# Using GitHub Codespaces in Visual Studio Code

**In this article**

You can develop in your codespace directly in Visual Studio Code by connecting the GitHub Codespaces extension with your account on GitHub Enterprise Cloud.

## About GitHub Codespaces in Visual Studio Code 🔗

You can use your local install of Visual Studio Code to create, manage, work in, and delete codespaces. To use GitHub Codespaces in VS Code, you need to install the Codespaces extension. For more information on setting up GitHub Codespaces in VS Code, see "[Prerequisites](#)."

By default, if you create a new codespace on GitHub.com, it will open in the browser. If you would prefer to open any new codespaces in VS Code automatically, you can set your default editor to be VS Code. For more information, see "[Setting your default editor for GitHub Codespaces](#)."
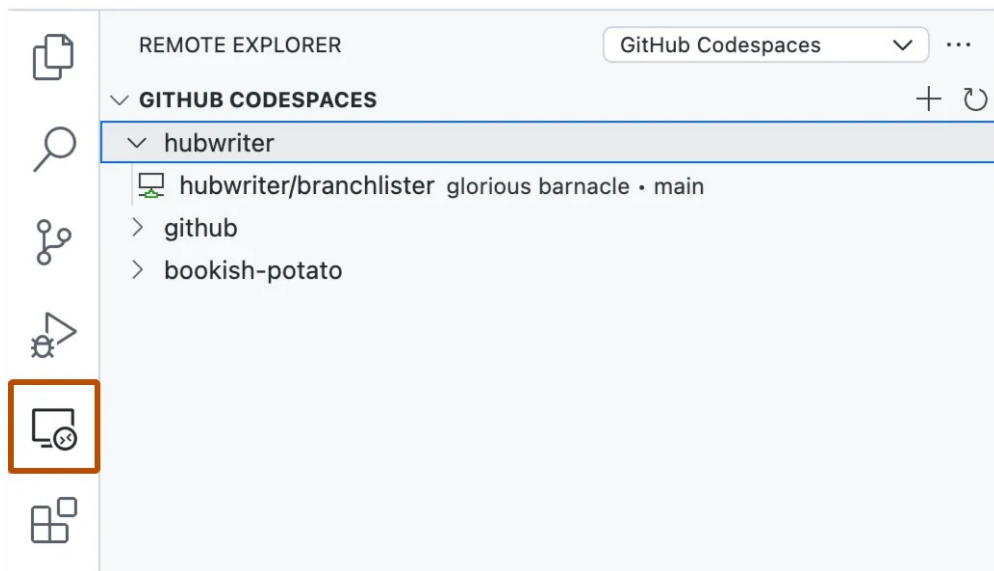
If you prefer to work in the browser, but want to continue using your existing VS Code extensions, themes, and shortcuts, you can turn on Settings Sync. For more information, see "[Personalizing GitHub Codespaces for your account](#)."

## Prerequisites 🔗

To develop in a codespace directly in VS Code, you must install and sign into the GitHub Codespaces extension with your GitHub Enterprise Cloud credentials. The GitHub Codespaces extension requires VS Code October 2020 Release 1.51 or later.

Use the Visual Studio Code Marketplace to install the [GitHub Codespaces](#) extension. For more information, see [Extension Marketplace](#) in the VS Code documentation.
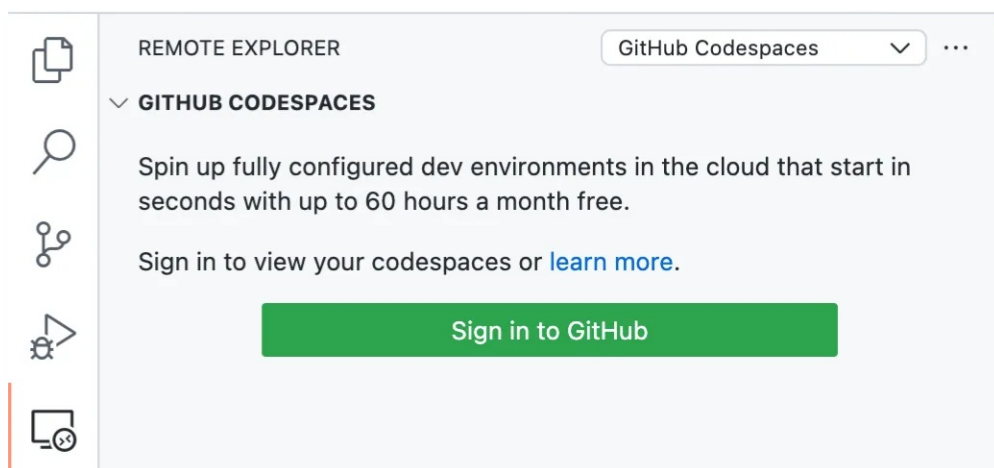
1 In VS Code, in the Activity Bar, click the Remote Explorer icon.

