

NEW App Platform: reimagining PaaS to make it simpler for you to build, deploy, and scale apps.



Community



TUTORIAL

Comment utiliser l'intégration de Git dans le code de Visual Studio

Git VS Code

By [James Quick](#)

Published on November 12, 2020

6.1k

Français



Introduction

Visual Studio Code (VS Code) est devenu l'un des éditeurs les plus populaires dans le domaine du développement web. Il a acquis une telle popularité grâce à ses nombreuses fonctionnalités intégrées telles que l'intégration du contrôle de source, notamment avec [Git](#). Exploiter la puissance de Git à partir de VS Code peut rendre votre flux de travail plus efficace et plus robuste.

Dans ce tutoriel, vous explorerez l'utilisation de l'intégration du contrôle de source dans VS Code avec Git.

Conditions préalables

Pour terminer ce tutoriel, vous aurez besoin des éléments suivants :

- Git installé sur votre machine. Pour plus de détails sur la réalisation de ce projet, consultez le tutoriel [Démarrer avec Git](#).
- La dernière version de [Visual Studio Code](#) installé sur votre machine.

Étape 1 — Se familiariser avec l'onglet Source Control

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics.



Enter your email address

Sign Up

Dans votre terminal, créez un répertoire pour un nouveau projet et changez dans ce répertoire :

```
$ mkdir git_test  
$ cd git_test
```

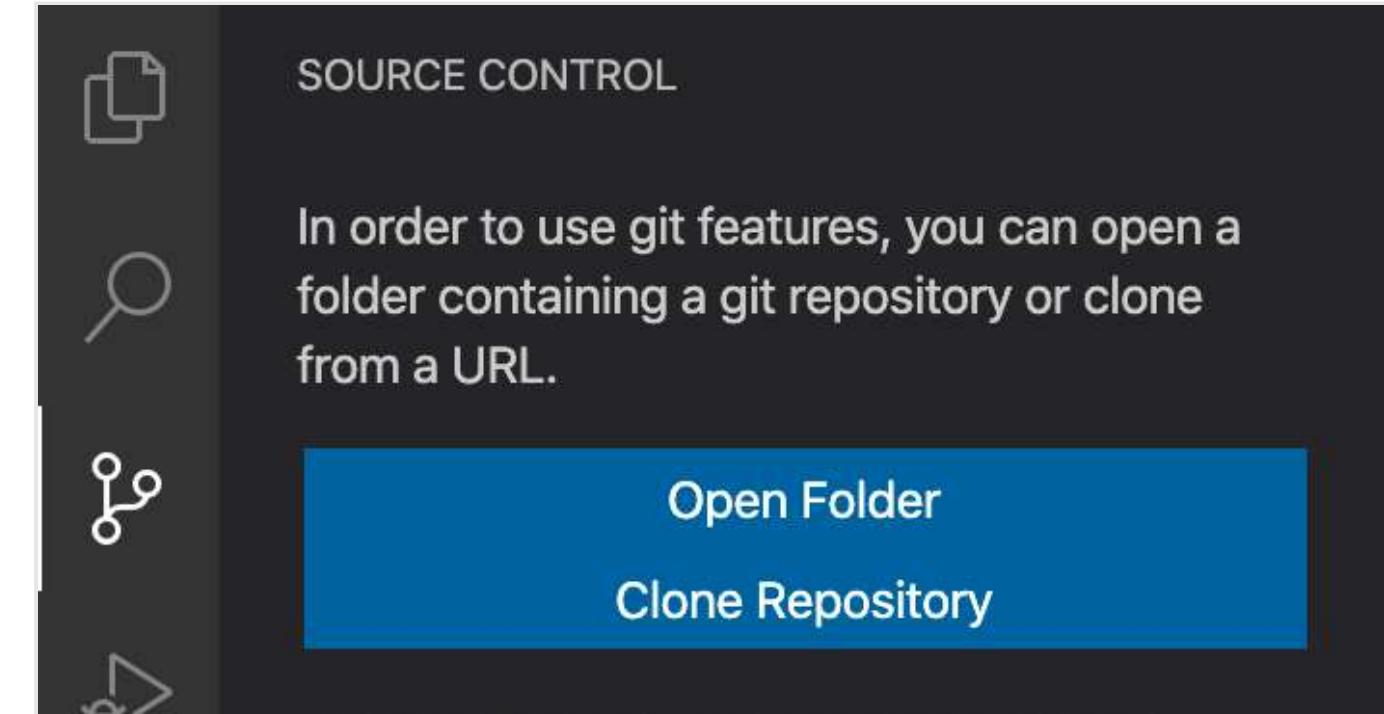
Ensuite, créez un référentiel Git :

```
$ git init
```

Une autre façon d'y parvenir avec Visual Studio Code consiste à d'ouvrir l'onglet Source Control (l'icône ressemble à **une fissure sur une route**) dans le panneau de gauche :



Ensuite, sélectionnez **Open Folder**:



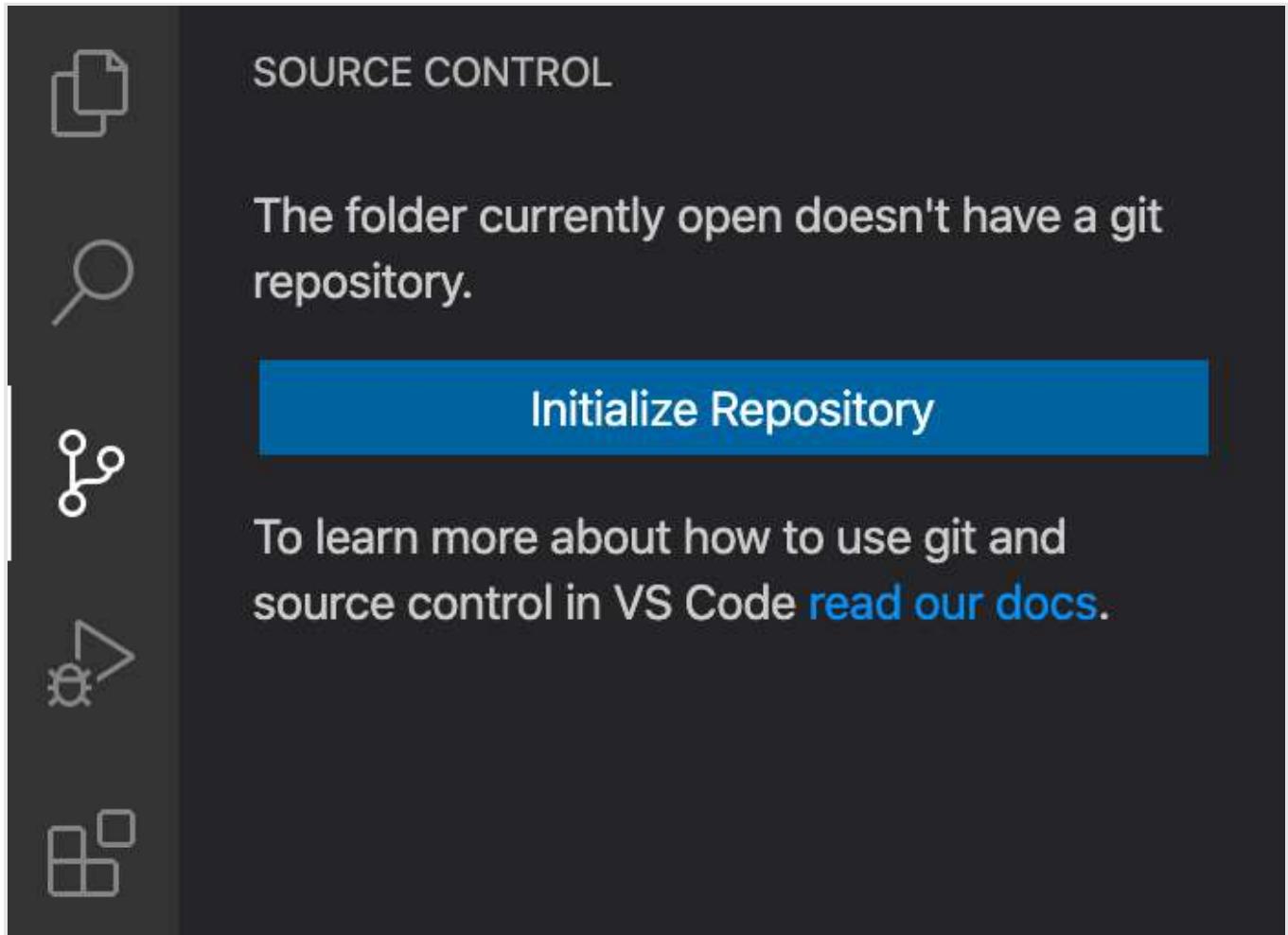
Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics.

Enter your email address

Sign Up

Cela ouvrira votre explorateur de fichiers au répertoire actuel. Sélectionnez le répertoire de projets préféré et cliquez sur **Open**.

Ensuite, sélectionnez **Initialize Repository**:



Si vous vérifiez maintenant votre système de fichiers, vous verrez qu'il comprend un répertoire `.git`. Pour ce faire, utilisez le terminal pour naviguer dans le répertoire de votre projet et lister tous les contenus :

```
$ ls -la
```

