Weslyn Wagner EE 5343 UTSA: Java Assignment #4 10/16/2016

|  |  |
| --- | --- |
| Class AnimationHw4 | |
| Requirements | * 8 parking spaces (90° wrt aisle) * Parking space length = 2x parking space width * Aisle width = 1.33x parking space length * Vehicle width = 0.8x space width * Vehicle should enter from left and travel to the right * Vehicle should park past middle of the lot * Vehicle should not touch any lines |
| Design | * Parking lot dimensions = 850x700 pixels * Car dimensions = 150x80 pixels * Total parking spaces = 8 * Space width = 100 pixels * Space length = 200 pixels * Aisle width = 266 pixels |

/\*

\*

\* @author Weslyn Wagner - zfs119

\* Sept 16, 2016

\*

\* Version 1.0 2016/10/06

\*

\* EE 5343 UTSA

\*/

import java.awt.event.\*;

import java.awt.Color;

import javax.swing.JFrame;

import javax.swing.JPanel;

import java.awt.Graphics;

import java.awt.Font;

import java.awt.Polygon;

import java.awt.Graphics2D;

import java.awt.\*;

import javax.swing.\*;

import java.awt.geom.AffineTransform;

public class AnimationHw4 {

public static void main(String[] args) {

JFrame frame = new JFrame("My First Frame");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

ColorJPanel colorJPanel = new ColorJPanel();

frame.add(colorJPanel);

//Creating the parking lot size

frame.setSize(850, 700);

frame.setVisible(true);

}

}

class ColorJPanel extends JPanel{

int i = 0;

int x = 0;

int y =327;

//Creating the car

Rectangle rect = new Rectangle(0,300,150,80);

public void paintComponent(Graphics g){

super.paintComponent(g);

//Setting the parking lot color to black

setBackground(Color.BLACK);

//Drawing the parking space lines

g.drawLine(100, 0, 100, 200);

g.drawLine(200, 0, 200, 200);

g.drawLine(300, 0, 300, 200);

g.drawLine(400, 0, 400, 200);

g.drawLine(500, 0,500, 200);

g.drawLine(600, 0,600, 200);

g.drawLine(700, 0,700, 200);

g.drawLine(100, 466, 100, 666);

g.drawLine(200, 466, 200, 666);

g.drawLine(300, 466, 300, 666);

g.drawLine(400, 466, 400, 666);

g.drawLine(500, 466, 500, 666);

g.drawLine(600, 466, 600, 666);

g.drawLine(700, 466, 700, 666);

Graphics2D g2d = (Graphics2D)g;

//Putting the car into motion with if/else statements

//Translate the car down the aisle at a rate of x+1 until it’s past the middle of the aisle

x = x + 1;

if(x<400){

g2d.translate(x, 0);

x = x + 1;

}

//Rotate and translate the car counterclockwise until it’s aligned with the parking space

else if(x<500 && i<90){

g2d.translate(x, 0);

i+=1;

g2d.rotate(Math.toRadians(-i),10,300);

}

//Translate the car upward to pull into the parking spot

else if (y>160){

y = y - 1;

g2d.translate(208, y);

g2d.rotate(Math.toRadians(-90));

}

//Keep the car parked in the parking spot

else{

//The else loop executes, but I couldn't get it to

//paint a stagnant rectangle in the parking spot

//Rectangle rect2 = new Rectangle(400,10,80,150);

}

//Setting the color of the car

g2d.setColor(Color.BLUE);

g2d.fill(rect);

}

public ColorJPanel() {

ActionListener animate = new ActionListener() {

public void actionPerformed(ActionEvent ae) {

repaint();

}

};

