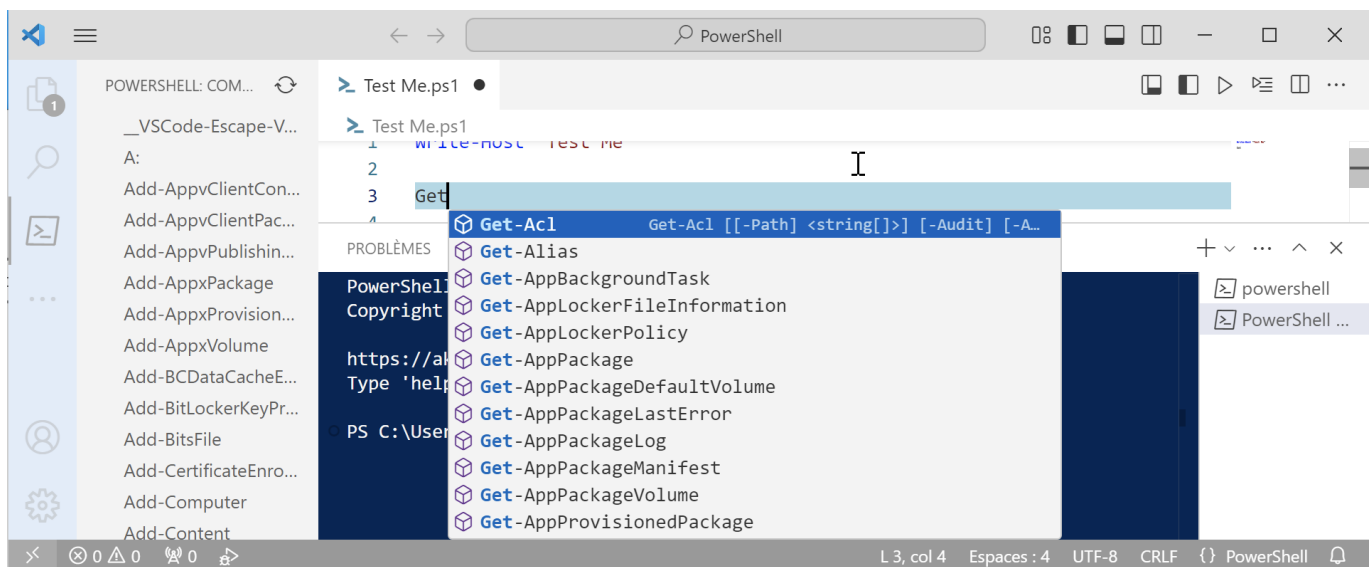


Configuration de Visual Studio Code pour PowerShell

Microsoft a abandonné PowerShell ISE (la dernière version de ISE n'est pas compatible avec PowerShell 6 et supérieur), une alternative est l'extension PowerShell de Microsoft de VS Code.

À la différence de l'éditeur PowerShell ISE, Visual Studio Code prend en charge à la fois Windows PowerShell et PowerShell Core. Bien que PowerShell ISE soit toujours utile et pratique dans certains cas (*éditer un script rapidement sur un serveur, par exemple*), VSCode est l'éditeur à préférer pour votre poste de travail.

Cette extension VSCode fournit des fonctionnalités telles que la coloration syntaxique, auto-suggestion, un mode ISE et un terminal PowerShell intégré. Exemple:

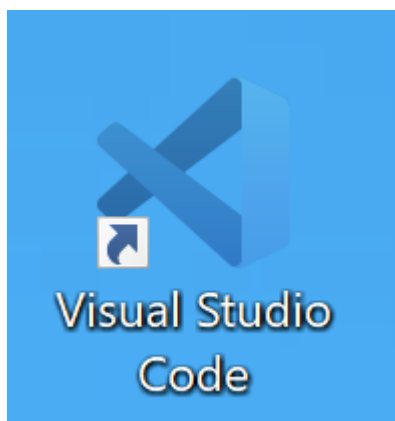


La fonction IntelliSense de VS Code est une fonction d'intelligence artificielle qui offre des suggestions de codage de PowerShell facilitant votre interaction avec ce langage.