> **Note**: If the Remote Explorer is not displayed in the Activity Bar:
>
> a.  Access the Command Palette. For example, by pressing `Shift` + `Command` + `P` (Mac) / `Ctrl` + `Shift` + `P` (Windows/Linux).
>
> b.  Type: `details` .
>
> c.  Click **Codespaces: Details**.

**2**  Select "GitHub Codespaces" from the dropdown at the top of the "Remote Explorer" side bar, if it is not already selected.

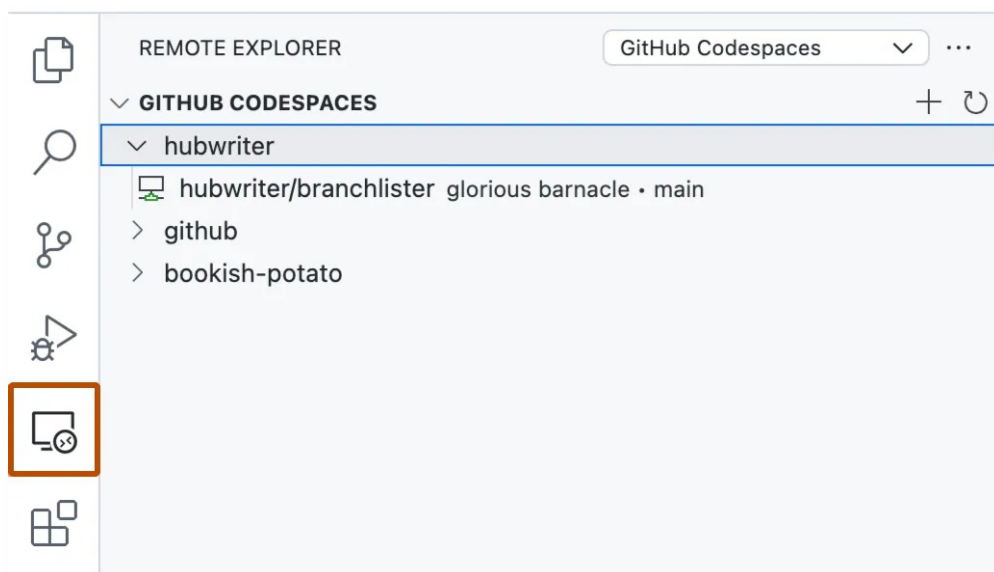**3**  Click **Sign in to GitHub**.



