



Visual Studio Code

Wprowadzenie

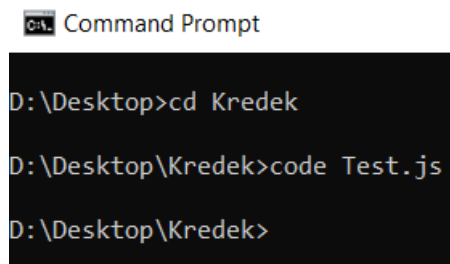
KRÓTKI WSTĘP

Visual Studio Code jest narzędziem które pod względem złożoności, funkcjonalności, znajduje się pomiędzy zwykłymi edytorami tekstu takimi jak Notatnik, czy Notepad++ a w pełni funkcjonalnymi zintegrowanymi środowiskami programistycznymi np. MS Visual Studio. Narzędzie to jest wygodne w sytuacjach gdy potrzebujemy debuggera, systemu kontroli wersji – git oraz narzędzia, które pomoże nam usprawnić pracę, podpowiadając i uzupełniając pisany kod – IntelliSense (IntelliSense znajduje się również w MS Visual Studio więc funkcjonalność pod względem podpowiedzi w kodzie jest ta sama). Wszystkie wymienione wyżej funkcjonalności są dostarczane przez VS Code. Ponadto mechanizm rozszerzeń w edytorze pozwoli na dodatkową personalizację i rozszerzenie funkcji programu, mechanizm ten zostanie opisany w dalszej części wprowadzenia.

URUCHAMIANIE VISUAL STUDIO CODE

Visual Studio Code możemy otworzyć używając CMD/Terminala. Aby to zrobić otwieramy CMD/Terminal, wybieramy potrzebną nam ścieżkę do folderu w, którym znajduje się nasz plik, który chcemy otworzyć. Wpisujemy komendę:

- 'code naszPlik.rozszerzenie '



```
cmd Command Prompt
D:\Desktop>cd Kredek
D:\Desktop\Kredek>code Test.js
D:\Desktop\Kredek>
```

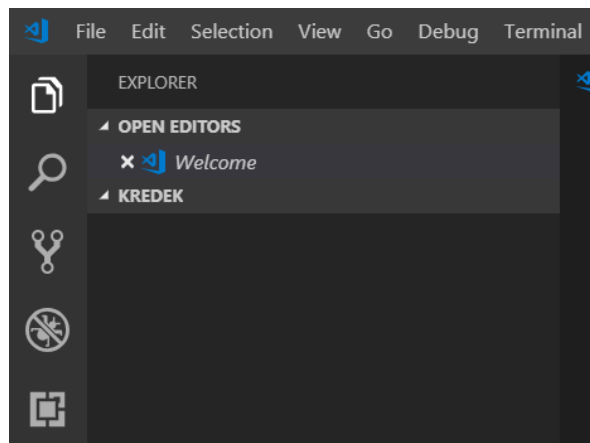
Jeśli w CMD/terminalu wyświetla się informacja, że instrukcja 'code' nie została znaleziona, trzeba zrobić jeszcze jedną rzecz, tzn. dodać komendę 'code' do ścieżki PATH. Aby to zrobić otwieramy edytor, wyszukujemy wszystkich poleceń skrótem 'ctrl + shift + p' w wyszukiwarce wpisujemy "**Shell Command: Install 'code' command in PATH**" i wybieramy tę opcję.

Polecenie 'code' w CMD daje również możliwość otwierania edytora w wybranym folderze. Żeby otworzyć edytor w danych folderze należy przejść poleceniem 'cd' do folderu który nas interesuje, a następnie wykonać polecenie:

- 'code .'

```
Command Prompt

D:\Desktop>cd Kredek
D:\Desktop\Kredek>code .
D:\Desktop\Kredek>
```