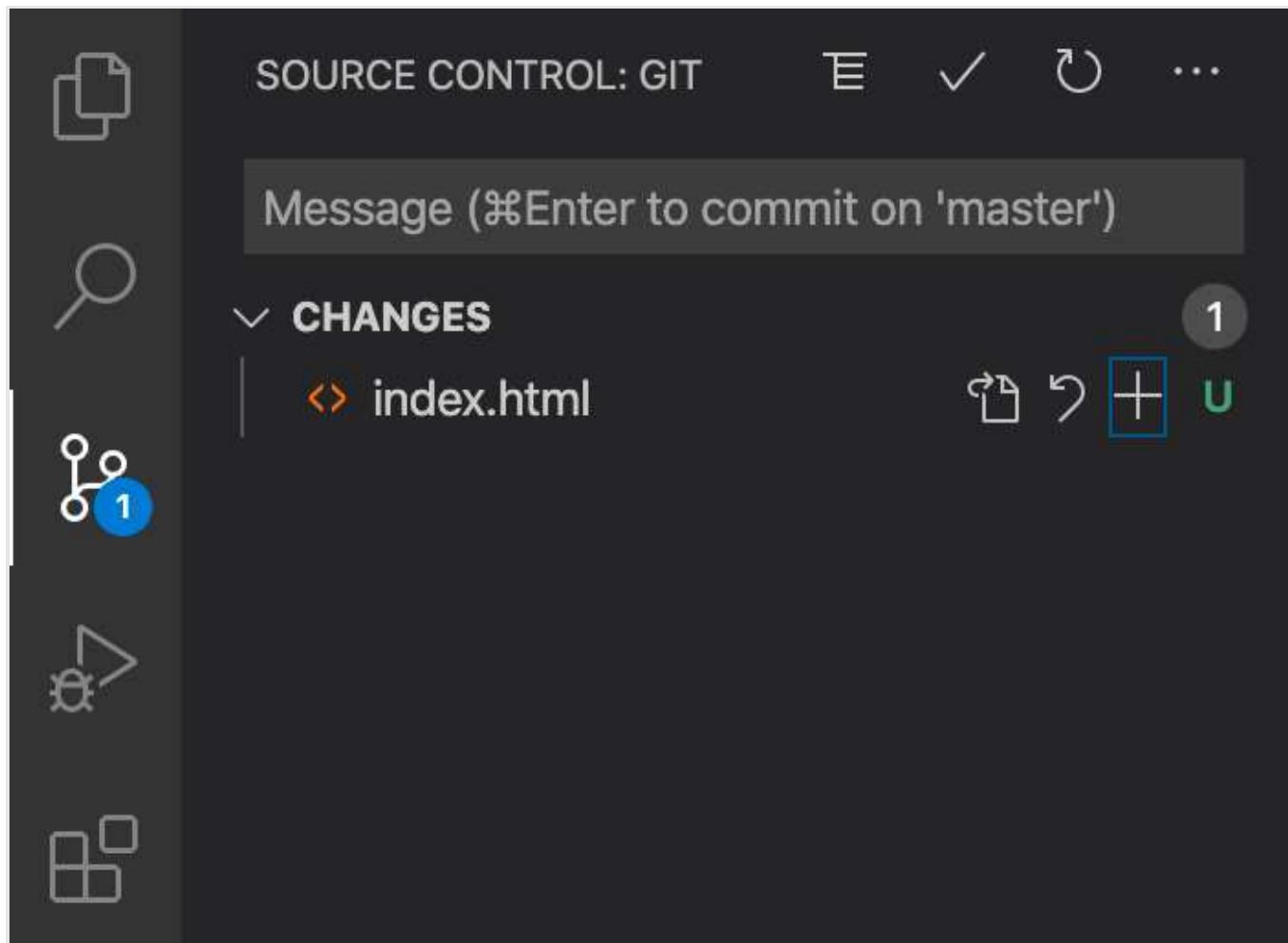
Vous verrez le répertoire `.git` qui a été créé :

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up

Après avoir fait cela, vous verrez dans le **panneau Source Control** que votre nouveau fichier apparaît avec la lettre **U** à côté. **U** signifie *untracked file*, c'est-à-dire un fichier qui est nouveau ou modifié (ou fichier non traqué), mais qui n'a pas encore été ajouté au référentiel :



Vous pouvez maintenant cliquer sur l'icône **plus** (+) de la liste des fichiers `index.html` pour suivre le fichier par le référentiel.

