VISUAL STUDIO CODE

Editeur de Code de MICROSOFT

Peut s'exécuter sur Windows, MacOs et Linux 

Pour modifier, débugger ou générer du code avec de nombreuses extensions pour améliorer ce logiciel gratuit

Installation de cet IDE

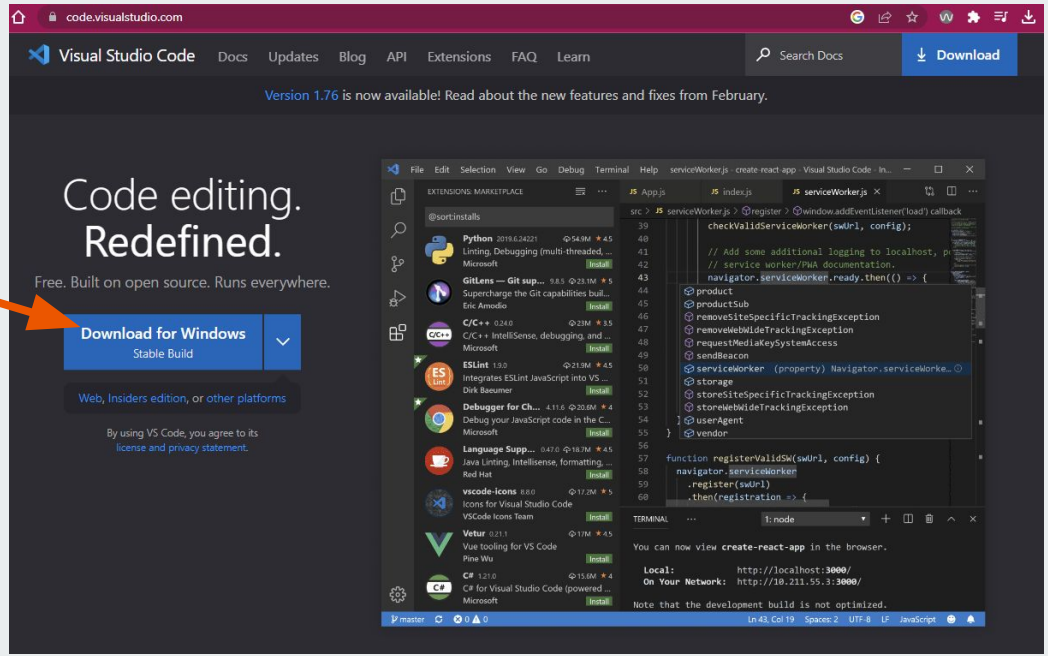
IDE: Environnement de Développement Intégré

C'est une application qui permet aux développeurs de créer (développer) efficacement un code logiciel.

Le lien officiel pour installer VSC : https://code.visualstudio.com/

En fonction de votre système d'exploitation, sélectionner avec le menu déroulant:

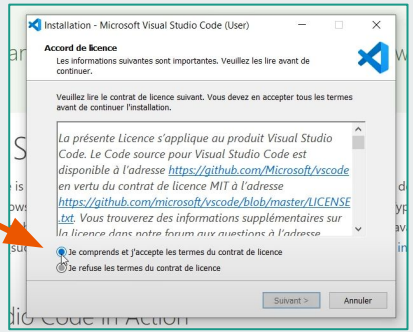
- soit **Windows** (pour PC)
- soit **Mac OS** (pour Apple)
- soit **Unix** (pour les serveurs)



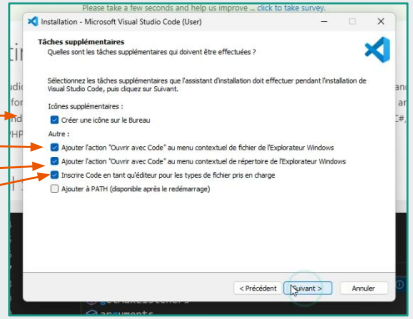
Pendant le téléchargement, une page de présentation rapide s'affiche.