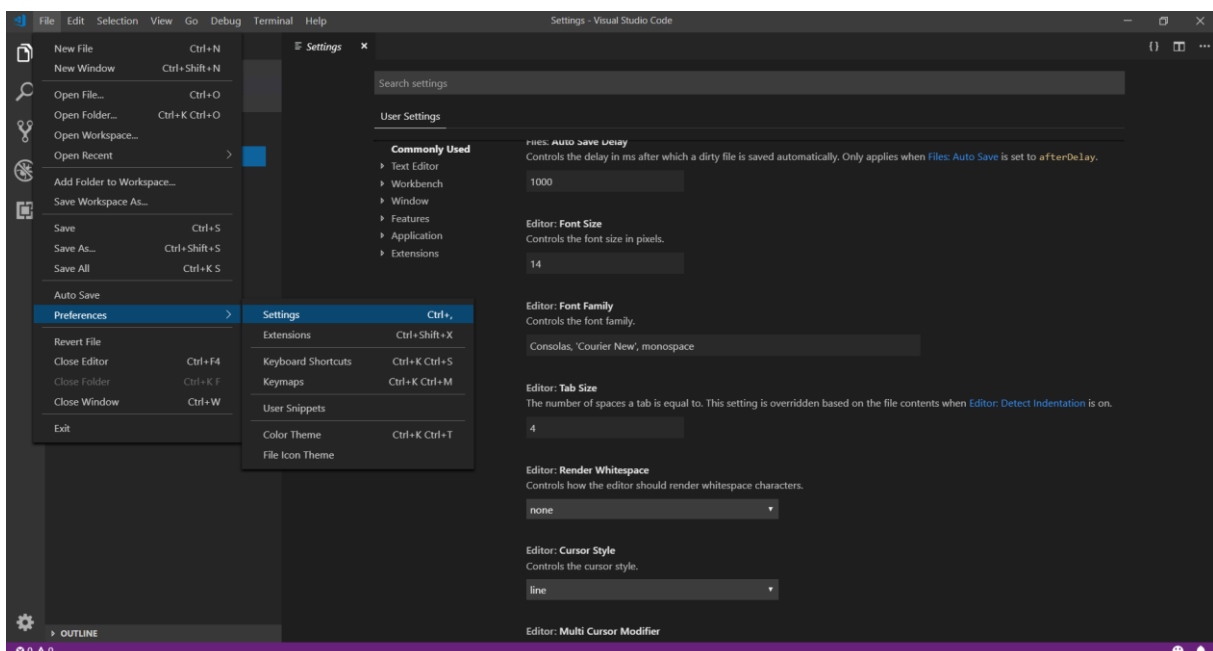


Teraz kiedy możemy nasz program otwierać jak prawdziwi programiści możemy zająć się środkiem edytora.

PERSONALIZACJA EDYTORA

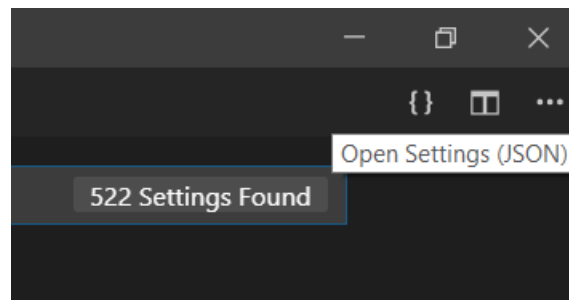
Osobiście pierwsze co robię po zaistalowaniu nowego edytora/IDE to szukanie ustawień pozwalających na spersonalizowanie edytora. VS Code daje tę możliwość – „File” -> „Preferences Settings” (na macOS – „Code” -> „Preferences Settings”) lub używamy skrótu:

- ‘Ctrl + ,’

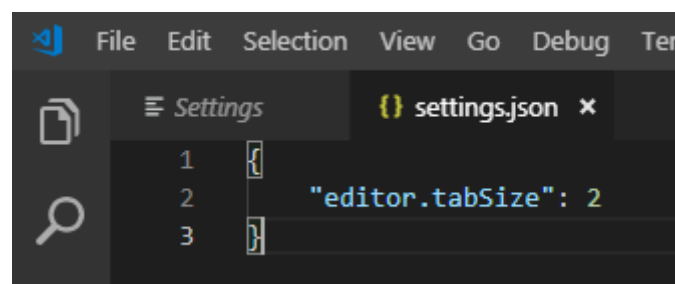


W tej zakładce znajdują się opcje ustawienia programu. Możemy edytować predefiniowane ustawienia edytora lub ustawienia dostarczone przez rozszerzenia w zakładce „Extensions”. Zalecam spędzić chwilę „bawiąc” się ustawieniami, pozwoli to na lepsze

zrozumienie edytora. Ponadto ustawieniami możemy manipulować z poziomu kodu, edytując specjalny plik .json który możemy otworzyć klikając na ikonkę „{ }”:



