Visual Studio Code

Larry O’Heron
Laboratory for Laser Energetics
Lead Analyst

Overview

I use VSE at work. I thought it would be interesting to learn about VSC.

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, Mac and Linux.

It's right for you if you don't need/want a heavy-weight development IDE.

It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Python, PHP) and runtimes.

Agenda

Download and Install

Node.js ‘Hello World’

Lint

Debugging

Node.js Simple Web App

GIT

C#

C++

Themes

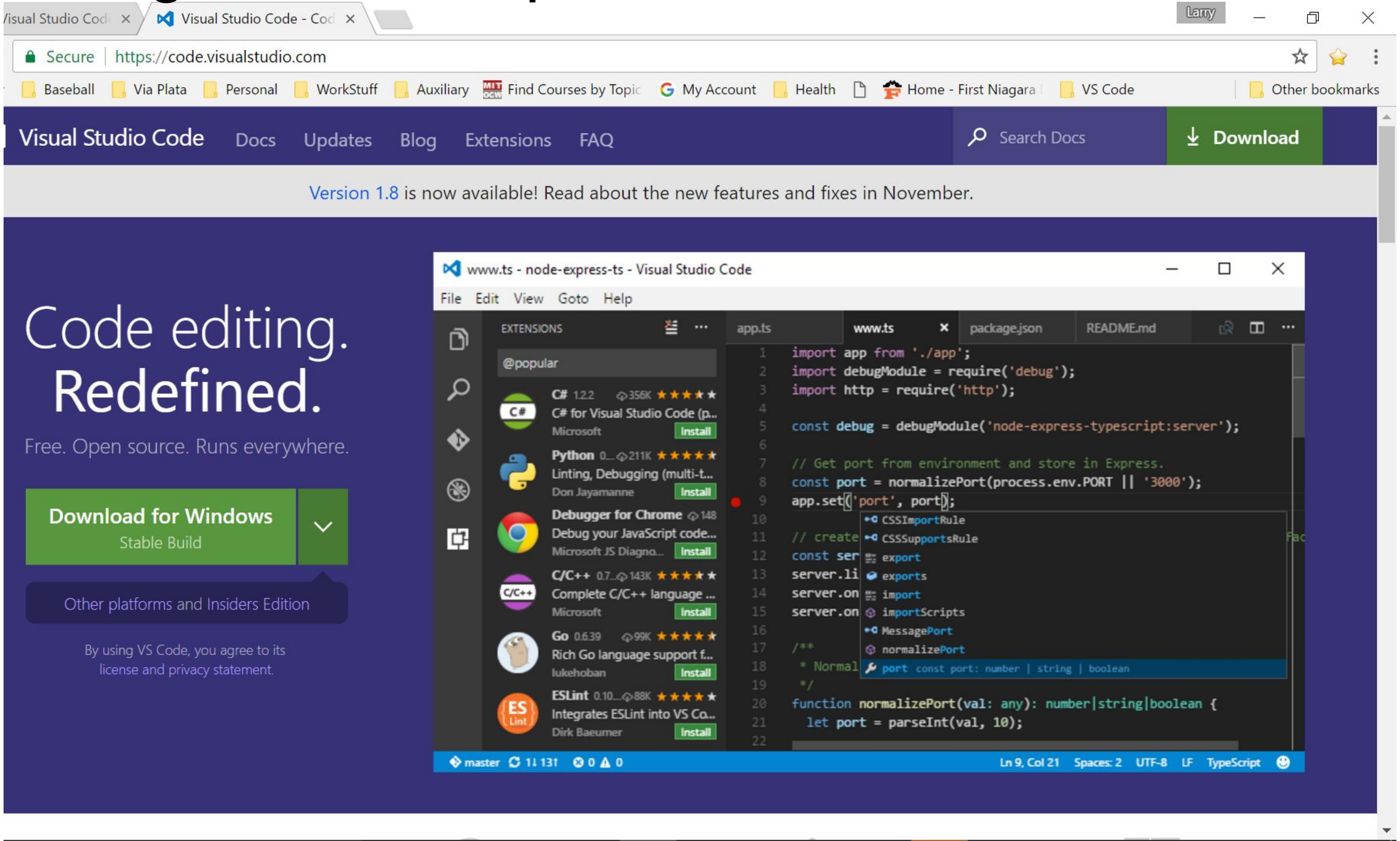
Settings

Asp.Net Core App

Snippets

Intellisense

Starting Place -- https://code.visualstudio.com/



Version 1.8 is now available! Read about the new features and fixes in November.

Code editing. Redefined.

Free. Open source. Runs everywhere.

By using VS Code, you agree to its license and privacy statement.

C# 1.2.2	356K ★★★★★
C# for Visual Studio Code (p...	
Microsoft	Install
Python 0...	211K ★★★★★
Linting, Debugging (multi-t...	
Don Jayamanne	Install
Debugger for Chrome	148
Debug your JavaScript code...	
Microsoft JS Diagno...	Install
C/C++ 0.7...	143K ★★★★★
Complete C/C++ language ...	
Microsoft	Install
Go 0.6.39	99K ★★★★★
Rich Go language support f...	
lukehoban	Install
ESLint 0.10...	88K ★★★★★
Integrates ESLint into VS Co...	
Dirk Baeumer	Install

```
import app from './app';
import debugModule = require('debug');
import http = require('http');

const debug = debugModule('node-express-typescript:server');

// Get port from environment and store in Express.
const port = normalizePort(process.env.PORT || '3000');
app.set('port', port);

// create
const ser
server.li
server.on
server.on

/**
 * Normal
 */
function normalizePort(val: any): number|string|boolean {
  let port = parseInt(val, 10);
```

