

Overview

This final homework will provide an overview of how to manage charting. In this homework, we:

- (1) Learn how to use chart.js to draw a line chart
- (2) Learn how to use chart.js to draw a bar chart
- (3) Learn how to use chart.js to draw a pie chart
- (4) Learn about the color picker
- (5) Learn how to use chart.js with PHP data

Exercise 1: Learning Chart.js


Chart.js is a JavaScript add-on library that helps you quickly make charts. In this library, you define an area of the screen you want the chart to be on with the canvas tag. You then use the Chart.js/JavaScript command `new Chart` to create the chart. There are a number of different arguments you can put into a chart, but only three are relevant for this class, the type, datasets, and options. Type defines the type of chart. Datasets specifies the data for the chart. Options specifies things like what the label of the axes will be.

Chart.js needs to render the chart AFTER the rest of the screen is built. As a result, we put the JavaScript for chart.js NEAR THE END of the HTML block, rather than at the beginning. Fundamentally, a chart.js chart will look something like this:

```
<body>
  <canvas id="mychartid" width="chart width" height="chart height" />
  <script>
    var ctx=document.getElementById("mychartid");
    var myChart=new Chart(ctx,
      {
        type: 'line' or 'bar' or 'pie',
        datasets: {datasets here},
        options: {options here}
      }
    );
  </script>
</body>
```

What is in datasets and options will vary by the chart type.

Download the file `mychart.php`. Run it just to see what's there.



Bar Chart	Line Chart	Pie Chart
<input type="radio"/> Include drinks for store 1Something here	<input type="radio"/> Include drinks for store 1Something here	<input type="radio"/> Store 1 vs Store 2 Drink Sales
<input type="radio"/> Include drinks for store 2Something here	<input type="radio"/> Include drinks for store 2Something here	<input type="radio"/> Store 1 vs Store 2 Food Sales
<input type="radio"/> Include food for store 1Something here	<input type="radio"/> Include food for store 1Something here	<input type="radio"/> Store 1 vs Store 2 Complaints
<input type="radio"/> Include food for store 2Something here	<input type="radio"/> Include food for store 2Something here	<input type="radio"/> Store 1: Something here Store 2: Something here
<input type="radio"/> Include complaints for store 1Something here	<input type="radio"/> Include complaints for store 1Something here	
<input type="radio"/> Include complaints for store 2Something here	<input type="radio"/> Include complaints for store 2Something here	

Draw Chart

For now, we're going to use the bottom black space to experiment with drawing charts.

To use chart.js, we have to link our code to the chart.js library. To do this, we need the line `<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.5.0/Chart.min.js"></script>` in our Javascript (it is already there).

Now, let's create our canvas.

Add the following lines to the code:

```
</table>
<canvas id="myChart" width="1000" height="400" style="background:
white;"></canvas>
</form>
```

The screenshot shows a configuration menu for a chart. It has three main sections: 'Bar Chart', 'Line Chart', and 'Pie Chart'. Each section contains several radio button options for different data series. A 'Draw Chart' button is located at the bottom of the menu. Below the menu is a large empty black rectangular area representing the canvas.

```
<script>
var ctx=document.getElementById("myChart");
var myChart=new Chart(ctx,{ //begin Chart
  type: 'line',
  data:{
    labels: [1,2,3,4,5],
    datasets: [
      { //begin dataset 1
        data: [30,40,20,30,50],
        label: "Store 1",
        borderColor: "#ff0000",
        fill: false
      }, //end dataset 1
      { //begin dataset 2
```

```

    data: [100,140,50,70,120],
    label: "Store 2",
    borderColor: "#00ff00",
    fill: false
  } //begin dataset 2
] //end datasets
},//end data
options: { //begin options
scales: { //begin scales
  xAxes: [{
    display: true,
    scaleLabel: { //begin scaleLabel
      display: true,
      labelString: 'Month'
    } //end scaleLabel
  }], //end xAxes
  yAxes: [{ //begin yAxes
    display: true,
    scaleLabel: { //begin scaleLabel
      display: true,
      labelString: 'Quantity'
    } //end scaleLabel
  } ] //end yAxes
} //end scales
} //end options
});//end Chart
</script>

