DEVWEEK

Having Fun with Visual Studio Code

Anthony Sneed

Twitter: @tonysneed

Email: tony.sneed@icloud.com

About Me

- **Status**: Married, three children
- **Locations**: Los Angeles, Dallas, Slovakia
- **Blog**: blog.tonysneed.com
- **Open Source Frameworks**:
  Trackable Entities, Simple MVVM Toolkit

KLIANT

kliant.com

Wintellect®
Know how.

wintellect.com

Global Knowledge.

globalknowledge.com

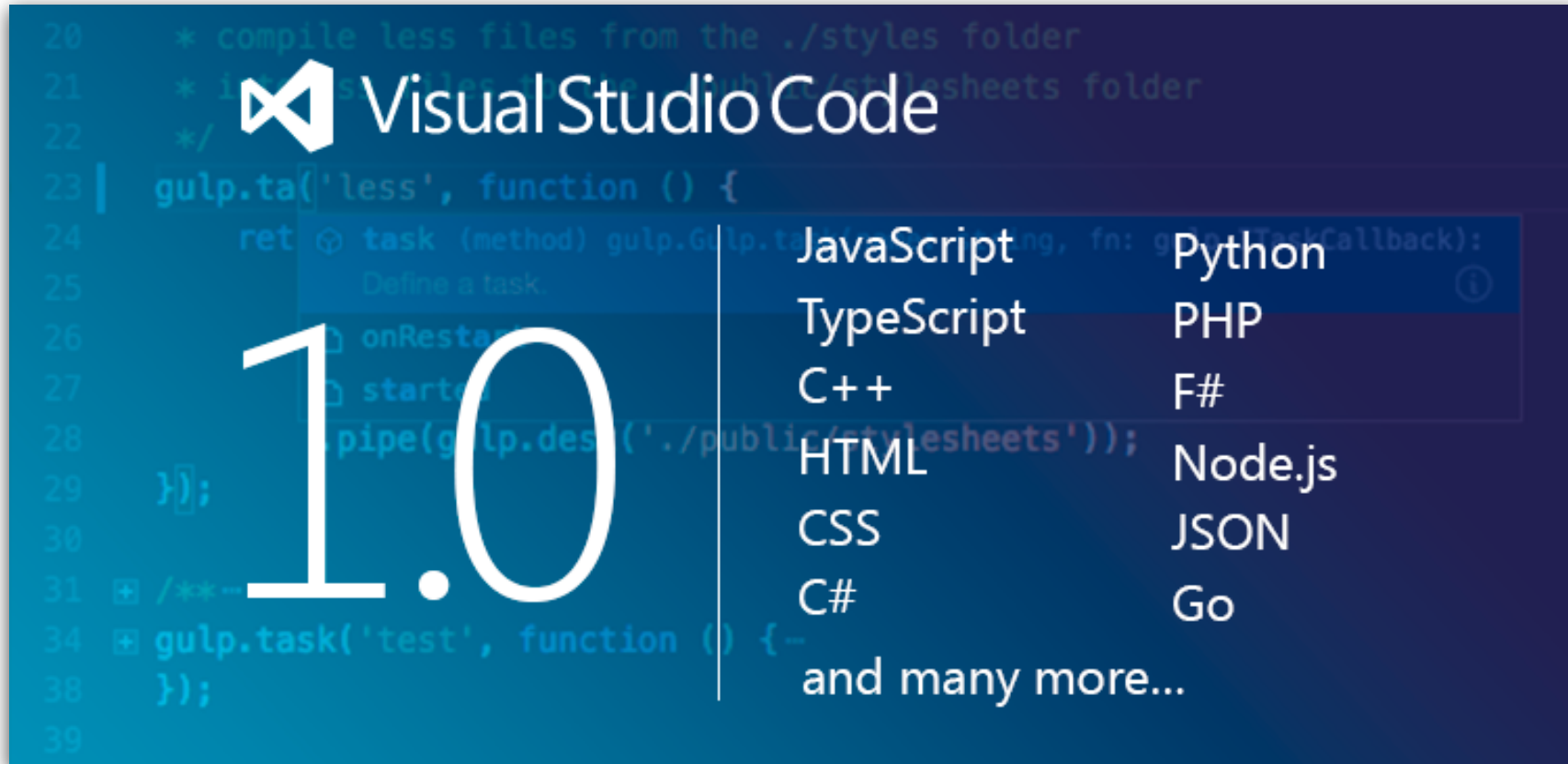# Get the Bits

github.com / **tonysneed** /

DevWeek.2016.VSCode-Fun

Agenda

- What is **Visual Studio Code**?

- **Languages** and **Features**

- Using the **Command Palette**

- **Navigation**, code **editing**, running **tasks**

- **Git** integration and **version control**

- **Node** and **C# Debugging**

There's a New Kid on the Block

What Is Visual Studio Code?

