



## Having Fun with Visual Studio Code

**Anthony Sneed**

Twitter: [@tonysneed](https://twitter.com/tonysneed)

Email: [tony.sneed@icloud.com](mailto:tony.sneed@icloud.com)

# About Me

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



[kliant.com](http://kliant.com)



[wintellect.com](http://wintellect.com)



[globalknowledge.com](http://globalknowledge.com)

# Get the Bits



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



## Visual Studio Code

1.0

JavaScript	Python
TypeScript	PHP
C++	F#
HTML	Node.js
CSS	JSON
C#	Go
and many more...	

```
20 * compile less files from the ./styles folder
21 * install the public/stylesheets folder
22 */
23 | gulp.task('less', function () {
24   ret task (method) gulp.Gulp.task('less', function (cb) {
25     Define a task.
26   });
27   onRestart
28   started
29   pipe(gulp.dest('./public/stylesheets'));
30   });
31 + /**~
34 + gulp.task('test', function () {~
38   });
39
```

# 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



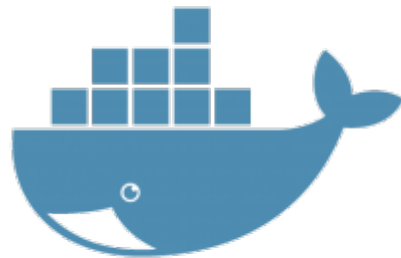
JavaScript



TypeScript



Markdown

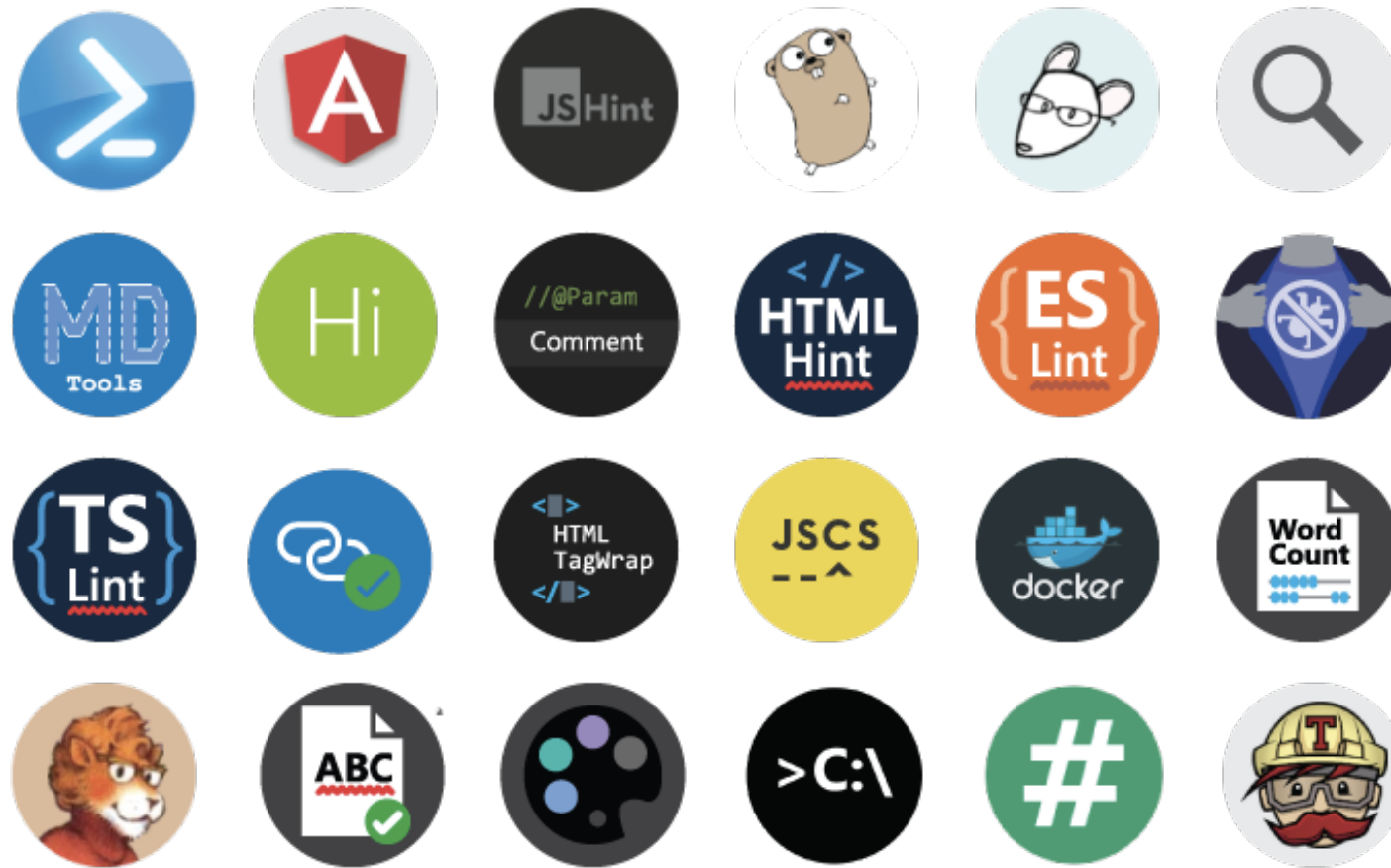


Dockerfile





# 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**
  - Debugger *visualizers*
  - *Conditional* breakpoints
  - *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>

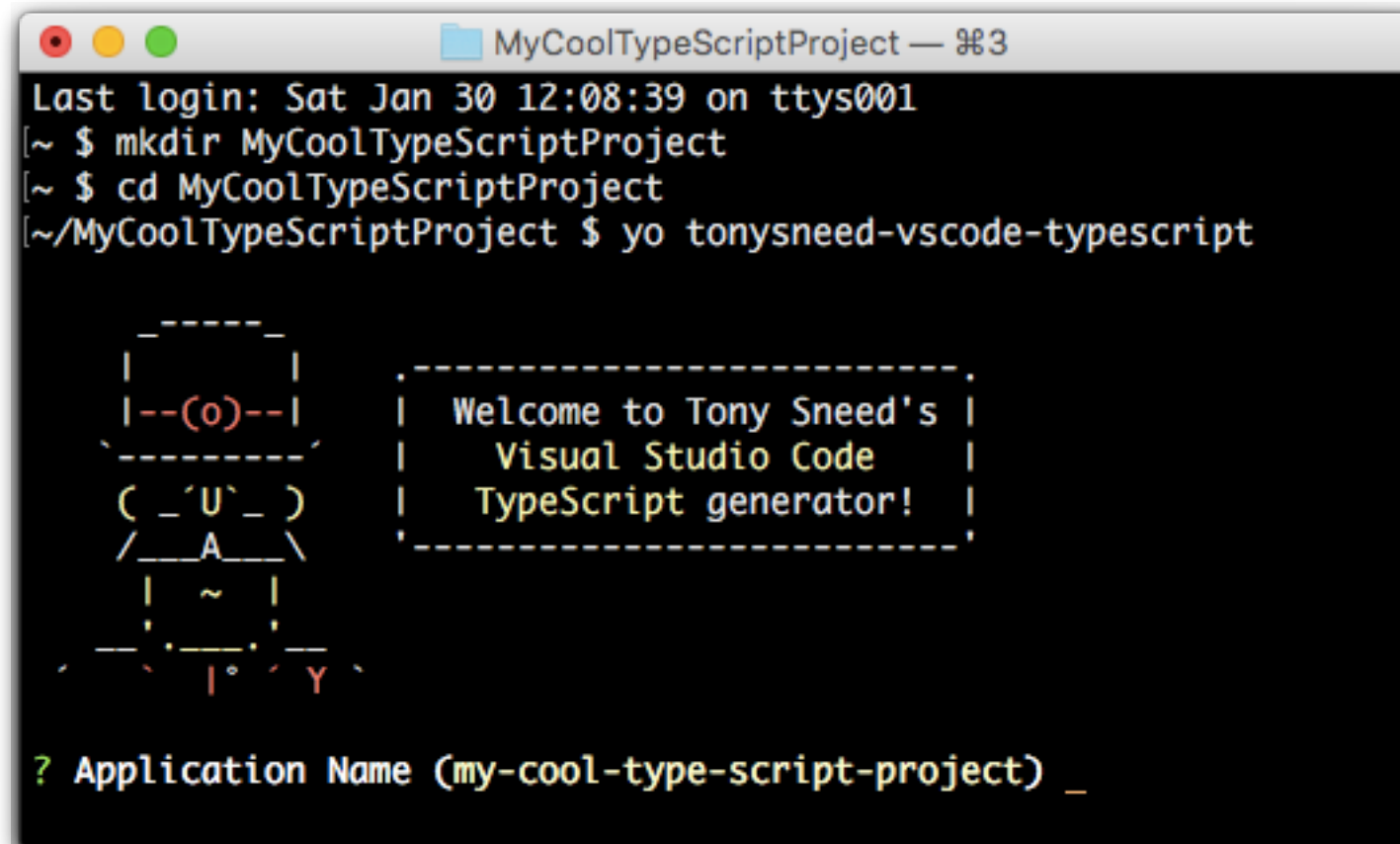
# File New Project = Yeoman



- 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 tonyneed-vscode-typescript
```

# Example: New TypeScript Project



