

Advanced TypeScript Training

Course content

Quick Review of TypeScript

- Why Typescript is such a stonkingly good idea
- Remember – Typescript is (currently) design-time only

How to get there from here

- Adding TypeScript to existing JS files
- Use as little or as much TypeScript as you like
- Towards an agreed code style: using TypeScript consistently
- Get out of jail free with `<any>` type and its relatives (and when not to use them)

Advanced TypeScript

- Checking against permitted members with Type Guards
- Partial, ReturnType, Required and other Utility types
- Assigning Type based on conditions
- Deriving Types from collection index members
- New types for old with maps types

Close of Day Discussion and Q&A

Leading-Edge ECMAScript

- Keeping up with the code-ashians: ECMAScript just keeps getting better every year
- What's new since ES6:
- ...and you *are* making use of back ticks and interpolation, aren't you?
- So how do we do it *this* week?
- Who's driving the ECMAScript agenda (and how you can have your say)
- What's coming

Functional Programming

- Why pure functions turn out to be a good idea
- Scalable encapsulation, testability and stateful apps
- What that means for coding style

Mixins, Generics and Testing

- Are we there yet? What the RxJS library is teaching ECMAScript

- Exploring functional helpers and the underlying thinking
- Popular helpers from lodash
- Benefits when it comes time to test
- Close of Day Discussion and Q&A

Function Overloading and Type overloading

- Can we fix it? Yes we can
- Can we fix it? Yes we can and so can my wife
- These types are *really* cool ... and flexible!
- Modules and Namespaces
- Import, Export (and not a trade deal in sight)
- Dependency injection, inversion of dependency and all that jazz
- These aren't just old components, they're *namespaced* components

Project Structuring

- Towards architectural conventions
- Automated tree-shaking, dehydration and module refactoring (oh my!)
- When you say compile, you *really* mean...
- The wonderful world of automated testing

ES2020 and beyond

- So how do we do it *this* week?
- Who's driving the ECMAScript agenda (and how you can have your say)
- What's coming next