

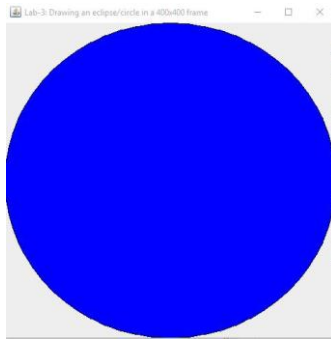
Lab 3 – Object Oriented Programming and Design Constructors in Graphics

1.

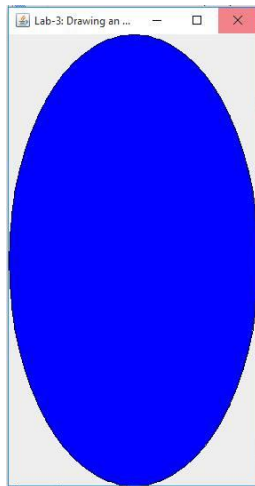
```
Circle100.java CViewer.java
1 /**
2  * Devante Wilson - 100554361
3  * October 29, 2015
4  *
5  * Program fills window with a circle graphic
6  */
7
8 // import packages
9 import java.awt.Graphics;
10 import java.awt.Graphics2D;
11 import java.awt.Color;
12 import java.awt.geom.Ellipse2D;
13 import javax.swing.JComponent;
14
15 // This component draws an ellipse tied to the component boundary
16 public class Circle100 extends JComponent
17 {
18     // Paints the ellipse
19     // @param g the graphics context
20     public void paintComponent(Graphics g)
21     {
22         // create objects
23         // cast Graphics object as Graphics2D object
24         Graphics2D g2 = (Graphics2D) g;
25         Ellipse2D.Double ellipse = new Ellipse2D.Double(0,0,getWidth(),getWidth());
26
27         g2.setColor(Color.BLUE);
28         g2.fill(ellipse);
29         g2.setColor(Color.BLACK);
30         g2.draw(ellipse);
31     }
32 }
```

```
Circle100.java CViewer.java
1 /**
2  * Devante Wilson - 100554361
3  * October 29, 2015
4  *
5  * Viewer for Ellipse demo
6  */
7
8 // import packages
9 import javax.swing.JFrame;
10
11 public class CViewer
12 {
13     public static void main(String[] args)
14     {
15         // create JFrame object
16         JFrame frame = new JFrame ();
17
18         // set frame properties
19         frame.setSize(500,500);
20         frame.setTitle("Lab-3: Drawing an ellipse/circle in a 500x500 frame");
21         frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
22
23         // create Circle100 object and add to frame
24         Circle100 component = new Circle100();
25         frame.add(component);
26
27         // make frame visible
28         frame.setVisible(true);
29     }
30 }
```