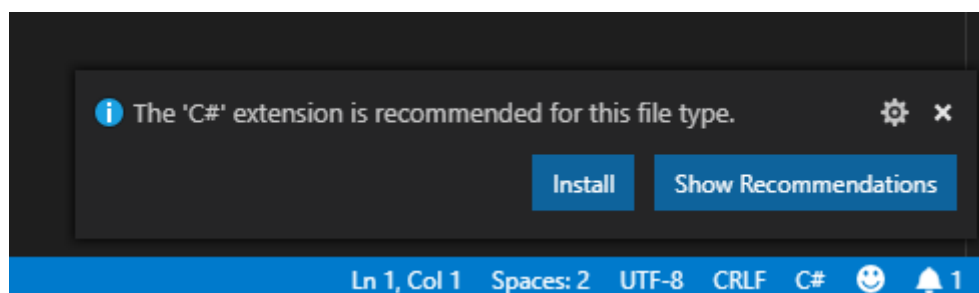
Dla przykładu zmieniłem wielkość domyślnego tabulatora z odległości 4 spacji do 2, jak widzimy w pliku 'settings.json' została umieszczona linijka kodu odpowiedzialna za tą zmianę:



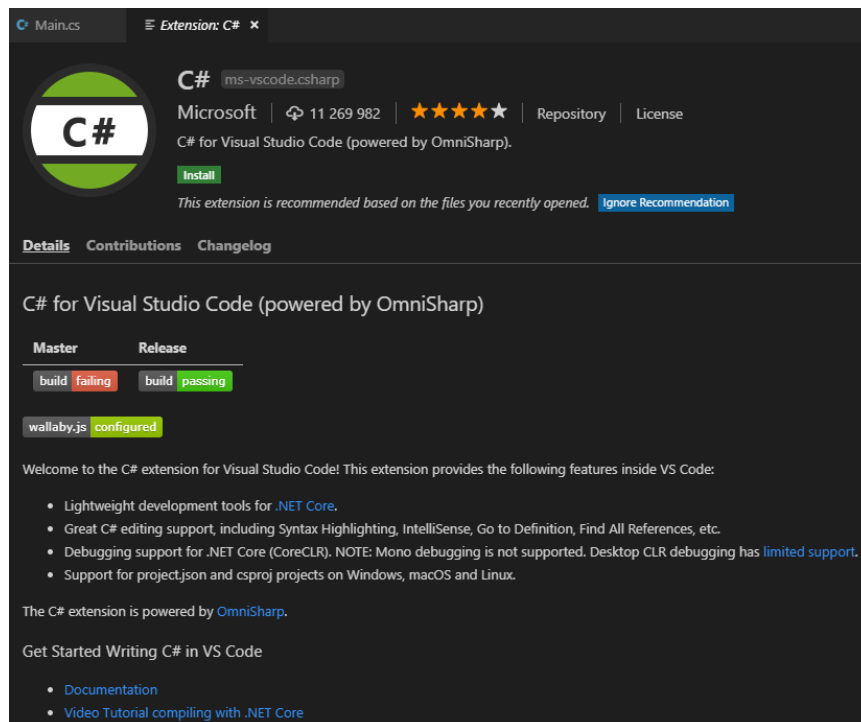
Jak widać na pierwszym zrzucie ekranu – w tym podrozdziale – możemy również manipulować skrótami klawiszowymi, proces zmiany skrótów jest podobny do opisanego procesu zmiany ustawień.

ROZSZERZENIA

Następną znaczącą opcją w VS Code jest opcja dodawania rozszerzeń. Jako że Visual Studio Code jest projektem open – source, programiści mogą pisać i wdrażać swoje własne usprawnienia w postaci rozszerzeń. Jest to potężny mechanizm pozwalający „wycisnąć” z VS code naprawdę niemało. Pierwszy raz zetknąłem się z tym mechanizmem kiedy próbowałem utworzyć nowy plik .cs żeby napisać prostą aplikację konsolową w C#. VS Code „podpowiedział” mi, że warto by zainstalować rozszerzenie „C#”:

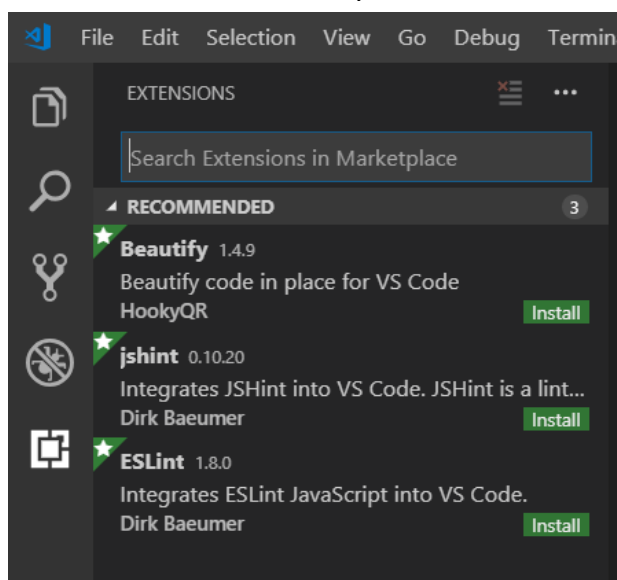


Zanim zdecydowałem się na instalację sprawdziłem opis tego rozszerzenia:



Jak widzimy, autorem rozszerzenia jest Microsoft, poniżej wyszczególniono autora i krótkiego opisu znajduje się zielony przycisk „Install” – to on jest odpowiedzialny za dodanie rozszerzenia do naszych rozszerzeń. Poniżej znajduje się opis rozszerzenia, zwykle tam znajdziemy informacje dotyczące działania, i ustawiania opcji rozszerzenia.

Rozszerzeń możemy szukać również klikając na ikonę w lewym panelu VS Code:



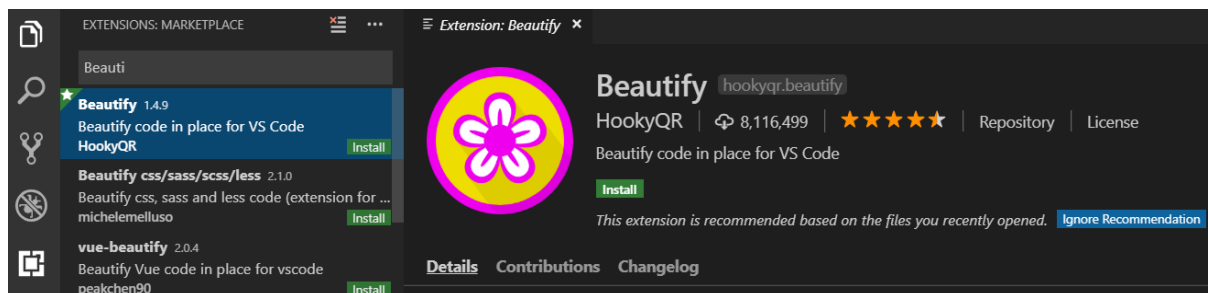
Ikonka znajduje się na samym dole paska, po kliknięciu w nią otworzy się panel z polem do wyszukiwania rozszerzeń, oraz kilka rekomendowanych rozszerzeń. Tutaj możemy wyszukiwać tych rozszerzeń które ktoś nam polecił lub te które sami odkryliśmy. Aby zainstalować rozszerzenie wystarczy kliknąć zielony przycisk „Install”.

Równie dobrze, w celu otworzenia panelu do wyszukiwania i instalowania rozszerzeń, możemy użyć skrótu klawiszowego:

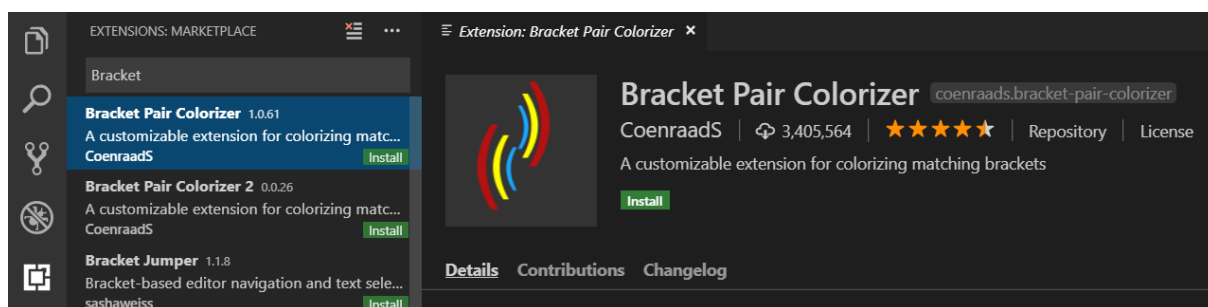
- ‘ Ctrl + Shift + X ’