```

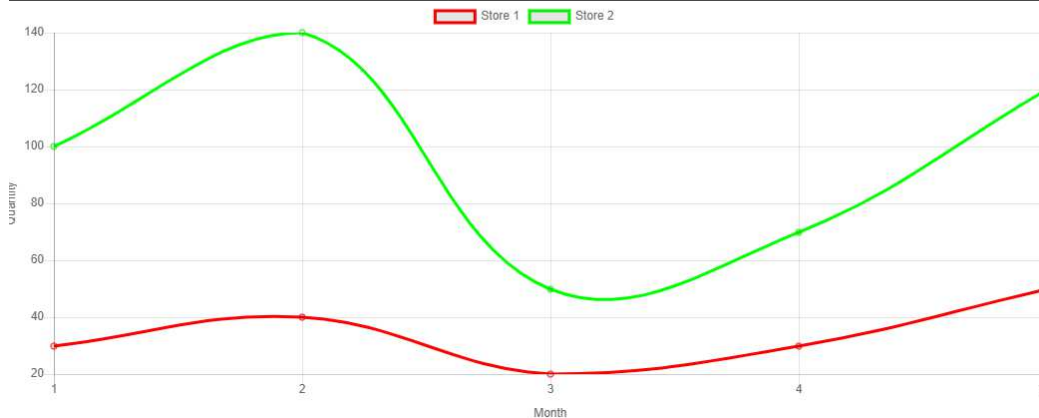
I know it is really difficult to type all of this, but you just aren't going to be able to see the complete chart unless all of this stuff goes in there. Note the //begin and //end remarks as you type and make sure you have the correct open parentheses in the begin to match the end parentheses at the end. If you typed everything correctly, your webpage will now look like this:

Bar Chart
 Include drinks for store 1Something here
 Include drinks for store 2Something here
 Include food for store 1Something here
 Include food for store 2Something here
 Include complaints for store 1Something here
 Include complaints for store 2Something here

Line Chart
 Include drinks for store 1Something here
 Include drinks for store 2Something here
 Include food for store 1Something here
 Include food for store 2Something here
 Include complaints for store 1Something here
 Include complaints for store 2Something here

Pie Chart
 Store 1 vs Store 2 Drink Sales
 Store 1 vs Store 2 Food Sales
 Store 1 vs Store 2 Complaints
 Store 1: Something here Store 2: Something here

Draw Chart



What does all this mean?

type: 'line' means the type of chart (i.e., a line chart).

labels: [1,2,3,4,5] are the labels on the X axis.

```

{ //begin dataset 1
data: [30,40,20,30,50],
label: "Store 1",
borderColor: "#ff0000",
fill: false
}, //end dataset 1
    
```

Is the first line on the chart. The border color denotes the color red. There are five data points (30, 40, 20, 30, 50) corresponding to our five labels.

```

{ //begin dataset 2
data: [100,140,50,70,120],
label: "Store 2",
borderColor: "#00ff00",
fill: false
} //begin dataset 2
    
```

Is our green line.

The long options list basically defines our X and Y axes. Those commands tell the chart that we are going to draw the X and Y axes and label them as “Month” and “Quantity” respectively.