1.1

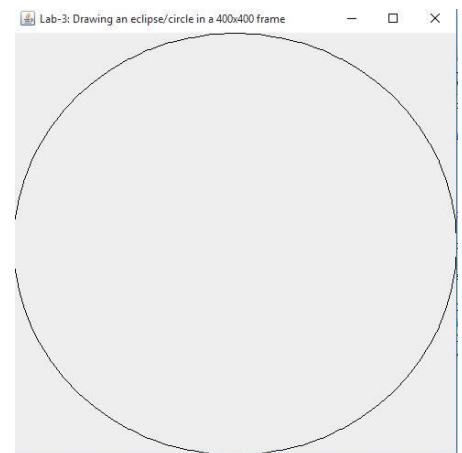


1.2

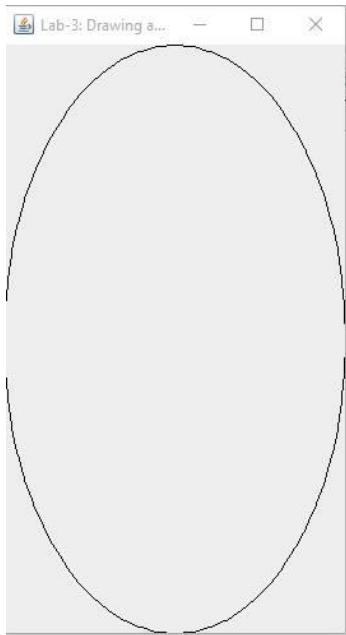


1.3

```
Circle100.java  CViewer.java
1  /**
2   * Devante Wilson - 100554361
3   * October 29, 2015
4   *
5   * Program fills window with a circle graphic
6   */
7
8  // import packages
9  import java.awt.Graphics;
10 import java.awt.Graphics2D;
11 import java.awt.Color;
12 import java.awt.geom.Ellipse2D;
13 import javax.swing.JComponent;
14
15 // This component draws an ellipse tied to the component boundary
16 public class Circle100 extends JComponent
17 {
18     // Paints the ellipse
19     // @param g the graphics context
20     public void paintComponent(Graphics g)
21     {
22         // create objects
23         // cast Graphics object as Graphics2D object
24         Graphics2D g2 = (Graphics2D) g;
25         Ellipse2D.Double ellipse = new Ellipse2D.Double(0,0,getWidth(),getWidth());
26
27         g2.setColor(Color.BLUE);
28         //g2.fill(ellipse);
29         g2.setColor(Color.BLACK);
30         g2.draw(ellipse);
31     }
32 }
```



1.4



2.

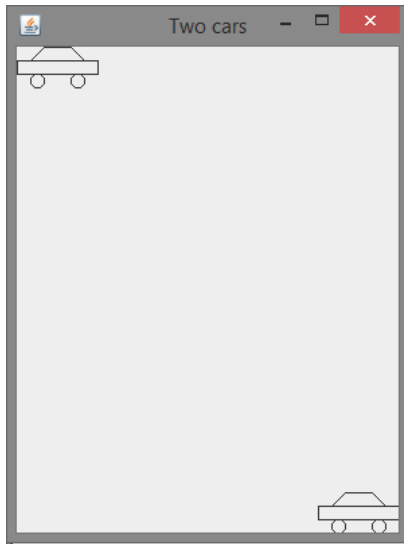
```
Circle100.java  CViewer.java  CarViewer.java  CarComponent.java  Car.java

1  /**
2   * Devante Wilson - 100554361
3   * October 29, 2015
4   *
5   * Program draws two cars in window at specific positions
6   * First at the origin (0,0) and the second at a distance of f
7   *
8   * CarViewer class specifically runs the program with the main method
9   */
10
11 // import packages
12 import javax.swing.JFrame;
13
14 public class CarViewer
15 {
16     public static void main(String[] args)
17     {
18         // create new JFrame object
19         JFrame frame = new JFrame();
20
21         // set frame properties
22         frame.setSize(300,400);
23         frame.setTitle("Two cars");
24         frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
25
26         // create new CarComponent object and add to frame
27         CarComponent component = new CarComponent();
28         frame.add(component);
29
30         frame.setVisible(true);
31     }
32 }
```

```
Circle100.java  CViewer.java  CarViewer.java  CarComponent.java  ⌕
1  /**
2   * Devante Wilson - 100554361
3   * October 29, 2015
4   *
5   * CarComponent draws the two cars
6   */
7
8  // import packages
9  import java.awt.Graphics;
10 import java.awt.Graphics2D;
11 import javax.swing.JComponent;
12
13 public class CarComponent extends JComponent
14 {
15     public void paintComponent(Graphics g)
16     {
17         // create objects and variables
18         Graphics2D g2 = (Graphics2D) g;
19
20         Car car1 = new Car(0,0);
21
22         int x = getWidth() - 60;
23         int y = getHeight() - 30;
24
25         Car car2 = new Car (x,y);
26
27         // draw objects to graphics context
28         car1.draw(g2);
29         car2.draw(g2);
30     }
31 }
```