Le **téléchargement peut prendre plusieurs minutes…**

- Acceptez les termes du contrat

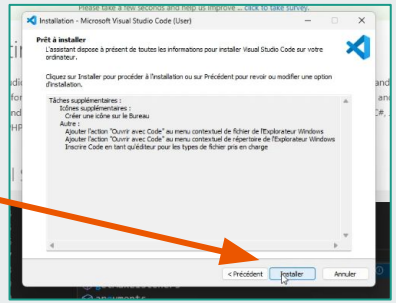


- Choisir les raccourcis de VSC
  - SUR LE BUREAU
  - EN OPTION SUR LE FICHIER
  - EN OPTION SUR LE RÉPERTOIRE
  - OUVRIR POUR TOUS LES FICHIERS

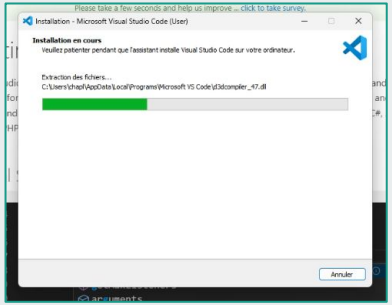


- Cliquez sur *suivant*

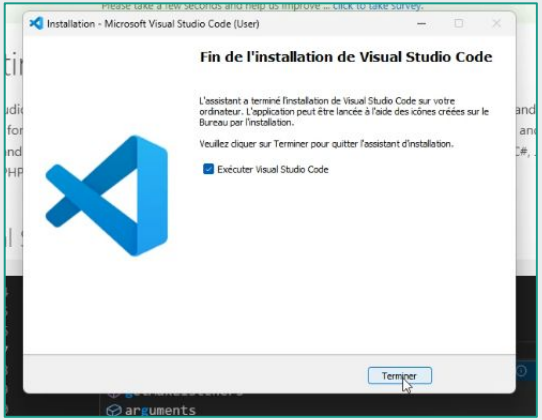
- Récapitulatif des raccourcis VSC
  Cliquez sur *installer*



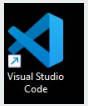
- Installation de Visual Studio Code en cours 👍



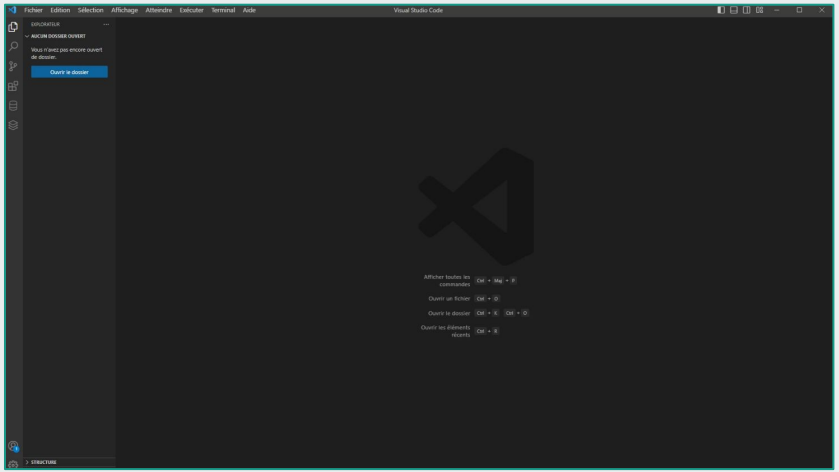
Visual Studio Code est installé
Cliquez sur *terminer*



VSC peut s’ouvrir:



- en cliquant sur l'icône du bureau de VSC
- en cliquant droit sur le fichier à ouvrir / *ouvrir avec* / *Visual Studio Code*

