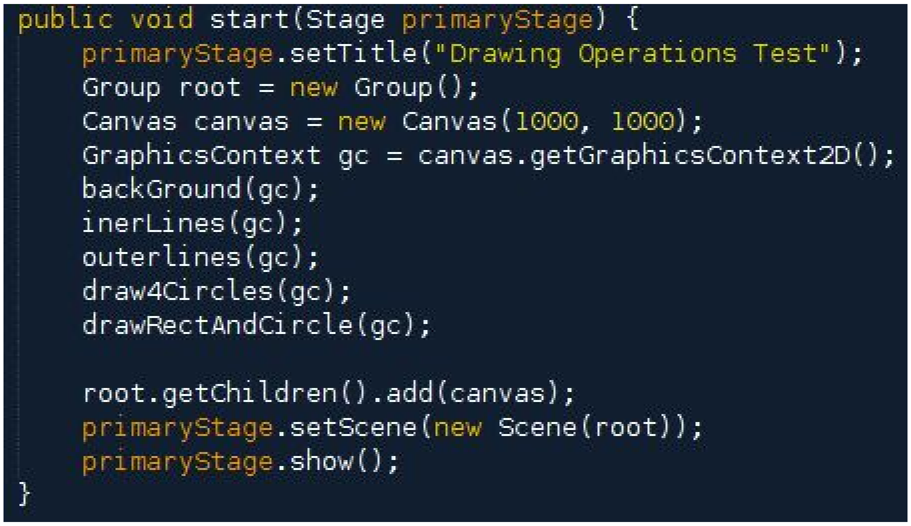
Methods Madness Essay

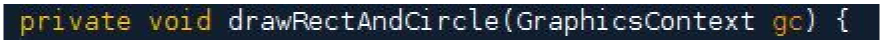
In the last couple of days we been leaning hot to use methods.Now in this new project called method madness we had to crate art in java.FX. To do so we had to implement everything we have been learning for the last couple of weeks, especially methods.

In my project I implemented circle inside the square in the middle surrounded with four circles in each four corners of the canvas.The four circles are connected with line then dose line are connected with the inner square. All of this was added inside my canvest that was created in the program.I think my project is cool i was created by me who didn't really get what to do or how to stated. But with the help of my peers and and the example you gave us i understood it.

In my project you could see the use of encapsulation through *private void drawRectAndCircle*(gc); a public method because it is calling a private method *private void drawRectAndCircle*(Graphicsontext gc).

