*drawAreaChart*();  
*drawBarChart*();  
  
  
**function** *drawAreaChart*() {  
  
 *// Margin object with properties for the four directions***var** margin = {**top**: 40, **right**: 10, **bottom**: 40, **left**: 60};  
  
 *// Chart size***var** width = 620 - margin.**left** - margin.**right**,  
 height = 400 - margin.**top** - margin.**bottom**;  
  
 *// Formatting function to convert strings to date objects***var** parseDate = **d3**.**timeParse**(**"%Y-%m-%d"**);  
  
 *// Formatting function display a date object as a string***var** formatDate = **d3**.**timeFormat**(**"%Y-%m-%d"**);  
  
 *// Once we have the xValue (mouse event), we can use 'array bisector' to find the index of the value in the array***var** bisectDate = **d3**.**bisector**(**function**(d) { **return** d.**date**; }).**left**;  
  
 *// SVG drawing area (corresponds to the D3 margin convention)***var** svg = **d3**.select(**"#area-chart"**).**append**(**"svg"**)  
 .**attr**(**"width"**, width + margin.**left** + margin.**right**)  
 .**attr**(**"height"**, height + margin.**top** + margin.**bottom**)  
 .**append**(**"g"**)  
 .**attr**(**"transform"**, **"translate("** + margin.**left** + **","** + margin.**top** + **")"**);  
  
 *// Load CSV file***d3**.**csv**(**"data/zaatari-refugee-camp-population.csv"**, **function**(data){  
  
 *// Convert strings to numbers/dates*data.forEach(**function**(d) {  
 d.**population** = +d.**population**;  
 d.**date** = parseDate(d.**date**);  
 });  
  
  
 *// Create scale and axis functions***var** x = **d3**.**scaleTime**()  
 .range([0, width])  
 .domain(**d3**.extent(data, **function**(d) { **return** d.**date**; }));  
  
 **var** y = **d3**.scaleLinear()  
 .range([height, 0])  
 .domain([0, **d3**.max(data, **function**(d) { **return** d.**population**; })]);  
  
 **var** xAxis = **d3**.axisBottom()  
 .scale(x)  
 .tickFormat(**d3**.**timeFormat**(**"%b %Y"**));  
  
 **var** yAxis = **d3**.axisLeft()  
 .scale(y);  
  
  
 *// The area function transforms data points into a shape***var** area = **d3**.area()  
 .**x**(**function**(d) { **return** x(d.**date**); })  
 .y0(height)  
 .y1(**function**(d) { **return** y(d.**population**); });  
  
 *// Append a path and call the area function  
 // D3 uses each data point and passes it to the area function. The area function translates the data into positions on the path in the SVG.***var** timePath = svg.**append**(**"path"**)  
 .**datum**(data)  
 .**attr**(**"class"**, **"area"**)  
 .**attr**(**"d"**, area);  
  
  
 *// Draw axes and chart title***var** xAxisGroup = svg.**append**(**"g"**)  
 .**attr**(**"class"**, **"x-axis axis"**)  
 .**attr**(**"transform"**, **"translate(0,"** + height + **")"**)  
 .call(xAxis);  
  
 **var** yAxisGroup = svg.**append**(**"g"**)  
 .**attr**(**"class"**, **"y-axis axis"**)  
 .call(yAxis)  
  
 **var** chartTitle = svg.**append**(**"text"**)  
 .**attr**(**"class"**, **"chart-title"**)  
 .**attr**(**"y"**, -30)  
 .**attr**(**"x"**, width/2)  
 .**attr**(**"dy"**, **".71em"**)  
 .**style**(**"text-anchor"**, **"middle"**)  
 .text(**"Camp Population"**);  
  
  
 *// Create a group for all the tooltip elements and hide it***var** focus = svg.**append**(**"g"**)  
 .**attr**(**"class"**, **"focus"**)  
 .**style**(**"display"**, **"none"**);  
  
 *// Append a vertical tooltip line*focus.**append**(**"line"**)  
 .**attr**(**"stroke"**, **"#824C2A"**)  
 .**attr**(**"y1"**, 0)  
 .**attr**(**"y2"**, height)  
 .**attr**(**"x1"**, 0)  
 .**attr**(**"x2"**, 0);  
  
 *// Append an empty SVG text element for the tooltip population value*focus.**append**(**"text"**)  
 .**attr**(**"class"**, **"focus-population"**)  
 .**attr**(**"x"**, 10)  
 .**attr**(**"y"**, 10)  
 .**attr**(**"dy"**, **".35em"**);  
  
 *// Append an empty SVG text element for the tooltip date value*focus.**append**(**"text"**)  
 .**attr**(**"class"**, **"focus-date"**)  
 .**attr**(**"x"**, 10)  
 .**attr**(**"y"**, 30)  
 .**attr**(**"dy"**, **".35em"**);  
  