Download and Install

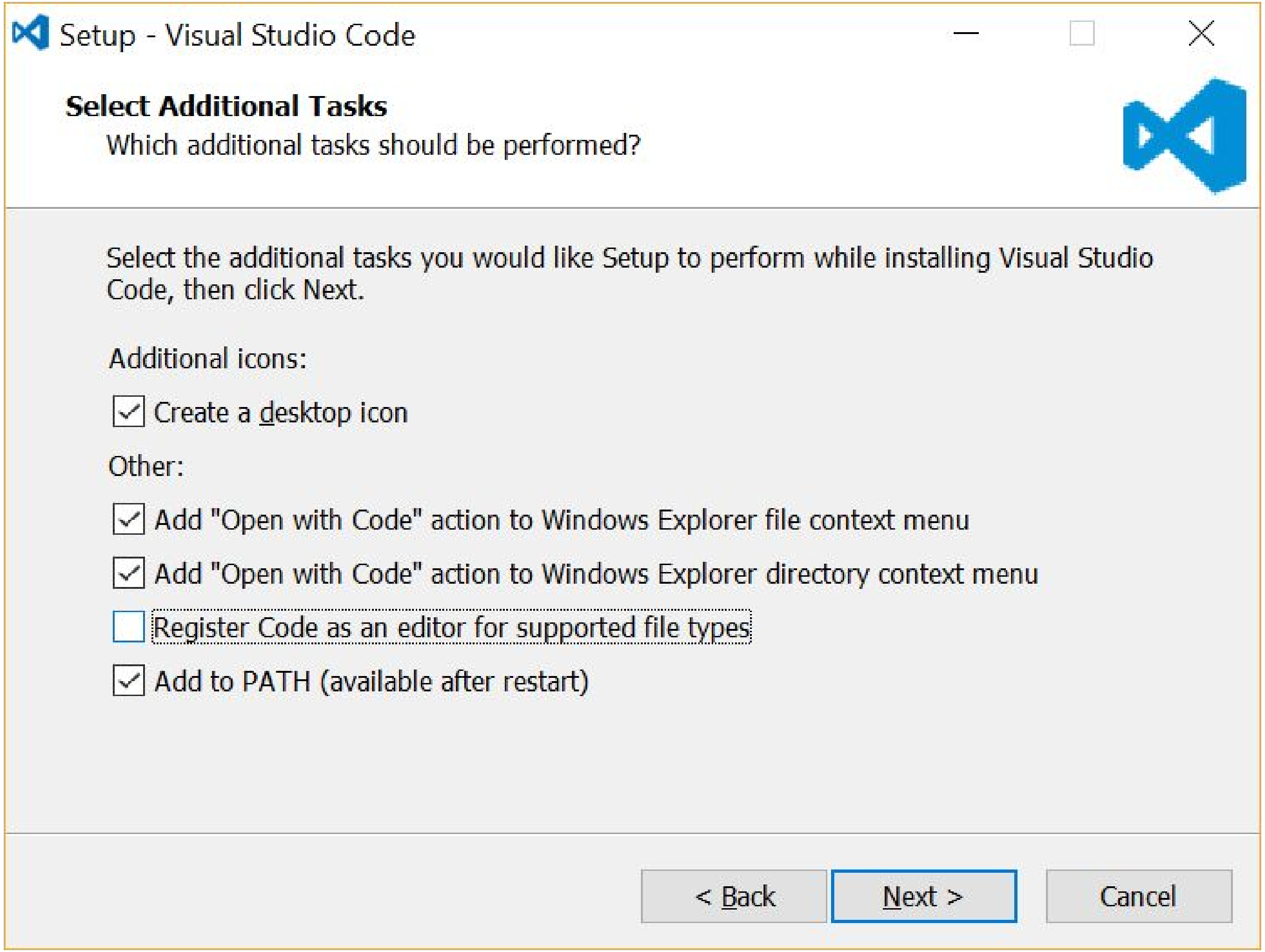
Download link: https://code.visualstudio.com/docs/?dv=win

What’s new?

- **Intellisense**
- **Debugging in VSC**
- **Linting, multi-cursor editing, parameter hints**
- **Git support**
- **Peek and navigate to definition**

Requirement: .Net Framework 4.5.2 (Win 7 users take note!)

‘Open with Code’ means that one can r/c on a folder in Windows Explorer to open Visual Studio Code



Setup - Visual Studio Code

Select Additional Tasks

Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing Visual Studio Code, then click Next.

Additional icons:

☑ Create a desktop icon

Other:

☑ Add "Open with Code" action to Windows Explorer file context menu

☑ Add "Open with Code" action to Windows Explorer directory context menu

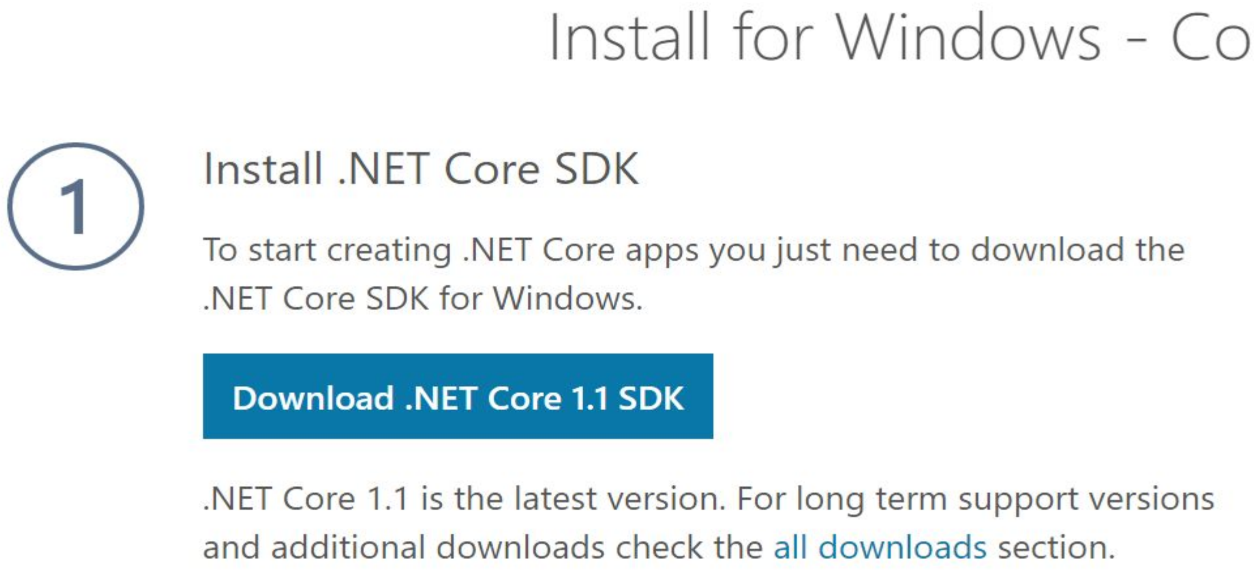
□ Register Code as an editor for supported file types

