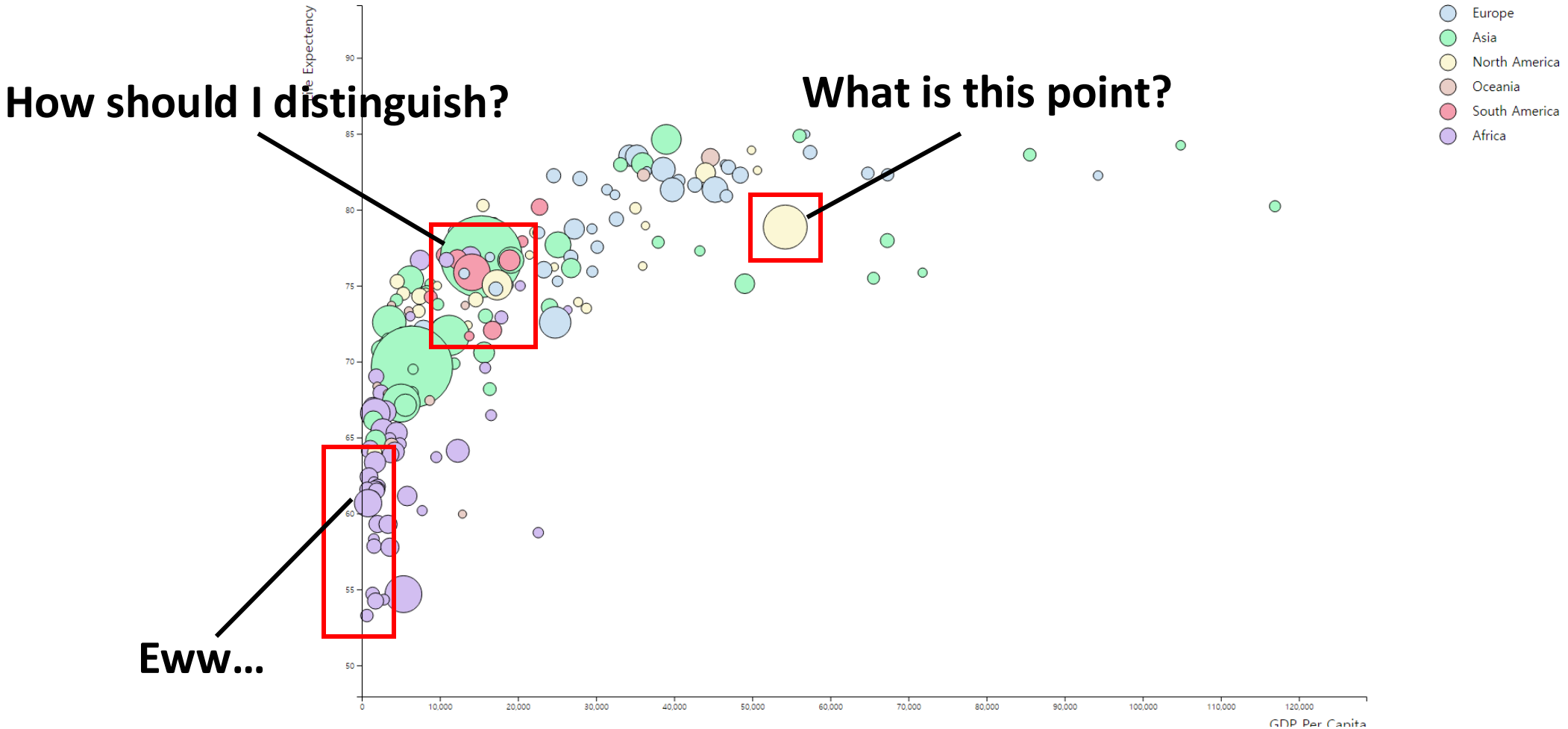
Information Visualization
Coding Session 2

2024.04.24.