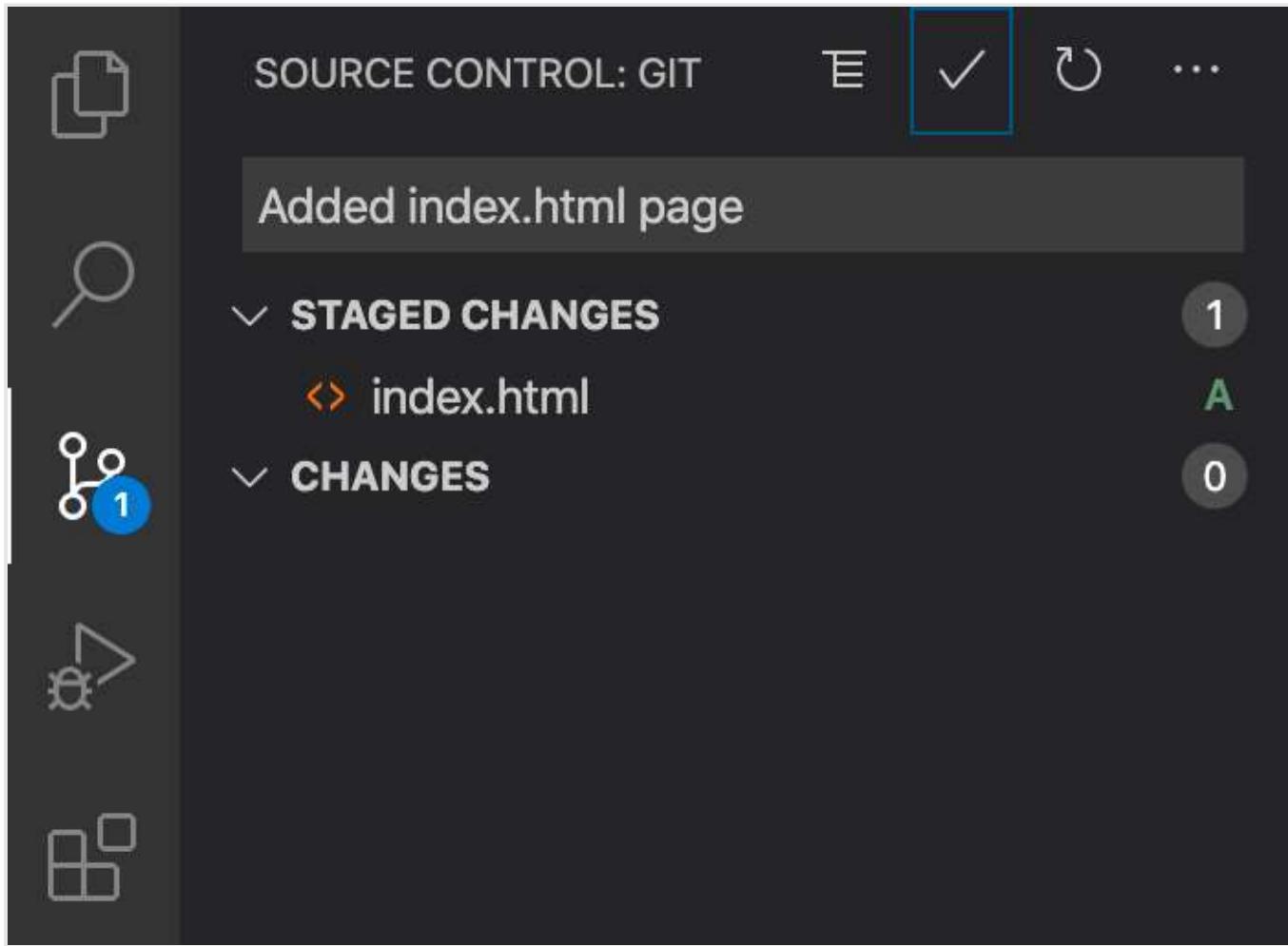
Une fois ajoutée, la lettre à côté du fichier se transforme en **A**. **A** représente un nouveau fichier qui a été ajouté au référentiel.

Pour valider vos modifications, tapez un message de commit dans la zone de saisie située en haut du **panneau Source Control**. Ensuite, cliquez sur l'icône **check** pour effectuer le commit.

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up



Vous constaterez alors qu'il n'y a pas de changements en cours.

Ensuite, ajoutez un peu de contenu à votre fichier `index.html`.