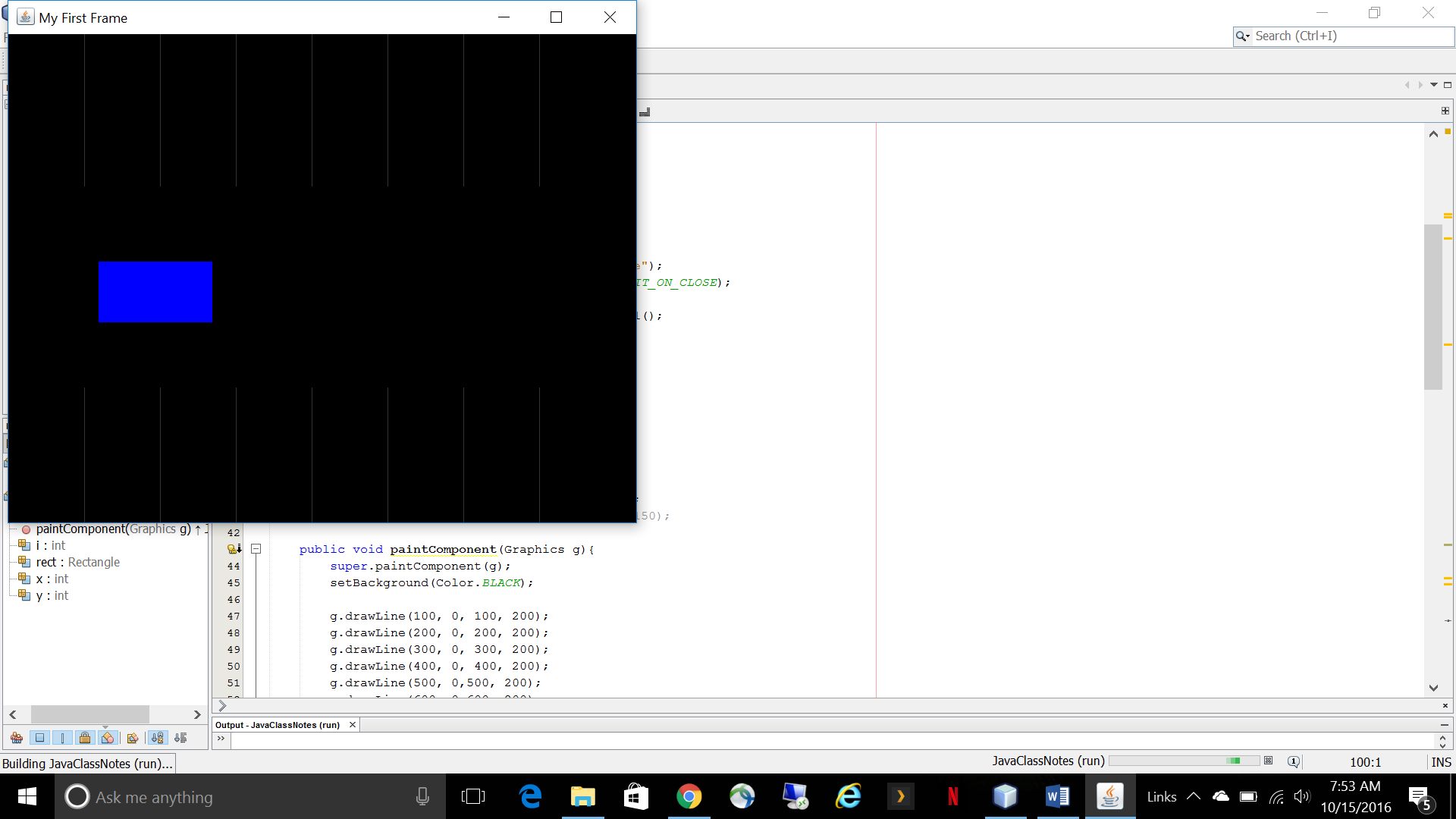
//Controls how quickly the actions are performed

Timer timer = new Timer(10,animate);