VS Code is a
cross-platform,
code-editor centric
development tool

VS Code Architecture

- Written in **TypeScript**
- Built on the **GitHub Electron Shell**
  - Based on **Node.js** and **Chromium**
  - Lets you write cross-platform desktop apps in **JavaScript**, **HTML**, **CSS**
- Leverages several **tool services**
  - TypeScript, Roslyn C# Compiler Service
  - Visual Studio debugging engine
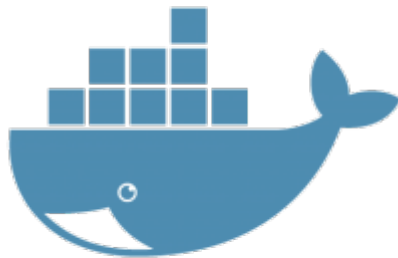
Some Supported Languages
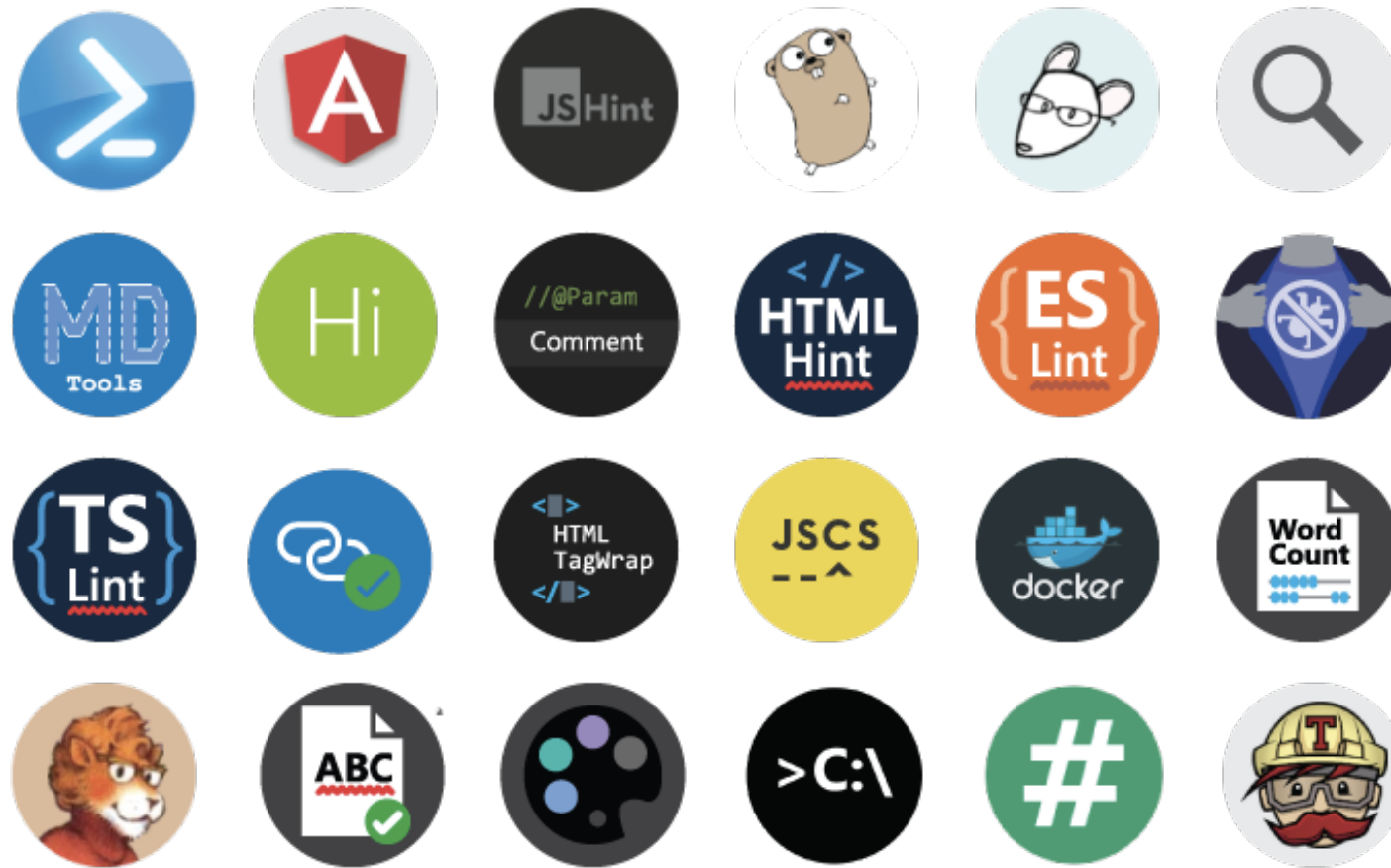

JavaScript

TypeScript

Markdown

Dockerfile

HTML & CSS

Extensions for Many More ...

Why VS Code?

