D3 Cheat Sheet

Scott Murray — @alignedleft — alignedleft.com — January 6, 2014

Selections

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

Data

`selection``.data()`	Binds an array of data values to the selection	`d3.selectAll("circle").data(dataset).enter()…`
`selection``.enter()`	Returns a selection of "new" placeholder elements	`d3.selectAll("circle").data(dataset).enter()…`

Use anonymous functions to access data values bound to elements via `d`.

```
d3.selectAll("rect")
  .attr("height", function(d) {
      return d.value;  // Set the height to 'value'
  });
```

Optionally, include `i` to get the index value of each element in the selection.

```
d3.selectAll("rect")
  .attr("x", function(d, i) {
      return i * 10;  // Move each rect to the right
  });
```

Transitions

`selection``.transition()`	Initiates a new transition	`d3.selectAll("circle").transition().attr("cx", …`
`selection``.duration()`	Sets the transition duration, in milliseconds	`d3.selectAll("circle").transition().duration(2000)…`

Scales

`d3.scale.linear()`	Creates a new linear scale function	`var xScale = d3.scale.linear()`
`scale``.domain()`	Sets the scale's input domain	`.domain([ 0, 2000 ])`
`scale``.range()`	Sets the scale's output range	`.range([ 0, width ]);`
`d3.min()`	Returns the smallest value in an array	`d3.min([ 10, 20, 70, 35 ])  // Returns 10`
`d3.max()`	Returns the largest value in an array	`d3.max([ 10, 20, 70, 35 ])  // Returns 70`

Axes

`d3.svg.axis()`	Creates a new axis generator function	`var xAxis = d3.svg.axis()`
`axis``.scale()`	Specifies the scale to be used with this axis	`.scale(xScale)`
`axis``.orient()`	Specifies the orientation for this axis	`.orient("bottom")`
`axis``.ticks()`	Suggests a number of ticks for this axis	`.ticks(5);`
`selection``.call()`	Calls a method; used to generate an axis	`svg.append("g").call(xAxis);`

Interactivity

`selection``.on()`	Binds an event listener to a selection	`d3.select("#button").on("click", function() { … });`

Within an anonymous function, `this` refers to "the element being acted upon."

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

Other Useful JavaScript

`Math.random()`	Returns a random value between 0.0 and 1.0	`Math.random() * 100  // Could return 61.87844036612…`
`Math.floor()`	Rounds down to the nearest integer	`Math.floor(61.87844036612)  // Returns 61`
`array``.push()`	Appends a new value to an existing array	`var numbers = [ 2, 3, 4, 5 ];`

```
numbers.push(6);  // Now numbers is [ 2, 3, 4, 5, 6 ]
```