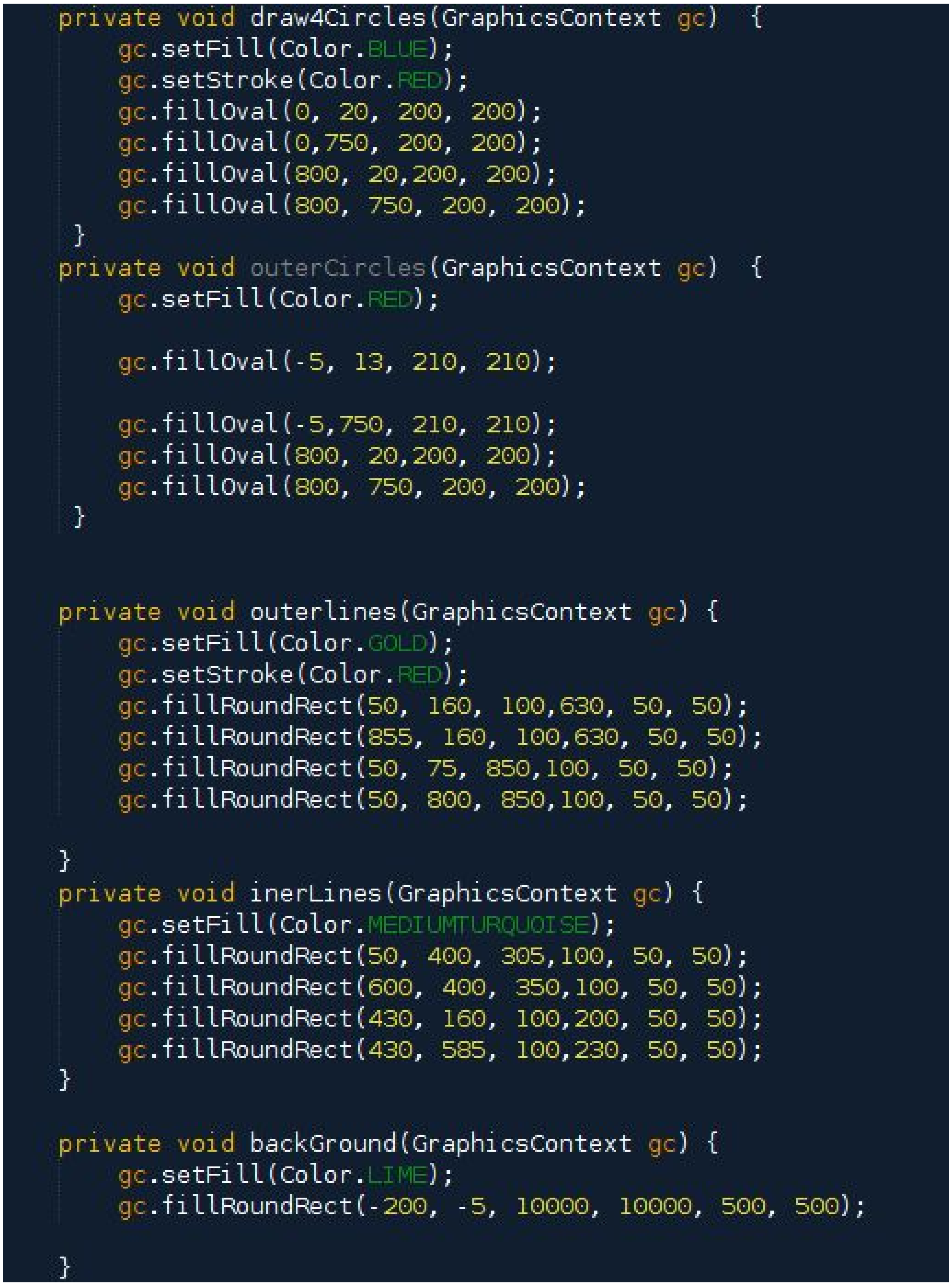


```
public void start(Stage primaryStage) {
    primaryStage.setTitle("Drawing Operations Test");
    Group root = new Group();
    Canvas canvas = new Canvas(1000, 1000);
    GraphicsContext gc = canvas.getGraphicsContext2D();
    backGround(gc);
    inerLines(gc);
    outerlines(gc);
    draw4Circles(gc);
    drawRectAndCircle(gc);

    root.getChildren().add(canvas);
    primaryStage.setScene(new Scene(root));
    primaryStage.show();
}
```

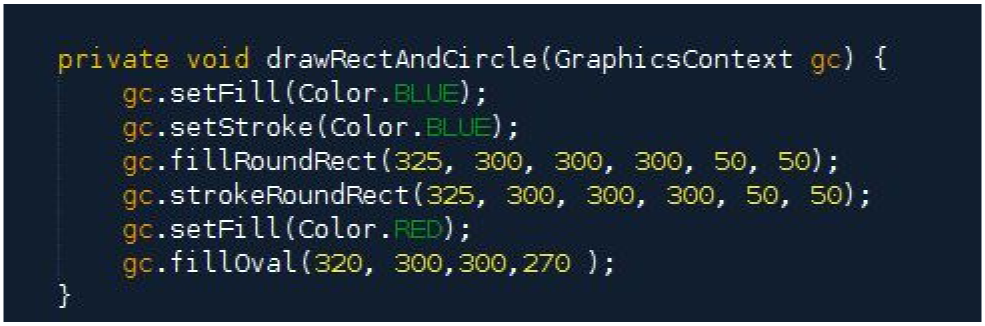


```
private void drawRectAndCircle(GraphicsContext gc) {
```

These are the rest of the methods used in the program:



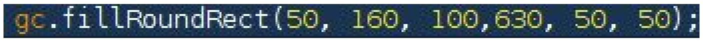
```
private void draw4Circles(GraphicsContext gc)  {
    gc.setFill(Color.BLUE);
    gc.setStroke(Color.RED);
    gc.fillOval(0, 20, 200, 200);
    gc.fillOval(0,750, 200, 200);
    gc.fillOval(800, 20,200, 200);
    gc.fillOval(800, 750, 200, 200);
 }
private void outerCircles(GraphicsContext gc)  {
    gc.setFill(Color.RED);

    gc.fillOval(-5, 13, 210, 210);

    gc.fillOval(-5,750, 210, 210);
    gc.fillOval(800, 20,200, 200);
    gc.fillOval(800, 750, 200, 200);
 }


private void outerlines(GraphicsContext gc) {
    gc.setFill(Color.GOLD);
    gc.setStroke(Color.RED);
    gc.fillRoundRect(50, 160, 100,630, 50, 50);
    gc.fillRoundRect(855, 160, 100,630, 50, 50);
    gc.fillRoundRect(50, 75, 850,100, 50, 50);
    gc.fillRoundRect(50, 800, 850,100, 50, 50);

}
private void inerLines(GraphicsContext gc) {
    gc.setFill(Color.MEDIUMTURQUOISE);
    gc.fillRoundRect(50, 400, 305,100, 50, 50);
    gc.fillRoundRect(600, 400, 350,100, 50, 50);
    gc.fillRoundRect(430, 160, 100,200, 50, 50);
    gc.fillRoundRect(430, 585, 100,230, 50, 50);
}

private void backGround(GraphicsContext gc) {
    gc.setFill(Color.LIME);
    gc.fillRoundRect(-200, -5, 10000, 10000, 500, 500);

}
```