- **Lightweight**, blazing **fast**
- Cross-platform - Windows, OS X, Linux
- Customizable, **extensible**
- It's a *real* code-editor
  - Syntax highlighting
  - Bracket matching
  - Auto indent
  - Code snippets

What's Missing?

- Project and item **templates**
- Visual **designers** (XAML)
- **Razor** syntax highlighting
- **ReSharper-like** goodness
- Advanced **Debugging**
  - Set *next statement*
  - Debugger *visualizers*
  - *Threads* window

Folder-Based Project System

- **File**, **Open**, select a **Folder**
  - Everything in the **folder** is *included*
- Open VS Code from **command line**: `code .`
- Can also open VS Code from the **Finder** or **File Explorer**
  - Windows: installer has option to include shell command
  - Mac OS X: http://bit.ly/mac-open-in-vscode

File New Project = Yeoman

- Use **Yeoman** to scaffold new projects
  - Install **Node**: https://nodejs.org
  - Use **npm** to install Yeoman
  - Search: http://yeoman.io/generators
  - Use npm to install **generators**

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

Example: New TypeScript Project

Yo Extension for VS Code

- Install **Yo** **extension** for VS Code
  - Create *new project* with a Yeoman **generator** - star
  - Add *new items* with a Yeoman **sub-generator**

1. VS Quick Open: **Cmd + P**
2. Type: **ext install yo**
3. Open Cmd Palette: **F1**
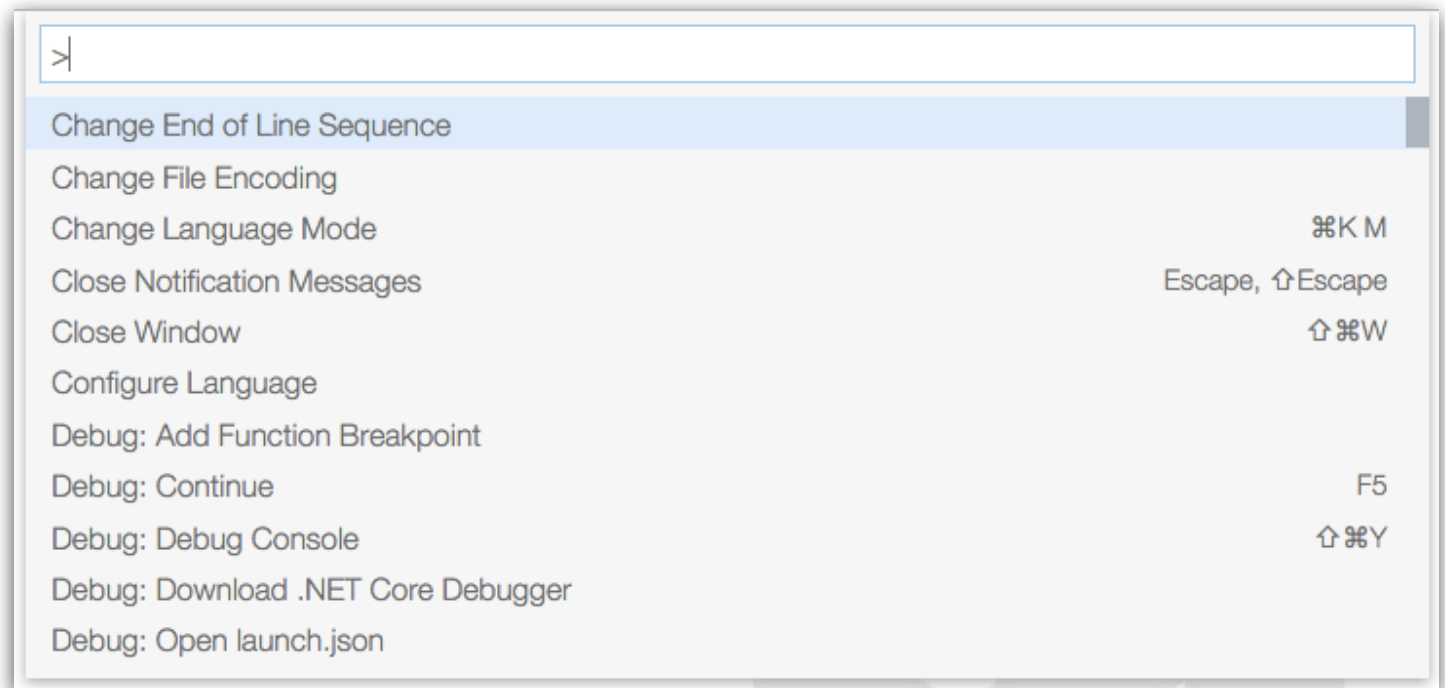4. Type: **yo** [(**sub**)**generator**]