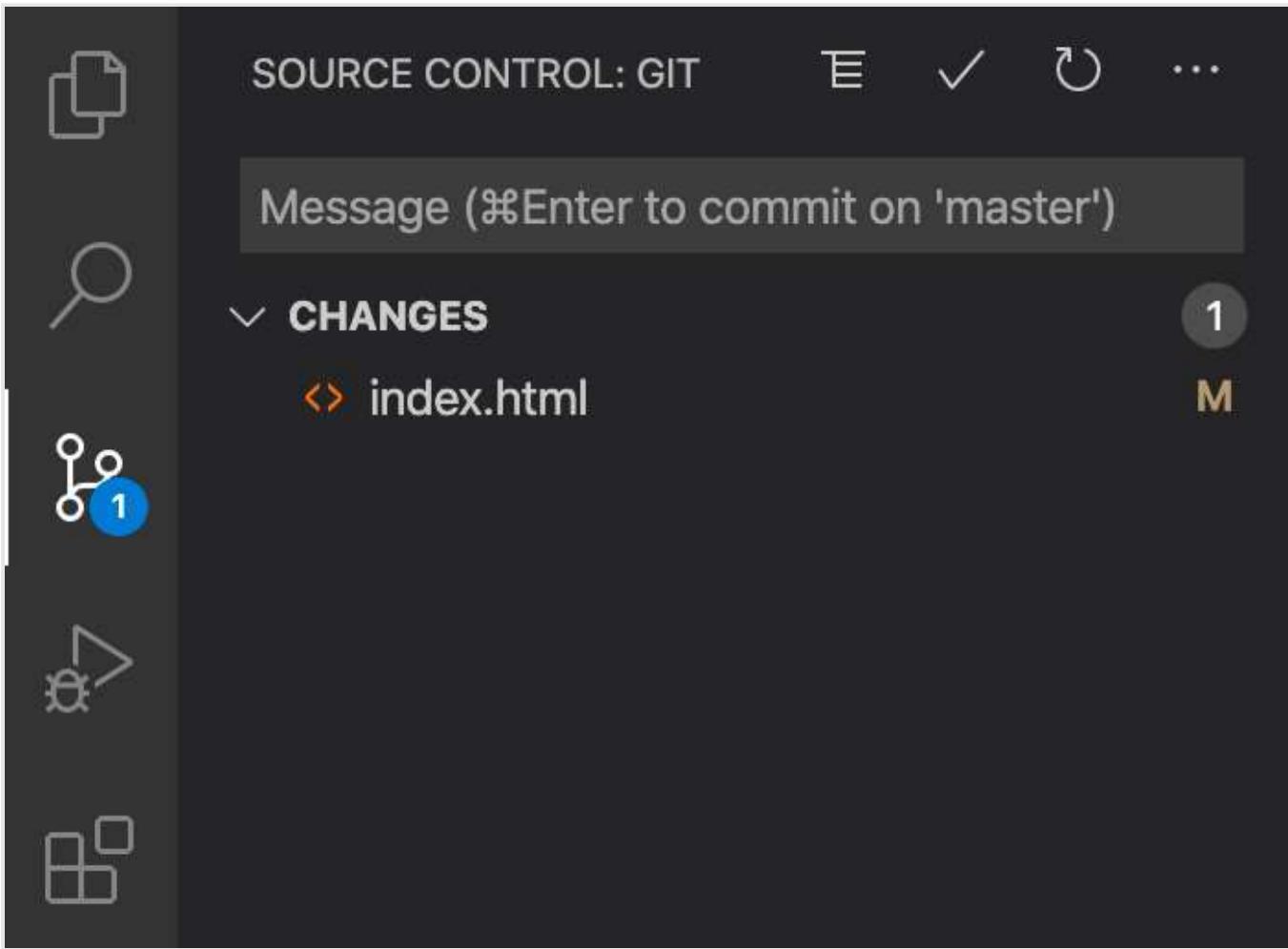
Vous pouvez utiliser un raccourci Emmet pour générer un squelette HTML5 en VS Code en appuyant sur le bouton ! touche suivie de la touche Tab . Allez-y et ajoutez quelque chose dans le `<body>` comme un en-tête `<h1>` et le sauvegarder.

Dans le panneau Source Control, vous verrez que votre fichier a été modifié. La lettre **M** apparaîtra à côté et représentant un dossier qui a été *modifié* :

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up



Pour la pratique, allez-y et validez aussi ce changement.

Maintenant que vous êtes familiarisé avec le panneau de contrôle de source, vous allez passer à l'interprétation des indicateurs de gouttière.

Étape 2 — Interpréter les indicateurs de gouttière

Au cours de cette étape, vous examinerez ce que l'on appelle la *gouttière* en VS Code. La gouttière est la zone étroite située à droite du numéro de ligne.

Si vous avez déjà utilisé le pliage de code, les icônes **de maximise** et de **minimise** sont situées dans la gouttière.

Commençons par apporter une petite modification à votre fichier `index.html`, comme par exemple une modification du contenu de la balise `<h1>`. Vous remarquerez alors une

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up

gouttière qu'il y a un triangle rouge. Le triangle rouge signifie qu'une ligne ou un groupe de lignes a été supprimé.

Enfin, au bas de votre section <body> ajoutez une nouvelle ligne de code et remarquez la barre verte. La barre verte verticale signifie qu'une ligne de code a été ajoutée.

Cet exemple illustre les indicateurs de gouttière pour une ligne modifiée, une ligne supprimée et une nouvelle ligne :

```
git_test > index.html > ...
1   <!DOCTYPE html>
2   <html lang="en">
3
4   <head>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <meta http-equiv="X-UA-Compatible" content="ie=edge">
8     <title>Git Test</title>
9   </head>
10
11  <body>
12    <!-- the following line has been modified -->
13    <h1>This is an Updated GIT test</h1>
14
15  <!-- the following line has been removed -->
16
17  <!-- the following line has been added -->
18  <h2>This is a new line of code</h2>
19  </body>
20
21  </html>
```

Étape 3 — Les fichiers diff

VS Code a également la capacité d'effectuer un diff sur un fichier. Pour ce faire, vous devez généralement télécharger un outil de comparaison distinct. Cette fonction intégrée peut donc vous aider à travailler plus efficacement.