**4**  If you are not currently signed in to GitHub you'll be prompted to do so. Go ahead and sign in.

**5**  When you're prompted to specify what you want to authorize, click the **Authorize** button for "GitHub."

**6**  If the authorization page is displayed, click **Authorize Visual-Studio-Code**.

## Creating a codespace in VS Code 🔗

After you connect your account on GitHub.com to the GitHub Codespaces extension, you can create a new codespace. For more information about the GitHub Codespaces extension, see the [VS Code Marketplace](#).
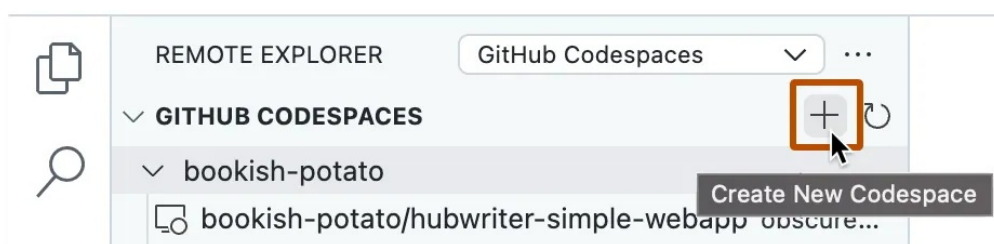
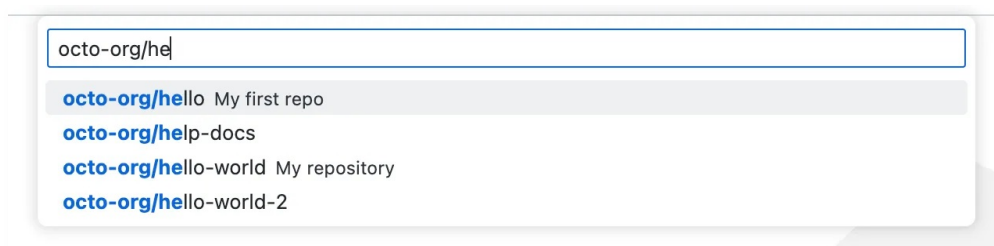1. In VS Code, in the Activity Bar, click the Remote Explorer icon.



> **Note**: If the Remote Explorer is not displayed in the Activity Bar:
>
> a.  Access the Command Palette. For example, by pressing `Shift` + `Command` + `P` (Mac) / `Ctrl` + `Shift` + `P` (Windows/Linux).
>
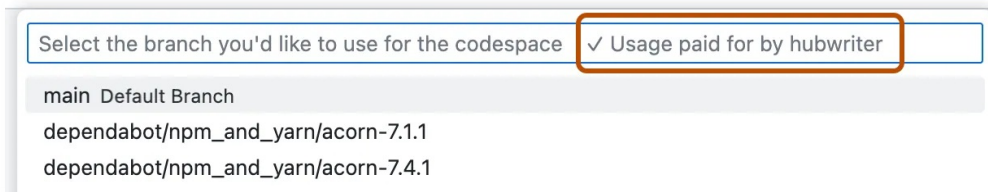> b.  Type: `details` .
>
> c.  Click **Codespaces: Details**.

2. Hover over the "Remote Explorer" side bar and click +.



3. In the text box, type the name of the repository you want to develop in, then select it.



A message is displayed at the right side of subsequent prompts telling you who will pay for the codespace.
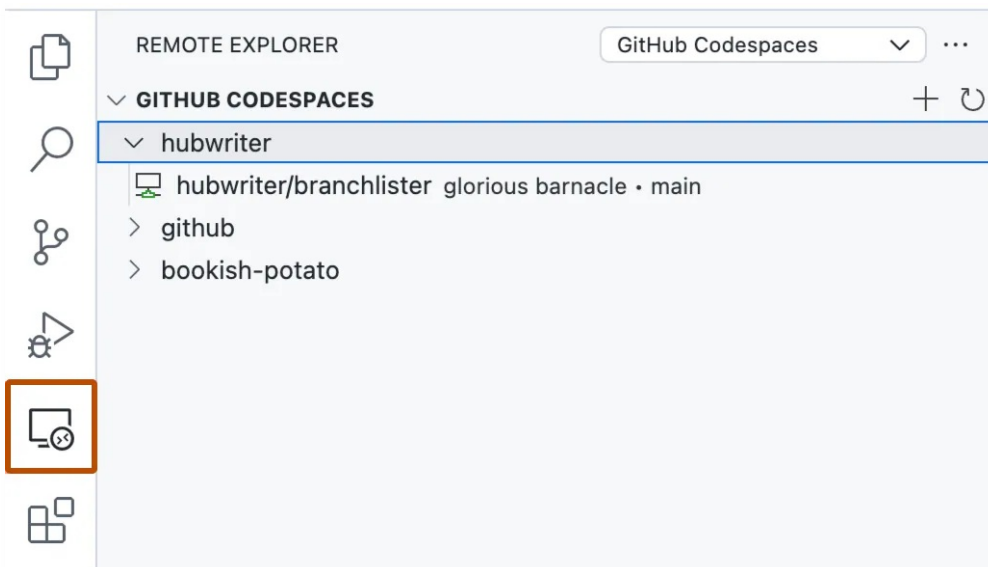
4. Click the branch you want to develop on.