Demo: Using VS Code with Yeoman

The Command Palette

- Provides quick access to **all functions** in VS Code

- **F1** or **Cmd + Shift + P**

- Enter command **name**

- *Note keyboard* ***shortcuts****!*

Navigation Basics

Open a file	Cmd + P
Cycle recent files	Ctrl + Tab (forward) Ctrl + Shift + Tab (reverse)
Go to symbol Go to line number	Cmd + Shift + O Ctrl + G
Go back Go forward	Ctrl + - Ctrl + Shift + -
Go to definition, Peek definition Find all references	F12, Alt + F12 Shift + F12
Show & cycle errors and warnings	Cmd + Shift + M F8 (forward), Shift + F8 (reverse)

Trigger Intellisense

Trigger intellisense - also works with *package*/*project* JSON	Ctrl + Space

Show Parameter Info

Show parameter info	Cmd + Shift + Space
Cycle through overloads	Up and down arrows

Code Snippets

- Start typing or press **Cmd + Space** to use **code snippets**
- Built-in snippets for **TypeScript, C#**
- Install other snippets as **extensions**
- *Create your own!*
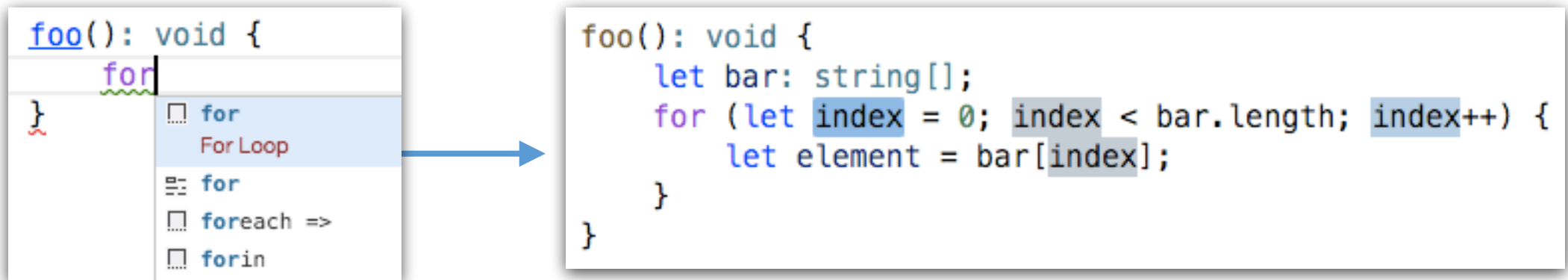
Format Code, Comments, Markdown

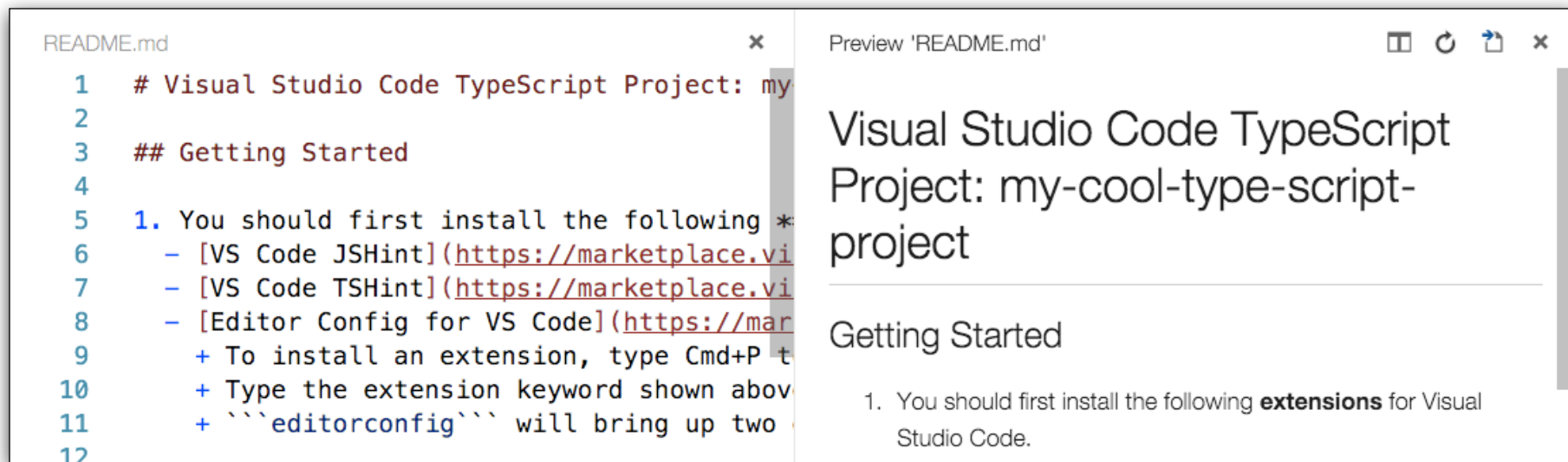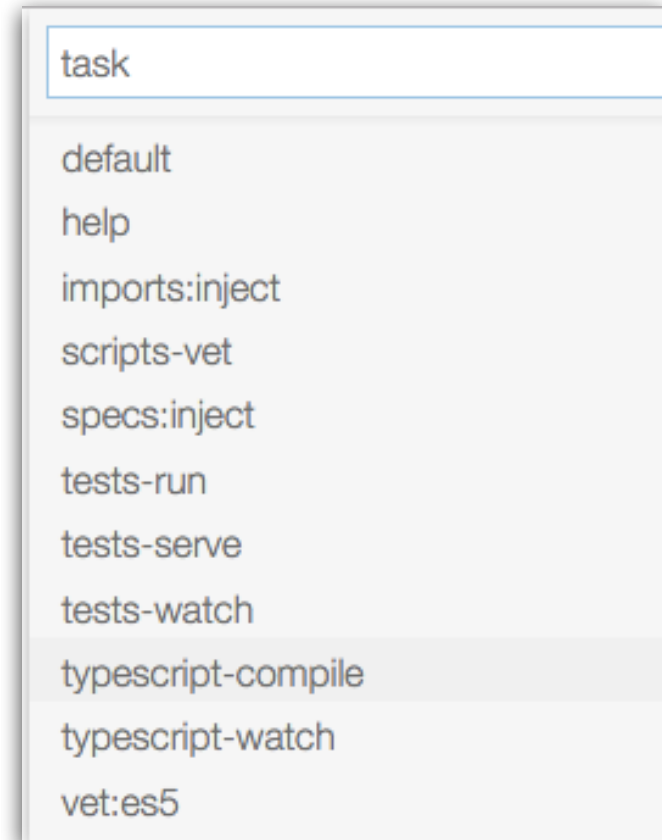|	
Format code, toggle comment	Shift + Alt + F, Cmd + /
Preview Markdown	Cmd + Shift + V
Split window, switch windows	Cmd + \, Cmd + 1, 2, 3