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up

Cet exemple montre qu'une ligne a été ajoutée dans la version actuelle :

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Git Test</title>
</head>
<body>
    <h1>This is a GIT test</h1>
    <h1>This is a new line of code</h1>
</body>
</html>

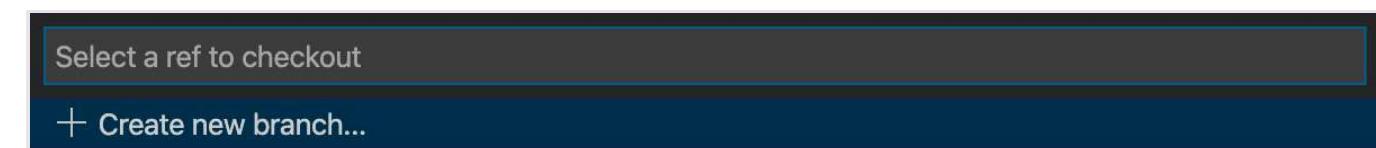
```

Étape 4 — Travailler avec les branches

En vous déplaçant vers la barre inférieure, vous avez la possibilité de créer et de changer de branche. Si vous regardez tout en bas à gauche de l'éditeur, vous devriez voir l'icône de **contrôle de source** (celle qui ressemble à une fissure dans la route) suivie très probablement par `master` ou le nom de la branche en cours de fonctionnement.



Pour créer une branche, cliquez sur le nom de cette branche. Un menu devrait s'afficher pour vous permettre de créer une nouvelle branche :



Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up

Maintenant, modifiez votre fichier `index.html` qui signifie que vous êtes dans la nouvelle branche `test`, comme l'ajout du texte `this is the new test branch`.

Validez ces changements à la branche `test`. Ensuite, cliquez à nouveau sur le nom de la branche en bas à gauche pour revenir à la branche `master`.

Après être retourné(e) à la branche `master`, vous remarquerez que le texte `this is the new test branch` appliqué à la branche `test` n'est plus présent.

Étape 5 — Travail avec les référentiels distants

Ce tutoriel ne l'abordera pas en profondeur, mais grâce au panneau Source Control, vous avez accès pour travailler avec les référentiels à distance. Si vous avez déjà travaillé avec un référentiel distant, vous remarquerez des commandes familières comme « `pull` », « `sync` », « `publish` », « `stash` », etc.

Étape 6 — Installer des extensions utiles

Non seulement VS Code est livré avec de nombreuses fonctionnalités intégrées pour Git, mais il existe également plusieurs extensions très populaires pour ajouter des fonctionnalités supplémentaires.

Git Blame

Cette extension permet de visualiser les informations Git Blame dans la barre d'état de la ligne actuellement sélectionnée.