```
MyCoolTypeScriptProject — 3
Last login: Sat Jan 30 12:08:39 on ttys001
[~ $ mkdir MyCoolTypeScriptProject
[~ $ cd MyCoolTypeScriptProject
[~/MyCoolTypeScriptProject $ yo tonysneed-vscode-typescript

  _ _ _ _ _
 |         |
 |  --(o)--  |
 |         |
 |  _ _ 'U' _ _  |
 |         |
 |  _ _ A _ _  |
 |         |
 |  _ _ ~ _ _  |
 |         |
 |  _ _ . _ _  |
 |         |
 |  _ _ I _ _  |
 |         |
 |  _ _ Y _ _  |
 |         |

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

? Application Name (my-cool-type-script-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]**

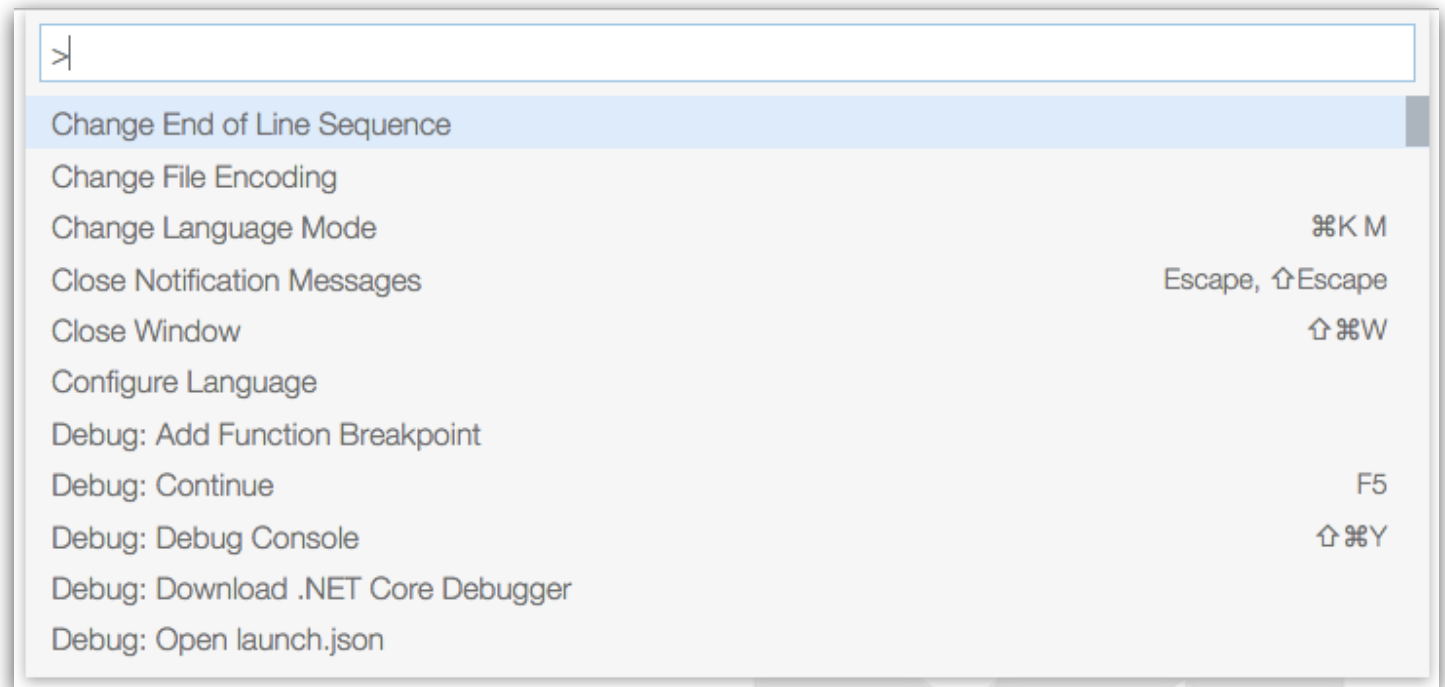
# Demo: Using VS Code with Yeoman





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

# Trigger Intellisense

Trigger **intellisense**

- also works with *package/project* **JSON**

Ctrl + Space

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

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

# Show Parameter Info

Show **parameter** info

Cmd + Shift + Space

Cycle through **overloads**

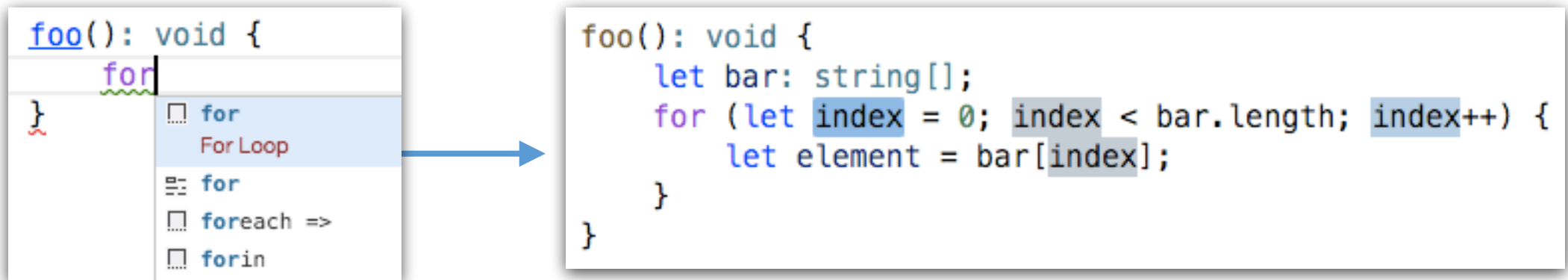
Up and down arrows

```
public class Program
{
    0 references
    public static void Main
    {
        Console.WriteLine()
    }
}
```

void Console.WriteLine(string format, object arg0, object arg1, object arg2)  
^ 15 <member  
v name="M:System.Console.WriteLine(System.String,System.Object,System.C