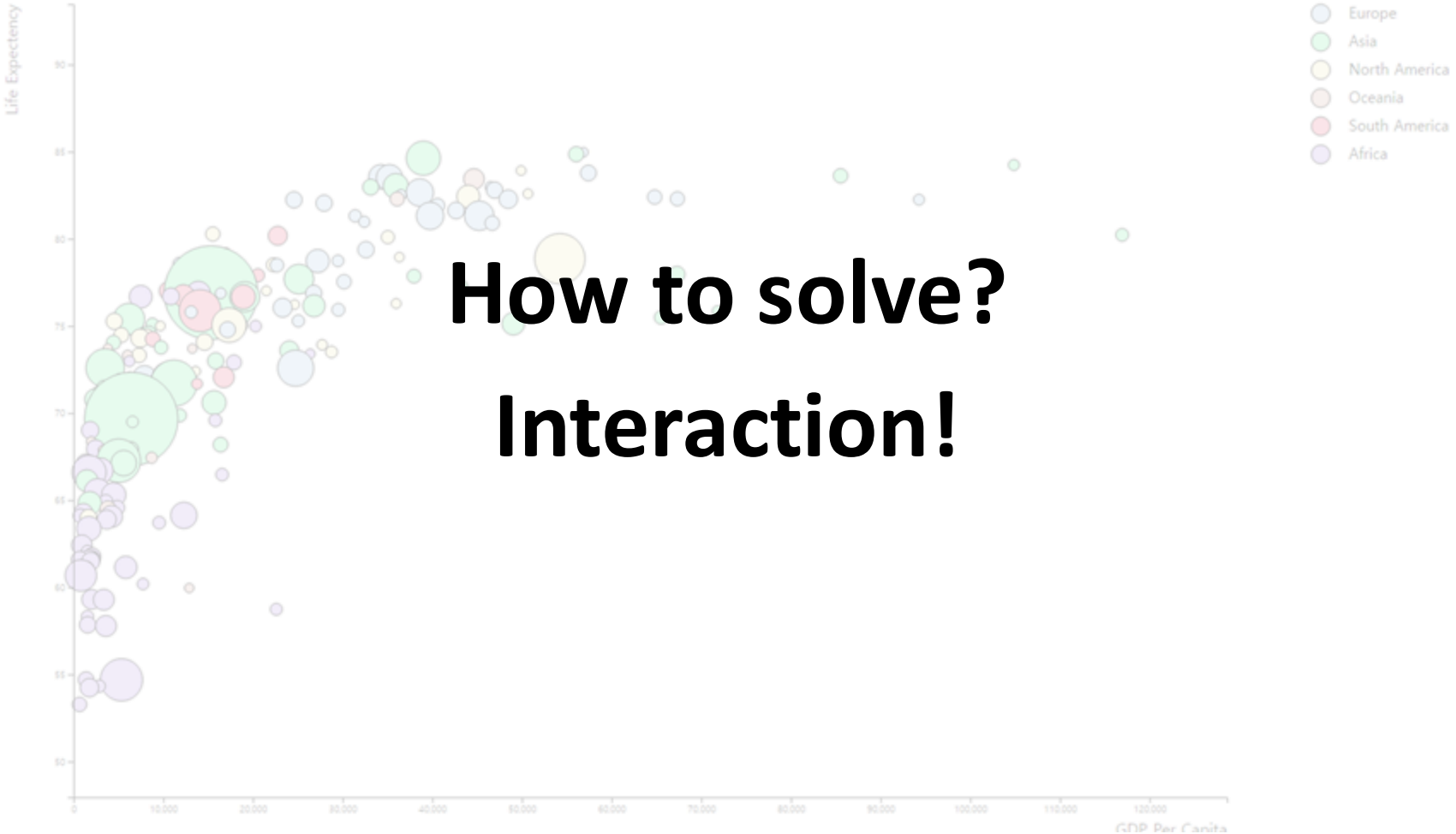
Seongouk Kim

HAiV

Motivation


How should I distinguish?
What is this point?
Eww…
Europe
Asia
North America
Oceania
South America
Africa
Life Expectency
GDP Per Capita
90
85
80
75
70
65
60
55
50
0
10,000
20,000
30,000
40,000
50,000
60,000
70,000
80,000
90,000
100,000
110,000
120,000

Motivation

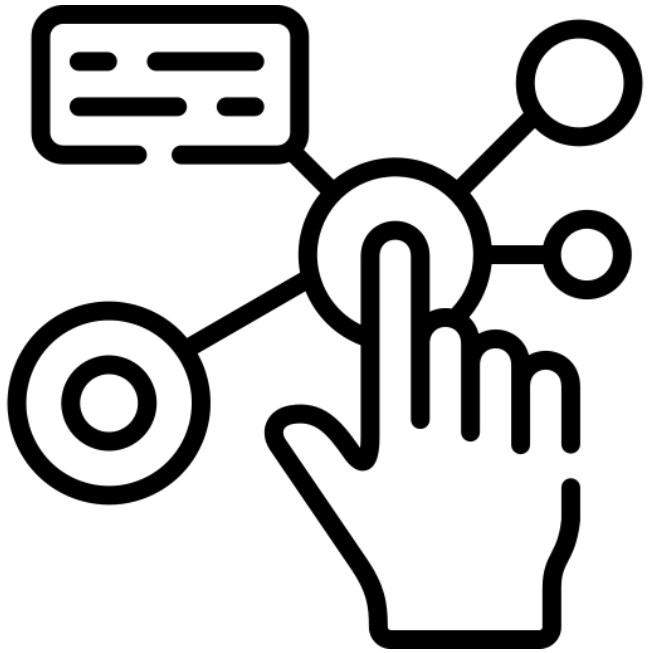


How to solve?

Interaction!

HAiV

Interaction

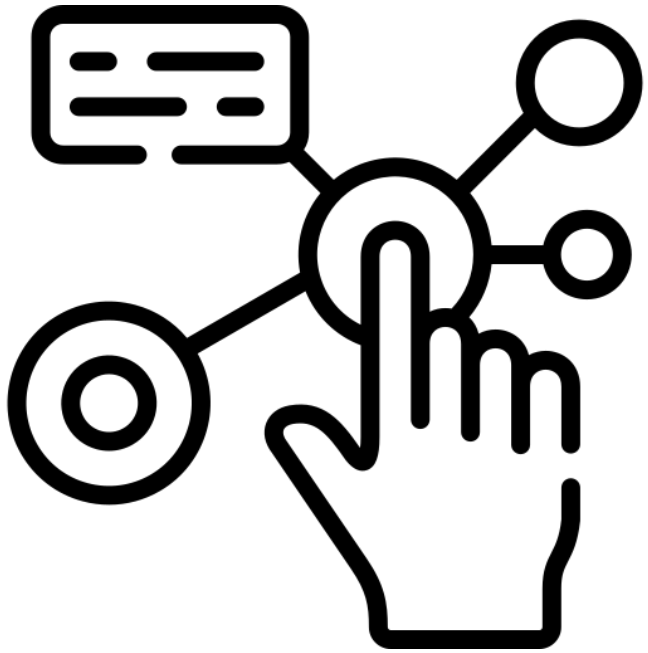


Event

+

Animation

Interaction



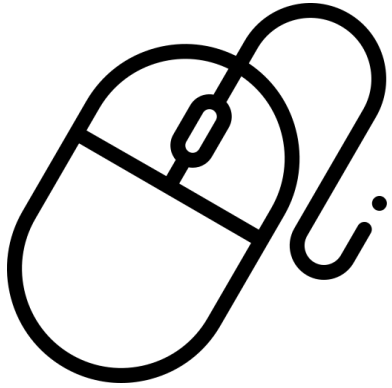
Event

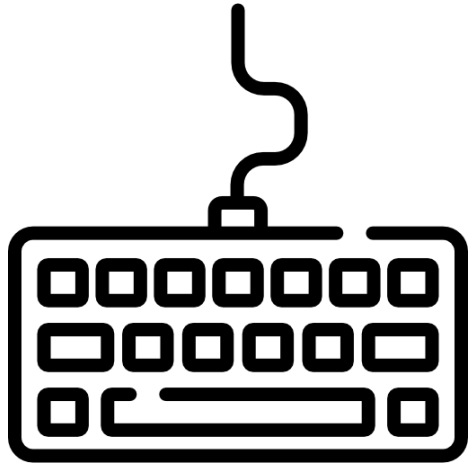
Animation

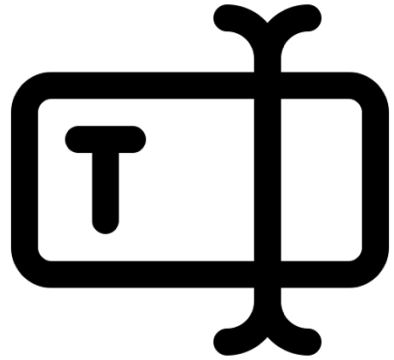
HAiV

Event

- Actions occurring in the system
- Handle with *selection*.on(*event*, *function*)







Interaction

Event

Animation

HAiV

Transition for Animation

- selection-like interface for animating changes to the DOM

Transition

```
svg.select("#sp4rect")
    .transition()
    .duration(1000)
    .delay(500)
    .ease(d3.easeLinear)
    .style("x", "400")
    .style("fill", "red")
```