PRZYKŁADOWE ROZSZERZENIA

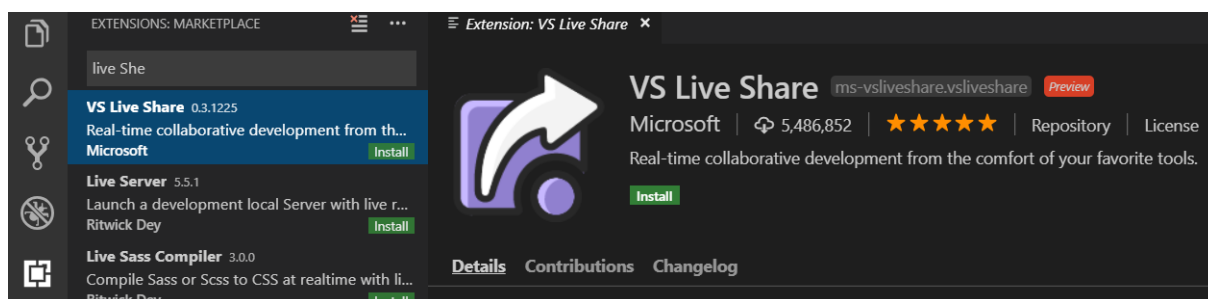
Przygotowując to wprowadzenie spotkałem się z wieloma rekomendacjami rozszerzeń, ponieważ wielu programistów posiada własne preferencje. Wybrałem więc subiektywnie te ciekawsze rozszerzenia, które moim zdaniem dodają praktyczne opcje do Visual Studio Code. Opiszę więc funkcje i usprawnienia zapewniane przez wyróżnione rozszerzenia.



- **Beautify** – rozszerzenie pozwalające na autoformatowanie kodu. Polecam to rozszerzenie szczególnie osobom z niezerowym doświadczeniem w pisaniu kodu, ponieważ osoby, które po raz pierwszy programują mogą wyrobić w sobie złe nawyki, pisania brzydkiego składniowo kodu.



- **Bracket Pair Colorizer** – rozszerzenie kolorujące pary nawiasów „wąsatych ☺”. W taki sposób, że zagnieżdżone nawiasy są oznaczone innymi kolorami, pozwala to na lepsze rozpoznanie w kodzie.



- **VS Live Share** – rozszerzenie pozwalające na ‘live coding’/‘pair programming’, aby użytkownicy mogli razem wspólnie pracować nad projektem, każdy z osobna musi się zalogować – poprzez konto MS lub Github’a, a jeden z użytkowników musi udostępnić innym link pozwalający na połączenie z jego projektem.

Wszystkie opisane rozszerzenia można spersonalizować we własnym zakresie, należy przeczytać ze zrozumieniem zakładkę 'Details', w niej znajdują się instrukcje jak ustawić/zmienić niektóre funkcje w rozszerzeniach, więc do przeczytania tych treści odsyłam zainteresowanych.

PRZYDATNE LINKI

W tak krótkim wprowadzeniu nie da się zawrzeć nawet znacznej części udogodnień jakie daje Visual Studio Code. Praca w nowym środowisku wymaga – pracy w nim ☺, dlatego przedstawione wyżej funkcje VS Code mają na celu skierować czytającego na odpowiednie tory, które pomogą lepiej zrozumieć ten edytor. Czytelnik musi sam na własną rękę i potrzeby, dostosować edytor. Dlatego aby ułatwić znalezienie materiałów potrzebnych do tego odsyłam do następujących stron:

- <https://code.visualstudio.com/docs> - oficjalna dokumentacja od twórców - posiada informacje od genezy powstawania ikony programu, do informacji o trikach i radach.
 - <https://code.visualstudio.com/docs/editor/versioncontrol> - opisanie wbudowanego mechanizmu systemu kontroli wersji – Git oraz innych możliwości kontroli wersji.
 - <https://code.visualstudio.com/docs/editor/intellisense> - materiały szerzej opisujące podpowiadanie składni.
 - <https://code.visualstudio.com/docs/editor/debugging> - opisana opcja debuggowania w VS Code.
- <https://github.com/Microsoft/vscode> - tak jak nadmieniałem wcześniej Visual Studio Code jest projektem open-source dlatego wklejam link do githuba projektu.
- <https://www.youtube.com/watch?v=yJw0SyKO9IU> - film z którego m.in korzystałem podczas procesu pisania tego wprowadzenia.
- <https://www.youtube.com/watch?v=rO8-cgtkZSw> – ciekawa i śmieszna prezentacja rozszerzenia pozwalającego na pisanie jak w znanym programie „hacker typer” – obrazuje to potęgę tego że VS Code jest open-source. Każdy może napisać swoje własne rozszerzenie pozwalające rozwiązać problem piszącego je.

Wklejanie większej liczby linków uważam za zbędne ponieważ dokumentacja od Microsoftu jest naprawdę dobrze napisana, znajduje się tam wszystko. Słowem końcowym niech będzie sparafrazowane zdanie – „Żeby dobrze programować, trzeba programować” – „Żeby dobrze opanować Visual Studio Code, trzeba go używać”.