# 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

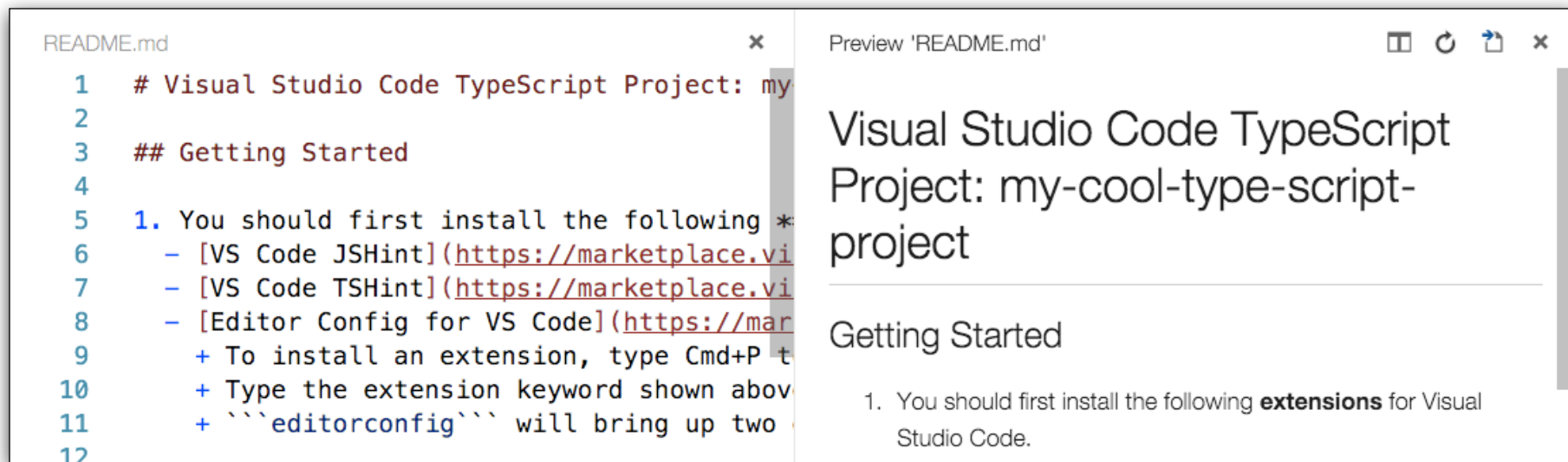
Shift + Alt + F, Cmd + /

Preview Markdown

Cmd + Shift + V

Split window, switch windows

Cmd + \, Cmd + 1, 2, 3

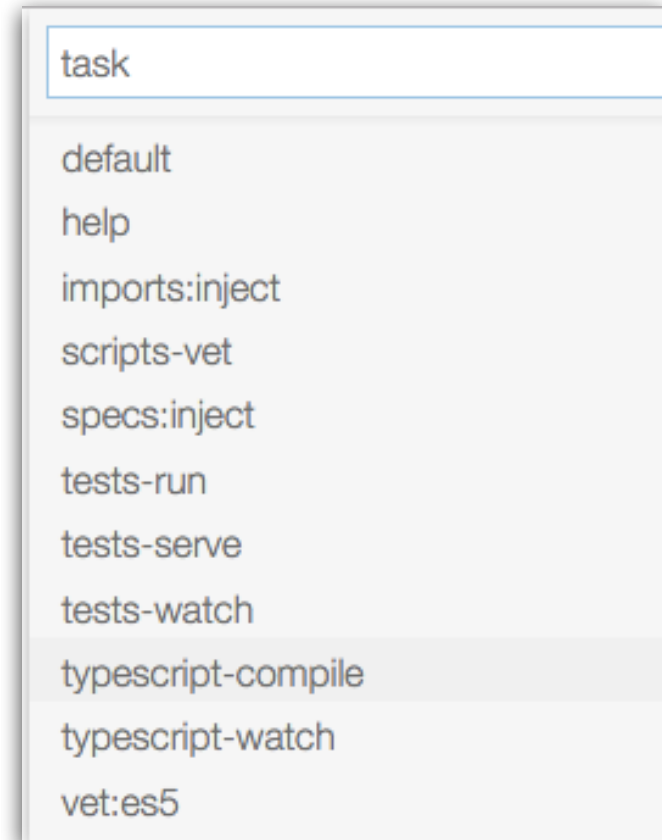