☑ Add to PATH (available after restart)

Install .NET Core

“Download .NET Core”. Ensure to not get .NET for Visual Studio.

- **fast and modular platform for creating server app that run on Windows, Linux and Mac.**



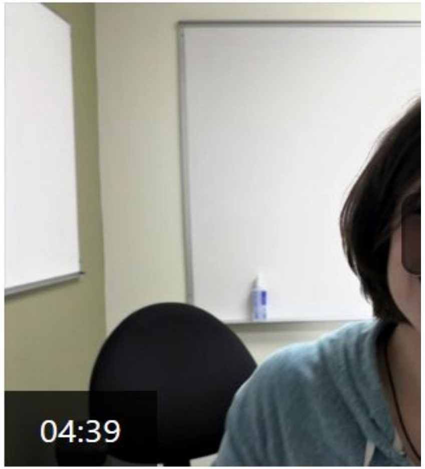
Install for Windows - Command Line

1

Install .NET Core SDK

To start creating .NET Core apps you just need to download the .NET Core SDK for Windows.

.NET Core 1.1 is the latest version. For long term support versions and additional downloads check the all downloads section.



Initial Configuration for .Net Core inside VSC

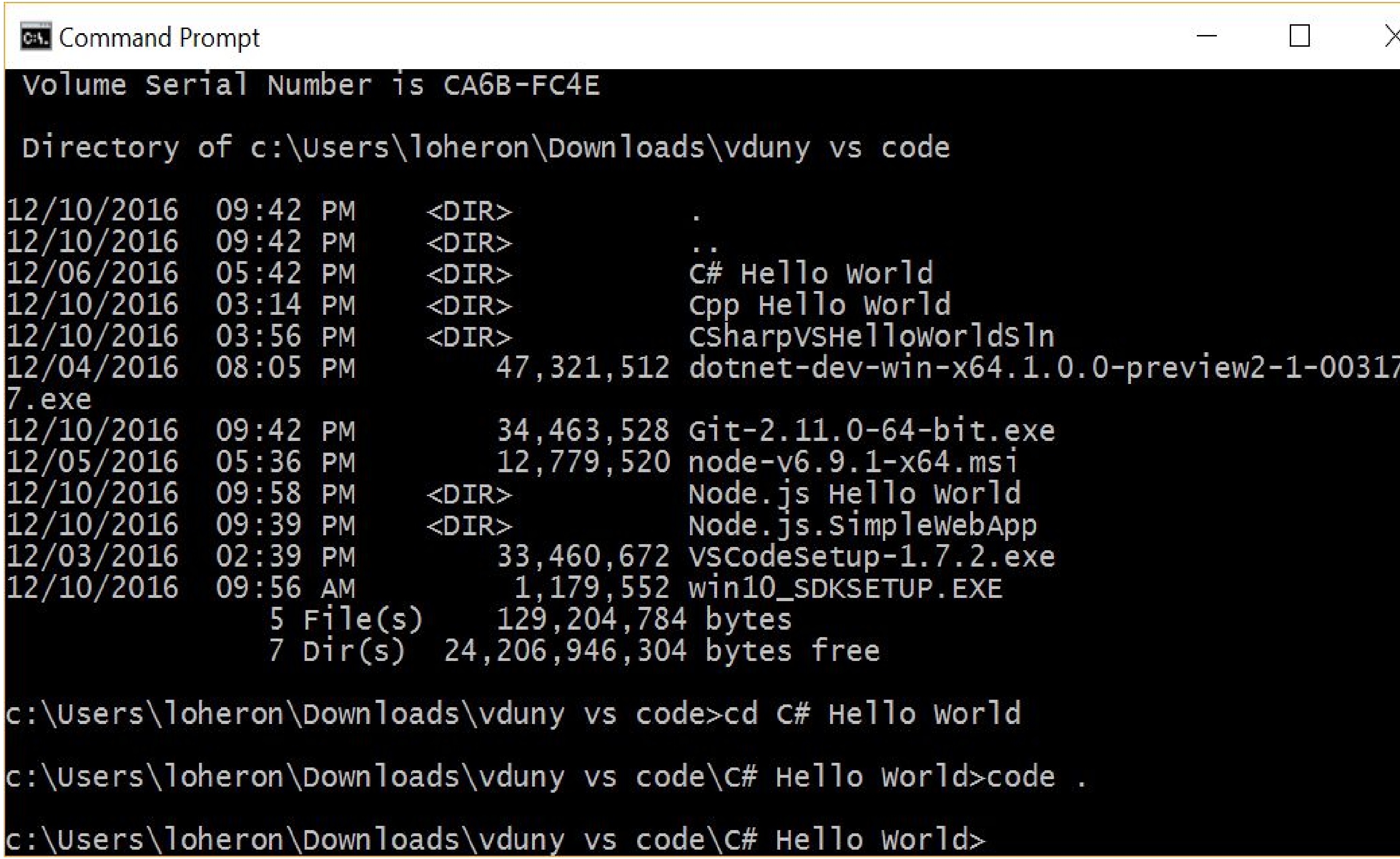
Ctrl ` → to open a terminal session.

dotnet new → configures the system. Only runs once.