Copy your chart code to someplace safe. We are going to need it again later, and if you lose it, you will have to retype it again. Then change the bolded text. Also delete the two fill: false lines and note the comma changes.

```

<script>
var ctx=document.getElementById("myChart");
var myChart=new Chart(ctx,{ //begin Chart
  type: 'bar',
  data: {
    labels: [1,2,3,4,5],
    datasets: [
      { //begin dataset 1
        data: [30,40,20,30,50],
        label: "Store 1",
        backgroundColor: "#ff0000"
      }, //end dataset 1
      { //begin dataset 2
        data: [100,140,50,70,120],
        label: "Store 2",
        backgroundColor: "#00ff00"
      } //begin dataset 2
    ] //end datasets
  },//end data
  options: { //begin options
    scales: { //begin scales
      xAxes: [{
        display: true,
        scaleLabel: { //begin scaleLabel
          display: true,
          labelString: 'Month'
        } //end scaleLabel
      }], //end xAxes
      yAxes: [{ //begin yAxes
        display: true,
        scaleLabel: { //begin scaleLabel
          display: true,
          labelString: 'Quantity'
        } //end scaleLabel
      }] //end yAxes
    } //end scales
  } //end options
});//end Chart
</script>

```

If you did everything correctly, you should see this:

Bar Chart

- Include drinks for store 1Something here
- Include drinks for store 2Something here
- Include food for store 1Something here
- Include food for store 2Something here
- Include complaints for store 1Something here
- Include complaints for store 2Something here

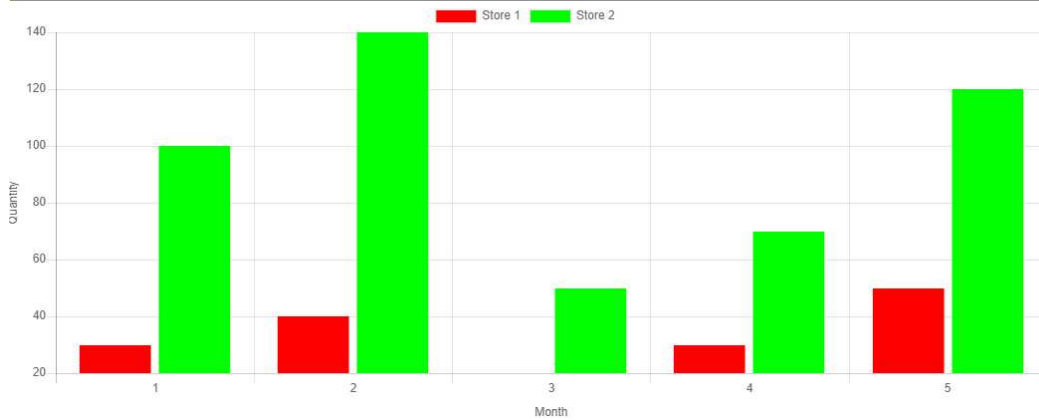
Line Chart

- Include drinks for store 1Something here
- Include drinks for store 2Something here
- Include food for store 1Something here
- Include food for store 2Something here
- Include complaints for store 1Something here
- Include complaints for store 2Something here

Pie Chart

- Store 1 vs Store 2 Drink Sales
- Store 1 vs Store 2 Food Sales
- Store 1 vs Store 2 Complaints

Store 1: Something here Store 2: Something here



Again, save everything in the script tags somewhere safe, and then replace everything in the script tags with the following code:

```

<script>
var ctx=document.getElementById("myChart");
var myChart=new Chart(ctx,{ //begin Chart
  type: 'pie',
  data: {
    labels: ['Store 1: 10','Store 2: 20', 'Store 3: 30'],
    datasets: [
      { //begin dataset 1
        data: [10, 20, 30],
        backgroundColor: ["#ff0000", "#00ff00", "#0000ff"]
      } //end dataset 1
    ] //end datasets
  } //end data
}); //end Chart
</script>

