Introduction to Programming II Project Log

Project title:	CM1010 ITP2 coursework by Devansh Gupta
Topic:	Week 10-14 log update to Share-TTWO.js

What progress have you made this topic?

- Calculations for the scaling feature.
- Implementation of candle draw function.
- Base for loop in draw for extracting data from excel sheet using an object has been refined and applied to draw function.
- Simple line to join candles function is ready.
- Brainstorming for more features like grids line, simple line graph as an overlay and more features to make use of scaling feature is underway.

What problems have you faced and were you able to solve them?

all candles were not following the scale plotting maths that well and it took a lot of tinkering with the map() function to get it working. A lot of rough sketches had to be drawn to understand the workings of the change in scale and the math behind it.
To solve these problems I have referenced helped from other data visualizations of stock markets to help me understand.
Eventually I was able to solve them and the candles responded correctly to changes in the mousewheel event.

What are you planning to do over the next few weeks?

I plan to research more for features that are interesting and doable according to the potential of this module. I will try implementing these features in a test file and see if I can figure them out before I can finally implement it in my main project folder. I have already looked at grid lines in the graph which should be easy to implement so I will be testing it out first in the coming weeks. More features for to use the of the scaling feature will also be researched upon.

Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?

So far I am a bit ahead of schedule, as my Gantt chart had been planned to complete the scaling function by week 14 but I have some time to do some brainstorming for extra features. Therefore I am not concerned about falling behind on the time frame given as my most important feature is mostly final.

Introduction to Programming II Project Log

Project title:	CM1010 ITP2 coursework by Devansh Gupta
Topic:	Week 14-18 log update to Share-TTWO.js.

What progress have you made this topic?

Made a list of features I'd like to implement-

1. Types of graphs:
2. line
3. histogram
4. Candle (try hollow as well)
5. bar
6. //color under line graph (area graph)
7. Market volume
8. Moving averages
9. Change interval for the price movement (candle length being 1 day to 1 year)
10. Draw function for the user to draw lines on the graph for extra custom visualization
11. Y axis elements can be changed to percentage change
12. Custom date range input
13. Pointer location on the graph
14. Add color coding to grids

- Grid lines in background.
- Added pointer that snaps to exact candle being hovered over
- Line graph as an overlay on the candle chart
- Draw feature that allows the user to draw a maximum of 3 lines on the graph.
- Made this feature only possible on a static chart, if the position of the graph changes the lines get erased. Also added a clear lines option and a menu to house these features.
- Added an info panel for the lines drawn on the graph.
- Added code for the line being drawn to snap to the candle hovered over.
- Made significant progress on the structure of the code itself.
- Added a dynamic daterange feature that displays the individual dates based on the zoom level of the graph.

What problems have you faced and were you able to solve them?

As my features started increasing, my data access structure started to struggle significantly giving me load errors and lag. So I spent more time on creating a structure that would aid me with these load errors. As most of the plotting and features revolve around the excel sheet from where the data is extracted, I have a for loop that has accesses the data and distributes it to all the functions according to what data each of them need using an object.
The draw feature caused some problems too. At first the methods I used to make it possible did not work. So a bit of googling took me to a solution that allowed me solve the problem. Reference to the source will be in my references document.

What are you planning to do over the next few weeks?

I plan to to continue brainstorming for more ideas and work on the list I have created. I will have to evaluate each task according to its difficulty as some of them such as the area graph seems difficult because it requires the startShape() with the vertex() command which changes my entire drawing method for the line and candle graphs. It would require me to rethink my entire drawing strategy. In these remaining weeks I plan to finish up my feature implementations and start the testing sequence. I plan to carry out usability and black box testing, with a bit of user testing at the end before submission.

Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?

At this point I am way ahead of schedule as this week I was to finalize my scaling feature and apply it to the candles, which I completely in week 13. Since then I have been looking into extra features. Therefore I am not concerned about completion and will take this 2 weeks to finish finalise these features and finish testing and writing report.

Introduction to Programming II Project Log

Project title:	CM1010 ITP2 coursework by Devansh Gupta
Topic:	Week 18 - 20 log update to Share-TTWO.js and tech-diversity-race.js

What progress have you made this topic?

- Evaluated and discarded some features because of difficulty of execution.
- Added a moving average feature.
- Added 5 options to the menu for moving average.
- Added an info panel for candles being hovered over, different from the lines info panel.
- Optimized code structure, made more functions to make code modular.
- Carried out extensive white box and stability testing.
- Did some user testing as well.
- Made progress on report.

What problems have you faced and were you able to solve them?

This week, through my testing I discovered a lot of underlying problems I had not encountered. I discovered that my candle plotting was still slightly wrong as the ranges in the map() were in the correct order. I found this out through the testing methods mentioned above. The moving average calculations were also difficult to solve. The console.log() was used a lot to see how the values were changing. Another problem I faced was in the draw feature. When I drew a line outside the main graph area, the entire system crashed or lagged. This was solved by limiting the space in which lines can be drawn.

What are you planning to do over the next few weeks?

I plan on completing testing after some of these bugs are fixed and completing the report.