***** Because I have already run this command, and because I have c# extensions already installed, I get a different result on my laptop.**

Opening from a Command Console:

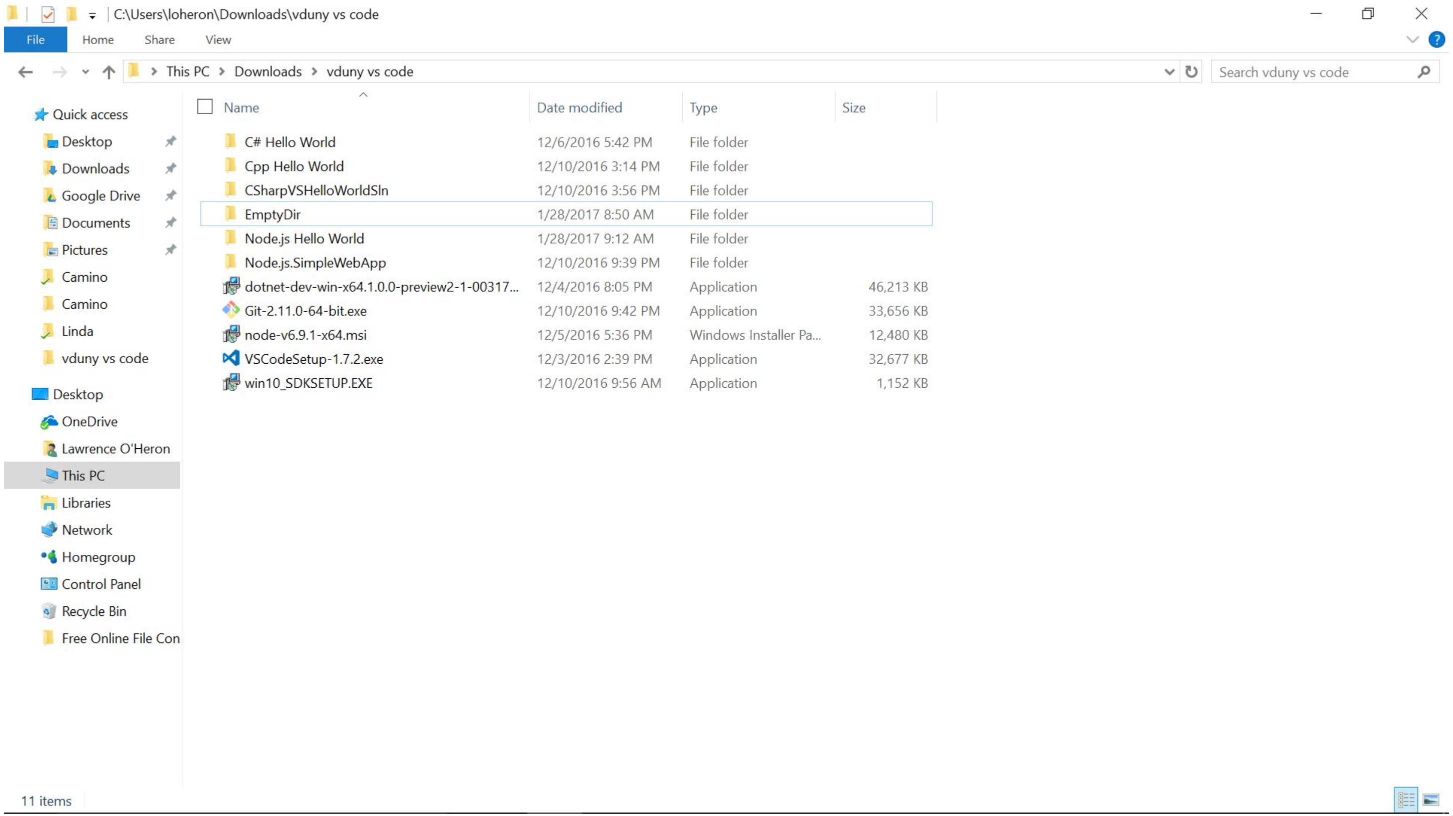
In project dir, type ‘code .’



```
 Volume Serial Number is CA6B-FC4E

 Directory of c:\Users\loheron\Downloads\vduny vs code

12/10/2016  09:42 PM    <DIR>          .
12/10/2016  09:42 PM    <DIR>          ..
12/06/2016  05:42 PM    <DIR>          C# Hello World
12/10/2016  03:14 PM    <DIR>          Cpp Hello World
12/10/2016  03:56 PM    <DIR>          CSharpVSHelloWorldSln
12/04/2016  08:05 PM        47,321,512 dotnet-dev-win-x64.1.0.0-preview2-1-00317
7.exe
12/10/2016  09:42 PM        34,463,528 Git-2.11.0-64-bit.exe
12/05/2016  05:36 PM        12,779,520 node-v6.9.1-x64.msi
12/10/2016  09:58 PM    <DIR>          Node.js Hello World
12/10/2016  09:39 PM    <DIR>          Node.js.SimpleWebApp
12/03/2016  02:39 PM        33,460,672 VSCodeSetup-1.7.2.exe
12/10/2016  09:56 AM         1,179,552 win10_SDKSETUP.EXE
               5 File(s)    129,204,784 bytes
               7 Dir(s)  24,206,946,304 bytes free

c:\Users\loheron\Downloads\vduny vs code>cd C# Hello World

c:\Users\loheron\Downloads\vduny vs code\C# Hello World>code .

c:\Users\loheron\Downloads\vduny vs code\C# Hello World>
```

Opening from Windows Explorer:

R/C on folder. Select ‘Open with Code’.