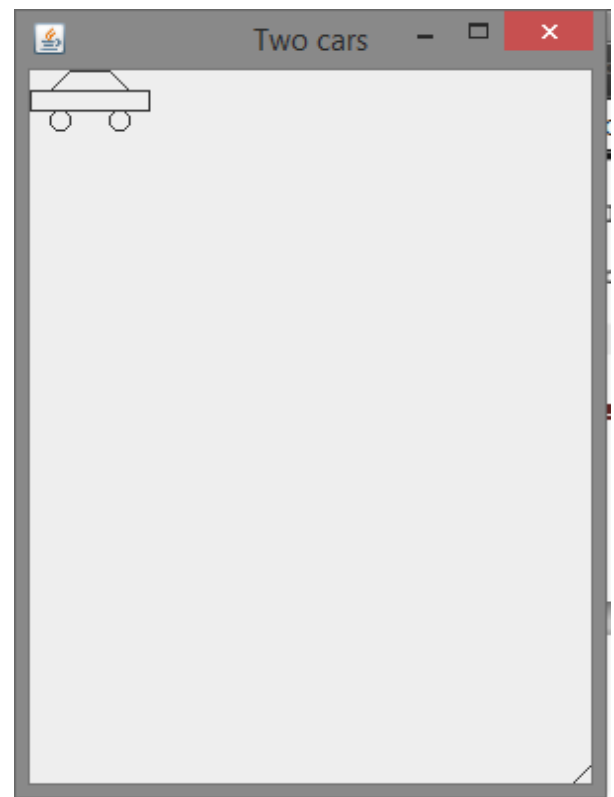
```
Circle100.java CViewer.java CarViewer.java CarComponent.java Car.java x
1  /**
2   * Devante Wilson - 100554361
3   * October 29, 2015
4   *
5   * Car class draws the car shape
6   */
7
8  // import packages
9  import java.awt.Graphics2D;
10 import java.awt.Rectangle;
11 import java.awt.geom.Ellipse2D;
12 import java.awt.geom.Line2D;
13 import java.awt.geom.Point2D;
14
15 // car shape that be positioned anywhere on the screen
16 public class Car
17 {
18     // declare variables
19     private int xLeft;
20     private int yTop;
21
22     // constructs a car with a given top left corner
23     // @param x the x coordinate of the top left corner
24     // @param y the y coordinate of the top left corner
25     public Car(int x, int y)
26     {
27         xLeft = x;
28         yTop = y;
29     }
30
31     // draws the car
32     // @param g2 the graphics context
33     public void draw(Graphics2D g2)
34     {
35         // body
36         Rectangle body = new Rectangle(xLeft, yTop + 10, 60, 10);
37         // front tire
38         Ellipse2D.Double frontTire
39             = new Ellipse2D.Double(xLeft + 10, yTop + 20, 10, 10);
40         // rear tire
41         Ellipse2D.Double rearTire
42             = new Ellipse2D.Double(xLeft + 40, yTop + 20, 10, 10);
43
44         // bottom of the front windshield
45         Point2D.Double r1 = new Point2D.Double(xLeft + 10, yTop + 10);
46         // front of the roof
47         Point2D.Double r2 = new Point2D.Double(xLeft + 20, yTop);
48         // rear of the roof
49         Point2D.Double r3 = new Point2D.Double(xLeft + 40, yTop);
50         // bottom of the rear windshield
51         Point2D.Double r4 = new Point2D.Double(xLeft + 50, yTop + 10);
52
53         // entire front windshield
54         Line2D.Double frontWindshield = new Line2D.Double(r1, r2);
55         // entire roof
56         Line2D.Double roofTop = new Line2D.Double(r2, r3);
57         // entire rear windshield
58         Line2D.Double rearWindshield = new Line2D.Double(r3, r4);
59
60         // draw objects to graphics context
61         g2.draw(body);
62         g2.draw(frontTire);
63         g2.draw(rearTire);
64         g2.draw(frontWindshield);
65         g2.draw(roofTop);
66         g2.draw(rearWindshield);
67     }
68 }
```

