

Advanced JavaScript

Course Summary

Description

The Advanced JavaScript course covers the programming with the JavaScript language, building web pages that interact with server-side resources, and accelerating development with popular application frameworks. Participants will learn how to use JavaScript and related debugging tools within a browser context, and how to use the web page as a dynamic user interface. Participants will then learn how to interact with web services to perform business operations and communicate results to a user, and how to leverage existing JavaScript frameworks to easily add advanced capabilities to their apps, while keeping them cross-platform and cross-browser compatible.

Topics

- Brief review of difficult JavaScript topics
- Intro to jQuery and jQuery UI
- JavaScript and graphics
- More HTML5 APIs
- JavaScript software engineering techniques

Audience

The course is intended for web designers and developers who wish to learn how create rich web user experiences using the client-side capabilities of JavaScript.

Prerequisites

To benefit from this course, participants should have experience developing with JavaScript and must be familiar with core programming constructs in any language, including branching, looping, variables and functions. Participants should also be familiar with the HTML, the basics of HTTP and the DOM. Experience with interactive debugging and more advanced aspects of programming and TCP/IP networking is helpful but not necessary.

Duration

Three days

Advanced JavaScript

Course Outline

- I. Brief review of difficult JavaScript topics**
 - A. Scope
 - B. Closures
 - C. Events, capture and bubbling
- II. Intro to jQuery and jQuery UI**
 - A. Widgets (buttons, dialogs, etc.)
 - B. Implementing drag and drop UIs
- III. JavaScript and graphics**
 - A. Raster graphics with Canvas API
 - B. Vector graphics with SVG and Raphael
 - C. Creating charts and graphs in HTML with gRaphael and dojox.charting
- IV. More HTML5 APIs**
 - A. Persisting data with local storage
 - B. Geolocation services
 - C. Future of AJAX/Comet with web socket
 - D. Background JavaScript tasks with web workers
 - E. Future of crossdomain messaging
 - F. Creating offline apps
- V. JavaScript software engineering techniques**
 - A. The feature detection pattern & modernizr
 - B. Organizing code for large JS projects
 - C. Unit and functional testing
 - D. Testing of async interactions
 - E. Developing with JavaScriptMVC