```

If you did everything correctly, you should get this:

```

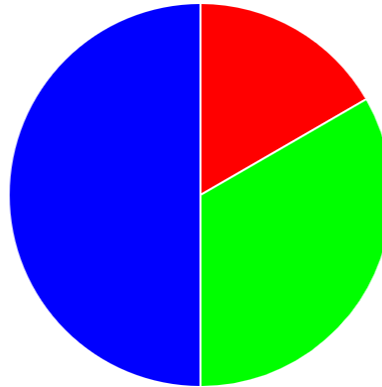
● Bar Chart
  ▣ Include drinks for store 1Something here
  ▣ Include drinks for store 2Something here
  ▣ Include food for store 1Something here
  ▣ Include food for store 2Something here
  ▣ Include complaints for store 1Something here
  ▣ Include complaints for store 2Something here

● Line Chart
  ▣ Include drinks for store 1Something here
  ▣ Include drinks for store 2Something here
  ▣ Include food for store 1Something here
  ▣ Include food for store 2Something here
  ▣ Include complaints for store 1Something here
  ▣ Include complaints for store 2Something here

● Pie Chart
  ● Store 1 vs Store 2 Drink Sales
  ● Store 1 vs Store 2 Food Sales
  ● Store 1 vs Store 2 Complaints
  Store 1: Something here Store 2: Something here
  
```

Draw Chart

■ Store 1: 10
 ■ Store 2: 20
 ■ Store 3: 30



Exercise 2: The Color Picker

You will observe that one of the things we need to do with our charts is determine the colors of the various things on the chart. JavaScript has a built-in color picker. The color picker allows the user to select a color and returns a six digit hexadecimal value that maps to that color.

In the webpage, you will notice the words “Color picker here” a lot. We are going to replace those with color pickers. For the bar chart/line chart, the color pickers should be called picker followed by a number. Their default colors should be red, orange, yellow, green, blue, purple (the colors of the rainbow). These are color codes #FF0000, #FFA500, #FFFF00, #008000, #0000FF, #800080

For example, the modification to the first bar/line color picker would be:

```

&nbsp;&nbsp;&nbsp;<input type="checkbox"
      id="param0"
      value="Y"
      disabled
    /><label for="param0">Include drinks for store 1</label>
      <input type="color"
        id="picker0"
        value="#FF0000"
        disabled
      />
    <br />
  
```

Then add the following lines to the function controlboxes.

```

function controlboxes(whichchart)
{
  if (whichchart=='p')
  
```

```

var ispie=true;
for (x=0; x<6; x++){
  document.getElementById("param"+x).disabled=ispie;
  document.getElementById("picker"+x).disabled=ispie;
  if (x % 2==0)
    document.getElementById("value"+x).disabled=!ispie;
  if (x<2)
    document.getElementById("piepick"+x).disabled=!ispie;
}
}

```

If you did everything correctly, this is what you should see:

Bar Chart
 Line Chart
 Pie Chart

Include drinks for store 1 

Include drinks for store 2 

Include food for store 1 

Include food for store 2 

Include complaints for store 1 

Include complaints for store 2 

Store 1 vs Store 2 Drink Sales
 Store 1 vs Store 2 Food Sales
 Store 1 vs Store 2 Complaints

Colors:  Store 1  Store 2

Exercise 3: Charts and the Database

Now, we can actually draw the charts.

We first need to prepare a number of SQL queries to do this. We basically need:

- An SQL query to generate the labels of the bar and line chart
- An SQL query to produce the data for the bar and line chart
- An SQL query to produce the pie chart

Create a PHP file called storemonth.php

```

<?php
  $mydb = new PDO('mysql:host=localhost;dbname=<your database here>', '<your
username here>', '<your password here>');
  $stmt=$mydb->prepare("select distinct storemonth "
                        "from storestats "
                        "order by storemonth"
                        );
  $stmt->execute();
  $result=$stmt->fetchAll();
  echo json_encode($result);
?>

```

Create a PHP file called storestats.php


```

<?php
    $store=$_REQUEST['store'];
    $statype=$_REQUEST['statype'];
    $mydb = new PDO('mysql:host=localhost;dbname=<your database name', '<your
username>', '<your password>');
    $stmt=$mydb->prepare("select stat ".
                        "from storestats ".
                        "where store=:store ".
                        "and statype=:statype ".
                        "order by storemonth"
                    );
    $stmt->bindParam("store",$store,PDO::PARAM_INT);
    $stmt->bindParam("statype",$statype,PDO::PARAM_STR);
    $stmt->execute();
    $result=$stmt->fetchAll();
    echo json_encode($result);
?>

