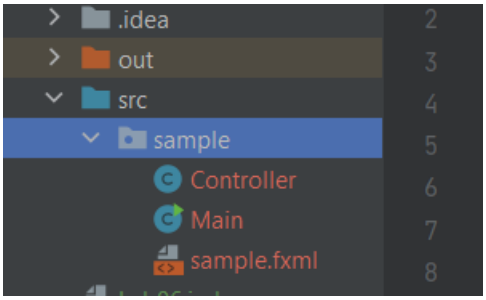
Lab09

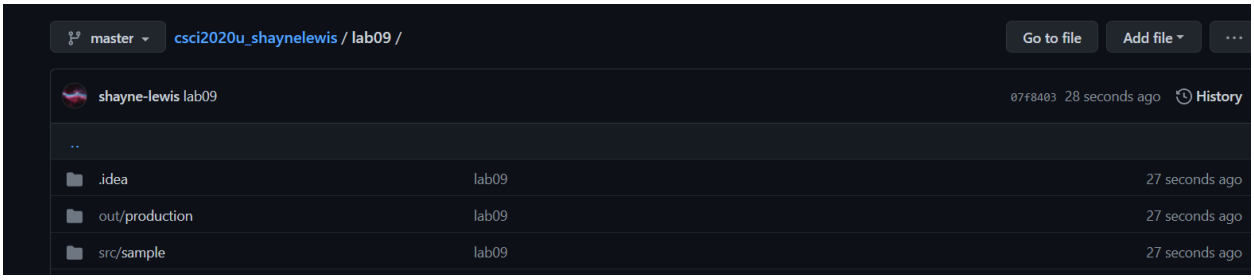
Shayne Lewis

100506658

Local Repository



Github Repository



https://github.com/shayne-lewis/csci2020u_shaynelewis/tree/master/lab09

Controller.java (only file with relevant changes)

```java
public class Controller {
    @FXML private Canvas mainCanvas;
    @FXML private GraphicsContext gc;

    private List<String[]> googleData = new ArrayList<String[]>();
    private List<String[]> appleData = new ArrayList<String[]>();
    private ArrayList<Float> googleClose = new ArrayList<>();
    private ArrayList<Float> appleClose = new ArrayList<>();

    @FXML public void initialize(){
        gc = mainCanvas.getGraphicsContext2D();
        downloadStockPrices();
        closeGoogle();
        closeApple();
        drawLinePlot(googleData, appleData);
    }
```

```java
    //gets list of columns with values
    public void downloadStockPrices(){
        try{
            //GOOGLE
            String sURL = "https://query1.finance.yahoo.com/v7/finance/download/GOOG?period1=" +
                    "1262322000&period2=1451538000&interval=1mo&events=history&includeAdjustedClose=true";
            URL netURL = new URL(sURL);

            URLConnection conn = netURL.openConnection();
            conn.setDoOutput(false);
            conn.setDoInput(true);

            InputStream inStream = conn.getInputStream();
            Scanner s = new Scanner(inStream);
            while(s.hasNext()){
                String line = s.nextLine();
                String[] columns = line.split( regex: ",");
                googleData.add(columns);
            }
```

```java
            //APPLE
            String sURL2 = "https://query1.finance.yahoo.com/v7/finance/download/AAPL?period1=" +
                    "1262322000&period2=1451538000&interval=1mo&events=history&includeAdjustedClose=true";
            URL netURL2 = new URL(sURL2);

            URLConnection conn2 = netURL2.openConnection();
            conn2.setDoOutput(false);
            conn2.setDoInput(true);

            InputStream inStream2 = conn2.getInputStream();
            Scanner s2 = new Scanner(inStream2);
            while(s2.hasNext()){
                String line2 = s2.nextLine();
                String[] columns2 = line2.split( regex: ",");
                appleData.add(columns2);
            }
        }
        catch (Exception e){
            e.printStackTrace();
        }
    }
```

```java
//adds close float values to an array list for google
public void closeGoogle(){
    boolean firstLine = true;
    for(int i = 0; i < googleData.size(); i++){
        if(!firstLine){
            googleClose.add(Float.valueOf(googleData.get(i)[4]));
        }
        else{
            firstLine = false;
        }
    }
}

//adds close float values to an array list for apple
public void closeApple(){
    boolean firstLine = true;
    for(int i = 0; i < appleData.size(); i++){
        if(!firstLine){
            appleClose.add(Float.valueOf(appleData.get(i)[4]));
        }
        else{
            firstLine = false;
        }
    }
}
```

```java
//makes axis and plots using lines
public void drawLinePlot(List googleData, List appleData){
    //x and y axis
    gc.setStroke(Color.BLACK);
    gc.strokeLine( v: 0,  v1: 50,  v2: 0,  v3: 950);
    gc.strokeLine( v: 0,  v1: 950,  v2: 800,  v3: 950);

    gc.setStroke(Color.RED);
    double x = 0;
    double y = 950;
    for(int i = 0; i < googleClose.size(); i++){
        try{
            plotLine(x, y: y - googleClose.get(i) + 50, xx: x + 10, yy: y - googleClose.get(i+1) + 50);
        }
        catch(IndexOutOfBoundsException e){
            //e.printStackTrace();
        }
        x += 10;
    }
```

```java
    //line plot apple
    double x2 = 0;
    double y2 = 900;
    gc.setStroke(Color.BLUE);
    for(int i = 0; i < appleClose.size(); i++){
        try{
            plotLine(x2, y: y2 - appleClose.get(i) + 50, xx: x2 + 10, yy: y2 - appleClose.get(i+1) + 50);
        }
        catch(IndexOutOfBoundsException e){
            //e.printStackTrace();
        }
        x2 += 10;
    }
}

//draws a line
public void plotLine(double x, double y, double xx, double yy ){
    gc.strokeLine(x,y,xx,yy);
}
}
```

Output