Demo: Navigation, Intellisense, Formatting

Running Tasks

- Tasks **automate** your dev workflow
  - Examples: *compilation, linting, tests, builds*
- VS Code **auto-detects** tasks from a **Gulpfile** (also Grunt, Jake, etc)
  - Can execute some tasks in the **background** (file watches, etc)
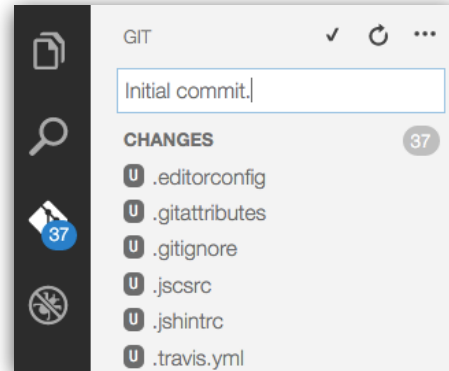- Config runner: **Build** and **Test** tasks
  - **Cmd + B, Cmd + T**

task

default
help
imports:inject
scripts-vet
specs:inject
tests-run
tests-serve
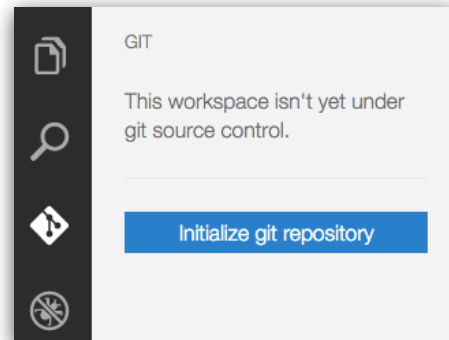tests-watch
typescript-compile
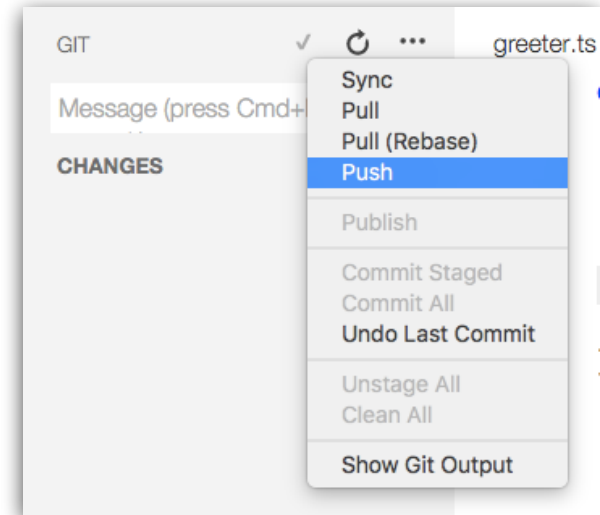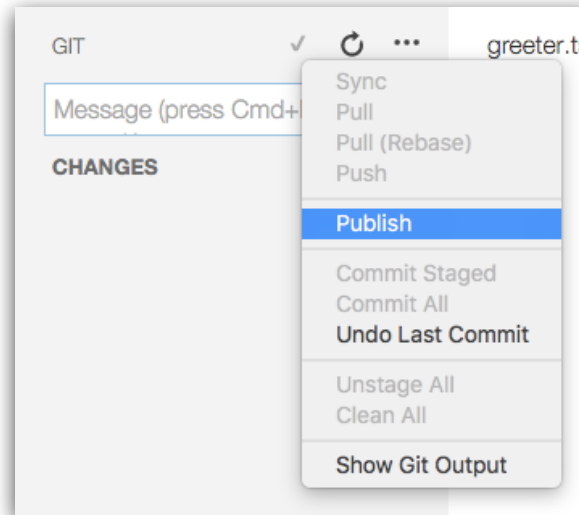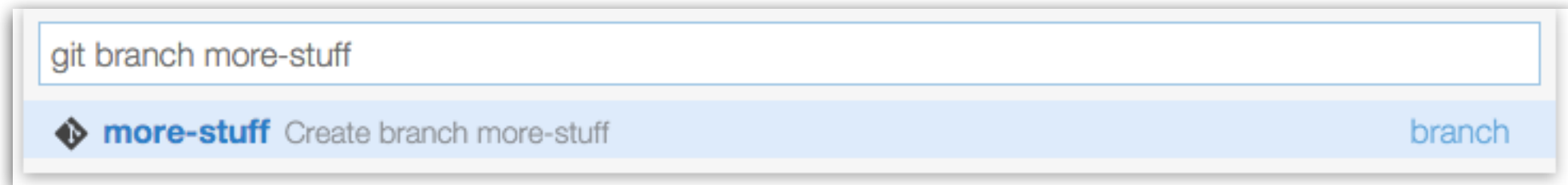typescript-watch
vet:es5