Cela peut sembler intimidant, mais ne vous inquiétez pas, l'[extension Git Blame](#) est bien plus une question de praticabilité qu'une question de culpabilité. L'idée de « blâmer » quelqu'un pour un changement de code consiste moins à lui faire honte qu'à trouver la bonne personne à qui poser des questions pour certains éléments du code.

Comme vous pouvez le voir sur la capture d'écran, cette extension fournit un message subtil lié à la ligne de code actuelle sur laquelle vous travaillez dans la barre d'outils inférieure, expliquant qui a fait le changement et quand il l'a fait.

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. 

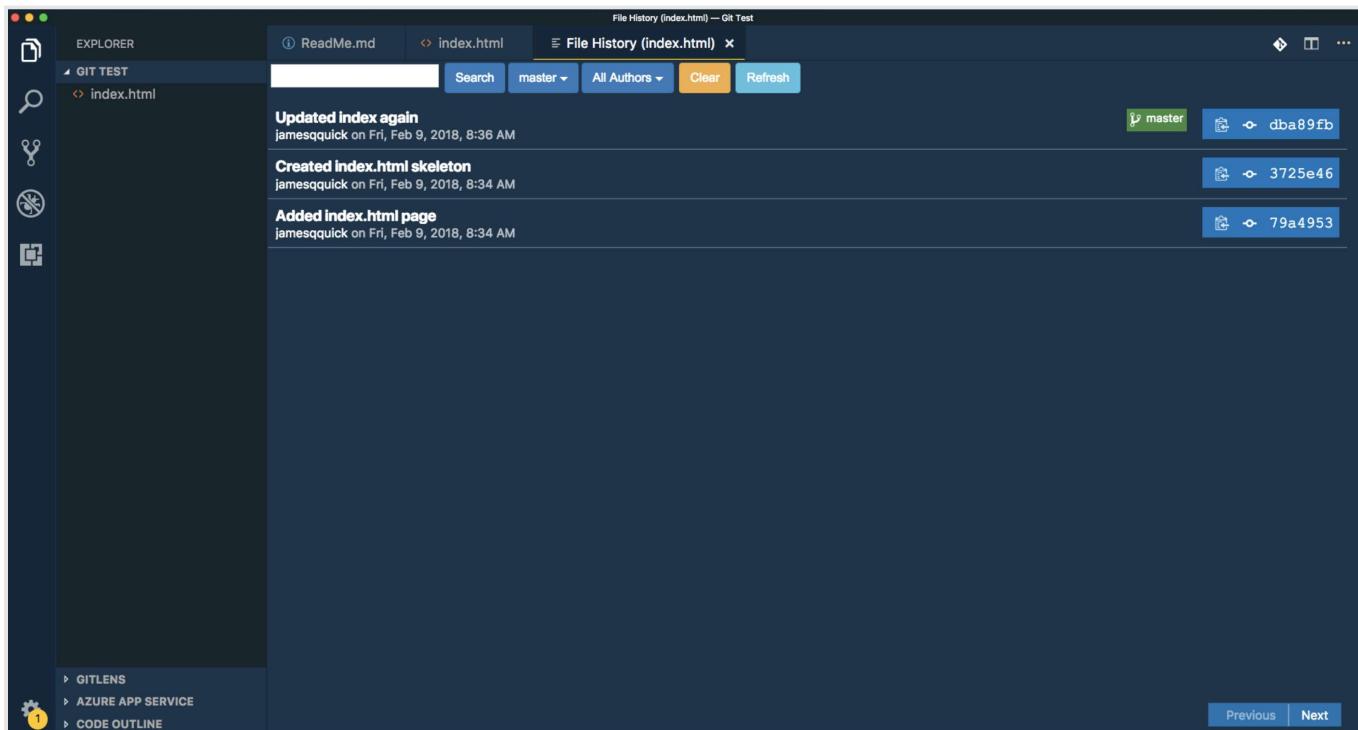
Enter your email address

Sign Up

Git History

Bien que vous puissiez visualiser les changements en cours, effectuer des diff's et gérer des branches grâce aux fonctionnalités intégrées dans VS Code, cela ne fournit pas une vue approfondie de Git History. L'[extension Git History résout ce problème](#).

Comme vous pouvez le voir dans l'image ci-dessous, cette extension vous permet d'explorer en profondeur l'historique d'un fichier, d'un auteur donné, d'une branche, etc. Pour activer la fenêtre Git History ci-dessous, faites un clic droit sur un fichier et choisissez **Git: View File History* *(Afficher l'historique du fichier) :



De plus, vous pouvez comparer les branches et les engagements, créer des branches à partir de commits, et plus encore.

Git Lens

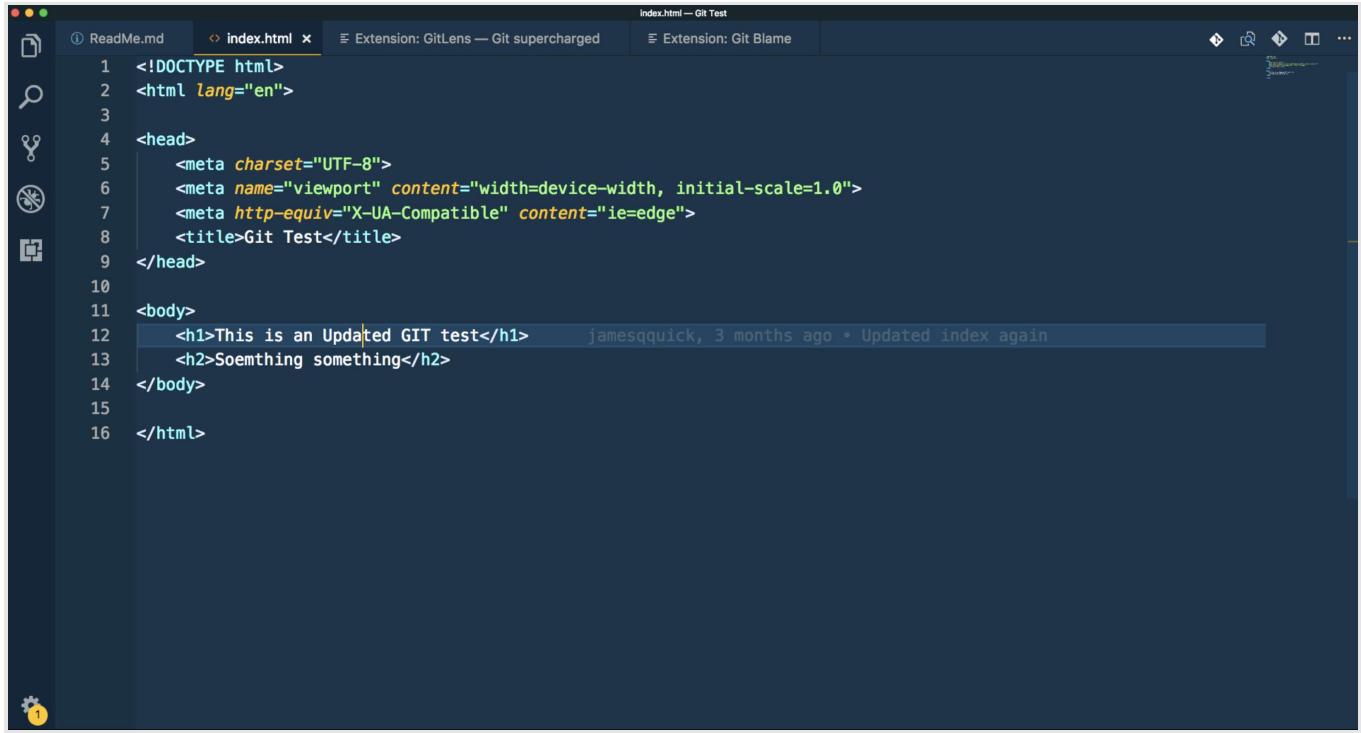
GitLens amplifie les capacités de Git intégrées dans Visual Studio Code. Il vous aide à visualiser la paternité du code en un coup d'œil via les annotations de Git Blame et code lens, à naviguer et à explorer les référentiels Git de manière transparente, à obtenir des informations précieuses via de puissantes commandes de comparaison, et bien plus

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up

Pour les informations de « blame », un message subtil apparaît à droite de la ligne sur laquelle vous travaillez actuellement pour vous informer de qui a fait le changement, quand il l'a fait, et le message de commit associé. D'autres informations apparaissent lorsque l'on survole ce message, comme le changement de code lui-même, l'horodatage, etc.