 *// Append a rectangle over the whole chart to capture 'mouse events'*svg.**append**(**"rect"**)  
 .**attr**(**"class"**, **"overlay"**)  
 .**attr**(**"width"**, width)  
 .**attr**(**"height"**, height)  
 .on(**"mouseover"**, **function**() { focus.**style**(**"display"**, **null**); })  
 .on(**"mouseout"**, **function**() { focus.**style**(**"display"**, **"none"**); })  
 .on(**"mousemove"**, *mousemove*);  
  
 *// Get the actual data of the current mouse position, update the coordinates and set the tooltip values***function** *mousemove*() {  
  
 **var** x0 = x.invert(**d3**.**mouse**(**this**)[0]);  
  
 **var** i = bisectDate(data, x0, 1);  
 *//var d0 = data[i - 1];***var** d = data[i];  
 *//var d = x0 - d0.date > d1.date - x0 ? d1 : d0;  
  
 // Shift the whole tooltip group on the x-axis*focus.**attr**(**"transform"**, **"translate("** + x(d.**date**) + **",0)"**);  
  
 *// Update the tooltip text properties*focus.select(**".focus-date"**).text(formatDate(d.**date**));  
 focus.select(**".focus-population"**).text(d.**population**);  
 }  
 });  
}  
  
**function** *drawBarChart*() {  
  
 *// Create an array with JSON objects***var** data = [  
 { **"shelter"**:**"Caravans"**, **"value"**:0.7968 },  
 { **"shelter"**:**"Combination\*"**, **"value"**:0.1081 },  
 { **"shelter"**:**"Tents"**, **"value"**:0.0951 }  
 ];  
  
  
 *// Specify the SVG drawing area***var** margin = {**top**: 40, **right**: 10, **bottom**: 40, **left**: 40};  
  
 **var** width = 400 - margin.**left** - margin.**right**,  
 height = 400 - margin.**top** - margin.**bottom**;  
  
 **var** svg = **d3**.select(**"#bar-chart"**).**append**(**"svg"**)  
 .**attr**(**"width"**, width + margin.**left** + margin.**right**)  
 .**attr**(**"height"**, height + margin.**top** + margin.**bottom**)  
 .**append**(**"g"**)  
 .**attr**(**"transform"**, **"translate("** + margin.**left** + **","** + margin.**top** + **")"**);  
  
  
 *// Create an ordinal scale for the three shelter types***var** x = **d3**.scaleBand()  
 .domain(data.map(**function**(d) { **return** d.**shelter**; }))  
 .rangeRound([0, width])  
 .paddingInner(0.1);  
  
 *// Create a linear scale for the percentage values (domain: 0 - 1)***var** y = **d3**.scaleLinear()  
 .domain([0, 1])  
 .range([height, 0]);  
  
  
 *// Draw axes***var** xAxis = **d3**.axisBottom()  
 .scale(x);  
  
 **var** yAxis = **d3**.axisLeft()  
 .scale(y)  
 .tickFormat(**d3**.*format*(**".0%"**));  
  
 **var** xAxisGroup = svg.**append**(**"g"**)  
 .**attr**(**"class"**, **"x-axis axis"**)  
 .**attr**(**"transform"**, **"translate(0,"** + height + **")"**)  
 .call(xAxis);  
  
 **var** yAxisGroup = svg.**append**(**"g"**)  
 .**attr**(**"class"**, **"y-axis axis"**)  
 .call(yAxis);  
  
 *// Draw the actual bars/columns of the bar chart***var** bar = svg.**selectAll**(**".bar"**)  
 .**data**(data)  
 .**enter**().**append**(**"rect"**)  
 .**attr**(**"class"**, **"bar"**)  
 .**attr**(**"x"**, **function**(d) { **return** x(d.**shelter**); })  
 .**attr**(**"y"**, **function**(d) { **return** y(d.**value**); })  
 .**attr**(**"height"**, **function**(d) { **return** height - y(d.**value**); })  
 .**attr**(**"width"**, x.bandwidth());  
  
 *// Append labels at the top of the bars***var** barLabel = svg.**selectAll**(**".bar-label"**)  
 .**data**(data)  
 .**enter**().**append**(**"text"**)  
 .**attr**(**"class"**, **"bar-label"**)  
 .**attr**(**"x"**, **function**(d) { **return** x(d.**shelter**) + (x.bandwidth()/2); })  
 .**attr**(**"y"**, **function**(d) { **return** y(d.**value**)-10; })  
 .**style**(**"text-anchor"**, **"middle"**)  
 .text(**function**(d) { **return** (d.**value** \* 100).toFixed(2) + **" %"** });  
  
 *// Append a chart title***var** chartTitle = svg.**append**(**"text"**)  
 .**attr**(**"class"**, **"chart-title"**)  
 .**attr**(**"y"**, -30)  
 .**attr**(**"x"**, width/2)  
 .**attr**(**"dy"**, **".71em"**)  
 .**style**(**"text-anchor"**, **"middle"**)  
 .text(**"Type of Shelter"**);  
}