Demo: Running Tasks

Version Control with Git


```
git remote add origin https://github.com/tonysneed/
MyCoolTypeScriptProject.git
```

More Git Goodness

Even More Git Goodness

- View **file diffs**, resolve **conflicts**


```
greeter.ts  src/greeter - Changes on working tree
1  export class Greeter {
2      constructor(public message: string) {
3      }
4
5      greet(): string {
6          return "Hello " + this.message;
7      }
8  }
```

```
1  export class Greeter {
2      constructor(public message: string) {
3      }
4
5      greet(): string {
6          return "Ciao " + this.message;
7      }
8  }
```

Limitations

- For other Git tasks, use a *real* **Git client** or the **command line**


```
git stash
git rebase HEAD~3 -i
git log --oneline
git reset --hard a9c7080
```

Use VS Code as Your Git Editor

- Configure Git to use **VS Code** for its **text editor**

```
git config --global core.editor "code --wait"
```

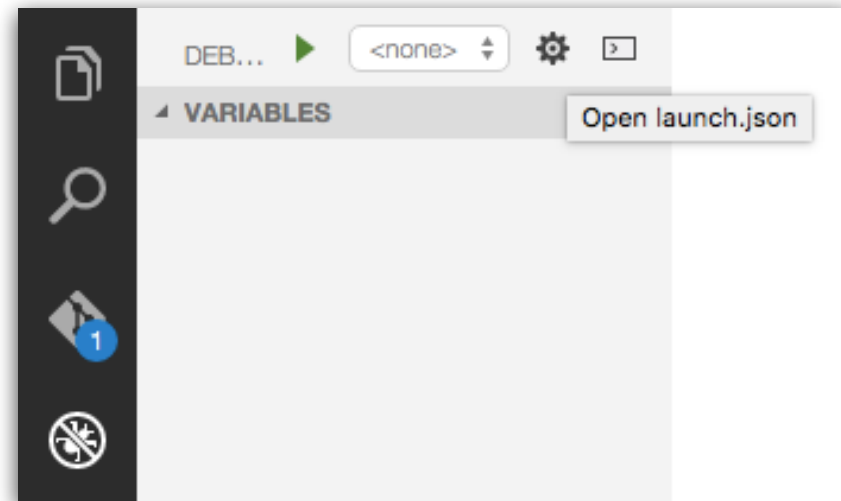- Git will then use VS Code for **interactive** add and rebase

```
git add -p
git rebase HEAD~3 -i
```