# Demo: Navigation, Intellisense, Formatting



# 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

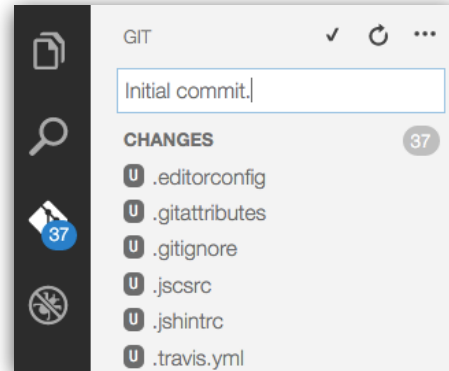
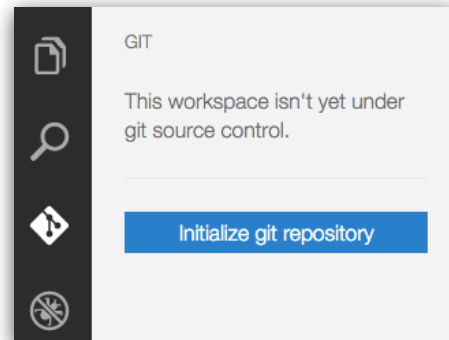




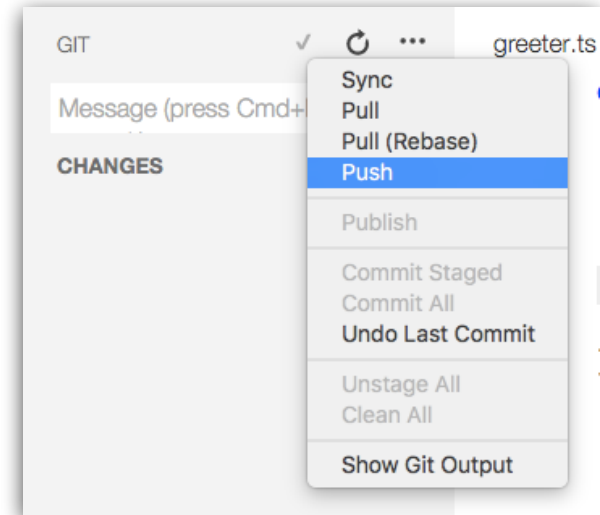
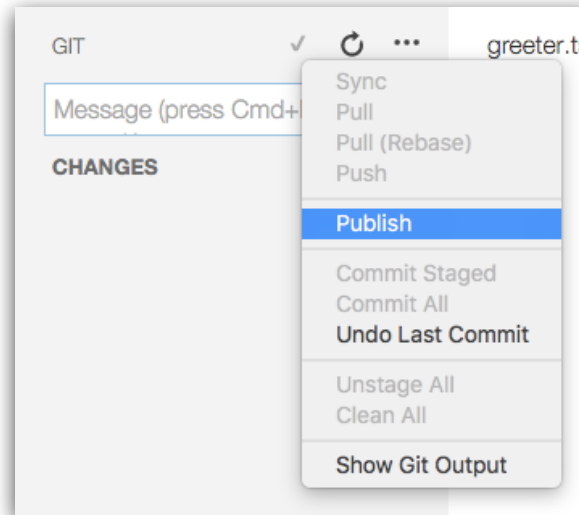
# Demo: Running Tasks