Attaching code snapshots below

	Final System Evaluation				
Test name	Full system test				
Description	Testing all the features in the program				
Test Status	complete				
Test case #	**Scenario**	**Steps**	**Expected Output**	**Actual output**	**Pass /Fail**
1.1	**Test date range menu.**	select each option one by one and change around other options on other menus to test for compatibility	it should change the number of days displayed on the graph according to the option, but it should remain static if zoom horizontally ias selected.	As expected the graph responds correctly	Pass; need to connect the zoom and scroll function better
1.2	**Test draw feature**	tap between 2 points anywhere on the canvas and see how it reponds	should only draw 3 lines inside the graph area and should snap to candles	caused a system breaking glitch when line was drawn outside graph area	fail
1.3	**Test zoom and scroll menu features**	use mousewheel to test all modes with other features	should work seemlessly without lag or disruption. Select date range shlould not work when zoom horizontally is activated	As expected the graph responds correctly	pass
1.4	**Test moving averages**	select feature in from the menu and test zooming and scrolling	should respond correctly to changes in range of the mapping function	responds correctly, however it produces a line at value 0$	pass; improvemen - to remove the line at 0$
1.5	**Repeat all tests with different types of graphs**	repeat all earlier tests for each type of graph	should respond correctly to changes all features	as expected	pass
1.6	**Switch between extensions with ease**	repeat all earlier tests and change between extensions	values should not reset and be at the position when left	as expected	pass

```
            }
        } else if(scrollOrZoomOpt == 4){
            if (event.deltaY > 0) {
                dateRange -= 2
            } else{
                dateRange += 2
            }
        }
}
```

```
//function to enable draw features. this function is connected to the mouseclicked event and
// menu4.value selection. max number of lines allowed is 10.
function drawFeature(event){
    if (drawFeatureOption == 'Enable Draw Feature' && arr.length <= 11){
        var pos = {xPos : mouseX, yPos : plotPoint, compareData : displayData}
        arr.push(pos)
    } else{
        console.log('arr full',arr.length)
    }
}
    this.setup = function(){

  //        initialising array for destruction of junk or previous data
         marketData = [];
```

```
472 function drawMovingAverage(days,datePos,date,prevDate){
473     var prevAverage = 0
474     var currentAverage = 0
475
476     if (datePos > days){
477         for (var i = datePos - days; i < datePos; i++){
478             prevAverage += marketData[i].prevMarketClose
479             currentAverage += marketData[i].marketClose
480         }
481         prevAverage = prevAverage/days
482         currentAverage = currentAverage/days
483     }
484
485     stroke(0,0,200)
486     line(prevDate,
487          map(prevAverage,yRange2,yRange1,20,500,true),
488          date,
489          map(currentAverage,yRange2,yRange1,20,500,true))
490     stroke(0)
491 }
```

```
    assigning the mouseWheel and mouseClicked function to canvas
    c.mouseWheel(changeScale);
    c.mouseClicked(drawFeature);

    array containing objects
    var t = this.table1;
    var table_rows = t.getRows();
    for (var i = 1; i <  table_rows.length; i++ ){
       object for storing data in structured manner and pushed to marketData array
        var data = {
            dateID:t.getString(i,0),
            marketOpen:t.getNum(i,1),
            prevMarketOpen:t.getNum(i - 1,1),
            high:t.getNum(i,2),
            low:t.getNum(i,3),
            marketClose:t.getNum(i,4),
            prevMarketClose:t.getNum(i - 1,4),
            volume:t.getNum(i,6),
            datePos:t.getNum(i,7),
            date:t.getString(i,8)
        }
        marketData.push(data);

function drawLineGraph(marketClose,date,prevMarketClose,prevDate){
    line(prevDate,map(prevMarketClose,yRange2,yRange1,20,500,true),date,map(marketClose,yRange2,yRange1,20,500,true))
}

function drawCandle(marketOpen,marketClose,date,high,low){
//    candle structure calculations
//    making the wick for the candle
    line(date,map(high,yRange2,yRange1,20,500,true),date,map(low,yRange2,yRange1,20,500,true))

//    making the body of the candle
//    marking the candles as red or green according to the values in the csv
    if (marketClose > marketOpen){
        graphColor = color(0, 220, 0);
        fill(graphColor)
        rect(date - 2.5,
             map(marketClose,yRange2,yRange1,20,500,true),
             5,
             map(marketOpen,yRange2,yRange1,20,500,true) - map(marketClose,yRange2,yRange1,20,500,true))
    } else {
        graphColor = color(220, 0, 0);
        fill(graphColor)
        rect(date - 2.5,
            map(marketOpen,yRange2,yRange1,20,500,true),
            5,
            map(marketClose,yRange2,yRange1,20,500,true) - map(marketOpen,yRange2,yRange1,20,500,true))
    }
    noFill()
}
```

```
//this  function is called when the mouseWheel event is triggered
function changeScale(event){

    if (scrollOrZoomOpt == 1){
        if (event.deltaY > 0) {
            yRange1 -= 5
            yRange2 += 5

        } else {
            yRange1 += 5
            yRange2 -= 5

            yRange1 = min(yRange1,120);
            yRange2 = max(yRange2,130);
        }

    } else if(scrollOrZoomOpt == 2){

        var temp1 = yRange1
        var temp2 = yRange2

        if (event.deltaY > 0) {

            yRange1 -= ((temp2 - temp1) / 20)
            yRange2 -= ((temp2 - temp1) / 20)

            } else {
                yRange1 += ((temp2 - temp1) / 20)
                yRange2 += ((temp2 - temp1) / 20)
            }
        } else if(scrollOrZoomOpt == 3){
            if (event.deltaY > 0) {
                datePos += 20
            } else{
                datePos -= 20
            }
```