Demo: Git Integration

Debugging: Getting Started

- Built-in debugging for **Node.js** (JavaScript, TypeScript, etc)
  - Debug **other languages** by installing an **extension**
  - **C#** debugging available for **Mono** and **.NET Core**
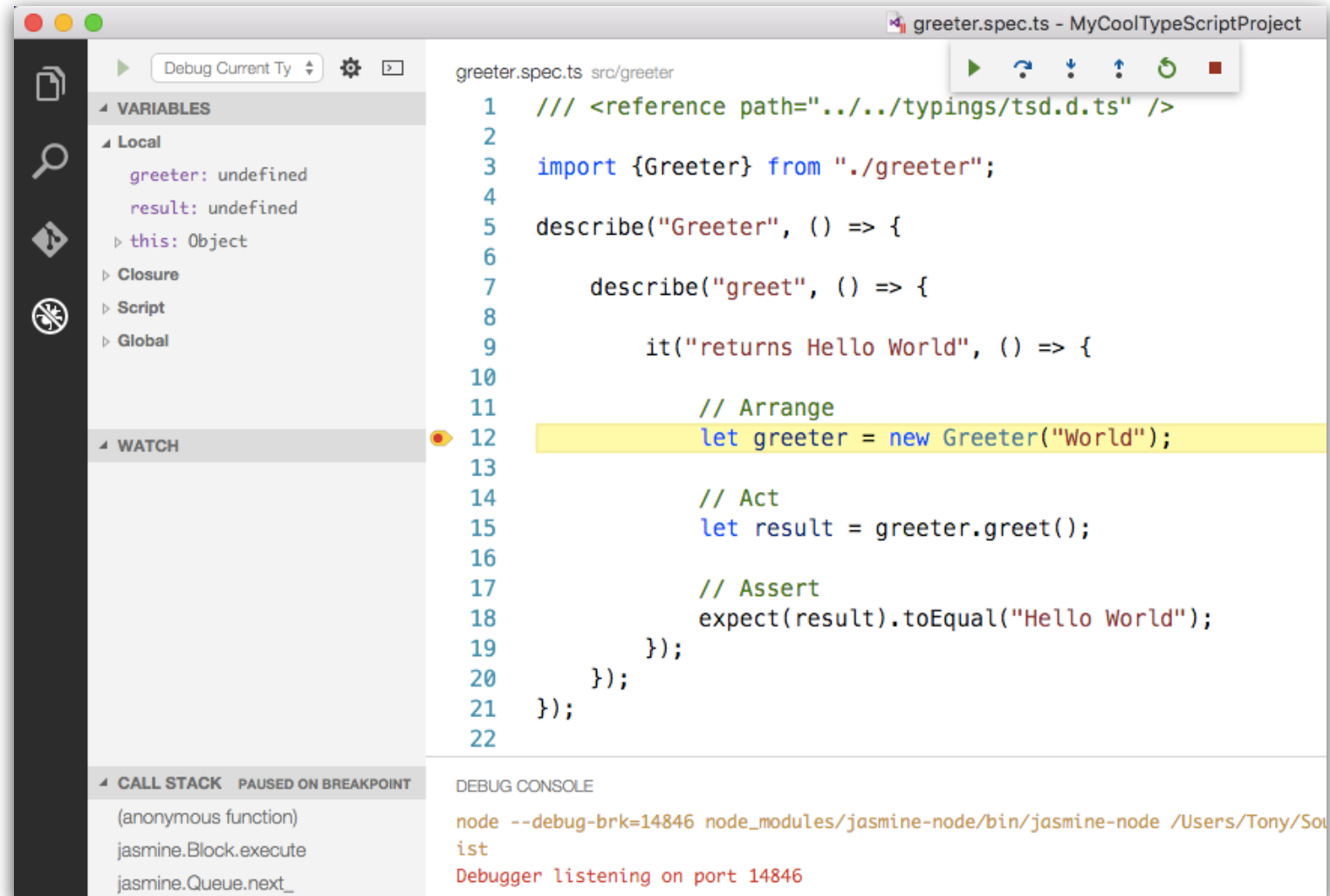
Debugging: Node - TypeScript

```json
{
    "compilerOptions": {
        "module": "commonjs",
        "target": "es5",
        "sourceMap": true,
        "declaration": true,
        "removeComments": true,
        "noImplicitAny": true,
        "rootDir": ".",
        "outDir": "../dist"
    },
    "exclude": [
        "node_modules"
    ]
}
```
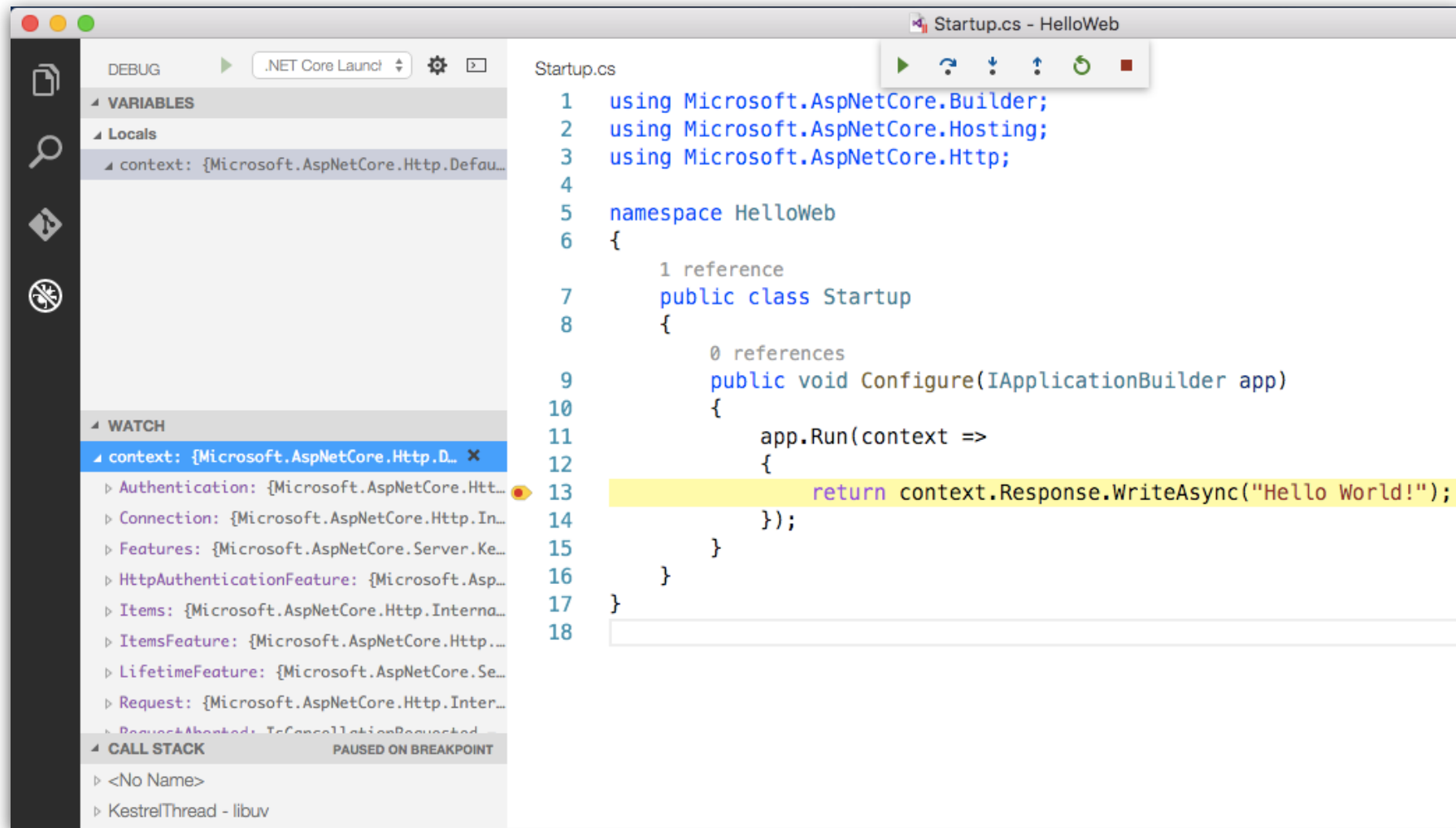
- Must configure **tsconfig.json**
  - Set **sourceMap** to **true**
  - Set **rootDir**, **outDir**