Declare Transition

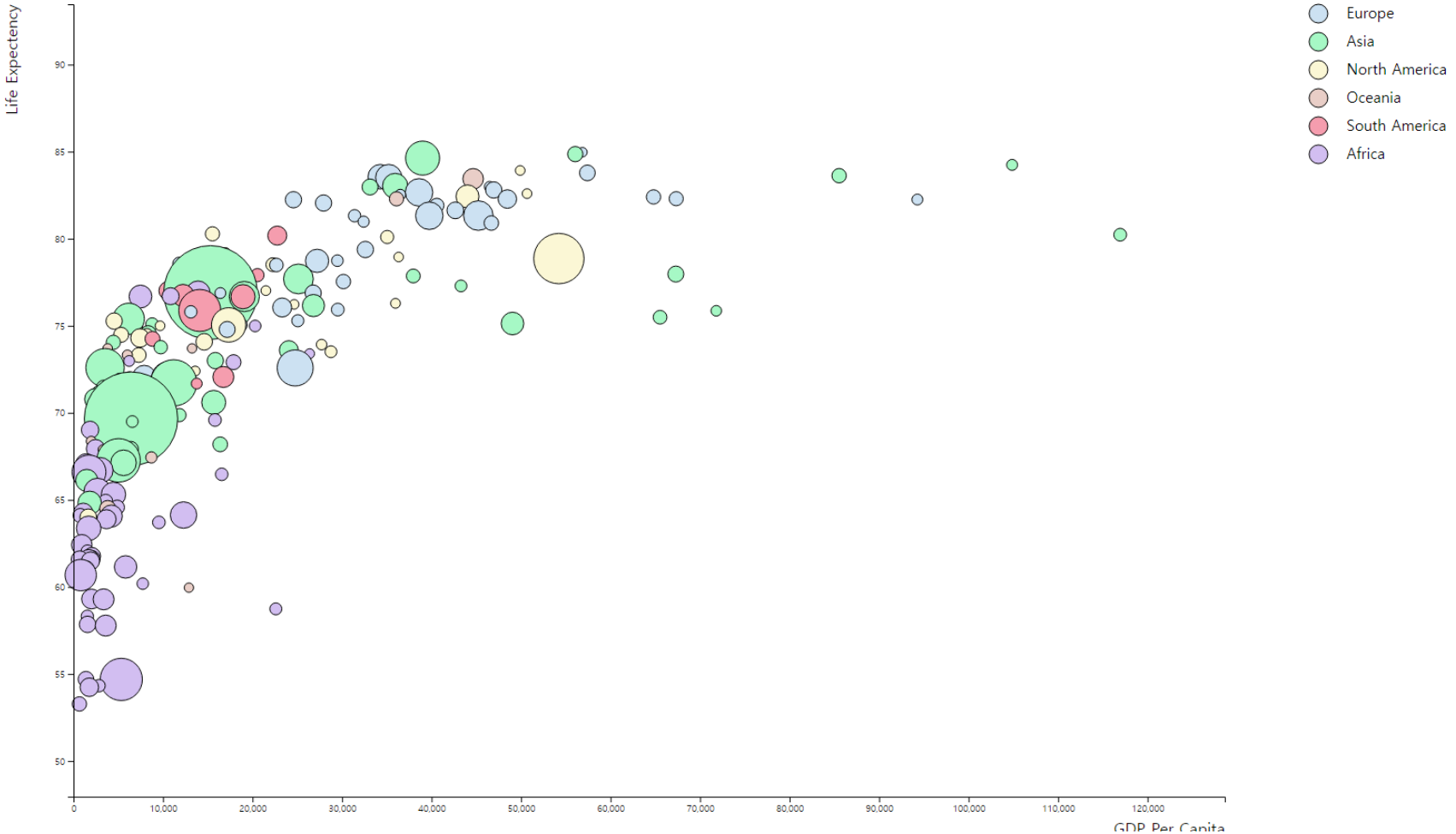
Set Duration

Set Delay

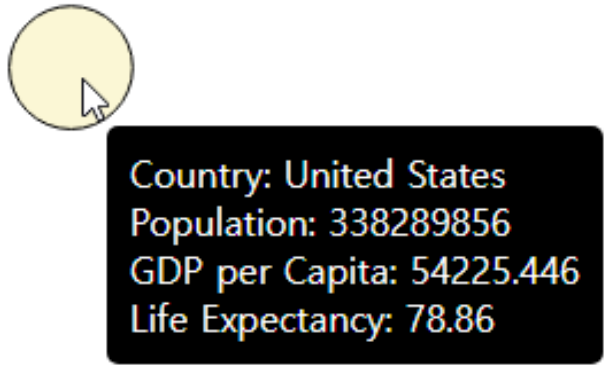
Set Ease

Set Target State

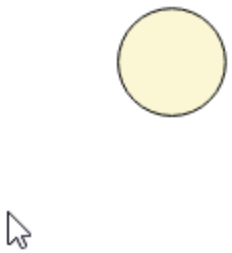
Example


Life Expectency
90
85
80
75
70
65
60
55
50
0
10,000
20,000
30,000
40,000
50,000
60,000
70,000
80,000
90,000
100,000
110,000
120,000
GDP Per Capita
Europe
Asia
North America
Oceania
South America
Africa

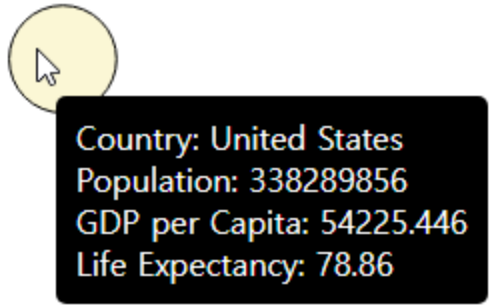
Tooltip


Country: United States
Population: 338289856
GDP per Capita: 54225.446
Life Expectancy: 78.86