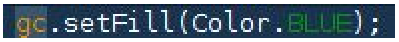
```
private void drawRectAndCircle(GraphicsContext gc) {
    gc.setFill(Color.BLUE);
    gc.setStroke(Color.BLUE);
    gc.fillRoundRect(325, 300, 300, 300, 50, 50);
    gc.strokeRoundRect(325, 300, 300, 300, 50, 50);
    gc.setFill(Color.RED);
    gc.fillOval(320, 300,300,270 );
}
```

All the values passed through each method for example like *gc .fill RoundRect(50,160,100,630,50,50);* which gives the shape a specific coordinates and the size depending of what you the in the canvest.

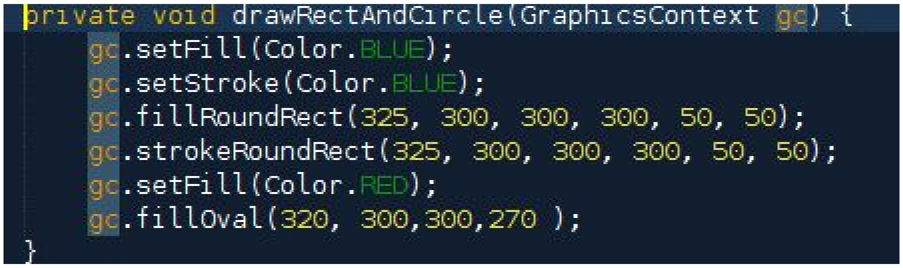
The method that give it the color is *gc.fillFill(Color.Gold);* that gives color to the shapes you gave it in these case it's the color gold.



```
gc.fillRoundRect(50, 160, 100,630, 50, 50);
```

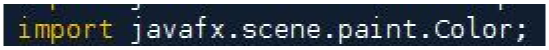


```
gc.setFill(Color.BLUE);
```



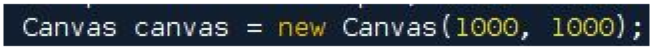
```
private void drawRectAndCircle(GraphicsContext gc) {
    gc.setFill(Color.BLUE);
    gc.setStroke(Color.BLUE);
    gc.fillRoundRect(325, 300, 300, 300, 50, 50);
    gc.strokeRoundRect(325, 300, 300, 300, 50, 50);
    gc.setFill(Color.RED);
    gc.fillOval(320, 300,300,270 );
}
```

The colors are given to us by *import javafx.scene.paint.Color;*



```
import javafx.scene.paint.Color;
```

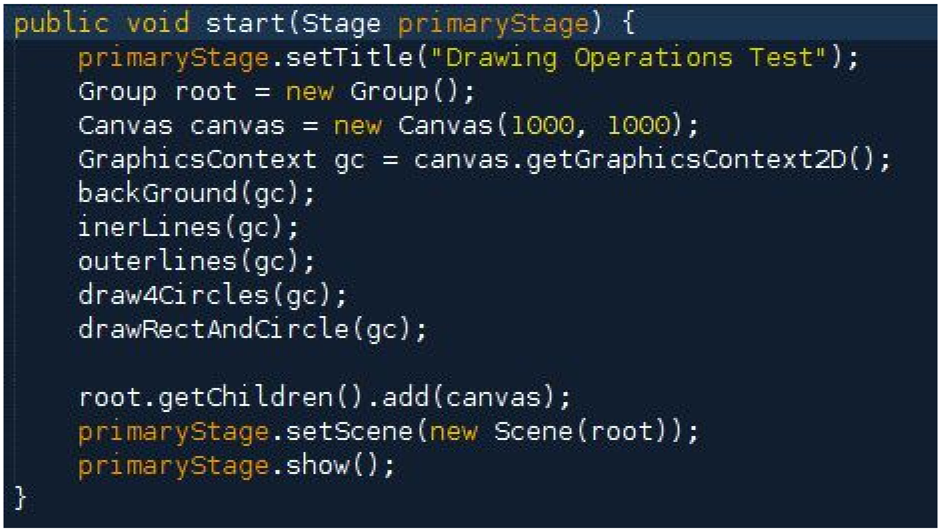
The i had a canvest were all my shapes went to. That had all the coordinated depending on how big it was. For example my canvas is 1000x1000 and its all provided to use by this *impoprt jacafx.scene.canvas.Canvas;*



```
Canvas canvas = new Canvas(1000, 1000);
```

