

Full Stack Web Development 

Using Visual Studio Code for TypeScript

Course Map: Day 1

1. ASP.NET Core Architecture
2. Getting Started with ASP.NET Core
3. Design Patterns, Unit Testing
4. Entity Framework Core



Course Map: Day 2

5. Introduction to TypeScript

6. Using VS Code for TypeScript

7. Angular 2 Architecture

8. Using Angular CLI for Client Apps



Agenda



- Folder-based projects
- Navigation
- Enabling Intellisense
- Compiler options
- Debugging
- Scaffolding with Yeoman

Get the Bits



[github.com](https://github.com/tonysneed) / **tonysneed** /

Kliant.AspNetCore-Angular

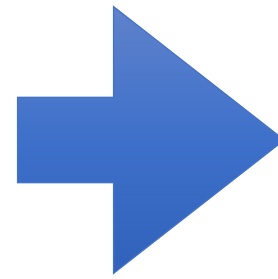
What Is Visual Studio Code?

- Lightweight code editor
- Cross-platform (Mac, Linux, Windows)
- Task runner, integrated terminal
- Intellisense, debugging, version control



Folder-based Project System

- Select File, Open
- Then choose a folder
- Project includes everything in the folder



Ways of Opening VS Code

- From the File Explorer or Finder
 - Windows: Installer option for shell support
 - Mac: <http://bit.ly/mac-open-in-vscode>
- From Command Prompt or Terminal
 - Enter: `code .`



Navigation: Windows, Linux

| Action | Command: Windows, Linux |
|--------------------------|---|
| Open a file | CTRL + |
| Cycle recent file | CTRL + TAB (forward) CTRL + SHIFT + TAB (reverse |
| Go to symbol | CTRL + SHIFT + |
| Go to line number | CTRL + |
| Go back | ALT + ←, CTRL + ALT + - |
| Go forward | ALT + →, CTRL + SHIFT + |
| Go to definition | F12 |
| Peek definition | ALT + F12, CTRL + SHIFT + F1 |
| Find all reference | SHIFT + F1 |
| Show errors and warning | CTRL + SHIFT + |
| Cycle errors and warning | F8 (forward), SHIFT + F8 (reverse |

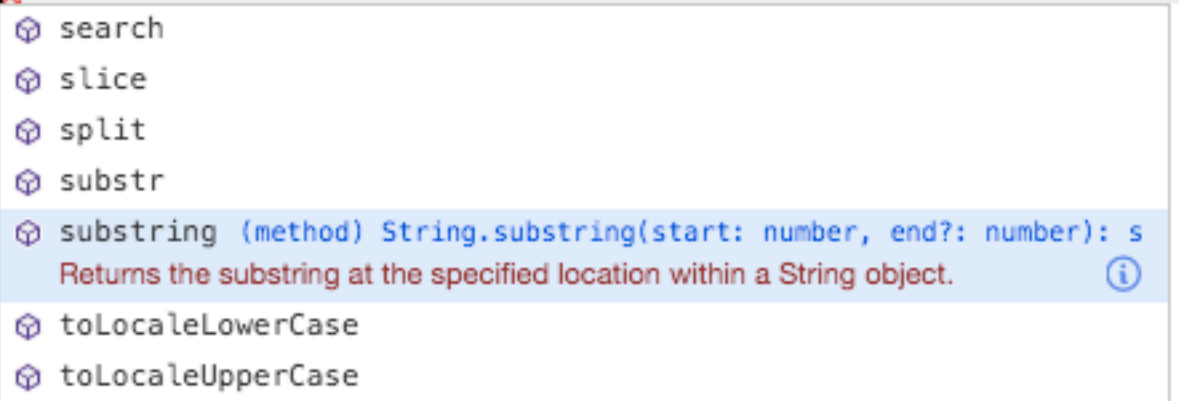
Navigation: Mac OS X

| Action | Command: Windows, Linux |
|--------------------------|---|
| Open a file | CMD + |
| Cycle recent file | CTRL + TAB (forward) CTRL + SHIFT + TAB (reverse |
| Go to symbol | CMD + SHIFT + |
| Go to line number | CTRL + |
| Go back | CTRL + - |
| Go forward | CTRL + SHIFT + |
| Go to definition | F12 |
| Peek definition | ALT + F1 |
| Find all reference | SHIFT + F1 |
| Show errors and warning | CMD + SHIFT + |
| Cycle errors and warning | F8 (forward), SHIFT + F8 (reverse |

Trigger Intellisense

- CTRL + Space

```
greet(): string {  
    return "Hello " + this.message.;  
}
```



The image shows a code editor with a dropdown menu for string methods. The menu is open, displaying a list of methods with a small icon to the left of each. The 'substring' method is highlighted in blue. The dropdown menu is positioned to the right of the code line 'return "Hello " + this.message.;'.