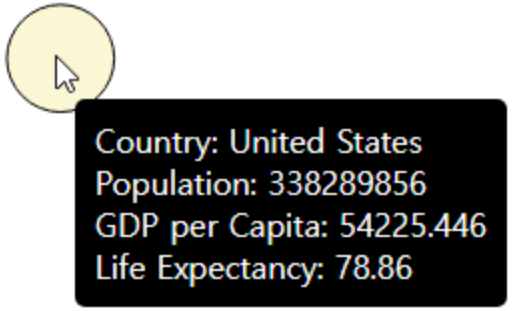
Tooltip



mouseover



mousemove



mouseout

Tooltip

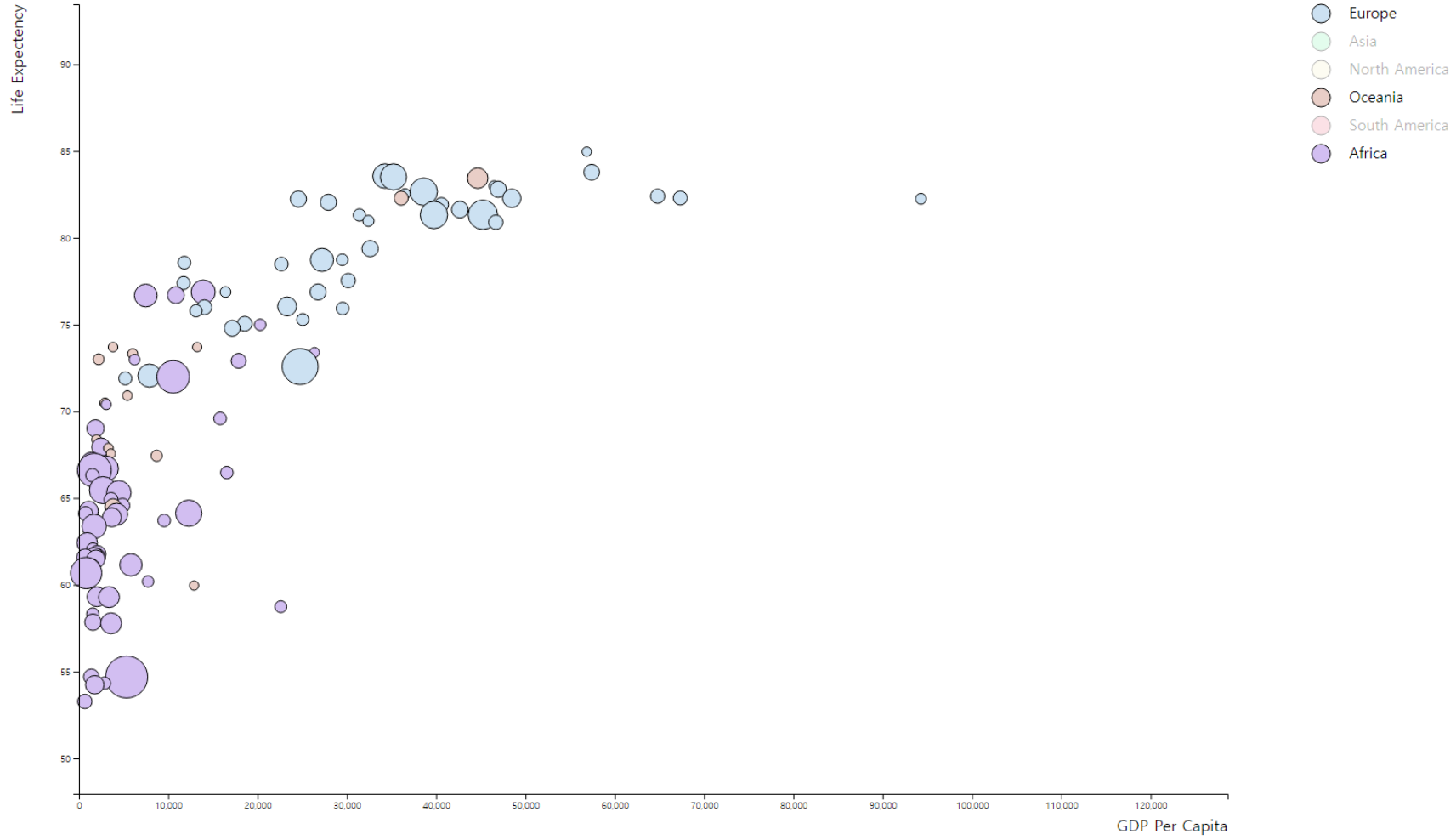
```
const tooltip = d3.select("#chart")
    .append("div")
    .style("opacity", 0)
    .attr("class", "tooltip")
    .style("background-color", "black")
    .style("border-radius", "5px")
    .style("padding", "10px")
    .style("color", "white")
    .style("display", "inline")
    .style("position", "fixed")
    .style("pointer-events", "none")
```

Tooltip

```
var circle = svg.append('g')
    .attr("clip-path", "url(#clip)")

var circle_enter = circle.append('g')
    .selectAll("dot")
    .data(data)
    .enter()

circle_enter.append("circle")
    .attr("class", function (d) { return "bubbles " + d.continent })
    .attr("cx", function (d) { return xScale(d.gdp_per_capita); })
    .attr("cy", function (d) { return yScale(d.life_expectancy); })
    .attr("r", function (d) { return sScale(d.population); })
    .style("fill", function (d) { return cScale(d.continent); })
    .attr("stroke", "black")
    // Add mouse over event
    .on("mouseover", showTooltip)
    .on("mousemove", moveTooltip)
    .on("mouseleave", hideTooltip)
```

```
function showTooltip(event, d) {
    tooltip
        .transition()
        .duration(10)
        .style("opacity", 1)
    tooltip
        .html("Country: " + d.location
            + "<br>Population: " + d.population
            + "<br>GDP per Capita: " + d.gdp_per_capita
            + "<br>Life Expectancy: " + d.life_expectancy)
        .style("left", (d3.pointer(event)[0] + 138) + "px")
        .style("top", (d3.pointer(event)[1] + 35) + "px")
}

function moveTooltip(event, d) {
    tooltip
        .style("left", (d3.pointer(event)[0] + 138) + "px")
        .style("top", (d3.pointer(event)[1] + 35) + "px")
}

function hideTooltip(event, d) {
    tooltip
        .transition()
        .duration(200)
        .style("opacity", 0)
}
```

