

**Introduction to D3.js**

**Course Number:** SCRPT-146  
**Duration:** 2 days

**Overview**

D3.js is a JavaScript library that renders data using HTML, SVG, and CSS.

Accelebrate’s Introduction to D3 training class teaches attendees how to create dynamic data visualizations that are viewable on almost any modern browser on a range of devices, allowing you to integrate sophisticated data displays into your web applications.

**Prerequisites**

Attendees should have a solid knowledge of HTML, JavaScript, and CSS. Prior experience with SVG is not presumed.

**Materials**

All D3.js training attendees receive comprehensive courseware.

**Software Needed on Each Student PC**

* Windows, macOS, Linux, or Solaris, with at least 8 GB RAM
* Git 2.x or later
* Preferred JavaScript code editor
* Modern Web Browser (one or more of the following)
  + Google Chrome (strongly preferred, one segment is Chrome specific)
  + Mozilla Firefox
* Either Node.js or Python

**Objectives**

* Understand the approach D3 takes to simplifying data visualization
* Create both static and dynamic visualizations that can vary based on modified data or user interaction
* Generate visualizations with several of the most common D3 layouts, including histogram, pie, cluster, stack,and treemap)
* How to maximize the performance of your visualizations and avoid slow rendering
* Master reusable skills for using future layouts

**Outline**

* Introduction
  + What is D3, who made it, and why do you care?
  + Examples and use cases
* Basics
  + Selections
  + Data
  + SVGs
    - Life without D3
  + Scales/Axes
  + Shapes
* The D3 Paradigm
  + Anonymous functions (brief review if necessary)
  + DOM data-binding
    - Adding, removing, and updating elements
    - Transitions
  + Code structure
* Layouts, SVG Helpers, and Time
  + Histogram, pie, cluster, stack, treemap
  + Lines, arcs, chords, and diagonals
  + Time
* Interaction
  + Mouse events
  + Keyboard events
  + Streaming data
  + Editing a dataset with D3
* Performant Visualizations
  + Why you should care about the performance of your visualizations
  + What makes a visualization slow
    - Data manipulation
    - Network conditions
    - Repaints
  + Patterns to avoid
  + Using Chrome Developer Tools to measure the performance of your visualizations
* Conclusion