Dans les explications ci-dessous, vous allez configurer et apprendre à utiliser VSCode pour écrire vos scripts PowerShell.

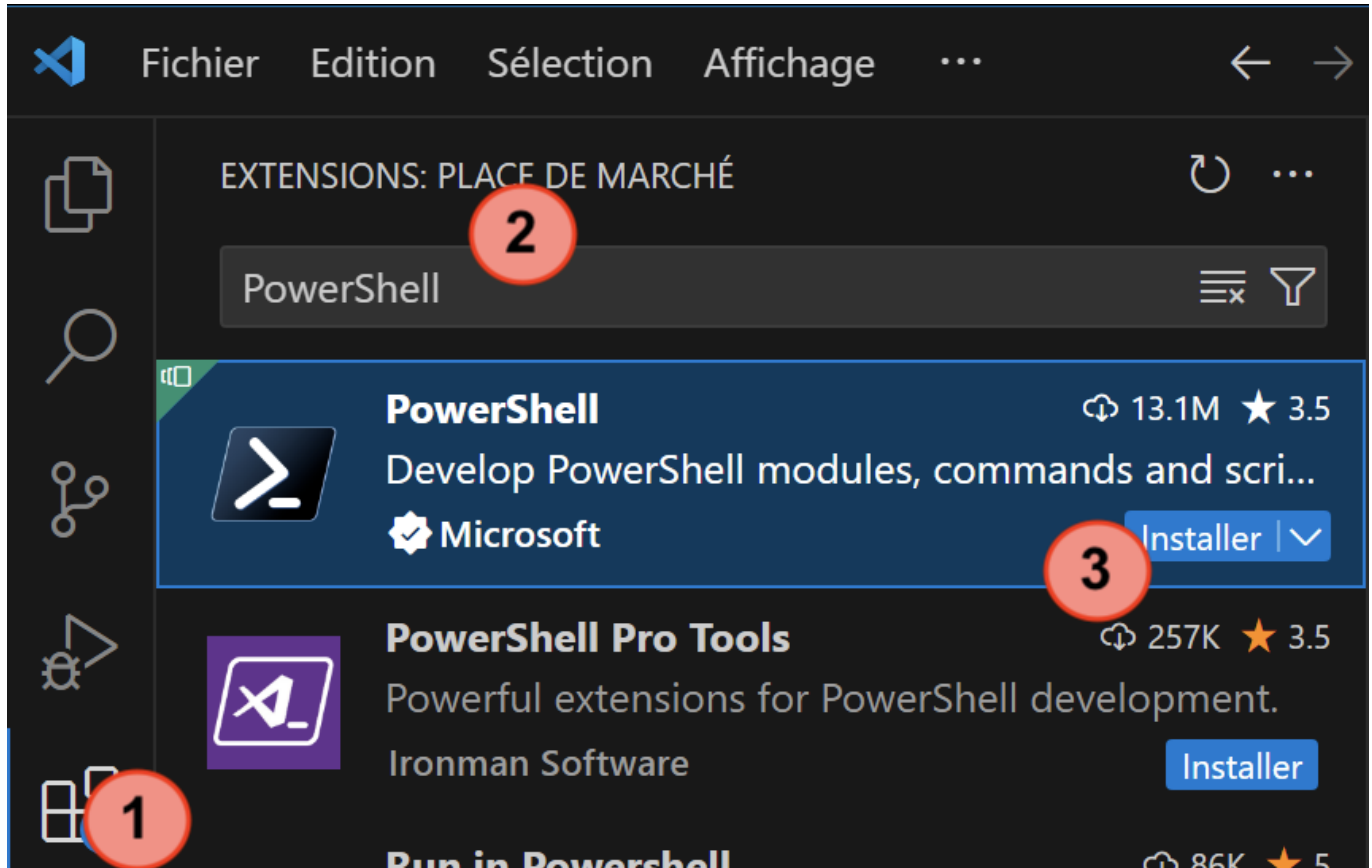
1. Installation de l'extension VSCode PowerShell

Prérequis: avoir installer [Visual Studio Code](#).