Toggle

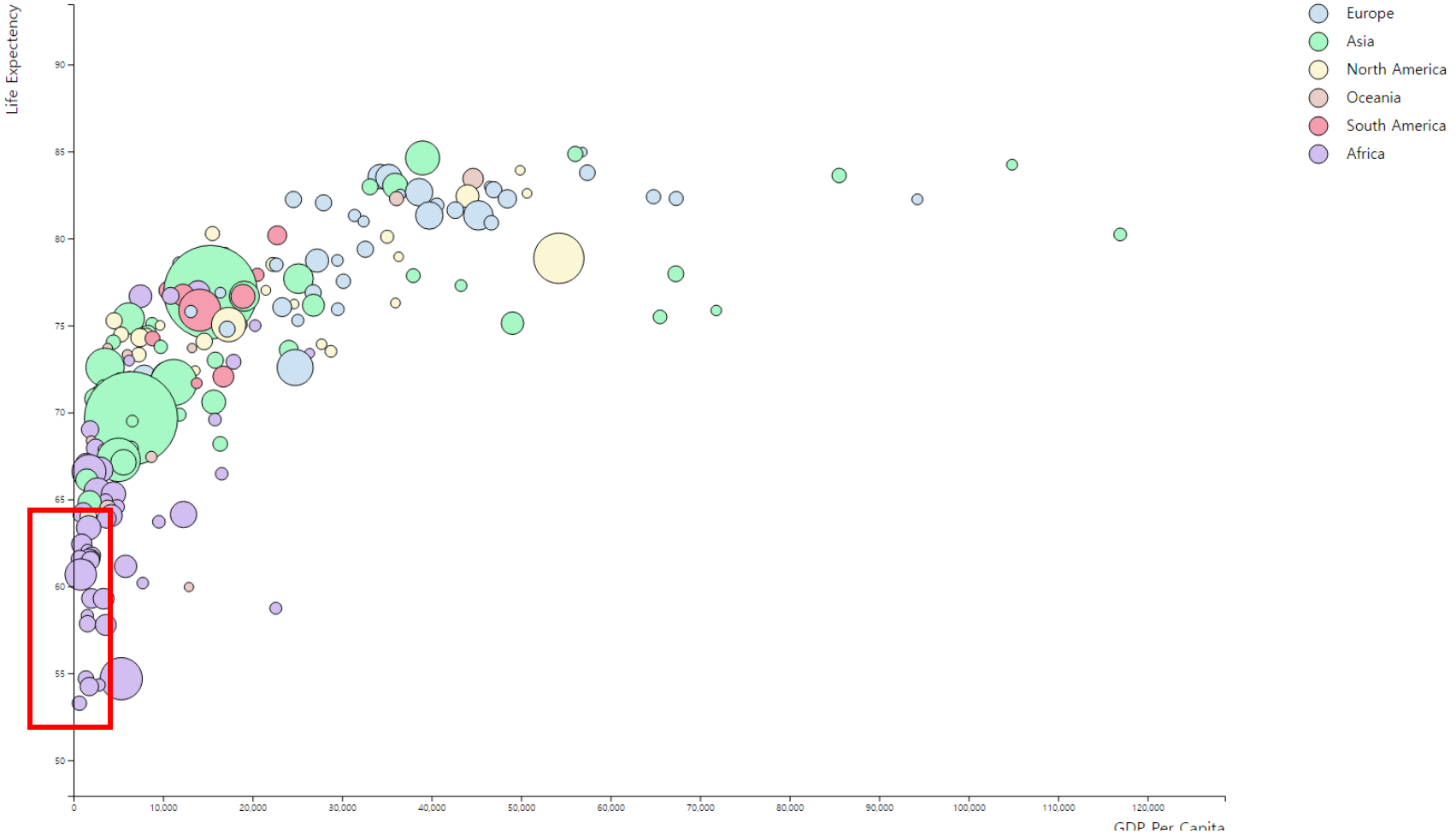

Life Expectency
90
85
80
75
70
65
60
55
50
0
10,000
20,000
30,000
40,000
50,000
60,000
70,000
80,000
90,000
100,000
110,000
120,000
GDP Per Capita
Europe
Asia
North America
Oceania
South America
Africa

Toggle

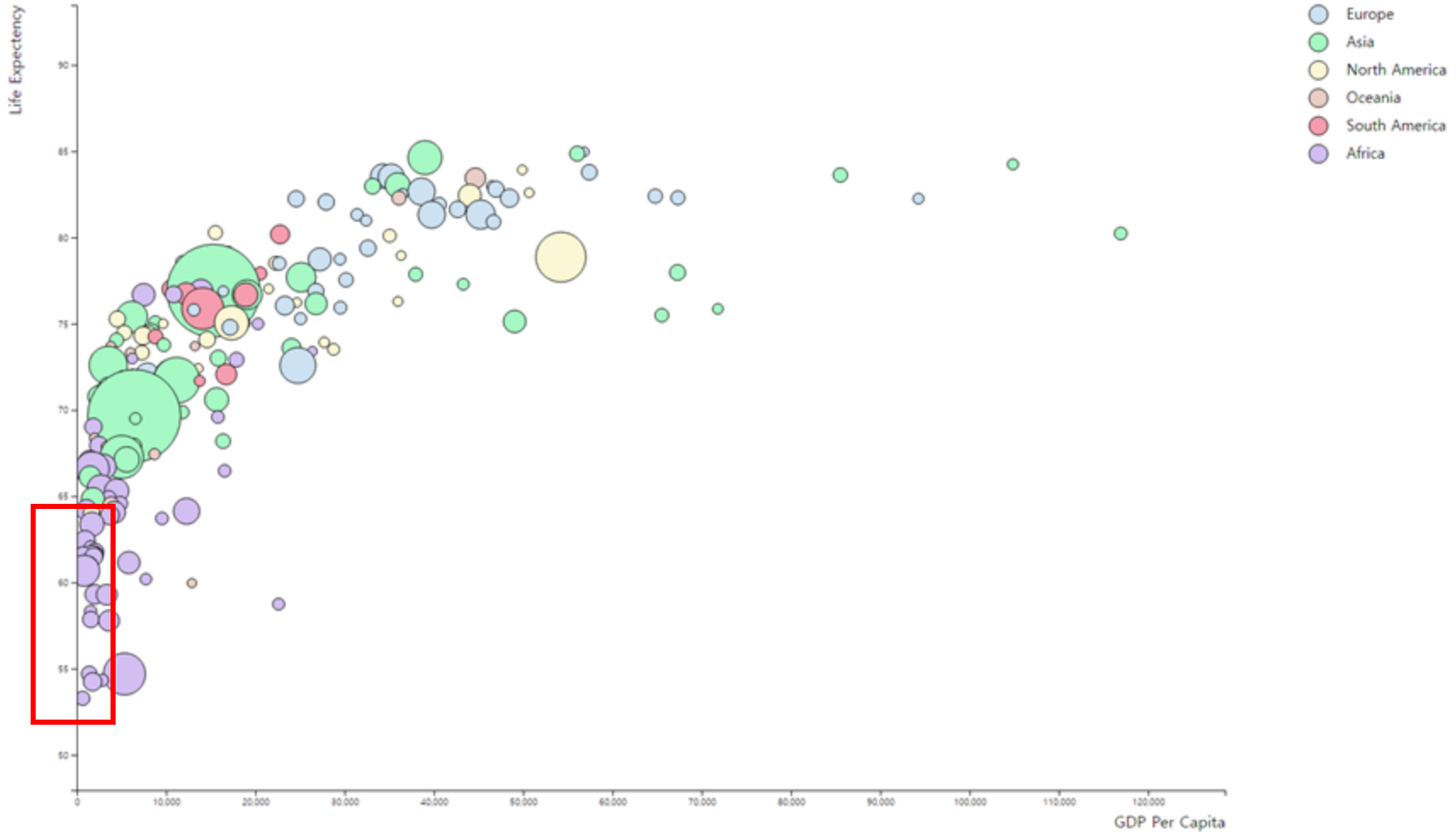
```
// Add legend
const size = 30
svg.selectAll("legend")
    .data(continentList)
    .enter()
    .append("circle")
    .attr("class", function (d) { return "legend " + d })
    .attr("cx", width + 100)
    .attr("cy", function (d, i) { return 10 + i * size })
    .attr("r", 10)
    .style("fill", function (d) { return cScale(d) })
    .attr("stroke", "black")
    .on("click", toggleContinent)

// Add legend texts
svg.selectAll("legend_label")
    .data(continentList)
    .enter()
    .append("text")
    .attr("class", function (d) { return "legendtext " + d })
    .attr("x", width + 100 + size)
    .attr("y", function (d, i) { return i * size + (size / 2) })
    .text(function (d) { return d })
    .attr("text-anchor", "start")
    .on("click", toggleContinent)
```

```
function toggleContinent(event, d) {

    // Parse continent into class name
    continentSplit = d.split(' ')
    className = continentSplit.join('.');

    // Get current opacity
    currentOpacity = d3.selectAll(".bubbles." + className).style("opacity")

    // Change the opacity: from 0 to 1 or from 1 to 0
    d3.selectAll(".bubbles." + className)
        .transition()
        .duration(200)
        .style("opacity", currentOpacity == 1 ? 0 : 1)
        .style("pointer-events", currentOpacity == 1 ? "none" : "auto")

    d3.select(".legend." + className)
        .transition()
        .duration(200)
        .style("opacity", currentOpacity == 1 ? 0.3 : 1)

    d3.select(".legendtext." + className)
        .transition()
        .duration(200)
        .style("opacity", currentOpacity == 1 ? 0.3 : 1)

}
```