2.1

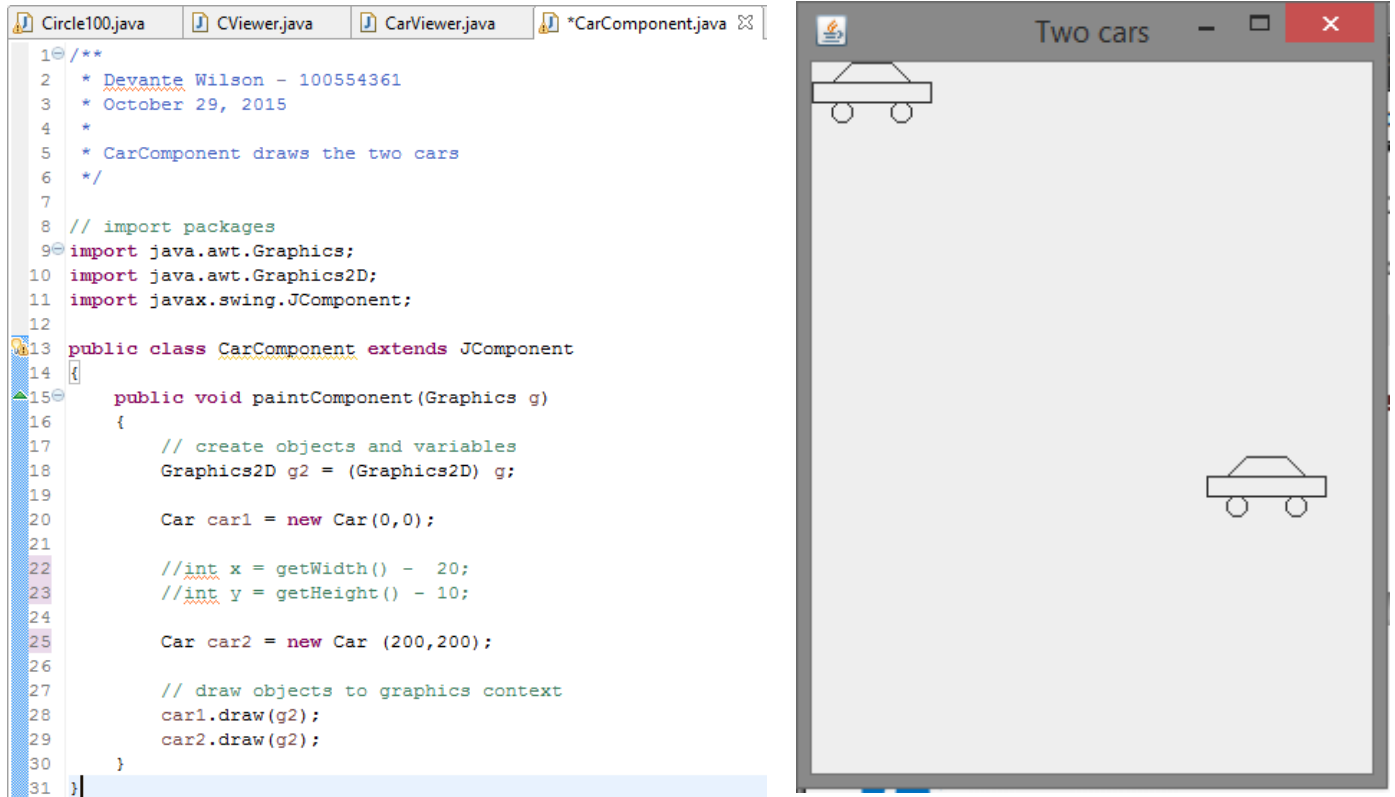


2.2

```
Circle100.java  CViewer.java  CarViewer.java  CarComponent.java
1  /**
2   * Devante Wilson - 100554361
3   * October 29, 2015
4   *
5   * CarComponent draws the two cars
6   */
7
8  // import packages
9  import java.awt.Graphics;
10 import java.awt.Graphics2D;
11 import javax.swing.JComponent;
12
13 public class CarComponent extends JComponent
14 {
15     public void paintComponent(Graphics g)
16     {
17         // create objects and variables
18         Graphics2D g2 = (Graphics2D) g;
19
20         Car car1 = new Car(0,0);
21
22         int x = getWidth() - 20;
23         int y = getHeight() - 10;
24
25         Car car2 = new Car (x,y);
26
27         // draw objects to graphics context
28         car1.draw(g2);
29         car2.draw(g2);
30     }
31 }
```



2.3



2.4

The purpose of lines 16 and 17 (or 22 and 23 in my program) of the `CarComponent` class is to retrieve the `JComponent`'s width and height (as the user may resize the frame) and to set the second car's location relative to those coordinates. In java, the top left of the component is defined to be the origin, so the coordinates that `getWidth()` and `getHeight()` return are relative to the origin.

2.5

The purpose of line 11 (line 24 in my program) of the `CarViewer` class is to initiate a certain operation once the user initiates a close on the frame (presses X). The default operation in this case is to exit the program (`EXIT_ON_CLOSE`).