- search
- slice
- split
- substr
- substring (method) String.substring(start: number, end?: number): string
Returns the substring at the specified location within a String object. ⓘ
- toLocaleLowerCase
- toLocaleUpperCase

Show Parameter Info

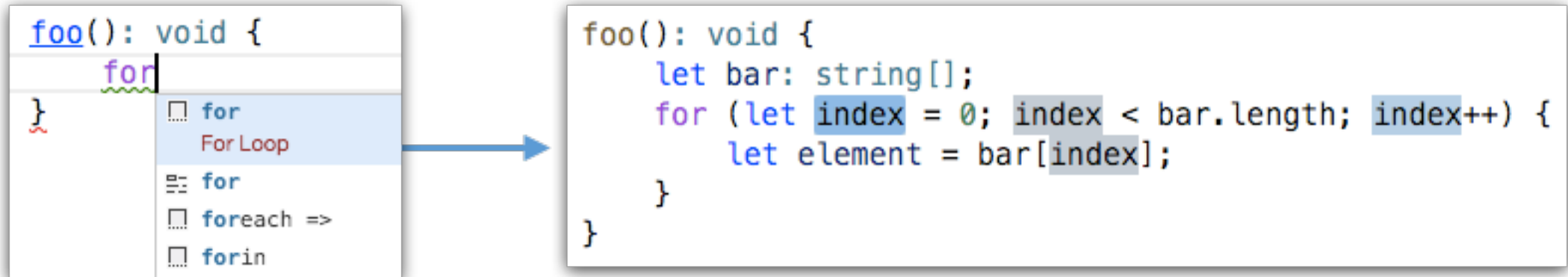
- Mac: CMD + SHIFT + SPACE
- Win/Linux: CTRL + SHIFT + SPACE

```
greet(): string {  
    return "Hello " + this.message.substring();  
}
```

`substring(start: number, end?: number): string`
Returns the substring at the specified location within a String object.
start: number The zero-based index number indicating the beginning of the substring.

Code Snippets

- Type keyword, then press Tab
- Press Tab to cycle through fields



Enabling Intellisense

- Use NPM to install type declarations
- Previous tools obsolete: tsd, typings

```
npm init --yes  
npm install lodash --save  
npm install @types/lodash --save
```

Add Import Statement

```
/// <reference path="../../node_modules/@types/lodash/index.d.ts" />
```

```
import * as _ from "lodash";
```

```
function hello(name: string): string {
```

```
    let message = _.p("Hello " + name, 40);
```

```
    return message;
```

```
}
```

pad (method) _.LoDashStatic.pad(string?: string, length?: number, _
Pads string on the left and right sides if it's shorter than length. Padding characters a...
padEnd
padStart
parseInt
Partial
partial
PartialRight
partialRight
partition
PH
pick
pickBy

Demo: Using Visual Studio Code



Define TypeScript Project

- Place a tsconfig.json file in a root directory
- To create file execute: `tsc --init`
- For compiler options press CTRL + Space

Specify Compiler Options

```
{
  "compilerOptions": {
    "module": "commonjs",
    "target": "es5",
    "noImplicitAny": false,
    "sourceMap": false,
  },
  // Comments allowed
  "exclude": [
    "node_modules",
    /* Globs supported */
    "**/*.spec.ts"
  ]
}
```

- ✎ allowJs
Allow javascript files to be compiled.
- ✎ allowSyntheticDefaultImports
- ✎ allowUnreachableCode
- ✎ allowUnusedLabels
- ✎ baseUrl
- ✎ charset
- ✎ declaration
- ✎ declarationDir
- ✎ diagnostics
- ✎ emitBOM
- ✎ emitDecoratorMetadata
- ✎ experimentalDecorators