□ Name	Date modified	Type	Size
C# Hello World	12/6/2016 5:42 PM	File folder	
Cpp Hello World	12/10/2016 3:14 PM	File folder	
CSharpVSHelloWorldSln	12/10/2016 3:56 PM	File folder	
EmptyDir	1/28/2017 8:50 AM	File folder	
Node.js Hello World	1/28/2017 9:12 AM	File folder	
Node.js.SimpleWebApp	12/10/2016 9:39 PM	File folder	
dotnet-dev-win-x64.1.0.0-preview2-1-00317...	12/4/2016 8:05 PM	Application	46,213 KB
Git-2.11.0-64-bit.exe	12/10/2016 9:42 PM	Application	33,656 KB
node-v6.9.1-x64.msi	12/5/2016 5:36 PM	Windows Installer Pa...	12,480 KB
VSCodeSetup-1.7.2.exe	12/3/2016 2:39 PM	Application	32,677 KB
win10_SDKSETUP.EXE	12/10/2016 9:56 AM	Application	1,152 KB

Installed Components

Small download by design with the minimum number of components shared across most development workflows.

35 MB install file & 150 MB directory in Program Files (x86)

VS 12 uses .iso file & takes 3 GB in Program Files (x86)

Base install has JavaScript/TypeScript language and Node.js debugger.

- **Has basic functionality (editor, file management, window management).**

Unlike large, monolithic development tools (IDEs), (SURPRISE!) scenarios aren't completely supported out of the box; there isn't a File > New Project dialog with pre-installed project templates.

What is Node.js

Node.js:

- **platform for building fast and scalable server applications.**
- **runtime for Javascript.**

NPM: Package Manager for Node.js modules.

VS Code supports JavaScript and TypeScript languages out-of-the-box.

Node.js runtime needs installation to run/debug Node.js.

Install node.js: https://nodejs.org/en/download/ . Restart VSC.

Open the terminal.

Type ‘node --help’ to verify that node.js installed correctly.

Type ‘npm --help’ to verify that Express Generator is correctly installed.

Example 1: Node.js “Hello World”

Create folder with File Explorer

Open VSC by R/C on new folder

Create file in VSC with File -> New File

Type code.

Terminal session: node app.js

Eslint

Use Extension icon to install.

Open Hello World Javascript folder in VSC.

Will be prompted to run command: “npm install -g eslint”.

Close and Open folder.

The console statements trigger error msgs.

‘eslint --init’ at any time to reconfigure. Answer prompts. /* App must build */

eslint *.js

Debugging

Just like Visual Studio Enterprise

Express Framework

Express is a very popular application framework for building and running Node.js applications.

You can scaffold (create) a new Express application using the Express Generator tool.

The Express Generator is shipped as an NPM module and installed by using the NPM command line tool `npm`.

```
Ctrl - `    // open a terminal

npm install -g express-generator
```

Node.js Web App

VS Code File Explorer: open a folder.

Terminal > express Node.js.SimpleWebApp

Open the folder in Folder Explorer > Go to the new folder Node.js.SimpleWebApp to see the files

Close folder.

Open folder SimpleWebApp

Terminal > npm install   // installs the apps dependencies

Example 2: Node.js Web App

Terminal > npm start

Chrome browser: http://localhost:3000

GIT Support

Install at least ver 2.2. /* https://desktop.github.com/ */

Restart VS Code.

Open a folder and initialize the repository. (Click GIT button.) … Commit.

Edit some code. Save file. Git window shows changes. Commit.