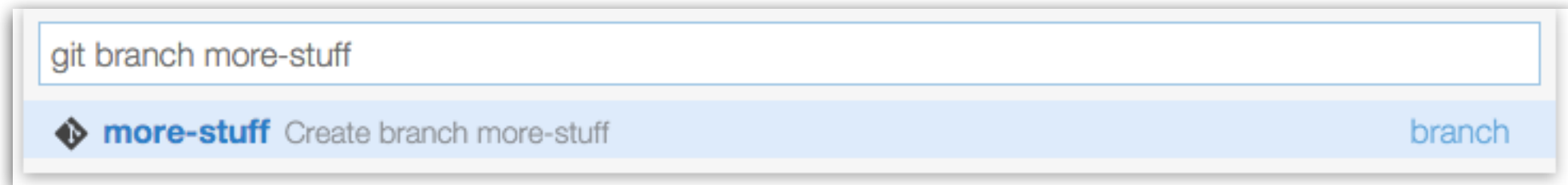
# Version Control with Git



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



# More Git Goodness



↑  
Checkout  
Branch

↑  
Publish  
Branch

# 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

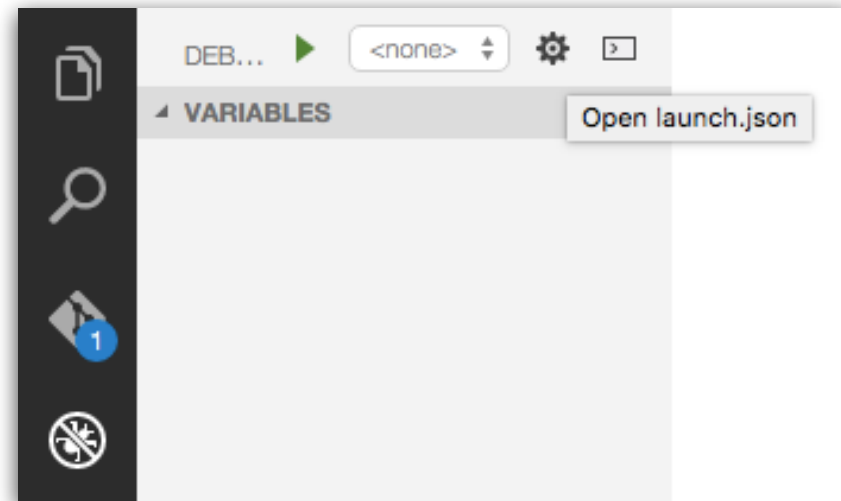
```
git add -p  
git rebase HEAD~3 -i
```