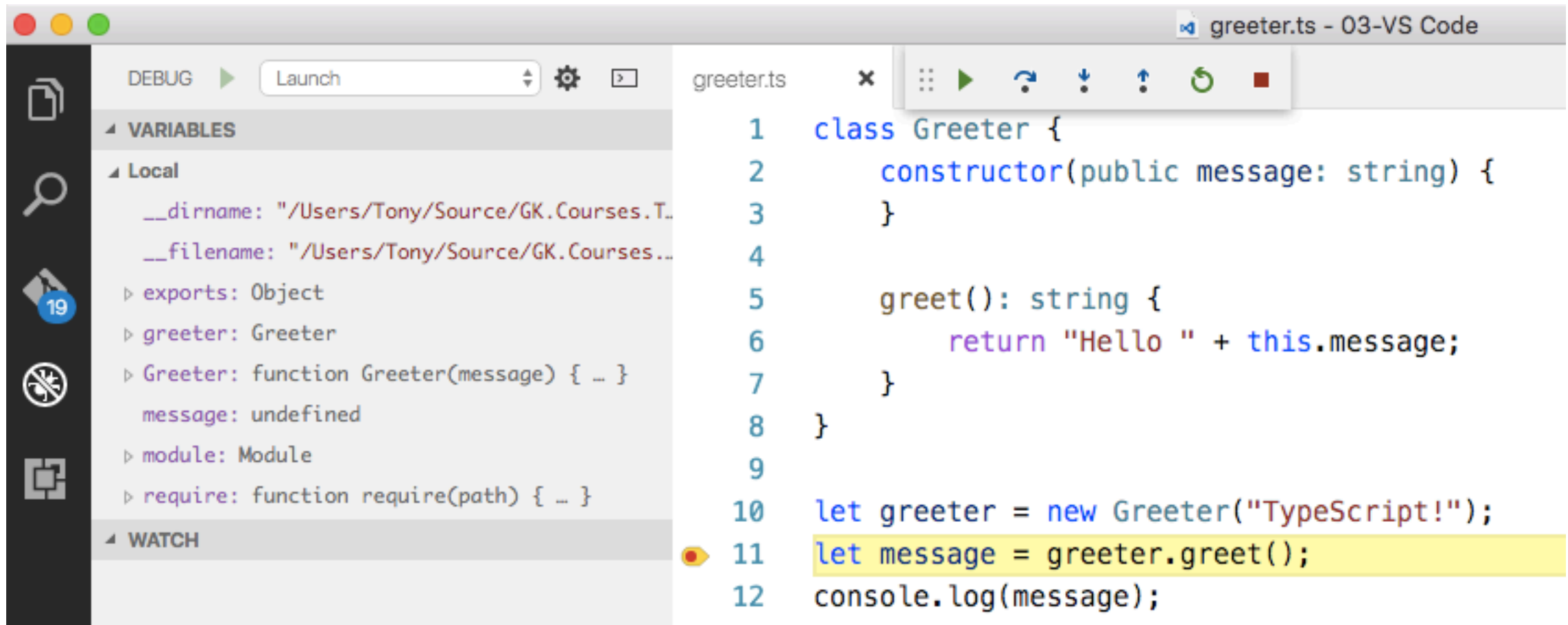
Compiling a TypeScript Project

- Configure Task Runner: add **tasks.json**
- Add **build** task
- Set **rootDir** and **outDir** in tsconfig.json

Enable Debugging

- Compiler options: enable **source maps**
- Add **launch.json** file
- Set **sourceMaps** and **outDir**

Debug TypeScript in VS Code



Demo: Compiler Options and Debugging



Where Is File New Project?



Use Yeoman to Scaffold New Projects

- Use NPM to install Yeoman
- Find a generator: <http://yeoman.io/generators>
 - Or create your own

```
npm install -g yo  
npm install -g generator-tonysneed-vscode-typescript  
cd MyCoolTypeScriptProject  
yo tonysneed-vscode-typescript
```


Sample TypeScript Yeoman Generator

```

Last login: Sat Jan 30 12:08:39 on ttys001
[~ $ mkdir MyCoolTypeScriptProject
[~ $ cd MyCoolTypeScriptProject
[~/MyCoolTypeScriptProject $ yo tonyneed-vscode-typescript

  _--_
  |   |
  |--(o)--|
  \--_--/
  ( _'U'_ )
  /___A___\
  |   ~   |
  |___|___|
  |___|___|
  |___|___|

Welcome to Tony Sneed's
Visual Studio Code
TypeScript generator!

? Application Name (my-cool-type-script-project) _

```

Demo: Scaffolding with Yeoman



Questions?