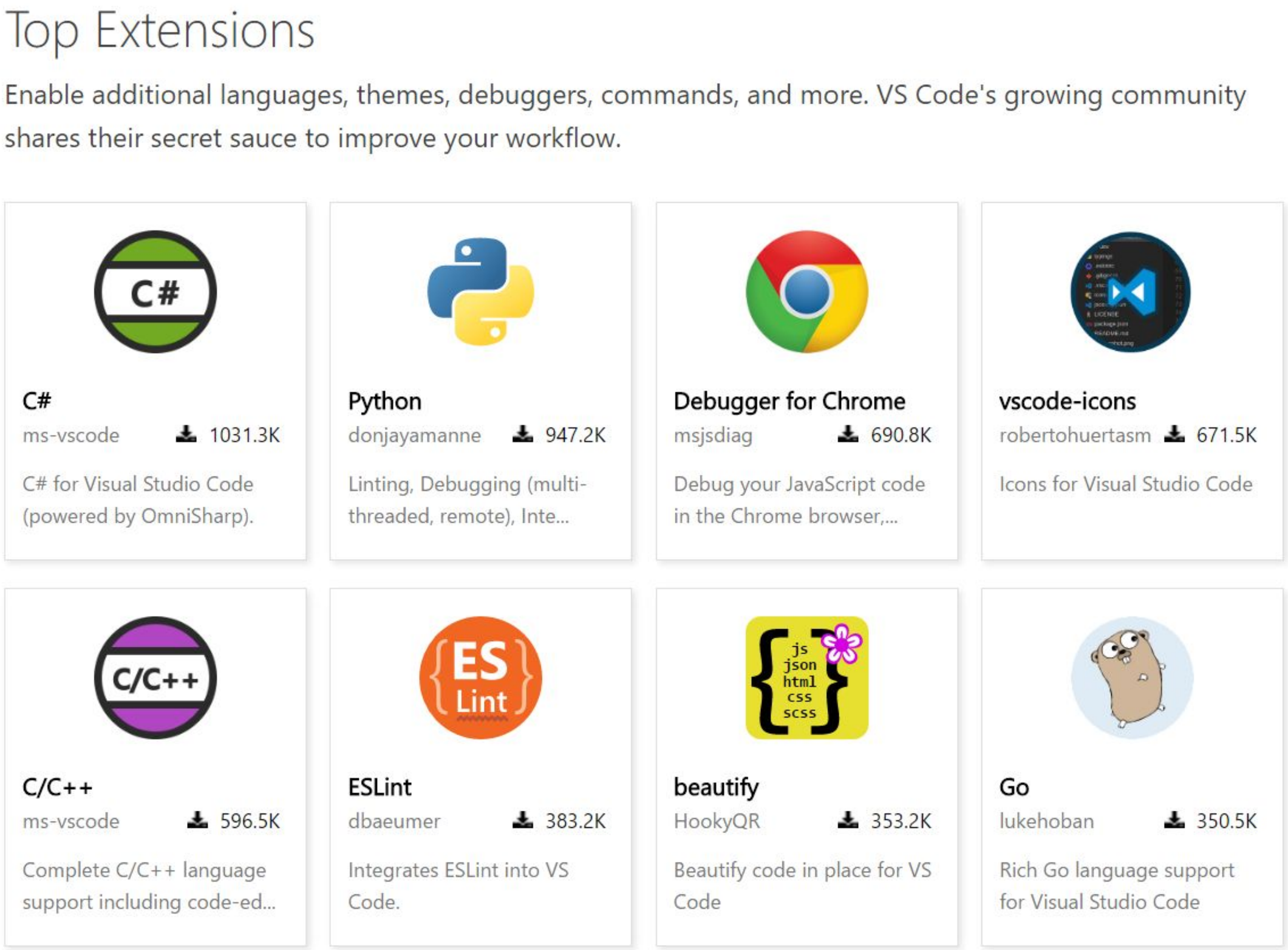
GIT Support

Edit code again. Save file. Select file under ‘CHANGES’. ‘Diff’ window.

File Name -> Rotating Arrow -> L/C -> Clean the file.

Undo last commit under GIT dropdown list … Select a file … Clean.

Repository is in the dir of the folder. To enable push/pull, need a remote hub.



Top Extensions

Enable additional languages, themes, debuggers, commands, and more. VS Code's growing community shares their secret sauce to improve your workflow.

C#
ms-vscode 1031.3K
C# for Visual Studio Code (powered by OmniSharp).

Python
donjayamanne 947.2K
Linting, Debugging (multi-threaded, remote), Inte...

Debugger for Chrome
msjsdiag 690.8K
Debug your JavaScript code in the Chrome browser,...

vscode-icons
robertohuertasm 671.5K
Icons for Visual Studio Code

C/C++
ms-vscode 596.5K
Complete C/C++ language support including code-ed...

ESLint
dbaeumer 383.2K
Integrates ESLint into VS Code.

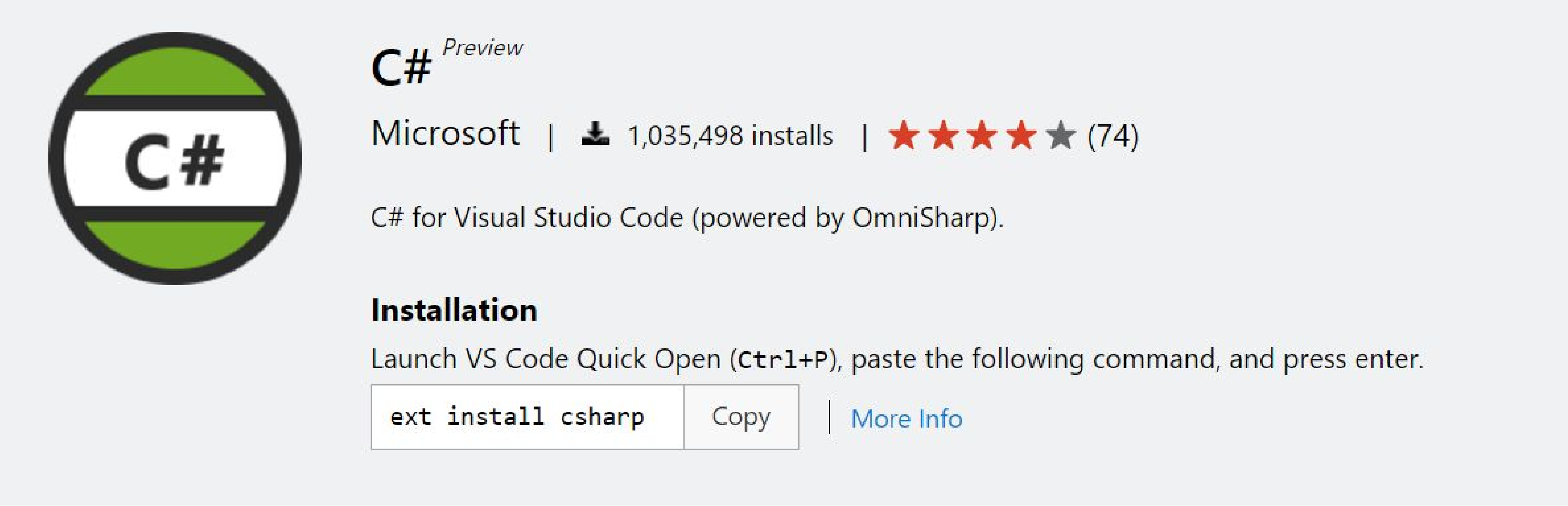
beautify
HookyQR 353.2K
Beautify code in place for VS Code

