<center>Project essay</center>

my project, was to make methods and create "art" with geometrical figures, I made methods to

create squares, rectangles, circles, triangle. First, I start off by taking the for loop that we made

from the beginning , this creates 500 (originally), but I set it to 100000, that way I can paint all

the canvas and also to make it select random colors in order have more colorful, all this

methods takes x and y values to place them at a certain position, however, the triangle does not.

The way the program starts is public void class that sets a tittle, length, width, and color the

background as well as methods are called in this part.

The
canvas
and tittle
are made
at this
point

Methods
are called
on this part

```java
@Override
public void start(Stage primaryStage) {
    primaryStage.setTitle("what am I doing?");
    Group root = new Group();
    Canvas canvas = new Canvas(300, 250);
    GraphicsContext gc = canvas.getGraphicsContext2D();
    //drawShapes(gc);
    gc.setFill(Color.BLACK);
    gc.fillRect(0, 0, canvas.getWidth(), canvas.getHeight());
    //drawCircle(gc);
    makeCircles(gc);
    makeTriangle(gc);
    makeSquare(gc,0,30);
    makeRect(gc,0,0);
    makeSquare(gc,270,30);
    makeRect(gc,240,0);
    makeSquare(gc,0,190);
    makeRect(gc,0,220);
    makeSquare(gc,270,190);
    makeRect(gc,240,220);
    root.getChildren().add(canvas);
    primaryStage.setScene(new Scene(root));
    primaryStage.show();
}
```

To breakdown more the program, make circles method creates 10000 with random x, y, and

colors. But before all of this happens it has to create one circle before creating the rest of the

circle these are set this way

```
void makeCircles(GraphicsContext gc){
    for (int i = 0; i<10000; i++){
        int x = randal.nextInt(400);
        int y = randal.nextInt(400);
        int r = randal.nextInt(255);
        int g = randal.nextInt(255);
        int b = randal.nextInt(255);
        drawCircle( gc, x, y, r, g, b);
    }
}

void drawCircle(GraphicsContext gc, int x, int y, int r, int g, int b){
    gc.setFill(Color.rgb(r,g,b));
    gc.fillOval(x, y, 42, 42);
}
```

then it is followed by the method make triangle that takes 7 values that controls the position of where the points are printed, it also has a stroke background in the outer lines and the inside its filled with a black background.

```
void makeTriangle(GraphicsContext gc){
    gc.setFill(Color.BLACK.brighter());
    gc.setStroke(Color.AQUA.brighter());
    gc.setLineWidth(1);
    gc.fillPolygon(new double[]{0, 80, 30, 0},
                   new double[]{210, 220, 220}, 3);
    gc.strokePolygon(new double[]{0, 80, 30, 0},
                   new double[]{210, 220, 220}, 3);
}
```

Next, the method make square has the same properties as triangle as long the background is concern, except the points this does take x and y and are not as complex as the triangle, also they can have curve ends at the corners, but I decided to make them without curves

```
void makeSquare(GraphicsContext gc, int x,int y){
    gc.setFill(Color.BLACK.brighter());
    gc.setStroke(Color.FORESTGREEN.brighter());
    gc.setLineWidth(1);
    gc.fillRoundRect(x, y, 30, 30, 0, 0);
    gc.strokeRoundRect(x, y, 30, 30, 0, 0);
}
```

Finally, the last method is the same as the square one, the only difference it creates a rectangle

```
void makeRect(GraphicsContext gc,   int x,int y){
    gc.setFill(Color.BLACK.brighter());
    gc.setStroke(Color.FORESTGREEN.brighter());
    gc.setLineWidth(1);
    gc.fillRoundRect(x, y, 60, 30, 0, 0);
    gc.strokeRoundRect(x, y, 60, 30, 0, 0);
}
```

Dimension are different to create a rectangle

with all this altogether, I created a simple image that creates rectangle and square

at the corners, and one triangle
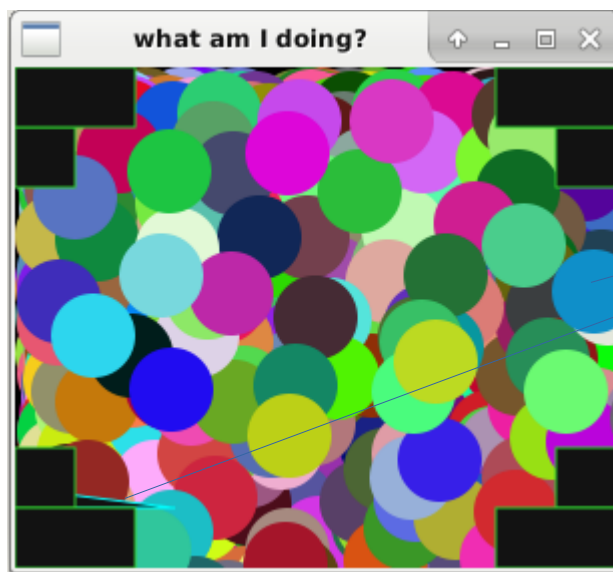
Rectangles,circles, triangles, and squares