

D3 Cheat Sheet

Scott Murray — d3book.com — 2022 June 8

Selections

	<i>Explanation</i>	<i>Example</i>
<code>d3.select()</code>	Returns the element found	<code>d3.select("svg")</code>
<code>d3.selectAll()</code>	Returns all found elements	<code>d3.selectAll("circle")</code>
<code>selection.append()</code>	Creates a new element inside the selection	<code>d3.select("svg").append("circle")</code>
<code>selection.remove()</code>	Removes the selection from the DOM	<code>d3.select("rect").remove()</code>
<code>selection.text()</code>	Sets the text content of the selection	<code>d3.select("#tooltip").text("")</code>
<code>selection.attr()</code>	Set an HTML attribute value on the selection	<code>d3.selectAll("circle").attr("r", 10)</code>
<code>selection.style()</code>	Set an inline CSS style on the selection	<code>d3.selectAll("circle").style("fill", "teal")</code>
<code>selection.classed()</code>	Adds or removes a class from the selection	<code>d3.select("circle").classed("highlight", true)</code>

Data

<code>selection.data()</code>	Binds an array of data values to the selection	<code>d3.selectAll("circle").data(dataset).enter()...</code>
<code>selection.enter()</code>	Returns a selection of “new” placeholder elements	<code>d3.selectAll("circle").data(dataset).enter()...</code>
Use anonymous functions to access data values bound to elements via <code>d</code> .		<code>d3.selectAll("rect")</code> <code>.attr("height", function(d) {</code> <code>return d.value; // Set the height to 'value'</code> <code>});</code>
Optionally, include <code>i</code> to get the index value of each element in the selection.		<code>d3.selectAll("rect")</code> <code>.attr("x", function(d, i) {</code> <code>return i * 10; // Move each rect to the right</code> <code>});</code>

Transitions

<code>selection.transition()</code>	Initiates a new transition	<code>d3.selectAll("circle").transition().attr("cx", ...</code>
<code>selection.duration()</code>	Sets the transition duration, in milliseconds	<code>d3.selectAll("circle").transition().duration(2000)...</code>

Scales

<code>d3.scaleLinear()</code>	Creates a new linear scale function
<code>scale.domain()</code>	Sets the scale's input domain
<code>scale.range()</code>	Sets the scale's output range
<code>d3.min()</code>	Returns the smallest value in an array
<code>d3.max()</code>	Returns the largest value in an array

```
var xScale = d3.scaleLinear()
    .domain([ 0, 2000 ])
    .range([ 0, width ]);
d3.min([ 10, 20, 70, 35 ]) // Returns 10
d3.max([ 10, 20, 70, 35 ]) // Returns 70
```

Axes

<code>d3.axisOrientation()</code>	Creates a new axis generator function
<code>axis.scale()</code>	Specifies the scale to be used with this axis
<code>axis.ticks()</code>	Suggests a number of ticks for this axis
<code>selection.call()</code>	Calls a method; used to generate an axis

```
var xAxis = d3.svg.axisBottom()
    .scale(xScale)
    .ticks(5);
svg.append("g").call(xAxis);
```

Interactivity

<code>selection.on()</code>	Binds an event listener to a selection
Within an anonymous function, this refers to “the element being acted upon.”	

```
d3.select("#button").on("click", function() { ... });
d3.selectAll("rect")
    .on("mouseover", function() {
        d3.select(this).classed("highlight", true);
    });
```

Other Useful JavaScript

<code>Math.random()</code>	Returns a random value between 0.0 and 1.0
<code>Math.floor()</code>	Rounds down to the nearest integer
<code>array.push()</code>	Appends a new value to an existing array

```
Math.random() * 100 // Could return 61.87844036612...
Math.floor(61.87844036612) // Returns 61
var numbers = [ 2, 3, 4, 5 ];
numbers.push(6); // Now numbers is [ 2, 3, 4, 5, 6 ]
```