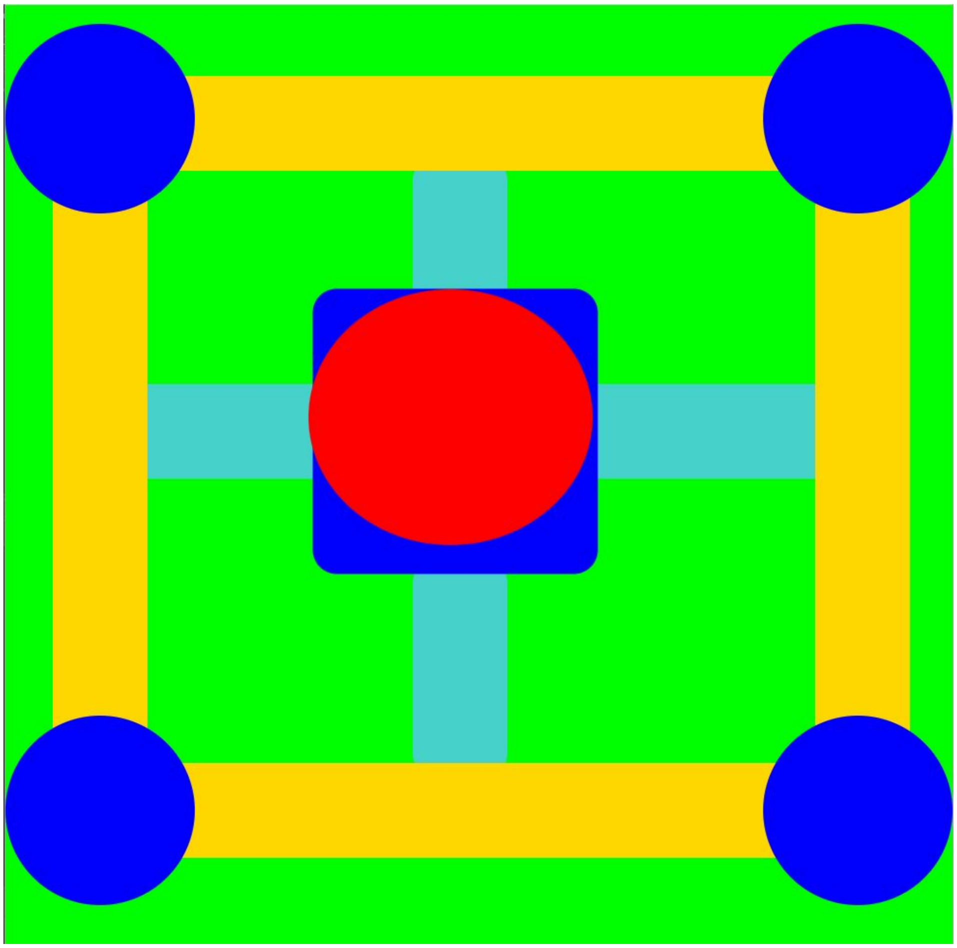
In my program I used methods and all of them were being called in the main(); if i didnt called them nothing would've ran.Also they needed to be called in order because otherwise the shape we're going to overlaps one and other or cover them completely.

```
public static void main(String[] args) {
```



```
public void start(Stage primaryStage) {
    primaryStage.setTitle("Drawing Operations Test");
    Group root = new Group();
    Canvas canvas = new Canvas(1000, 1000);
    GraphicsContext gc = canvas.getGraphicsContext2D();
    backGround(gc);
    inerLines(gc);
    outerlines(gc);
    draw4Circles(gc);
    drawRectAndCircle(gc);

    root.getChildren().add(canvas);
    primaryStage.setScene(new Scene(root));
    primaryStage.show();
}
```

All together all my methods created a geometrical art that is so bright and colorful that is some one just saw a glimpse of it they would have to turn their head again and say what is that.



At the end i think my project came out very well. I was able to successfully create what i had in mind by using methods. i had some road block on they way but with some fixing I overcame them. Now i have a way better understanding of creating art in JavaFX and hope to create way better things next time around with the new knowledge i have on JavaFX.