



CTD Intro Week 16

Node, NPM, Debug, and OpenAPI

Node and NPM

- JavaScript originally just ran in browsers
- In 2009, Ryan Dahl build a stand-alone JavaScript engine based on Google's V8 JavaScript engine.
 - Enabled server and desktop JavaScript applications
- JavaScript's asynchronous capabilities made it possible to build fast and efficient servers
- NPM (Node Package Manager) and other packages managers provide a rich ecosystem.



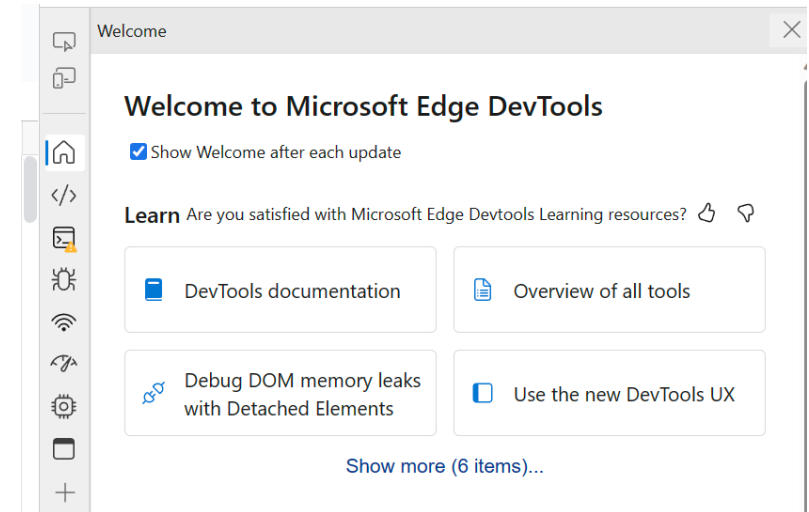
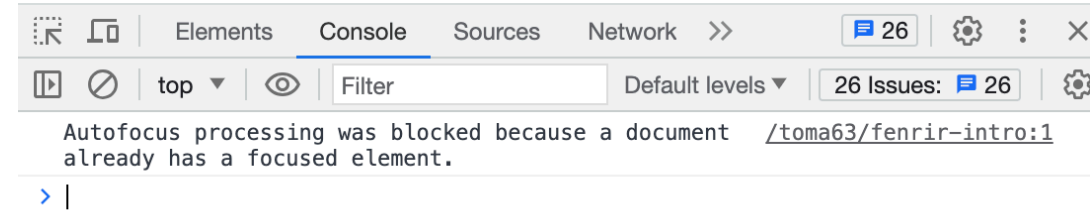


Debugging and Developer Tools

- Debugging
 - Finding and correcting errors in specification and/or implementation
- By inspection
- Accessing the developer tools, debugger, javascript console
 - Chrome
 - Click three vertical dots in the upper right corner
 - Select 'more tools'
 - Select 'developer tools'
 - Edge
 - Three horizontal dots in the upper right corner
 - Same menu selections as Chrome
 - Firefox
 - Three horizontal bars in the upper right corner
 - Select 'more tools'
 - Select 'Web Developer Tools'
 - For all, right-click and select 'inspect'

Developer Tools

- Select tools using tabs
 - Console
 - Interactive javascript
 - Error messages
 - Elements
 - Graphical browser for the Dom hierarchy
 - Source
 - Debugger
 - Single stepping
 - Breakpoints
 - Watchpoints
 - Call stack, function tracing
 - Network
 - Watch fetch requests



Debugger – interactive demo

- Some of
 - Stepping
 - Break on event
 - Line breakpoints
 - Debugging functions
 - Debugging exceptions
 - Breaking on DOM changes (watch expressions)
 - Mobile/Desktop

Open API project

- Adding a second fetch endpoint
- Two options for how to display additional information
- Show/hide data from the second fetch in a single page
 - Use the hidden attribute to show/hide additional information
- Load a second page
 - Use [window.location.href](#) to go to a new page
 - Needs to load its own css and js
 - Provide a back button
 - Pass information (e.g. a fetch URL) to the second page using search params
 - Use [const params = new URLSearchParams\(\)](#)
 - Use [params.append\("param-name", "string-data"\)](#)
 - ['?'](#) + [params.toString\(\)](#) added to the url for the other page
 - In the other js file:
 - [params = new URLSearchParams\(window.location.search\)](#)
 - Retrieve using [params.get\("param-name"\)](#)





Questions and Demo