Par défaut, Visual Studio Code peut éditer des fichiers textes, mais il n'est pas prêt à vous accompagner pour éditer du code PowerShell. Il faut savoir que VSCode s'appuie sur un système d'extension : chaque extension va venir ajouter des fonctionnalités à Visual Studio Code.

Cliquez sur l'icône des extensions sur la gauche (1), recherchez "powershell" (2) et cliquez sur l'extension "PowerShell" proposée par Microsoft (3).



Cliquez sur le bouton "Installer" et patientez pendant quelques secondes. Visual Studio Code ajoute l'extension PowerShell à votre machine.

Remarque : vous pouvez également installer l'extension en saisissant la commande "ext install powershell" après avoir utilisé le raccourci CTRL + P.

Les extensions sont installées dans le profil de l'utilisateur. Voici le répertoire utilisé sur chaque système :

- Windows : %USERPROFILE%.vscode\extensions.
- macOS : ~/.vscode/extensions.
- Linux : ~/.vscode/extensions.

A partir du moment où l'extension est installé, VSCode est prêt à éditer du code PowerShell !

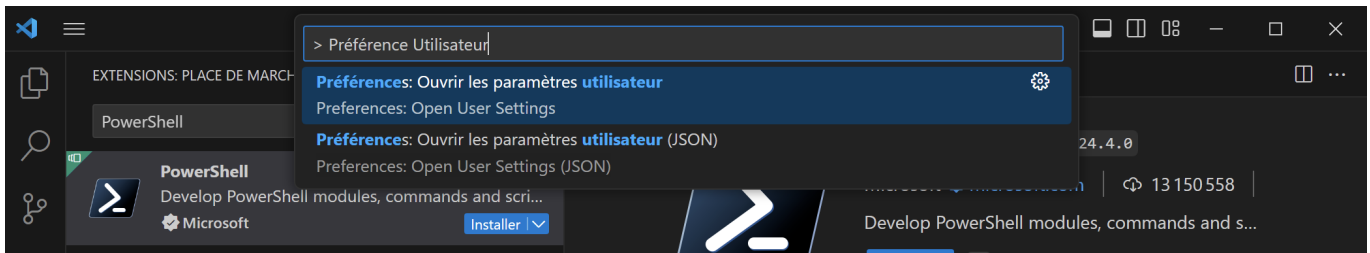
2. Utiliser PowerShell avec VSCode

A. Configurer VSCode pour PowerShell

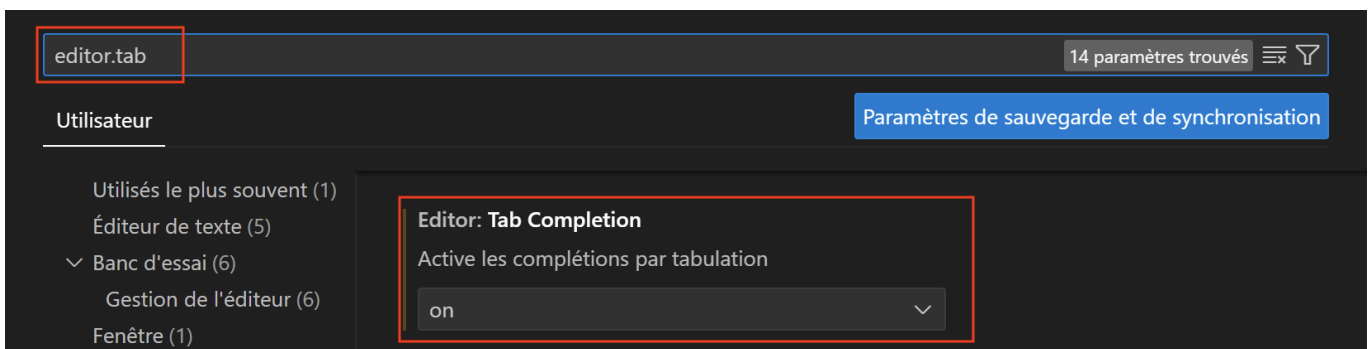
Avant de créer un script, nous allons modifier une option dans les paramètres de VSCode afin d'activer l'utilisation de la touche TAB pour l'autocomplétion (ce qui plaira d'autant plus à ceux habitués à PowerShell ISE).

