public void title()

{

try

{

JLabel title = new JLabel(new ImageIcon("title.PNG"));

cont.add(title);

title.setBounds(-60,-300,600,200);

do

{

title.setBounds(title.getX(),title.getY()+1,1000,1000);

Thread.sleep(15);

}

while(title.getY()<700);

}

catch(Exception e){}

}

public void paint(Graphics g)

{

super.paint(g);

g.setColor(Color.LIGHT\_GRAY);

g.fillRect(0,0,WIDTH,HEIGHT);

g.setColor(Color.GREEN);

Rectangle line0=new Rectangle(WIDTH/9,HEIGHT/2,(int)(((WIDTH/9)\*1.5)/2),HEIGHT/140);

Rectangle lineI=new Rectangle((WIDTH/9)+(int)(((WIDTH/9)\*1.5)/2),(HEIGHT/2)+(HEIGHT/10),(int)(((WIDTH/9)\*1.5)/2),HEIGHT/140);

g.drawImage(img11,left.x,left.y,this);

g.fillRect(left.x,left.y,left.width,left.height);

g.fillRect(right.x,right.y,right.width,right.height);

g.fillRect(top.x,top.y,top.width,top.height);

g.fillRect(bottom.x,bottom.y,bottom.width,bottom.height);

g.drawImage(img9,center.x,center.y,this);

g.fillRect(center.x,center.y,center.width,center.height);

g.setColor(Color.GREEN);

g.fillRect(obstacle.x,obstacle.y,obstacle.width,obstacle.height);

//g.fillRect(obstacle2.x,obstacle2.y,obstacle2.width,obstacle2.height);

//g.fillRect(obstacle3.x,obstacle3.y,obstacle3.width,obstacle3.height);

//g.setColor(Color.lightGray);

//g.fillRect(obstacle4.x,obstacle4.y,obstacle4.width,obstacle4.height);

//g.fillRect(obstacle5.x,obstacle5.y,obstacle5.width,obstacle5.height);

g.setColor(Color.WHITE);

g.fillRect(line0.x,line0.y,line0.width,line0.height);

g.fillRect(lineI.x,lineI.y,lineI.width,lineI.height);

//g.setColor(Color.BLACK);

//g.fillRect(l.x,l.y,l.width,l.height);

g.setColor(Color.WHITE);

g.fillRect(finish.x,finish.y,finish.width,finish.height);

g.setColor(Color.WHITE);

g.fill3DRect(p1.x,p1.y,p1.width,p1.height,true);

g.setColor(Color.RED);

g.fill3DRect(p2.x,p2.y,p2.width,p2.height,true);

g.drawImage(img9,center.x+90,center.y+30,this);

g.drawImage(img10,finish.x,finish.y,this);

if(p1Direction==UP)

g.drawImage(img5,p1.x,p1.y,this);

if(p1Direction==LEFT)

g.drawImage(img8,p1.x,p1.y,this);

if(p1Direction==DOWN)

g.drawImage(img7,p1.x,p1.y,this);

if(p1Direction==RIGHT)

g.drawImage(img6,p1.x,p1.y,this);

if(p2Direction==UP)

g.drawImage(img1,p2.x,p2.y,this);

if(p2Direction==LEFT)

g.drawImage(img4,p2.x,p2.y,this);

if(p2Direction==DOWN)

g.drawImage(img3,p2.x,p2.y,this);

if(p2Direction==RIGHT)

g.drawImage(img2,p2.x,p2.y,this);

}

private class Move1 extends Thread implements KeyListener

{

public void run()

{

addKeyListener(this);

while(true)

{

try

{

repaint();

if(p1.intersects(left)||p1.intersects(right)||p1.intersects(top)||p1.intersects(bottom)||p1.intersects(p2))//||p1.intersects(obstacle)||p1.intersects(obstacle2)||p1.intersects(obstacle3)||p1.intersects(obstacle4)||p1.intersects(obstacle5))

{

p1Speed=-4;

}

if(p1.intersects(center))

p1Speed=-2.5;

if(p1.intersects(finish)&&p1Direction==UP)