```
index.html — Git Test
ReadMe.md index.html Extension: GitLens — Git supercharged Extension: Git Blame
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <meta http-equiv="X-UA-Compatible" content="ie=edge">
8   <title>Git Test</title>
9 </head>
10
11 <body>
12   <h1>This is an Updated GIT test</h1> jamesqqquick, 3 months ago * Updated index again
13   <h2>Soemthing something</h2>
14 </body>
15
16 </html>
```

Pour les informations sur Git History, cette extension offre de nombreuses fonctionnalités. Vous avez facilement accès à des tonnes d'options, y compris l'affichage de l'historique des fichiers, l'exécution de différences avec les versions précédentes, l'ouverture d'une révision spécifique, et plus encore. Pour ouvrir ces options, vous pouvez cliquer sur le texte dans la barre d'état inférieure qui contient l'auteur qui a édité la ligne de code et la date à laquelle elle a été éditée.

Cela ouvrira la fenêtre suivante :

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up

Cette extension est pleine de fonctionnalités, et il faudra un certain temps pour qu'elle puisse accueillir tout ce qu'elle a à offrir.

Conclusion

Dans ce tutoriel, vous avez exploré comment utiliser l'intégration du contrôle de source avec VS Code. VS Code peut gérer de nombreuses fonctionnalités qui, auparavant, auraient nécessité le téléchargement d'un outil distinct.

Qu'avez-vous pensé de cette traduction? thumb up thumb down

Was this helpful?

Yes

No



[Report an issue](#)

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up

**JAMES QUICK**

Developer. Speaker. Teacher.
Developer Advocate at Auth0.

Still looking for an answer?

[Ask a question](#)[Search for more help](#)

RELATED

[Comment formater du code avec Prettier dans Visual Studio Code](#)

[Tutorial](#)

[Comment installer Git sur Ubuntu 20.04](#)

[Tutorial](#)

Comments

0 Comments

Leave a comment...

[Sign In to Comment](#)

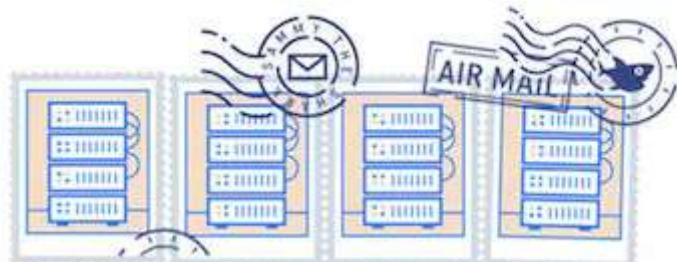
Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics.

Enter your email address

[Sign Up](#)



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.



GET OUR BIWEEKLY NEWSLETTER

Sign up for Infrastructure as a Newsletter.



HOLLIE'S HUB FOR GOOD

Working on improving health and education, reducing inequality, and spurring economic growth?
We'd like to help.

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

Sign Up

You get paid; we donate to tech
nonprofits.

Featured on Community Kubernetes Course Learn Python 3 Machine Learning in Python
Getting started with Go Intro to Kubernetes

DigitalOcean Products Virtual Machines Managed Databases Managed Kubernetes Block Storage
Object Storage Marketplace VPC Load Balancers

Welcome to the developer cloud

DigitalOcean makes it simple to launch in the cloud and scale up as you grow – whether you're running one virtual machine or ten thousand.

[Learn More](#)



Company

About

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. X

Enter your email address

[Sign Up](#)

Products

Pricing
Products Overview
Droplets
Kubernetes
Managed Databases
Spaces
Marketplace
Load Balancers
Block Storage
API Documentation
Documentation
Release Notes

Community

Tutorials
Q&A
Tools and Integrations
Tags
Product Ideas
Write for DigitalOcean
Presentation Grants
Hatch Startup Program
Shop Swag
Research Program
Open Source
Code of Conduct

Contact

Get Support
Trouble Signing In?
Sales
Report Abuse
System Status

Sign up for our newsletter Get the latest tutorials on SysAdmin and open source topics. 

Enter your email address

Sign Up