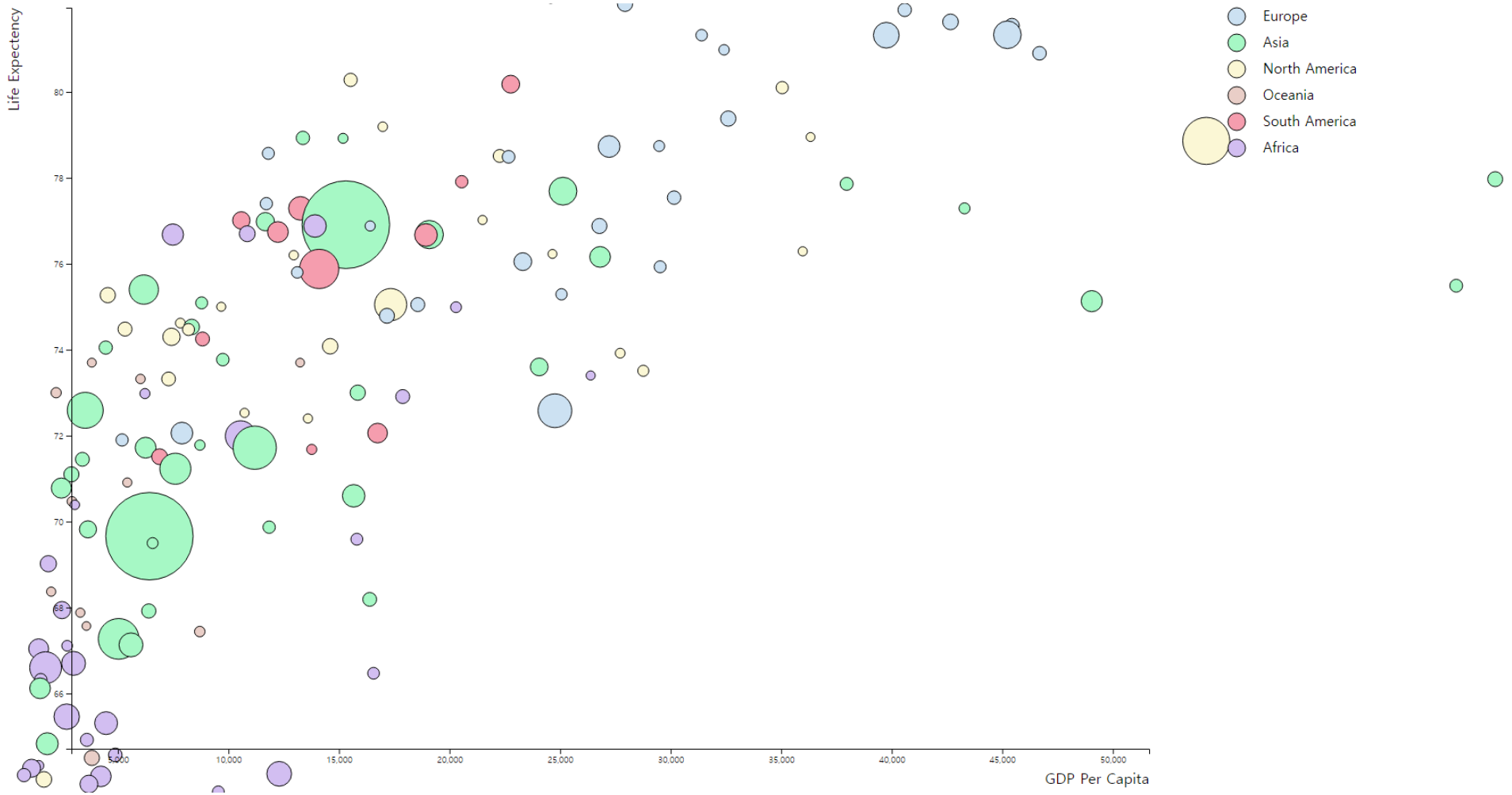
Before Zoom - clipPath



Before Zoom - clipPath

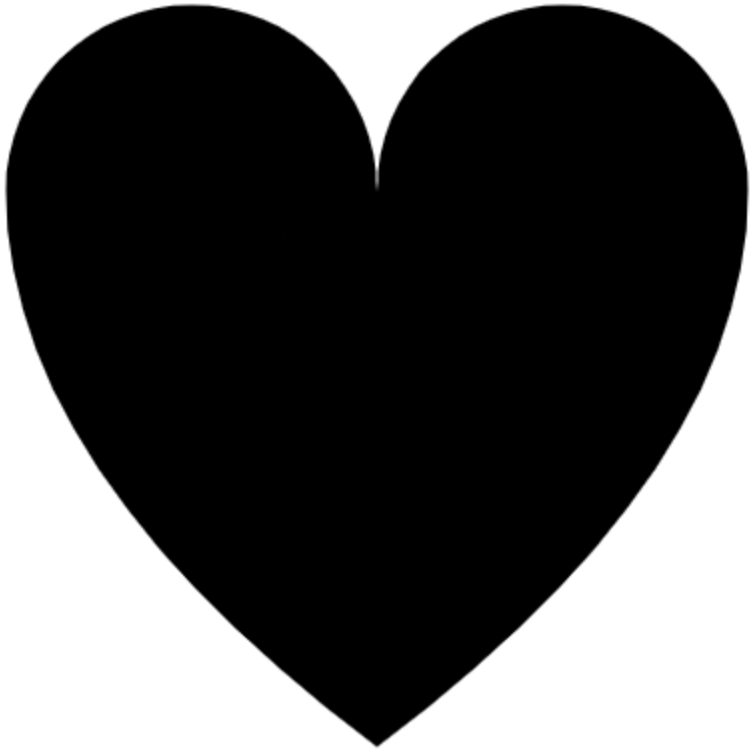


Before Zoom - clipPath



HAiV

Before Zoom - clipPath

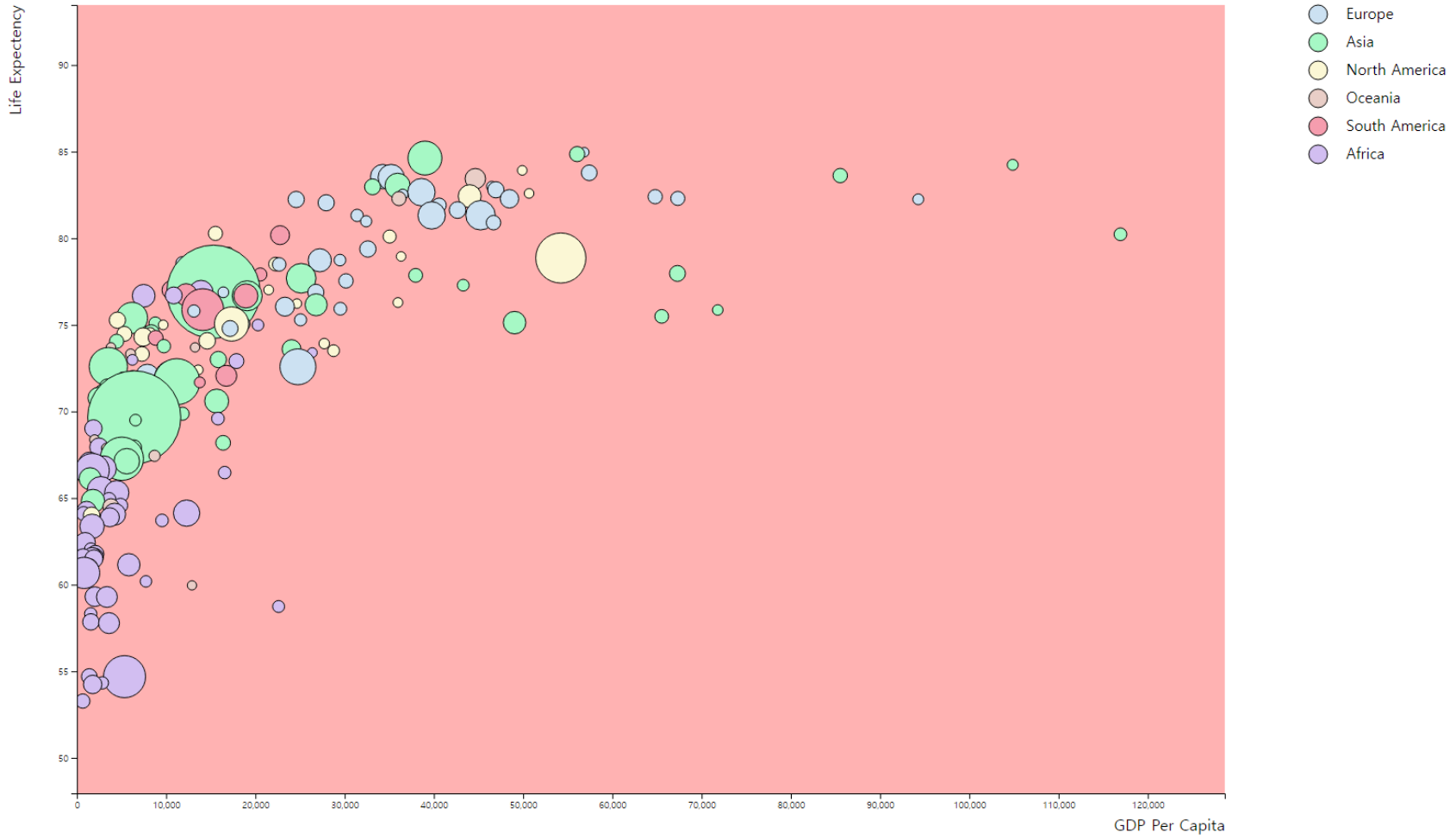


Before Zoom - clipPath

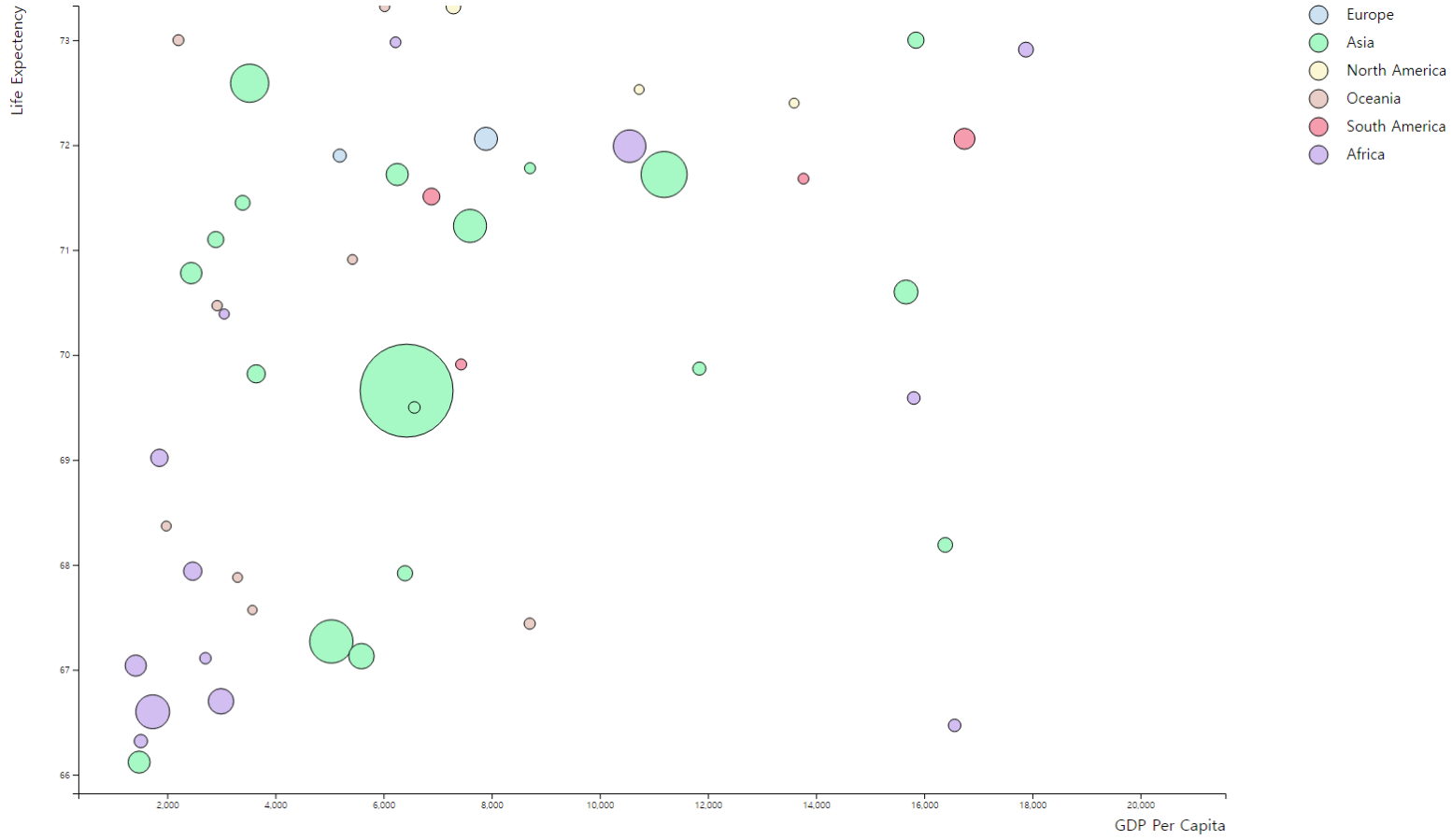
```
// Set the visible area of bubbles
var clip = svg.append("defs").append("clipPath")
    .attr("id", "clip")
    .append("rect")
    .attr("width", width)
    .attr("height", height)
```

```
var circle = svg.append('g')
    .attr("clip-path", "url(#clip)")

var circle_enter = circle.append('g')
    .selectAll("dot")
    .data(data)
    .enter()

circle_enter.append("circle")
    .attr("class", function (d) { return "bubbles " + d.continent })
    .attr("cx", function (d) { return xScale(d.gdp_per_capita); })
    .attr("cy", function (d) { return yScale(d.life_expectancy); })
    .attr("r", function (d) { return sScale(d.population); })
    .style("fill", function (d) { return cScale(d.continent); })
    .attr("stroke", "black")
    // Add mouse over event
    .on("mouseover", showTooltip)
    .on("mousemove", moveTooltip)
    .on("mouseleave", hideTooltip)
```