```

Create a file called piestats.php

```

<?php
    $statype=$_REQUEST['statype'];
    $jsonfile = file_get_contents('amazoniainfo.json');
    $dbinfo = json_decode($jsonfile, false);
    $mydb = new PDO('mysql:host=localhost;dbname='.$dbinfo->dbname, $dbinfo-
>dbuser, $dbinfo->dbpass);
    $stmt=$mydb->prepare("select store, sum(stat) as sm ".
                        "from storestats ".
                        "where statype=:statype ".
                        "group by store ".
                        "order by store"
                    );
    $stmt->bindParam("statype",$statype,PDO::PARAM_STR);
    $stmt->execute();
    $result=$stmt->fetchAll();
    echo json_encode($result);
?>

```

Now we are ready to create our chart.

Just before the end form tag, add the following lines:

```
<canvas id="drawhere" style="height: 500px; width: 500px; border: 1px solid">
```

```
</canvas>
```

Now, we have to put in the code to set up our charts:

```

    const storeparams=[1,2,1,2,1,2];
    const statype=['drinks','drinks','food','food','complaints','complaints'];
    var thechart=null;
    var bllabeldata=[];

```

```

function loadlabels(){
  const xhttp = new XMLHttpRequest();
  xhttp.onload = function() {
    var monthdata = JSON.parse(this.responseText);
    for (x=0; x<monthdata.length;x++)
      bllabeldata.push(monthdata[x].storemonth);
  }
  xhttp.open("GET", "storemonth.php", true);
  xhttp.send();
}

function drawpie(){
  drawhere.innerHTML="";
  if (!(thechart==null))
    thechart.destroy();
  thechart=new Chart(drawhere,{ //begin Chart
    type: 'pie',
    data:{
      labels: ['Store 1','Store 2'],
      datasets: [
        { //begin dataset 1
          data: [],
          backgroundColor: [piepick0.value,piepick1.value]
        } //end dataset 1
      ] //end datasets
    } //end data
  }); //end Chart
}

function drawbarorline(whichtype){
  drawhere.innerHTML="";
  if (!(thechart==null))
    thechart.destroy();
  var thetype="";
  if (whichtype=='l')
    thetype='line';
  else
    thetype='bar';
  thechart=new Chart(drawhere,{ //begin Chart
    type: thetype,
    data:{
      labels: bllabeldata,
      datasets: []
    } //end data
    options: { //begin options
      scales: { //begin scales
        x: {
          display: true,
          title: { //begin scaleLable

```

```

        display: true,
        text: "Month"
    } //end scaleLabel
}, //end xAxes
y: { //begin yAxes
    display: true,
    title: { //begin scaleLabel
        display: true,
        text: 'Quantity'
    } //end scaleLabel
    } //end yAxes
} //end scales
} //end options
}); //end Chart

```

```

}

```

The constant arrays storeparams and stattypes are information we are sending to the PHP files to perform the queries.

The function loadlabels runs the PHP query to do labels for the bar and line charts. Drawpie sets up the pie chart, while drawbarorline draws the basic bar or line chart.

Now, we need to link these functions to the right places.

Link loadlabels to the body onload event. Then, modify controlboxes as follows:

```

function controlboxes(whichchart)
{
    if (whichchart=='p')
        var ispie=true;
    for (x=0; x<6; x++){
        document.getElementById("param"+x).disabled=ispie;
        document.getElementById("picker"+x).disabled=ispie;
        if (x % 2==0)
            document.getElementById("value"+x).disabled=!ispie;
        if (x<2)
            document.getElementById("piepick"+x).disabled=!ispie;
    }
    if (ispie)
        drawpie();
    else
        drawbarorline(whichchart);
}