# 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**



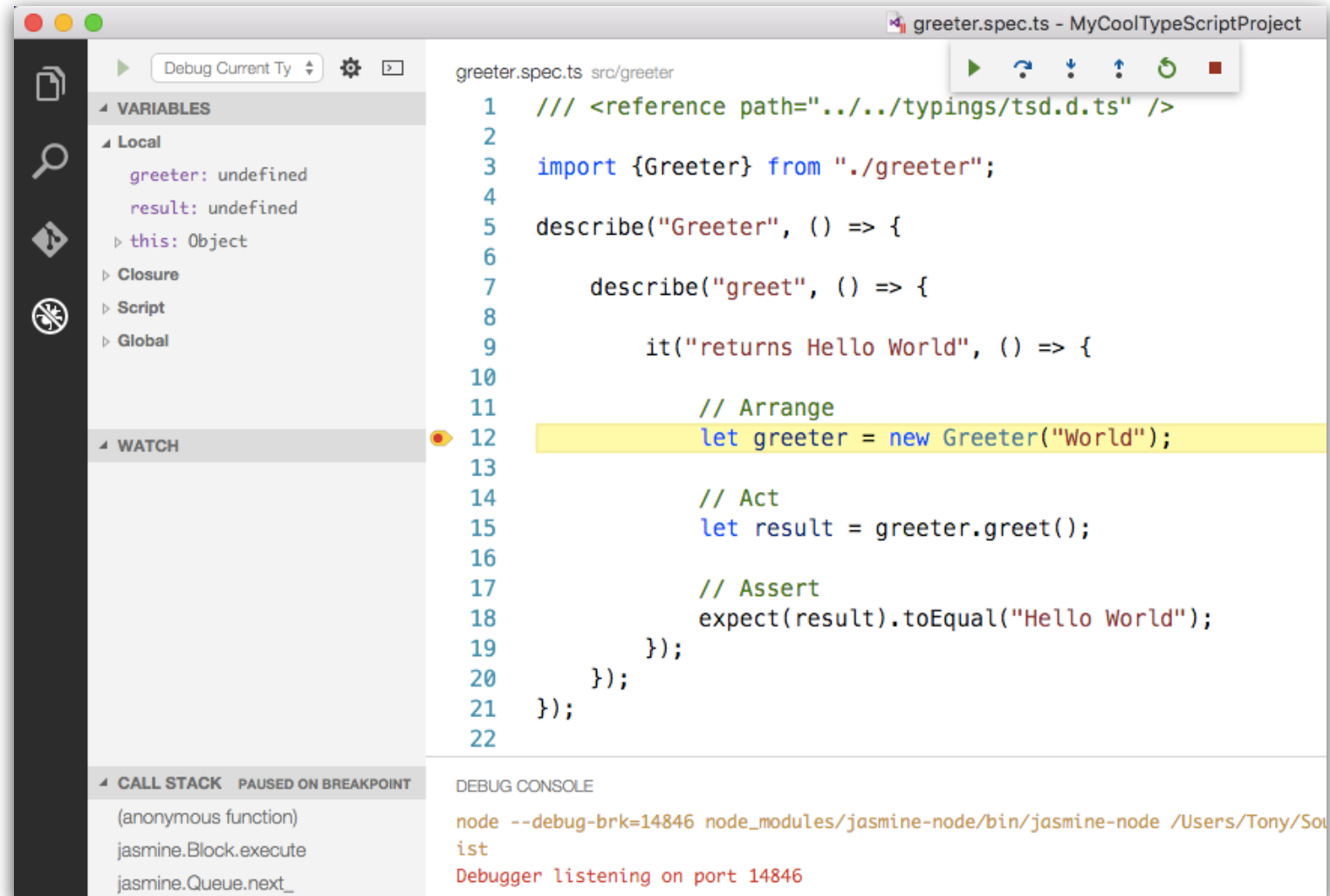


# Debugging: Node - TypeScript

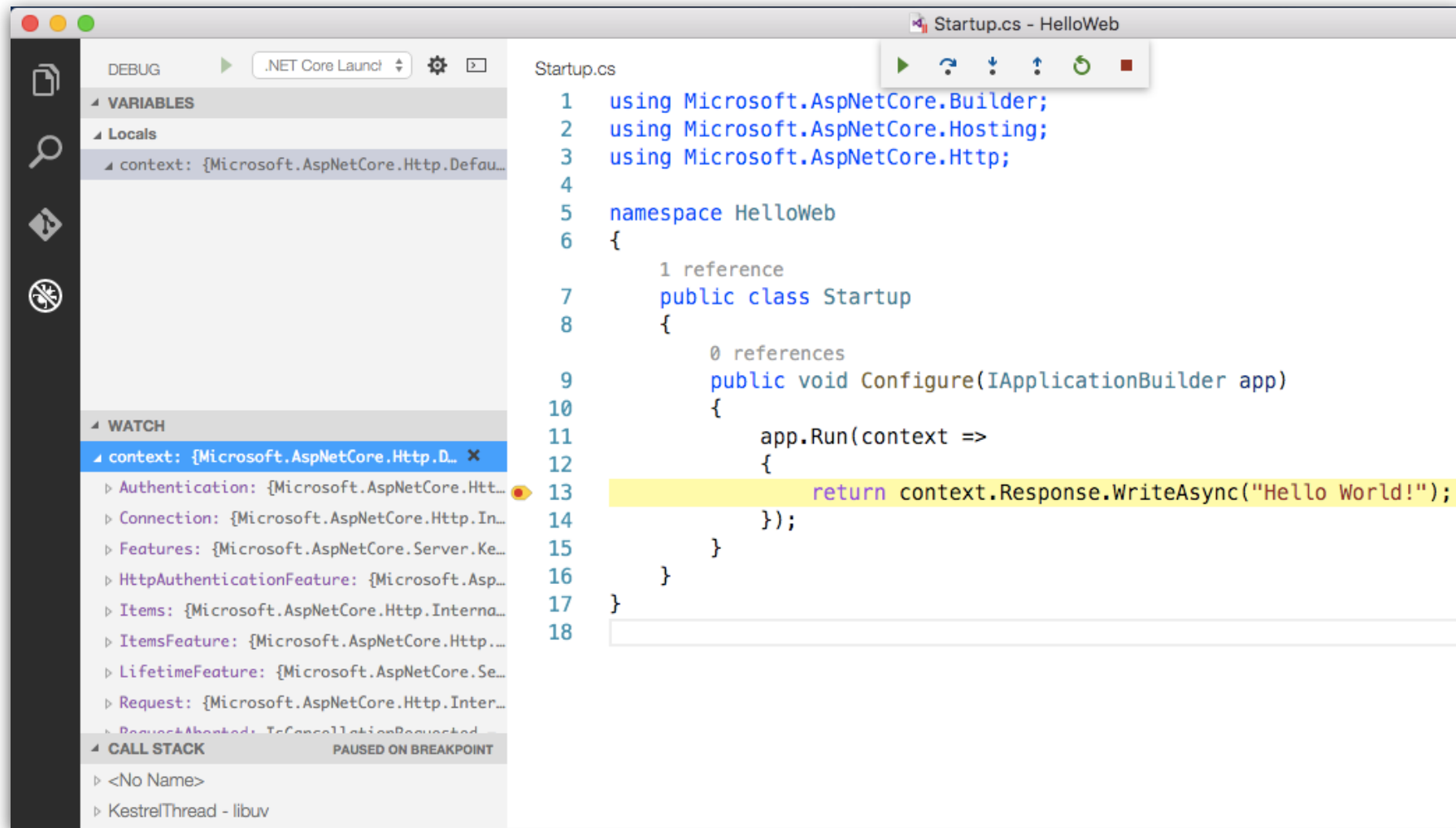
```
{
  "compilerOptions": {
    "module": "commonjs",
    "target": "es5",
    "sourceMap": true,
    "declaration": true,
    "removeComments": true,
    "noImplicitAny": true,
    "rootDir": ".",
    "outDir": "../dist"
  },
  "exclude": [
    "node_modules"
  ]
}
```

- Must configure **tsconfig.json**