Go
lukehoban 350.5K
Rich Go language support for Visual Studio Code

C# - Install The Extension

Inside Visual Studio Code, ctrl - P (quick open). Enter command →

ext install csharp



C# *Preview*

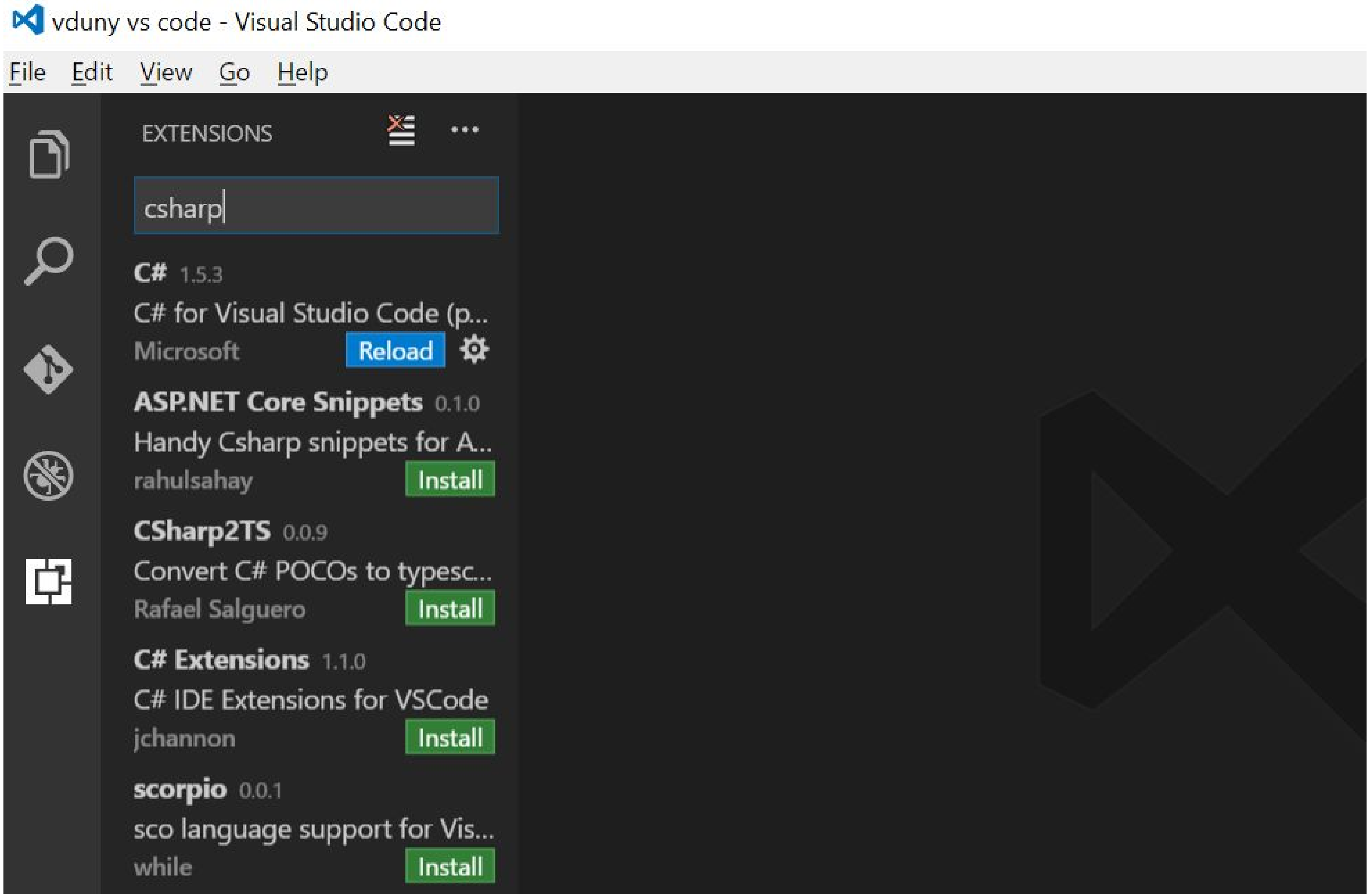
Microsoft | 1,035,498 installs | ★★★★☆ (74)

C# for Visual Studio Code (powered by OmniSharp).

Installation

Launch VS Code Quick Open (`Ctrl+P`), paste the following command, and press enter.

```
ext install csharp
```



csharp

C# 1.5.3
C# for Visual Studio Code (p...
Microsoft

ASP.NET Core Snippets 0.1.0
Handy Csharp snippets for A...
rahulsahay

CSharp2TS 0.0.9
Convert C# POCOs to typesc...
Rafael Salguero

C# Extensions 1.1.0
C# IDE Extensions for VSCode
jchannon

scorpio 0.0.1
sco language support for Vis...
while

Example 3: C# “Hello World” - Create and Run

Create a folder in Windows Explorer

VSC -> File -> Open Folder

Ctrl - `   // open a terminal session

dotnet new → creates *.cs file and a *.json file. /* basic ‘Hello World’ prgm */

dotnet restore → fixes dependencies /* status msg at top advises restore */

dotnet run

Example 3: C# “Hello World” - Debugging

If you don’t restore/fix dependencies, go to Debug Mode -> Look for ‘No Config’ or ‘Add Config’ in dropdown box.

Launch.json file opens. Requires editing.

- **Replace ‘<target-framework>/<project-name.dll>’ with ‘netcoreapp1.1/C# Hello World.dll’**

Create/configure Task Runner when prompted. No editing needed.

Set breakpoint. /* On my laptop, Fn - F10 */

Start debugging.

Example 4: C++ App

<ctrl + P> Quick Open Window

    ext install cpptools

Windows File Explorer → create directory for project

Visual Studio Code File Explorer → Open folder

New file → enter code → Save as *.cpp file.

Configure for build and runtime. Click the Debug button.

C++ - Configure the Environment

To enable code completion and navigation, you will need to generate a c_cpp_properties.json file:

- **Hover over any green squiggle in a source file (e.g. a #include statement).**
- **Click the lightbulb that appears underneath the mouse cursor.**
- **Click Add include path to settings.**
- **Save file**
- **On my laptop: C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\include**

C++ App

Select a compiler

Cygwin provides a full layer of POSIX compatibility to implement UNIX/Linux features,such as forking, on top of Windows. This slows things down, but ensures that your UNIX programs will work on Windows. Due to this compatibility layer, you must include a cygwin .dll file with your apps.