```

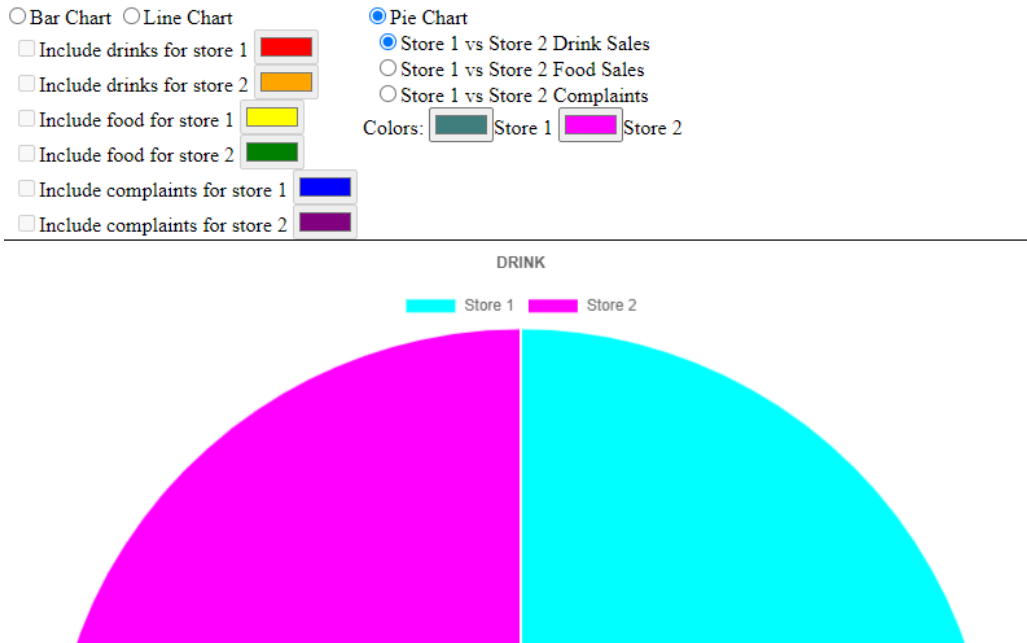
If you have done everything correctly, blank bar, line and pie charts will appear when you click those radio buttons:



Now let's actually fill in the data. The pie chart is significantly easier, so let's do that first. Create the following function:

```
function dopie(){
    var x=0;
    var haschecked=false;
    while (x<3 && ! haschecked)
        if (document.getElementsByName("whichpie").item(x).checked)
            haschecked=true;
        else
            x++;
    if (x<3){
        thechart.options.plugins.title.text=stattype[x*2].toUpperCase();
        thechart.options.plugins.title.display=true;
        const xhttp = new XMLHttpRequest();
        xhttp.onload = function() {
            var phpdata = JSON.parse(this.responseText);
            thechart.data.datasets[0].data=[];
            for (x=0; x<phpdata.length;x++)
                thechart.data.datasets[0].data.push(phpdata[x].sm);
            thechart.update();
        }
        xhttp.open("GET", "piestats.php?stattype="+stattype[x*2], true);
        xhttp.send();
    }
}
```

Link this function to the whichpie radio buttons. If you did everything correctly, you should get:



We also want to set it so changing the color picker values changes the color in the chart.

Add the following function:

```
function dopiecolor(whichcolor){
```

```
  thechart.data.datasets[0].backgroundColor[whichcolor]=document.getElementById("
  piepick"+whichcolor).value;
  thechart.update();
}
```