Pour accéder à la zone de saisie des commandes, utilisez le raccourci "F1" ou "CTRL + SHIFT + P". C'est un bon moyen d'exécuter des actions dans VSCode sans parcourir les menus.

Ici, recherchez "Paramètres Utilisateur" afin de cliquer sur "Préférences: Ouvrir les paramètres utilisateur". Vos préférences sont personnelles et enregistrées dans votre session.



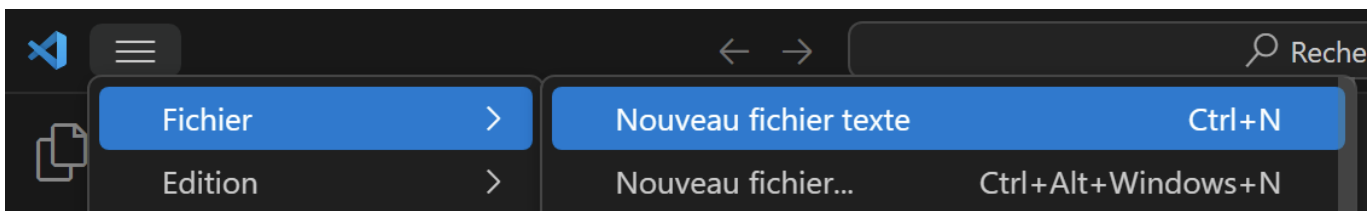
Recherchez le paramètre via le mot clé "editor.tab" et basculez sur "on" le paramètre nommé "Editor: Tab Completion". Il est à noter que VSCode bénéficie d'IntelliSense (comme PowerShell ISE) qui est une bonne aide quand on code.



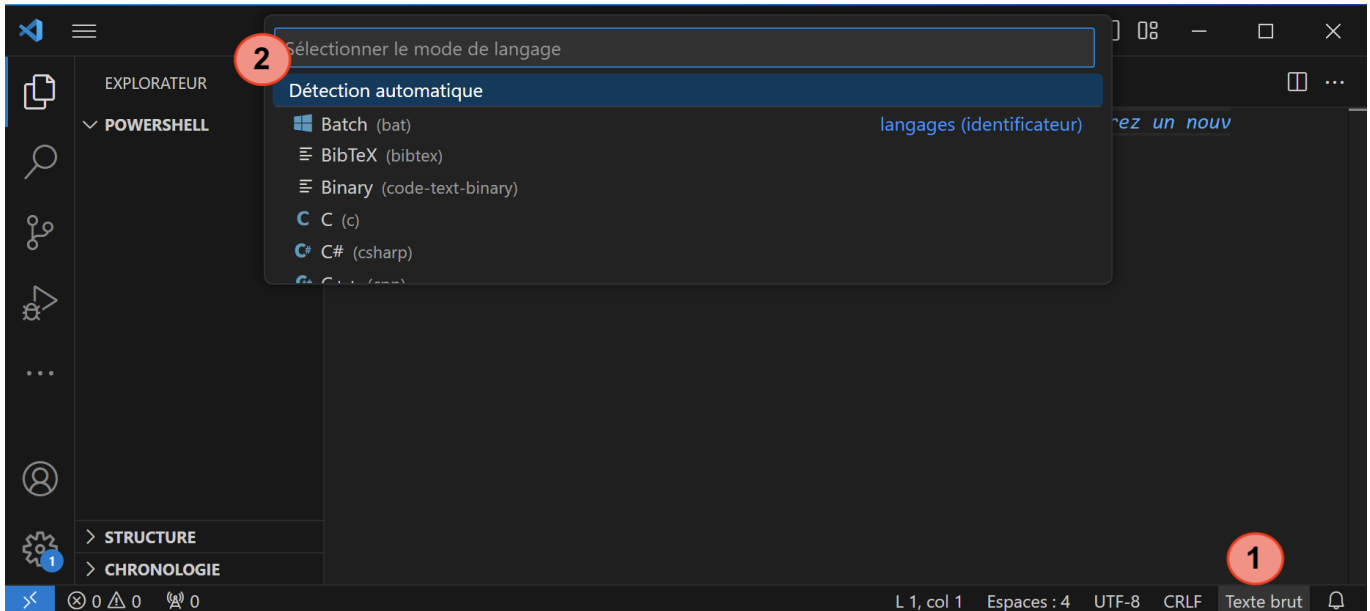
Il y a beaucoup d'autres paramètres disponibles et qui pourront vous permettre de personnaliser votre environnement. Nous allons nous satisfaire de cette option pour débiter.