5. If prompted to choose a dev container configuration file, choose a file from the list.

6. Click the machine type you want to use.

> **Note**: Your choice of available machine types may be limited by a number of factors. These can include a policy configured for your organization, or a minimum machine type specification for your repository. For more information, see "Restricting access to machine types" and "Setting a minimum specification for codespace machines."
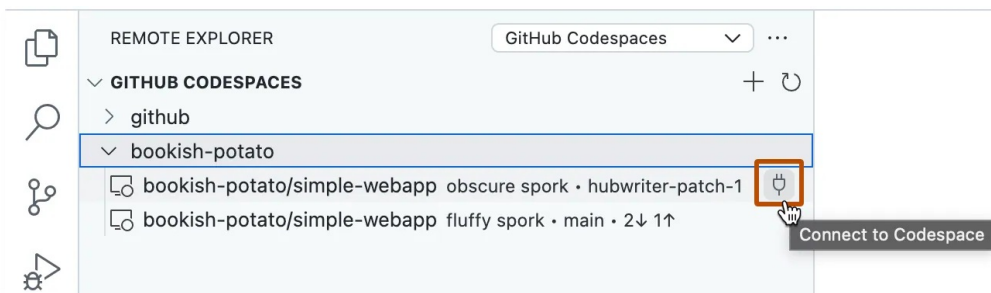
# Opening a codespace in VS Code 🔗

1. In VS Code, in the Activity Bar, click the Remote Explorer icon.



> **Note**: If the Remote Explorer is not displayed in the Activity Bar:
>
> a. Access the Command Palette. For example, by pressing `Shift` + `Command` + `P` (Mac) / `Ctrl` + `Shift` + `P` (Windows/Linux).
>
> b. Type: `details` .
>
> c. Click **Codespaces: Details**.

2. Under "GitHub Codespaces", hover over the codespace you want to develop in.
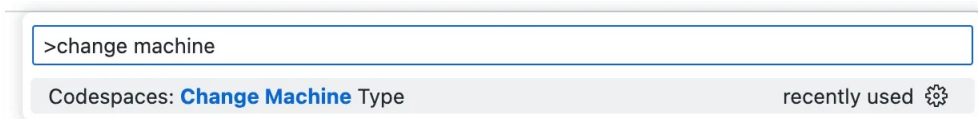
3. Click the connection icon (a plug symbol).

## Changing the machine type in VS Code 🔗

Typically, you can run your codespace on a choice of remote machine types. These machine types offer a choice of hardware specifications ranging from 2 cores to 32 cores, although the full range of machine types may not always be available. Each machine type has a different level of resources and a different billing tier. For information, see "About billing for GitHub Codespaces."
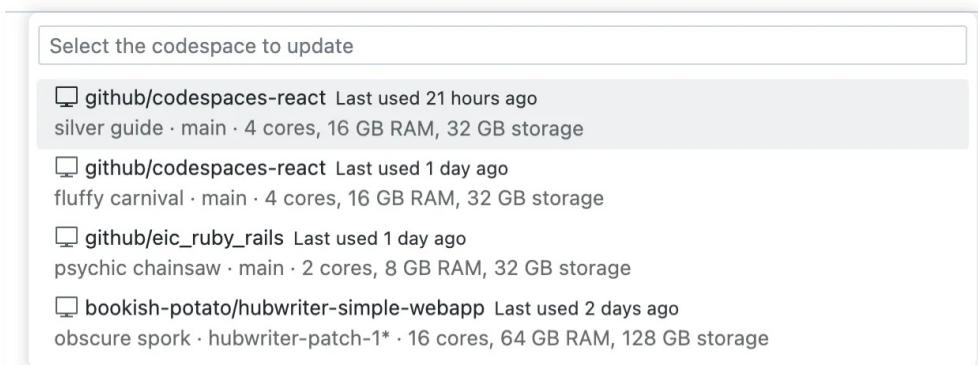
By default the machine type with the lowest valid resources is used when you create a codespace. You can change the machine type of your codespace at any time.

