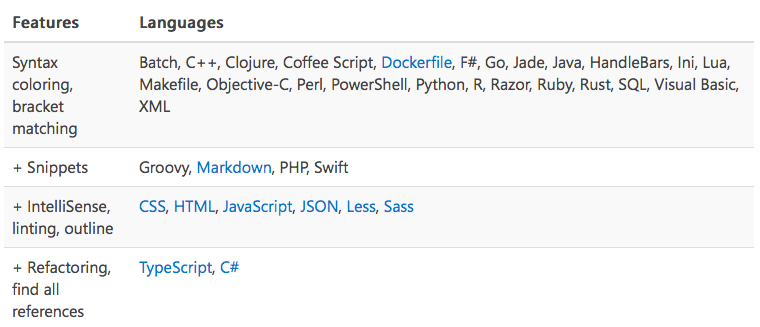
# Возможности редактора

Как я упоминал в предыдущей статье, Visual Studio Code позволяет работать с отельными файлами или каталогами. То есть, чтобы открыть проект, необходимы открыть каталог, где расположен проект. Если Вы хотите создать новый проект, то просто создайте новый каталог и откройте его в Code.

В зависимости от расширения файлов, Visual Studio Code определяет какой язык программирования используется и подсвечивает разметку для более чем 30-ти поддерживаемых языков. Полный список поддерживаемых языков Вы можете посмотреть по [данной ссылке](https://code.visualstudio.com/Docs/languages/overview).

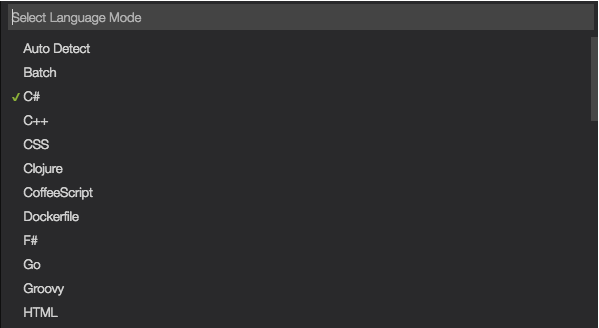


Данная таблица регулярно обновляется, но уже сейчас есть поддержка большого количества возможностей для таких языков как TypeScript, JavaScript, CSS, HTML, C# и др. При этом, если вы используете Node.js для запуска JavaScript или TypeScript приложений, то Visual Studio Code поддерживает отладчик. Также отладчик поддерживается для приложений, которые запускаются на Mono, это ASP.NET 5.0 и Unity3D приложения. Таким образом Вы можете разрабатывать современные Web приложения, имея доступ ко всем возможностям, начиная от поддержки IntelliSense системы и заканчивая отладчиком.

Используя строку состояния в Visual Studio Code вы можете видеть тот язык, который был выбран автоматически, базируясь на расширении активного файла:



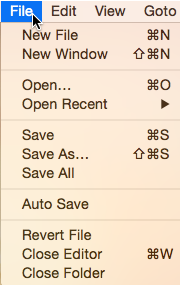
Если Code по каким-то причинам не определил язык правильно, то Вы можете вызвать **Select Language Mode** окно, щелкнув на текущем языке в строке состояния:



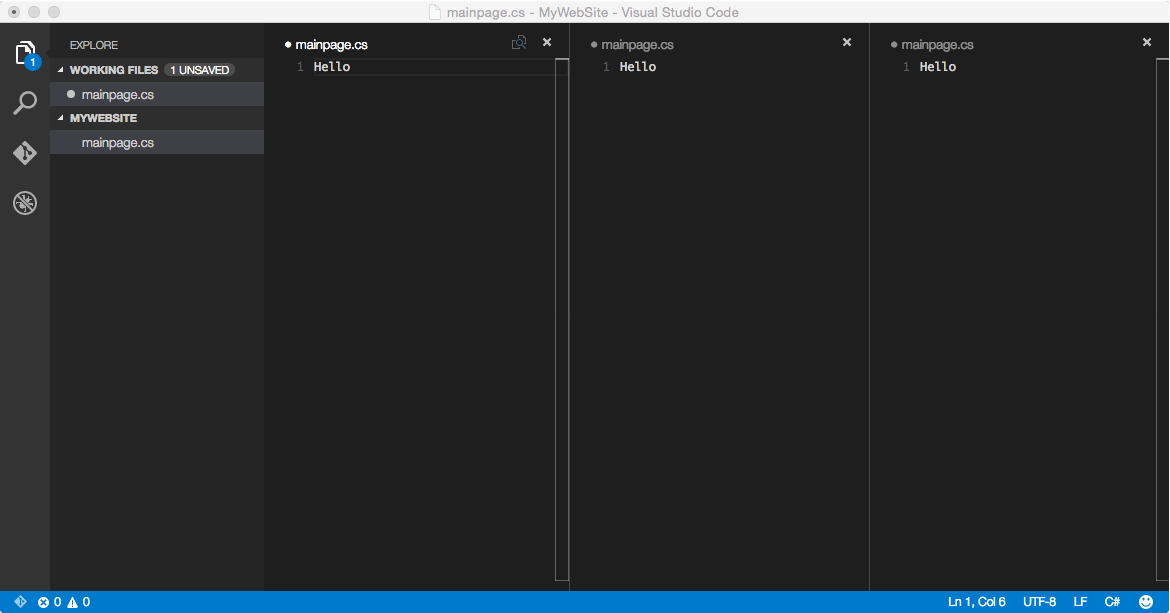
Мне очень нравится это окно, так как тут можно найти список всех поддерживаемых языков, даже если их забыли указать в документации.

Конечно, чтобы получить полную поддержку IntelliSense системы для JavaScript, TypeScript или C#, Visual Studio Code должен иметь информацию о проекте. Поэтому Code может автоматически вычитывать информацию из различных файлов и позволяет Вам выбирать активный проект. Например, **package.json** используется для Node.js, **project.json** для ASP.NET, **jsconfig.json** и **tsconfig.json** – для JavaScript и TypeScript проектов. Кроме того, Visual Studio Code способен понимать **.sln** файлы и **.csproj** файлы. В следующих статьях мы обсудим все эти файлы и проекты более детально.

Когда Вы работаете с файлами, то следует обратить особое внимание, что все изменения не сохраняются автоматически. Чтобы активировать автоматическое сохранение, нужно воспользоваться пунктом меню **File** и выбрать **Auto Save**:



Если Auto Save возможность отключена, то сможете увидеть количество не сохраненных файлов в браузере файлов (Explorer), а также на элементе Viewbar, позволяющем переключаться между различными отображениями:

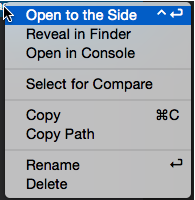


Visual Studio Code содержит следующие основные элементы интерфейса: Viewbar, Sidebar, строку состояния и окно редактирования кода.

Элемент Viewbar (панель с левой стороны редактора) позволяет переключаться между различными отображениями, а также может содержать дополнительную информацию, как количество не сохраненных файлов.

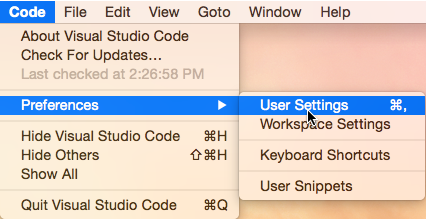
Элемент Sidebar поддерживает несколько отображения, таких как Explorer, Search, Git и Debug. Наиболее часто используемое отображение, это Explorer, которое позволяет переключаться между файлами в проекте и просматривать структуру основного каталога и всех подкаталогов, а также создавать новые файлы или удалять существующие. Search позволяет выполнять поиска по заданной строке внутри файлов каталога. Git окно предоставляет ряд команд, связанных с системой контроля версий Git. Наконец, Debug окно предоставляет информацию, связанную с отладкой приложения.