{

p1Laps++;

l.setText("p1LAPS: "+p1Laps/10);

}

if(p1Laps>=30)

{

if(!winnerChosen)

{

winnerChosen=true;JOptionPane.showMessageDialog(null,pnames1[i]+ ".....WINS !!");

set=1;

break;

}

else

{

JOptionPane.showMessageDialog(null,"PLAYER 1(BLUE):LOSER!:(\n"+"PLAYER2 (RED):WINNER!!!:D");

break;

}

}

if(p1Speed<=5)

p1Speed+=.2;

if(p1Direction==UP)

p1.y-=(int)p1Speed;

if(p1Direction==DOWN)

p1.y+=(int)p1Speed;

if(p1Direction==LEFT)

p1.x-=(int)p1Speed;

if(p1Direction==RIGHT)

p1.x+=(int)p1Speed;

Thread.sleep(75);

}

catch(Exception e)

{

break;

}

}

}

public void keyPressed(KeyEvent event)

{

}

public void keyReleased(KeyEvent event)

{

}

public void keyTyped(KeyEvent event)

{

if(event.getKeyChar()=='a')

p1Direction=LEFT;

if(event.getKeyChar()=='s')

p1Direction=DOWN;

if(event.getKeyChar()=='d')

p1Direction=RIGHT;

if(event.getKeyChar()=='w')

p1Direction=UP;}

}

private class Move2 extends Thread implements KeyListener

{

public void run()

{

addKeyListener(this);

while(true)

{

try

{

repaint();

if(p2.intersects(left)||p2.intersects(right)||p2.intersects(top)||p2.intersects(bottom)||p2.intersects(p1))//||p1.intersects(obstacle)||p1.intersects(obstacle2)||p1.intersects(obstacle3)||p1.intersects(obstacle4)||p1.intersects(obstacle5))

{

p2Speed=-4;

}

if(p2.intersects(center))

p2Speed=-2.5;

if(p2.intersects(finish)&&p2Direction==UP)

{

p2Laps++;

r.setText("p2LAPS: "+p2Laps/10);

}

if(p2Laps>=30)

{

if(!winnerChosen)

{

winnerChosen=true;JOptionPane.showMessageDialog(null,pnames2[i]+ ".....WINS !!");

set=2;

break;

}

else

{

JOptionPane.showMessageDialog(null,"PLAYER 2(RED):LOSER!:(\n"+"PLAYER1 (BLUE):WINNER!!!:D");

break;

}

}

if(p2Speed<=5)

p2Speed+=.2;

if(p2Direction==UP)

p2.y-=(int)p2Speed;

if(p2Direction==DOWN)

p2.y+=(int)p2Speed;

if(p2Direction==LEFT)

p2.x-=(int)p2Speed;

if(p2Direction==RIGHT)

p2.x+=(int)p2Speed;

Thread.sleep(75);

}

catch(Exception e)

{

break;

}

}

}

public void keyPressed(KeyEvent event)

{

}

public void keyReleased(KeyEvent event)

{

}

public void keyTyped(KeyEvent event)

{

if(event.getKeyChar()=='j')

p2Direction=LEFT;

if(event.getKeyChar()=='k')

p2Direction=DOWN;

if(event.getKeyChar()=='l')

p2Direction=RIGHT;

if(event.getKeyChar()=='i')

p2Direction=UP;}

}

public static void main(String args[])throws IOException

{

new GAME();

String text=null;

Writer output = null;

if(set==1)

{text = "\n>>"+pnames1[i];

}

else if(set==2)

{

text = pnames2[i];

}

File file = new File("WINNERDATA.txt");

output = new BufferedWriter(new FileWriter(file, true));

output.write(text);

output.close();

System.out.println("RECORDS UPDATED");

FileInputStream fstream = new FileInputStream("WINNERDATA.txt");

DataInputStream in = new DataInputStream(fstream);

BufferedReader br = new BufferedReader(new InputStreamReader(in));

String strLine;

while ((strLine = br.readLine()) != null)

System.out.println (strLine);

in.close();

}

}