MinGW provides a suite of GNU tools that run on Windows natively. This is faster than Cygwin and requires no extra DLL, but does not have the same amount of feature support. This means that not all your UNIX applications can be used on Windows.

Visual Studio Compiler

- **C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\bin**

Themes

File → Preferences → Color Theme   (Use cursor keys to preview the themes.)

Adding Themes From Extension Marketplace

Extensions Icon → Enter ‘themes’ in search bar.

Already downloaded the PreDawn themes. Must restart VS Code.

Settings

File - Preferences - Settings

Scroll ‘Search Settings’

In ‘User Settings’, set ‘editor.fontSize’ to 26.

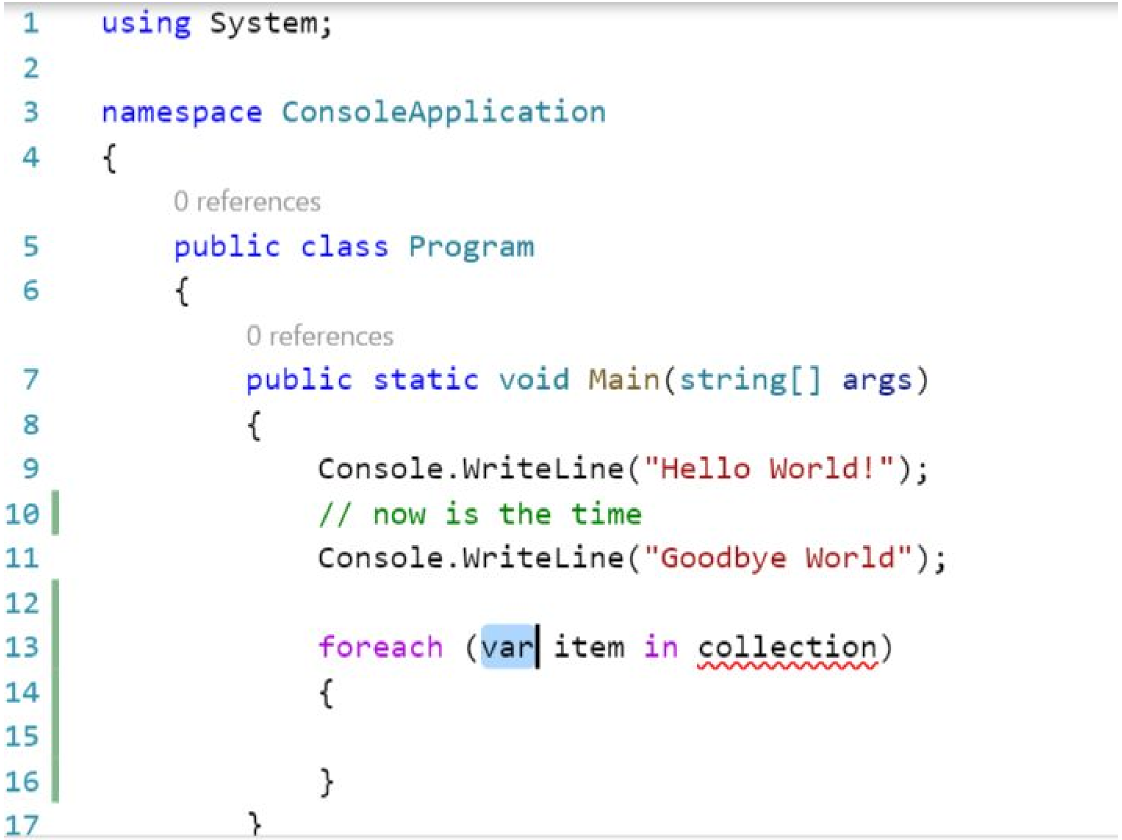
Difference between ‘User Settings’ and ‘Workspace Settings’

Takes effect on file save.

Restore ‘fontSize’ to 14.

Code Snippets

File → Preferences → User Snippets



```
using System;

namespace ConsoleApplication
{
    0 references
    public class Program
    {
        0 references
        public static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
            // now is the time
            Console.WriteLine("Goodbye World");

            foreach (var item in collection)
            {

            }
        }
```

Intellisense

VSC Intellisense goes way beyond most light-weight code editors.

In a C# project, type ‘Console’ and let Intellisense display ...

Settings Synchronization

Synchronize Settings, Snippets, launch, keybindings, workspaces and extensions across Multiple Machines using Github Gist.

Installation:

Launch VS Code Quick Open (Ctrl+P)

ext install code-settings-sync

ASP.Net Core App

To setup your development machine

- **download and install .NET Core and Visual Studio Code with the C# extension**
- **Node.js and npm is also required**
- **using yo aspnet to generate the Web Application Basic template (scaffolding)**
- **‘Bower’ is the client side package manager**

ASP.Net Core App

```
Console> npm install -g yo generator-aspnet@0.2.6 bower
```

```
Cosole> yo aspnet
```

- **Select Web Application Basic [without Membership and Authorization] and tap Enter**
- **Select Bootstrap (3.3.6) as the UI framework and tap Enter**
- **Use "MyFirstApp" for the app name and tap Enter**

When the generator completes scaffolding the files, it will instruct you to restore, build, and run the application.

Asp.Net Core App

The project is now created.

```
Console>      cd "MyFirstApp"
Console>      dotnet restore
Console>      dotnet build (optional, build will also happen with it's run)
Console>      dotnet run
```

Files are found in your user directory, e.g. ‘Larry OHeron’

Open browser: ‘localhost:5000’

Summary

VSC is a powerful and flexible light-weight code editor.

Suitable for projects that

- **do not require heavy user interface development**
- **find VS too muscular**