



Practice 4

TypeScript Basics

Topics

- New expressions in webapp development: TypeScript, WebPack, Grunt, Gulp, Bundle, Lint,
- One of the most dynamically changing areas of development
- Actual trends include higher level abstraction, types, OOP-like development



TypeScript

- For years, JavaScript was the most common tool for building webapplications
- Uncomfortable elements, feature requests -> need for a more abstract solution
- New paradigms, similar to object-oriented programming (class, interface, types)
- Transpiling to JavaScript -> usual use, easier high-level development



TypeScript tools

- **tsc**: TypeScript compiler – linking, transpiling
 - **tsconfig.json**: tsc will look for this in the folder it's running in (it's in the root of the project)
 - **tslint**: checking code quality
 - **typings**: non-native definitions for usage with different frameworks, libs
-
- Naturally a usual project has much more tools than this
 - Keywords: automation, modularization



npm

- npm = Node Package Manager
 - <https://www.npmjs.com/>
 - package.json
 - Contains all the informations about our project
 - Contains all the dependencies that the project needs with a specific version

Bundle and webpack

- A complex webapp consists of multiple classes, interfaces, definitions -> huge files
- Breakdown to smaller files, modules
- include, import, export tools needed to use them with each other
- Native JavaScript doesn't support it, running framework, or **bundle** file is required
- Webpack is a very popular solution for this
- webpack.config.js contains the settings

Gulp and Grunt

- <http://gulpjs.com/>
- Running a webapp takes many steps
- Just in our simple example project we've already used npm install, tsc, webpack, if we want to clean up, we have to individually delete the folders and files...
- Automatic task running with a single command!
- Local install needed in the project
- Task definition files - gulpfile, Gruntfile
- Running: `gulp <task_name> [default task else]`



More useful tools

- Visual Studio Snippet
 - In every project we need to write similar codeblocks over and over again -> insertable file/code templates
 - Better than copy-pasting example/old projects...
 - Download from marketplace or writing based on development team conventions
 - Autocomplete or Insert Snippet
- Developer Tools
 - Almost every browser has one
 - Breakpoints, console logs, call stack, local database, cache, source file browsing
 - Runtime debugging, many trivial (and not so trivial) bug and error can be easily detected with them