B. Créer un premier script PowerShell

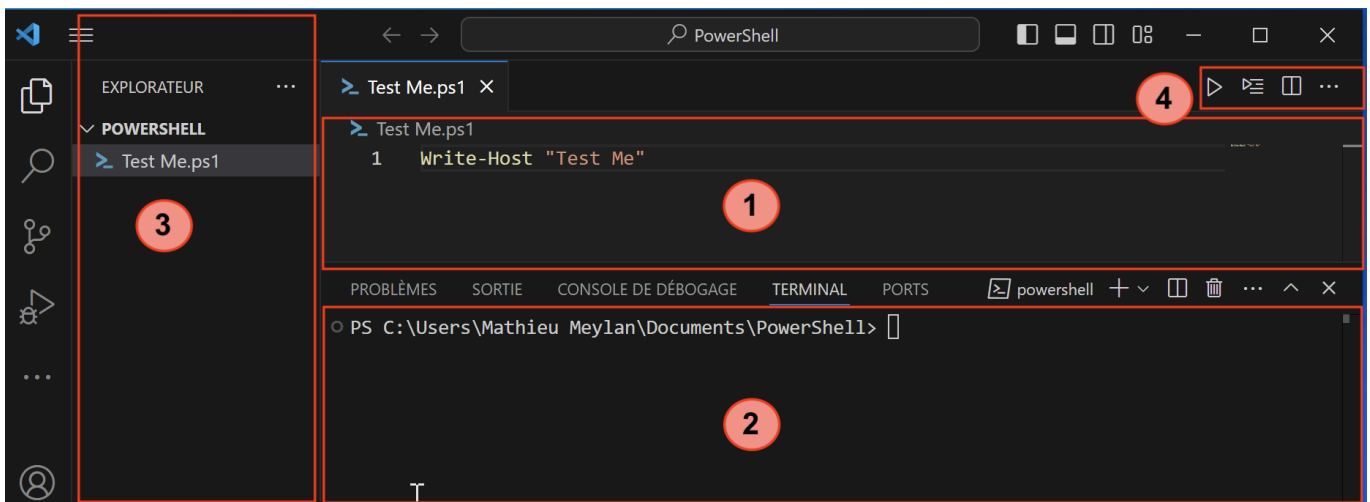
Dans Visual Studio Code, il y a plusieurs façons de créer un nouveau fichier. Tout simplement, vous pouvez cliquer sur le menu "Fichier" puis "Nouveau fichier texte", mais le plus rapide c'est d'utiliser le raccourci "CTRL + N" afin de créer un nouveau fichier.



Une fois votre fichier créé, il est nécessaire de préciser qu'il est de type "PowerShell". Pour ce faire cliquer en bas de page sur "Test brute" (1) et dans la fenêtre de sélection en haut chercher "PowerShell" (2).



Puis sauvez votre fichier avec l'extension ".ps1".



Un nouveau fichier s'ouvre dans un onglet dédié. Sans rentrer dans le détail, voici quelques informations sur l'organisation de l'interface de VSCode :

1. Zone où vous rédigez votre script PowerShell (ou dans un autre langage), c'est l'éditeur de code tout simplement.
2. Zone où se situe le terminal. C'est là que vous pouvez visualiser le résultat de votre script, ou saisir directement des commandes.
3. Zone de l'Explorer, où vous pouvez afficher le contenu d'un répertoire de votre PC (un répertoire où vous stockez vos scripts, par exemple). Ce répertoire sera associé à un workspace qui aura sa propre configuration. Vous pouvez créer plusieurs workspaces.
4. Zone avec des boutons d'action. Le premier bouton sert à exécuter tout le script, le second bouton sert à exécuter uniquement le code sélectionné du script, tandis que le troisième bouton permet de diviser l'éditeur de code en deux zones et ainsi avoir deux scripts côte à côte.