Before Zoom - clipPath



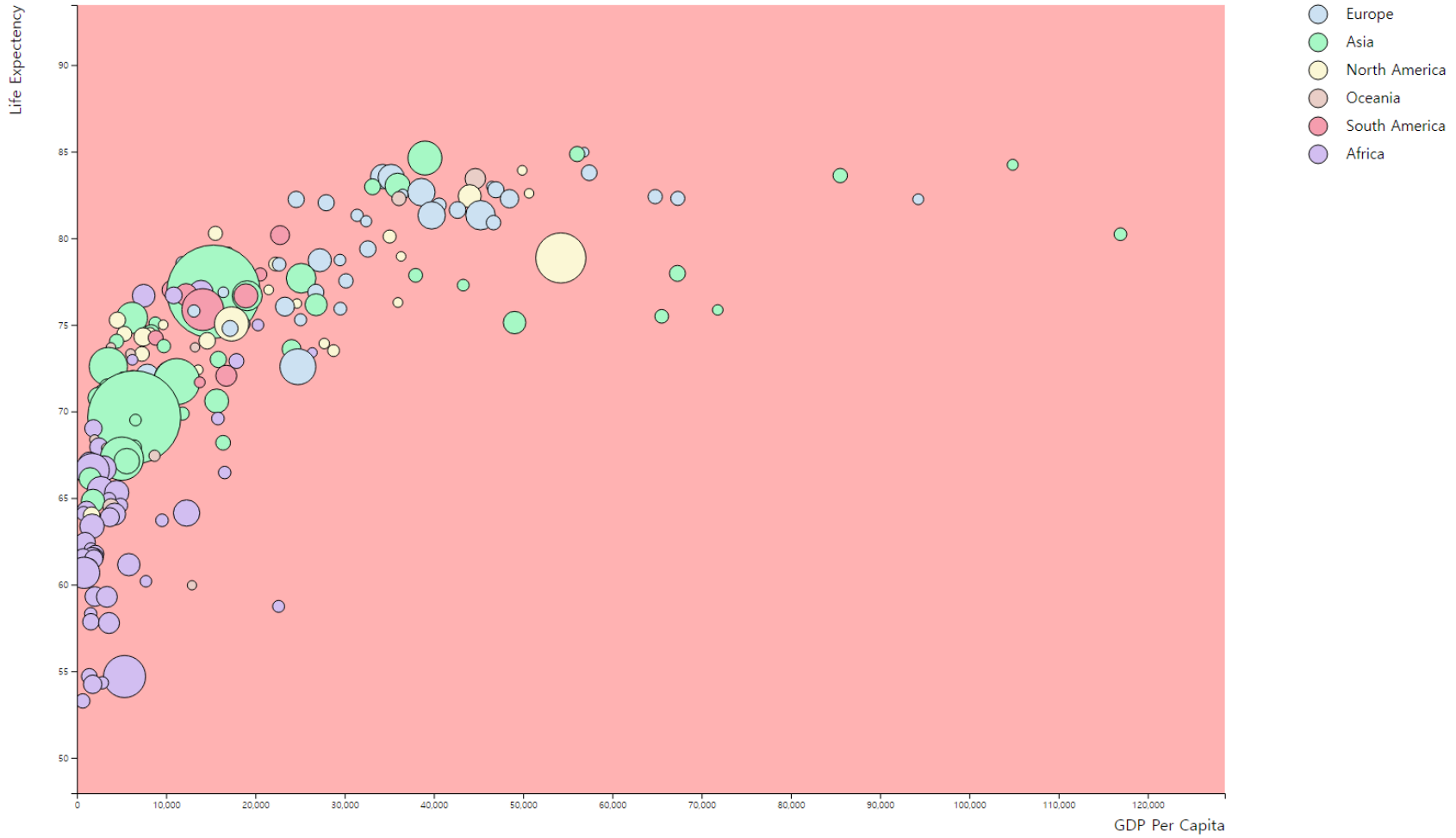
Zoom


Life Expectency
73
72
71
70
69
68
67
66
2,000
4,000
6,000
8,000
10,000
12,000
14,000
16,000
18,000
20,000
GDP Per Capita
Europe
Asia
North America
Oceania
South America
Africa

Zoom

```
// Add zoom event
const zoom = d3.zoom()
    .scaleExtent([.5, 20]) // zoom scale from x0.5 to x20
    .on("zoom", updateChart);

// Set the interaction area for zooming
svg.append("rect")
    .attr("width", width)
    .attr("height", height)
    .style("fill", "none")
    .style("pointer-events", "all")
    .call(zoom);
```

Zoom


Life Expectency
90
85
80
75
70
65
60
55
50
0
10,000
20,000
30,000
40,000
50,000
60,000
70,000
80,000
90,000
100,000
110,000
120,000
GDP Per Capita
Europe
Asia
North America
Oceania
South America
Africa

Zoom

```
function updateChart(event) {
    // create new scale
    const newX = event.transform.rescaleX(xScale);
    const newY = event.transform.rescaleY(yScale);

    // update axes
    xAxis.call(d3.axisBottom(newX))
    yAxis.call(d3.axisLeft(newY))

    // update bubble position
    circle
        .selectAll("circle")
        .attr('cx', function (d) { return newX(d.gdp_per_capita) })
        .attr('cy', function (d) { return newY(d.life_expectancy) });
}
```

HAiV

References

- https://d3js.org/d3-selection/events
- https://d3js.org/d3-transition
- https://d3js.org/d3-transition/timing
- https://d3-graph-gallery.com/bubble.html
- https://d3-graph-gallery.com/interactivity.html

Thank you!