Если говорить об окне редактора, то Visual Studio Code поддерживает до трех различных окон редактора. Чтобы отобразить второе окно, Вы можете воспользоваться контекстным меню и выбрать команду **Open to the Side**.



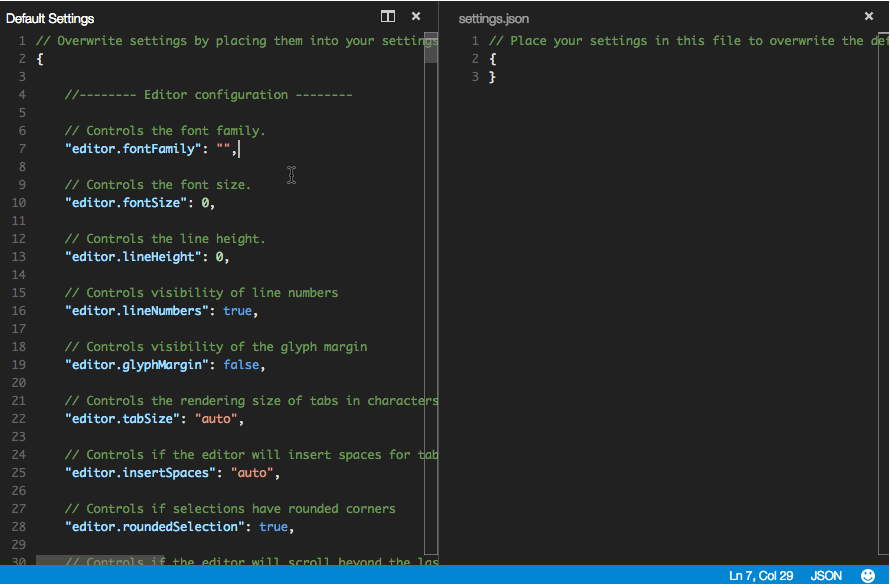
Используя контекстное меню, Вы можете найти еще две полезных команды, это **Reveal in Finder** и **Open in Console**. Первая команда позволяет открыть каталог с выбранным файлом в Finder , а вторая – открыть терминал в контексте текущего каталога.

Visual Studio Code поддерживает множество полезных настроек. Для модификации существующих настроек используйте **User Settings** или **Workspace Settings** элементы меню:



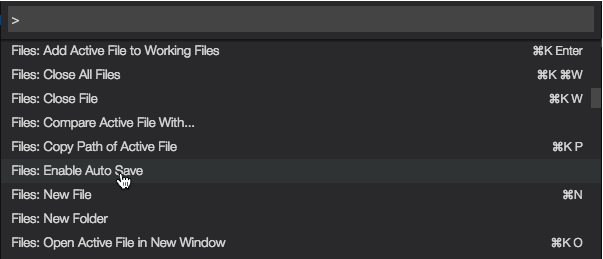
Первый пункт позволяет применить пользовательские настройки для всех проектов в Code, а второй – только для текущего рабочего пространства.

Вызвав один из этих пунктов, Visual Studio Code создаст для Вас новый файл, где Вы сможете перегрузить любые существующие настройки, а чтобы облегчить Вам жизнь, в другом окне Code откроет уже существующие настройки, которые Вы можете использовать в качестве примера:



Среди настроек Вы можете найти параметры редактора, информацию о расширениях файлов, которые нужно скрыть из Explorer, настройки IntelliSense и другое.

Обычно я люблю использовать мышь для доступа к любой команде, располагающейся в меню. Но Visual Studio Code позволяет выполнять все команды без использования мыши. Во-первых, тут можно использовать горячие клавиши, а во-вторых, Code поддерживает специальное окно Command Palette, где Вы сможете посмотреть и выполнить любую из доступных команд в Code. Для вызова Command Palette используете комбинацию клавиш ⇧⌘P:



Используя Command Palette Вы сможете увидеть и комбинации клавиш, которые назначены той или другой команде. Если Вы хотите изменить существующие комбинации клавиш, то Вы можете использовать **Open Keyboard Shortcuts** команду.