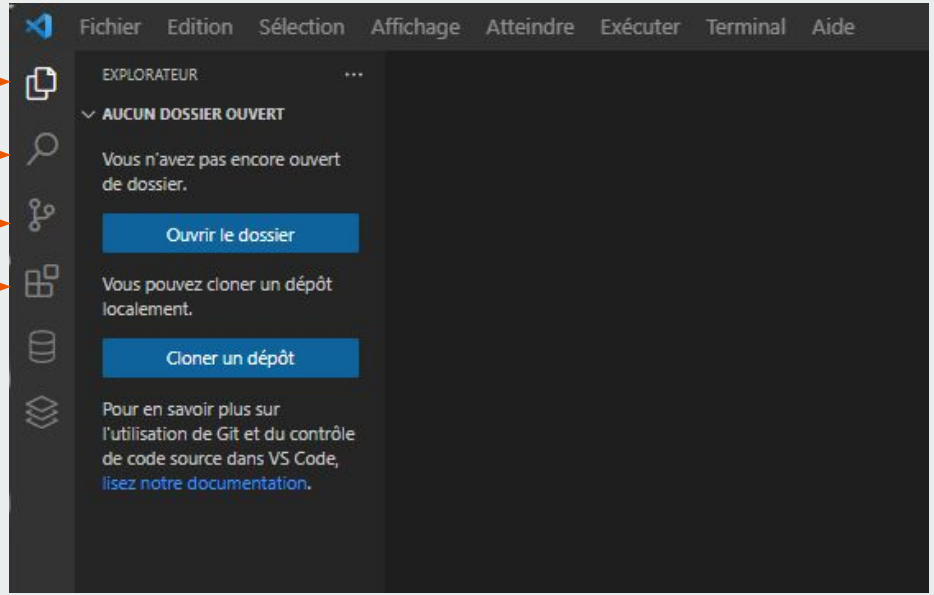


Rechercher son dossier

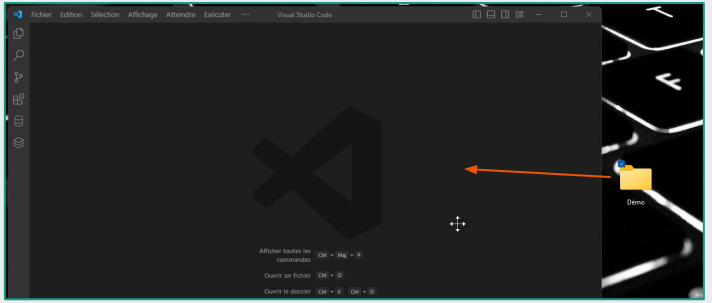
Loupe de recherche

Git

Gérer les extensions

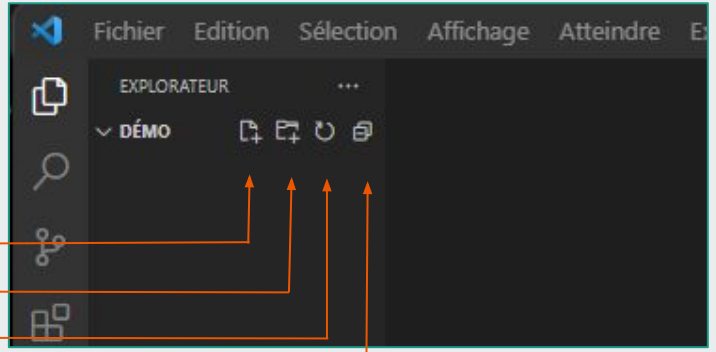


Créons un dossier *Démo* sur notre bureau et déplacez-le directement dans VSC.

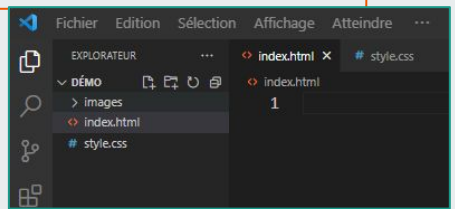


Le dossier s'affiche dans la colonne latérale de gauche.
Au survol de la souris, 4 icônes apparaissent

- Créer un nouveau fichier
- Créer un nouveau dossier
- Réactualiser
- Réduire la taille

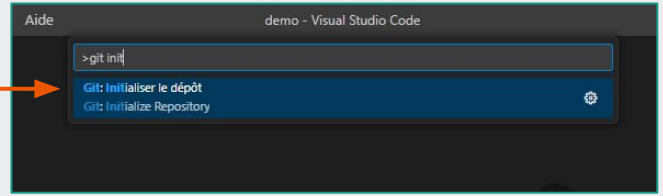


Créons un dossier images et deux fichiers index.html et style.css en cliquant sur les icônes correspondants.