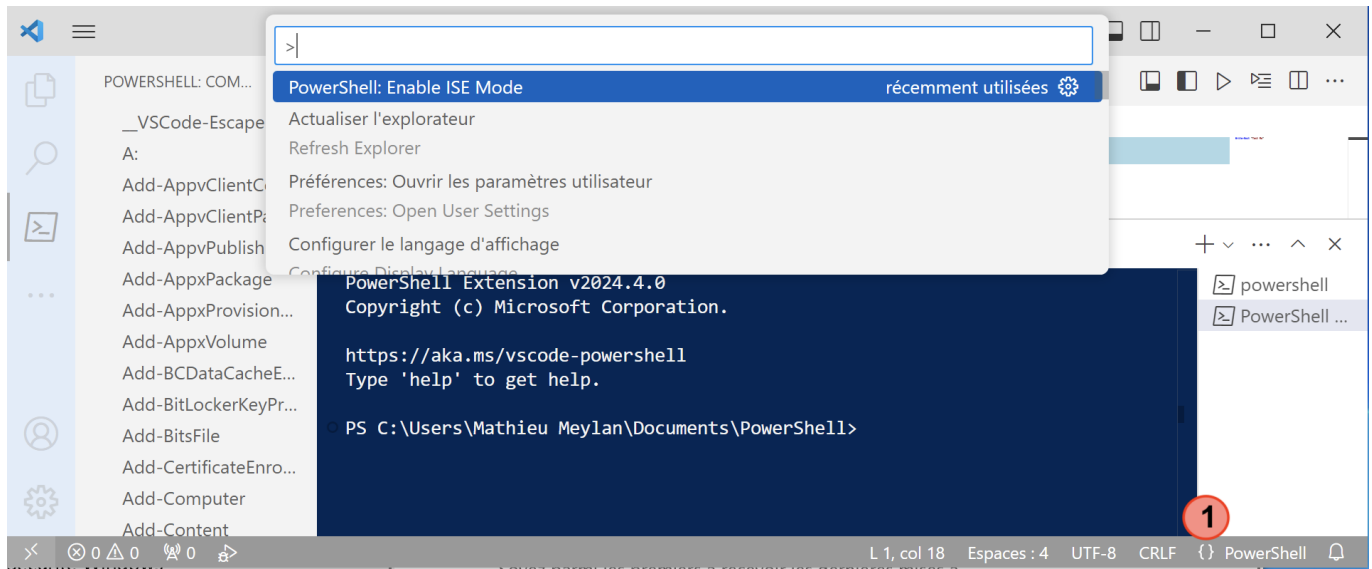
Il ne vous reste plus qu'à coder ! Pensez à enregistrer votre script avec les touches raccourci "CTRL + S".

C. Le mode ISE dans VSCode

Pour les personnes habituées à PowerShell ISE, sachez que Visual Studio Code intègre un mode nommé tout bonnement "Mode ISE".

Pour l'activer, il suffit d'appeler l'invite de commande VSCode via "F1" ou "CTRL + SHIFT + P" et de saisir "ISE Mode" afin de cliquer sur "PowerShell: Enable ISE Mode". Pour le désactiver, il suffit de répéter l'opération pour sélectionner "PowerShell: Disable ISE Mode".

Voici un aperçu :



Pour passer de Windows PowerShell à PowerShell utiliser le raccourci "{}" (1).

D. Conclusion

Vous êtes maintenant en mesure d'utiliser Visual Studio Code en tant qu'éditeur de code pour PowerShell ! Il y a beaucoup d'autres fonctionnalités à explorer, car VSCode est une application riche et personnalisable !

En attendant la suite, pour aller plus loin, vous pouvez consulter la documentation VSCode :

- [PowerShell avec VSCode](#)

Je vous souhaite de profiter de toute la puissance de PowerShell avec VSCode !