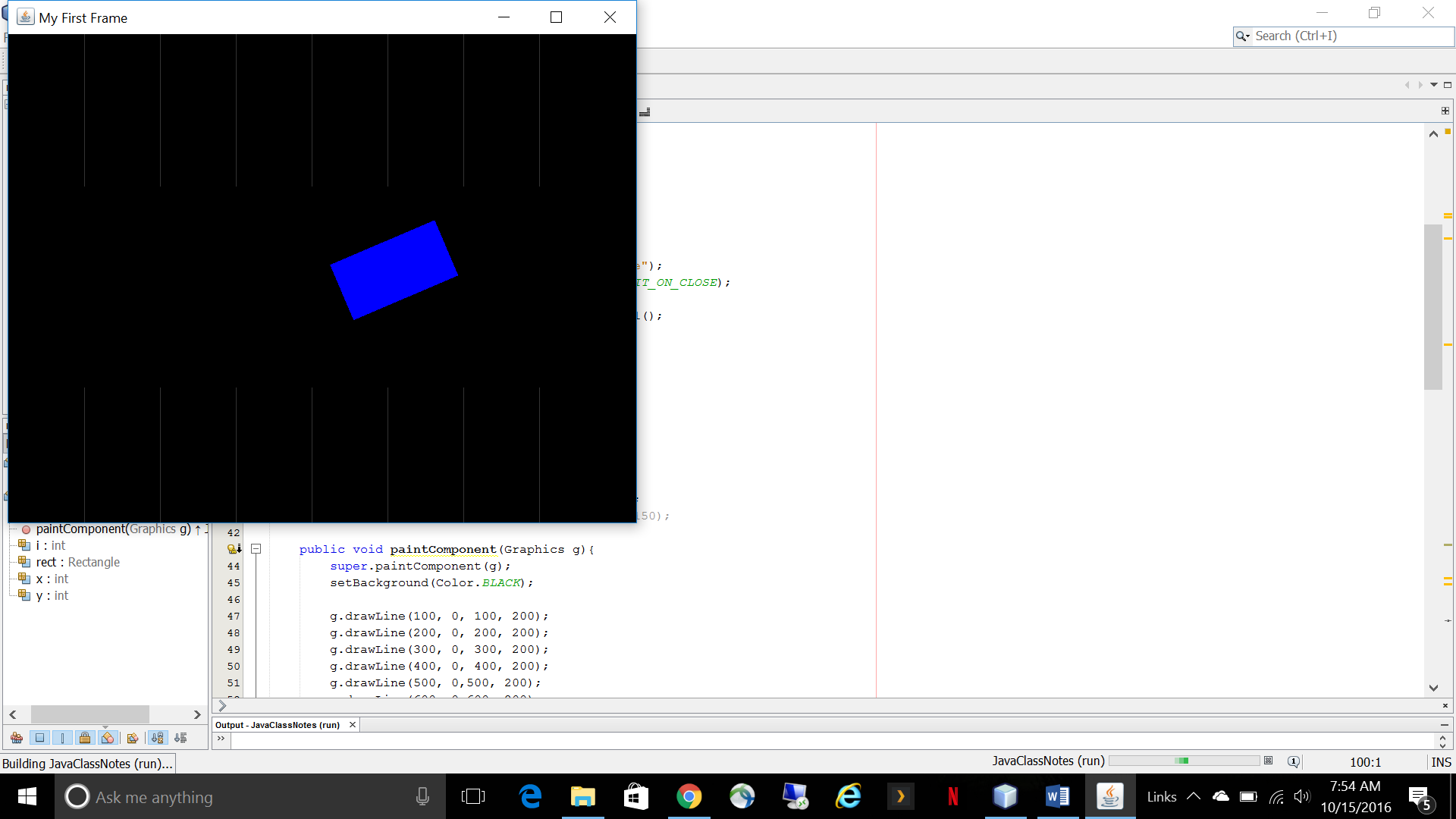
timer.start();