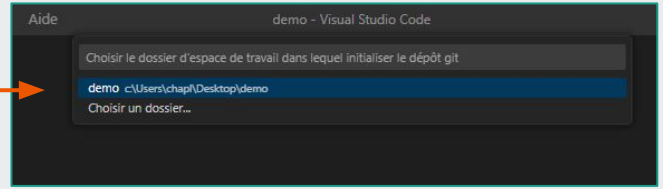


Utiliser le versionning de GIT directement sur VISUAL STUDIO CODE

Ctrl + Shift + P => permet d'accéder au panneau de commande
Taper **Git init** (permet d'initialiser un projet) puis **Entrée**

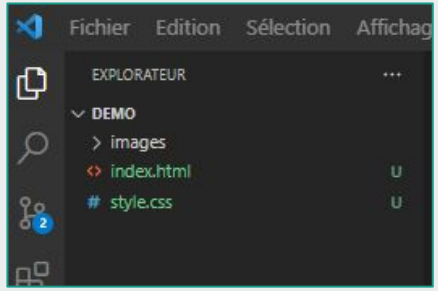


Sélectionner le chemin du projet



📌 Il y a différents changements car on vient d'initialiser le projet:

- les dossiers et fichiers sont en vert
- Il y a une lettre à droite des dossiers (ici: U)
- Sur l'icône du contrôle de code source, il y a un nombre, c' est le versionning



Au fil de votre versioning, il y aura différentes lettres à droite des fichiers

A un nouveau fichier a été ajouté

M un fichier existant a été modifié

D un fichier a été supprimé

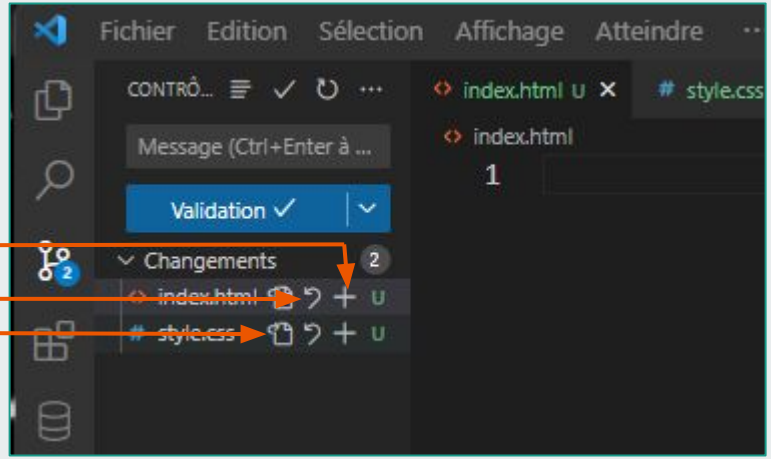
U un fichier est nouveau ou a été modifié mais n'a pas encore été ajouté

C il y a un conflit dans le fichier

R le fichier a été renommé

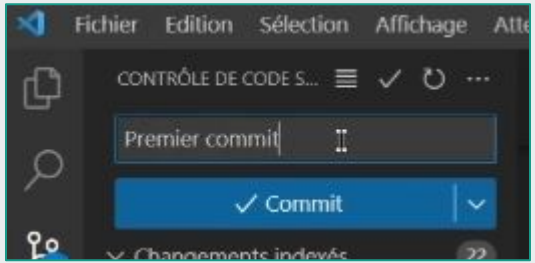
Sur le badge se trouve l'ensemble des fichiers qui ne sont pas suivis:

- on peut les ajouter
- on peut annuler les modifications
- On peut ouvrir le fichier dans l'éditeur

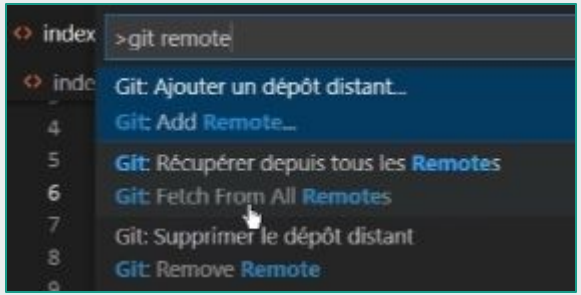


Pour tout valider, cliquer sur le **+** à côté de *changements*

Ecrire le nom du commit dans la fenêtre au dessus de *Commit*

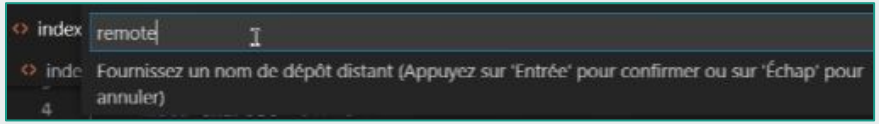


On peut publier sur un dépôt distant
Avec **Ctrl + shift + P**
Ecrire Git remote
Et cliquez sur *Ajouter un dépôt distant*

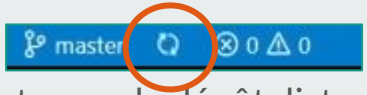


Copiez votre URL de GitHub ou GitLab ou GitBucket

Saisissez un nom (par convention on écrit *remote*)



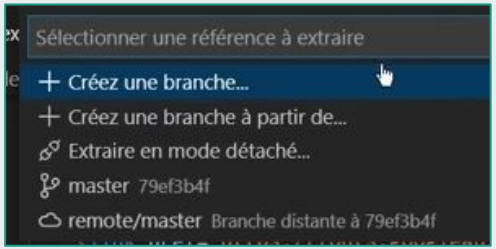
Cliquer sur *Publier Branch*, VSC va l'envoyer à distance

En bas, à gauche se trouve un icône rond 

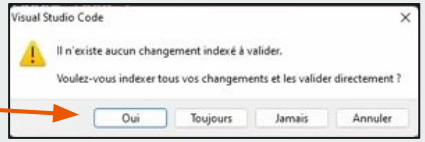
qui permet de synchroniser les changements avec le dépôt distant.
Validez la pop-up avec *OK*.

A gauche de l'icône est écrit la branche sur laquelle vous versionez, ici **master.** 

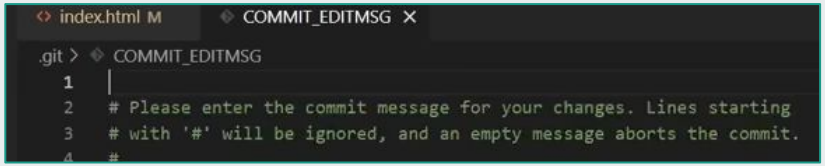
Pour choisir une autre branche ou créer une nouvelle branche, cliquez sur master et effectuez vos opérations.



📌Vous devez valider tous vos changements avant de commiter avec le **+** sinon VSC ouvrira une pop-up pour vous suggérer de le faire. Dans ce cas, cliquez sur *Oui*.