Debugging: Node - TypeScript
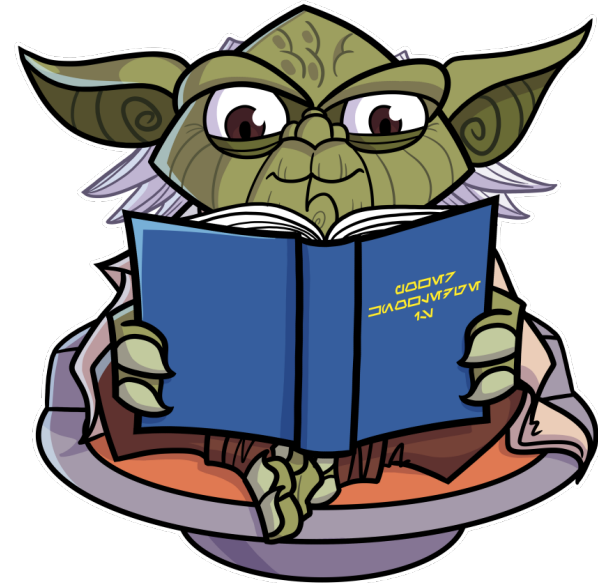
Debugging: C# - .NET Core

Demo: Node Debugging

Demo: .NET Core Debugging

Additional Resources

- VS Code **Docs**:
  http://bit.ly/vscode-docs

- "Official" **Tips and Tricks**:
  http://bit.ly/vscode-tips

- VS Code **eBook** from MS:
  http://bit.ly/vscode-ebook

- VS Code **Extensions** Marketplace:
  https://marketplace.visualstudio.com/VSCode

Questions?

Get the Bits

github.com / **tonysneed** /

DevWeek.2016.VSCode-Fun

Contact Me

- Email: **tony.sneed@icloud.com**

- Blog: **blog.tonysneed.com**

- Twitter: **@tonysneed**