  - Set **sourceMap** to true
  - Set rootDir, outDir

# Debugging: Node - TypeScript



# Debugging: C# - .NET Core



# Demo: Node Debugging

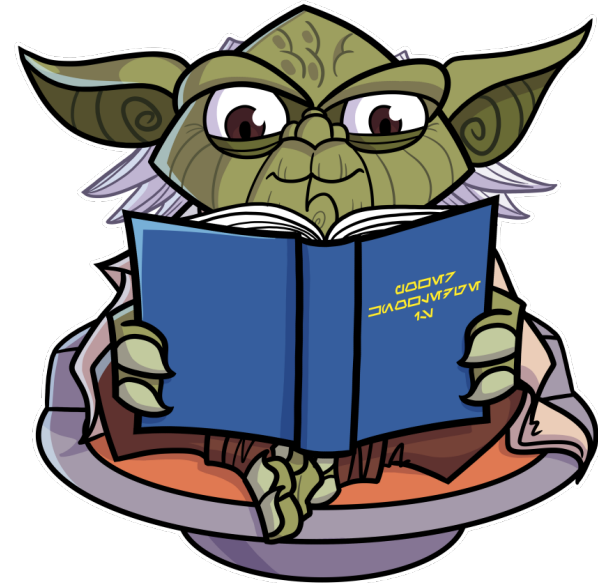


# Demo: .NET Core Debugging



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

**DevWeek.2016.VSCode-Fun**



# Contact Me



- Email: [tony.sneed@icloud.com](mailto:tony.sneed@icloud.com)



- Blog: [blog.tonysneed.com](http://blog.tonysneed.com)



- Twitter: [@tonysneed](https://twitter.com/tonysneed)