Entrez un nom de commit

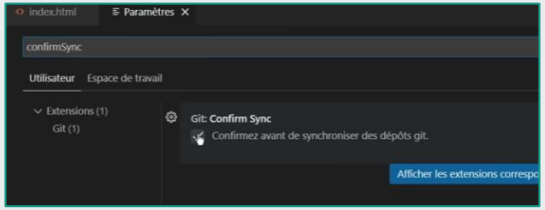


Vous sauvegardez, vous fermez. En bas, l'icône s'est modifié 

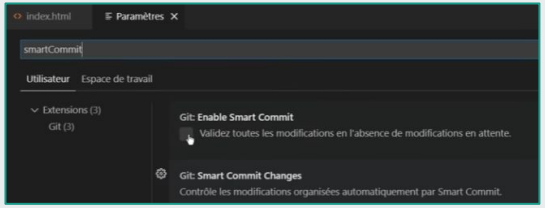
Il y a 0 commit à récupérer (flèche vers le bas) et 1 commit à envoyer (flèche vers le haut).
Cliquez ensuite sur le commit à envoyer. 

Et voilà 👍

Pour désactiver la pop up qui peut être pénible:
Ctrl + shift + p / écrire **Préférence** / **confirmSync**
Décocher la case

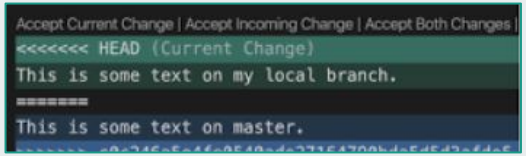


Pour ne plus valider les changements avant le commit :
Ctrl + shift + p / écrire **Préférence** / **smartCommit**
Cocher la case

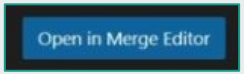


GIT MERGE ET FUSION

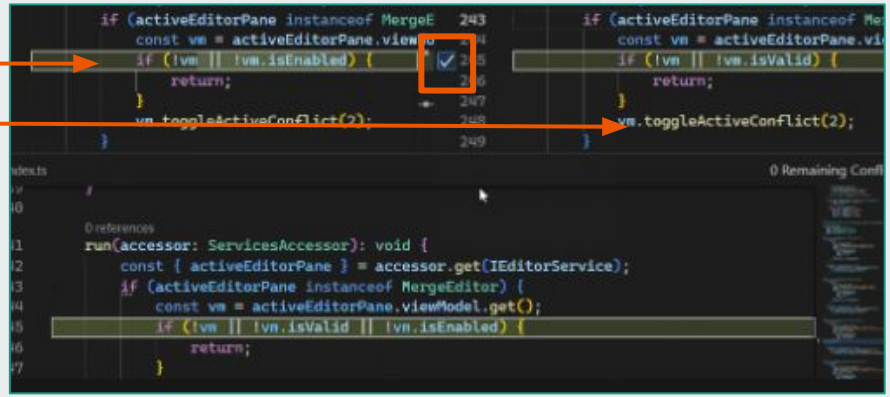
Un conflit entre le local et le dépôt distant se caractérise ainsi:



Il existe une autre interface en cliquant en bas à droite sur *Open in Merge Editor*

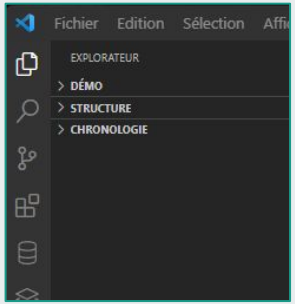


A gauche : le **dépôt distant**
A droite: vos **modifications en local**
A gauche du numéro de la ligne, il y a une case à cocher. Cochez la/les modifications que vous souhaitez conserver via ces cases à cocher.
En bas, le résultat s'affiche.
Si vous cochez sur les deux, VSC fera une fusion des deux.



Puis cliquer sur *Accepter la fusion*
Expliquez dans le titre du commit, les choix que vous avez faits et commitez / pushez

Dans la partie chronologie se trouvent les modifications qui ont été apportées au projet.
C'est une manière rapide et simple de voir la modification, la date de celle-ci ainsi que son auteur.



Une extension pratique : **GIT GRAPH** (permet de visualiser sous forme de graphique et de naviguer parmi les commits et les branches, de nombreuses opérations possibles depuis l'interface de Git Graph).

Un autre extention : **GIT LENS** (permet de visualiser les commits directement en fin de ligne de code en donnant les détails du commits). Cette extension se personnalise très facilement.