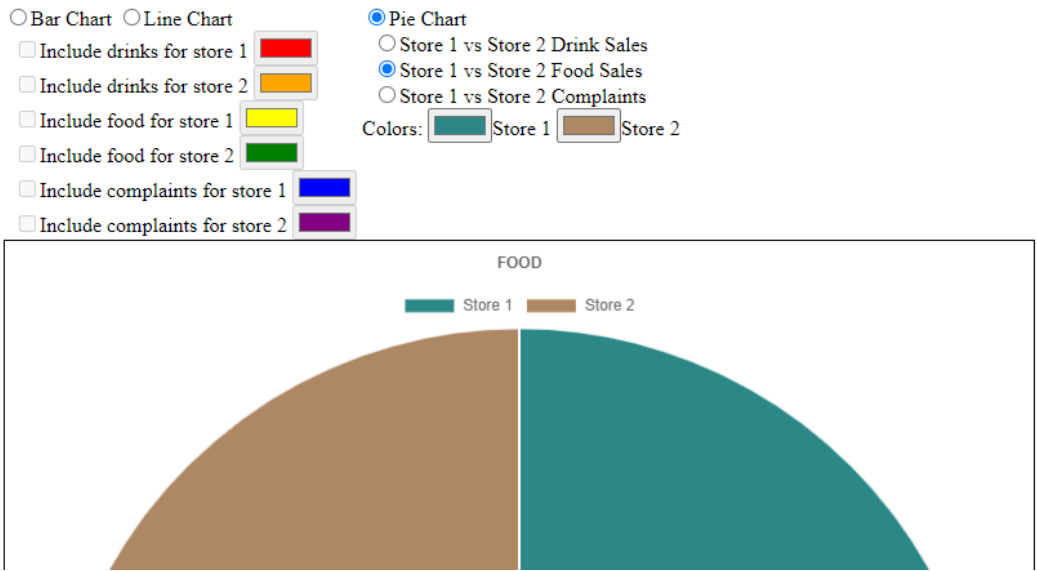
Link this to the color pickers associated with the pie chart:

```

<input type="color"
  id="piepick0"
  value="#00FFFF"
  oninput="dopiecolor(0);"
  disabled
/><label for="piepick0">Store 1</label>
<input type="color"
  id="piepick1"
  value="#FF00FF"
  oninput="dopiecolor(1);"
  disabled
/><label for="piepick1">Store 2</label>

```

If you did this correctly, changing the color picker will change the pie chart colors.



Now let's draw the line and bar graphs:

```

var bllinedata=[,,,,,];

function dobarline(whichcheck){

    if (document.getElementById("param"+whichcheck).checked){
        const xhttp = new XMLHttpRequest();
        xhttp.onload = function() {
            var phpdata = JSON.parse(this.responseText);
            var linedata=[];
            for (x=0; x<phpdata.length;x++)
                linedata.push(phpdata[x].stat);
            bllinedata[whichcheck]=linedata;
            refreshbarline();
        }
        xhttp.open("GET",

"storestats.php?store="+storeparams[whichcheck]+"&stattype="+stattype[whichcheck],
                true
            );
        xhttp.send();
    }
    else
        refreshbarline();
}

function refreshbarline(){
    var x=0;
    var y=0;
    var chartdatatsruc=null;
    thechart.data.datasets=[];
    for (x=0; x<storeparams.length; x++){

```

```

    if (document.getElementById("param"+x).checked){
        if (barchart.checked)
            chartdatastruc={ data: [],
                label: 'Store '+storeparams[x]+' '+stattype[x],
                backgroundColor:
document.getElementById("picker"+x).value,
                fill: true
            };
        else
            chartdatastruc={ data: [],
                label: 'Store '+storeparams[x]+' '+stattype[x],
                borderColor: document.getElementById("picker"+x).value,
                fill: false
            };

        chartdatastruc.data=bllinedata[x];
        thechart.data.datasets.push(chartdatastruc);
    }
}
thechart.update();
}

```

We also need to add one line to drawbarorline:

```

function drawbarorline(whichtype){
.
.
.
refreshbarline();
}

```

Finally, for all the bar/line color pickers, set the oninput event to
oninput="refreshbarline()";

If you did everything correctly, you should be able to do this:

Bar Chart Line Chart

Include drinks for store 1

Include drinks for store 2

Include food for store 1

Include food for store 2

Include complaints for store 1

Include complaints for store 2

Pie Chart

Store 1 vs Store 2 Drink Sales

Store 1 vs Store 2 Food Sales

Store 1 vs Store 2 Complaints

Colors: Store 1 Store 2