}

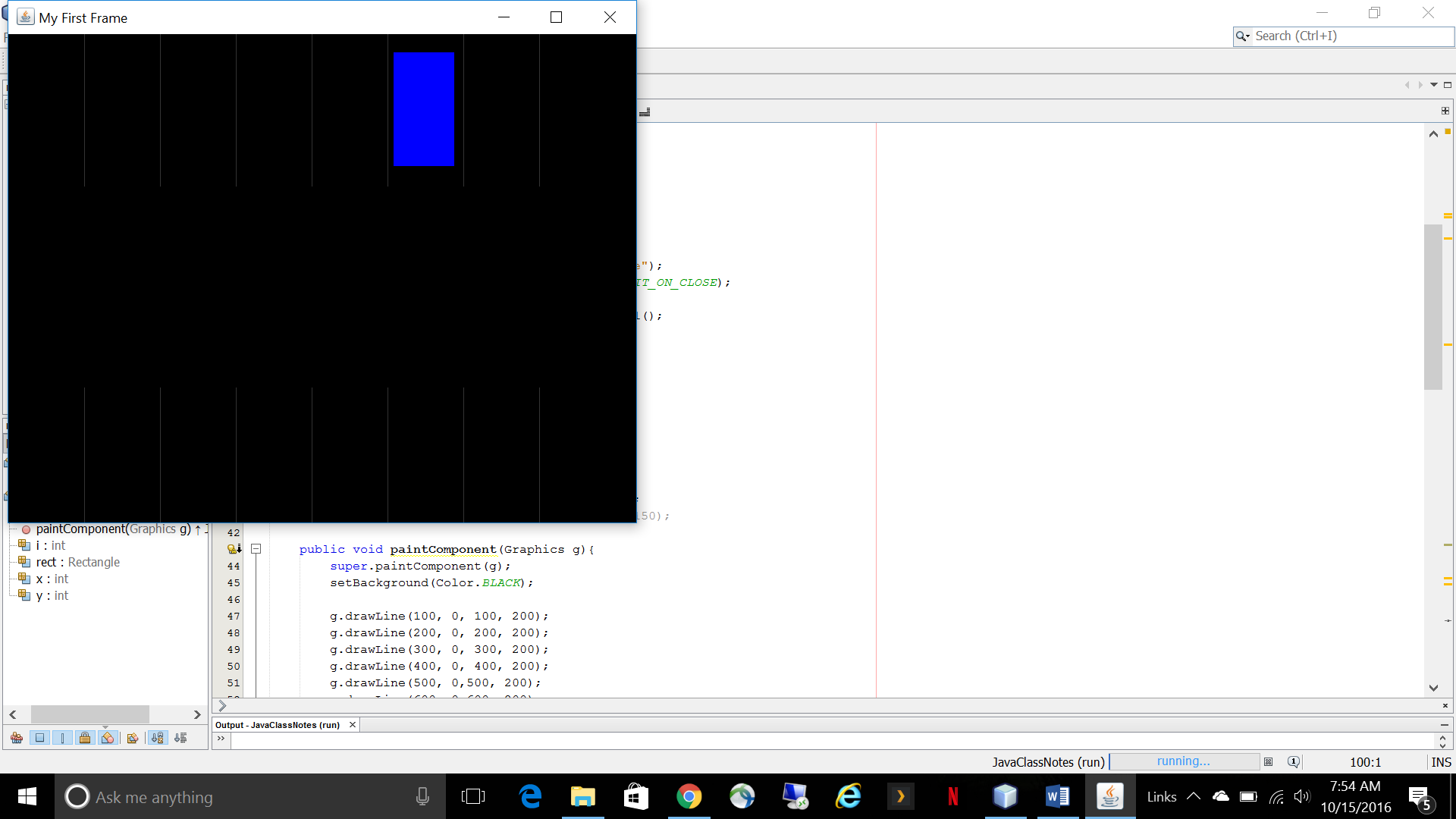
}



**Figure 1: Translating the car to the right**



**Figure 2: Rotating the car to align it with the parking space**



**Figure 3: Translating the car upward to pull into the parking space**