> **Note**: Your choice of available machine types may be limited by a number of factors. These can include a policy configured for your organization, or a minimum machine type specification for your repository. For more information, see "Restricting access to machine types" and "Setting a minimum specification for codespace machines."

① In VS Code, open the Command Palette with `Command` + `Shift` + `P` (Mac) or `Ctrl` + `Shift` + `P` (Windows/Linux).

② Search for and select "Codespaces: Change Machine Type."



③ If you're not following these instructions within a codespace, click the codespace that you want to change.



If you are following these instructions in a codespace, the change will apply to the codespace you're working in.

④ Select the machine type you want to use.

**5** If you're changing to a machine type with a different storage capacity, a prompt will appear asking if you would like to continue. Read the prompt and click **Yes** to accept.
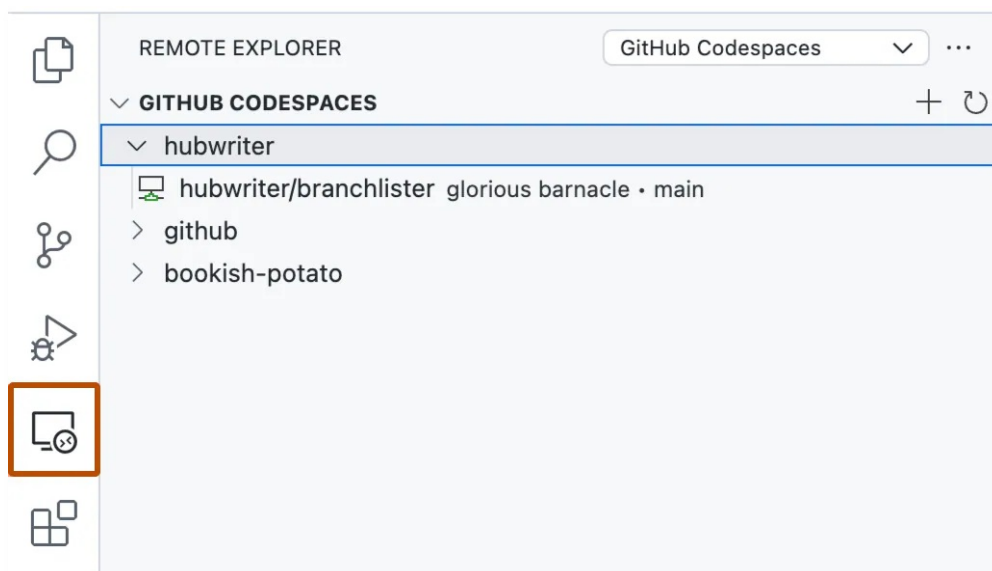
If you changed to a virtual machine with a different storage capacity (for example, from 32 GB to 64 GB), your codespace will be unavailable for a short time while the machine type is changed. If the codespace is currently active, it will automatically be stopped. When the change is complete, you'll be able to restart the codespace running on the new machine type.

If you changed to a virtual machine with the same storage capacity, the change will be applied the next time you restart the codespace. An active codespace will not be stopped automatically. For more information on restarting a codespace, see "Stopping and starting a codespace."

## Deleting a codespace in VS Code 🔗

You can delete codespaces from within VS Code when you are not currently working in a codespace.

**1** In VS Code, in the Activity Bar, click the Remote Explorer icon.



> **Note**: If the Remote Explorer is not displayed in the Activity Bar:
>
> a. Access the Command Palette. For example, by pressing `Shift` + `Command` + `P` (Mac) / `Ctrl` + `Shift` + `P` (Windows/Linux).
>
> b. Type: `details`.
>
> c. Click **Codespaces: Details**.

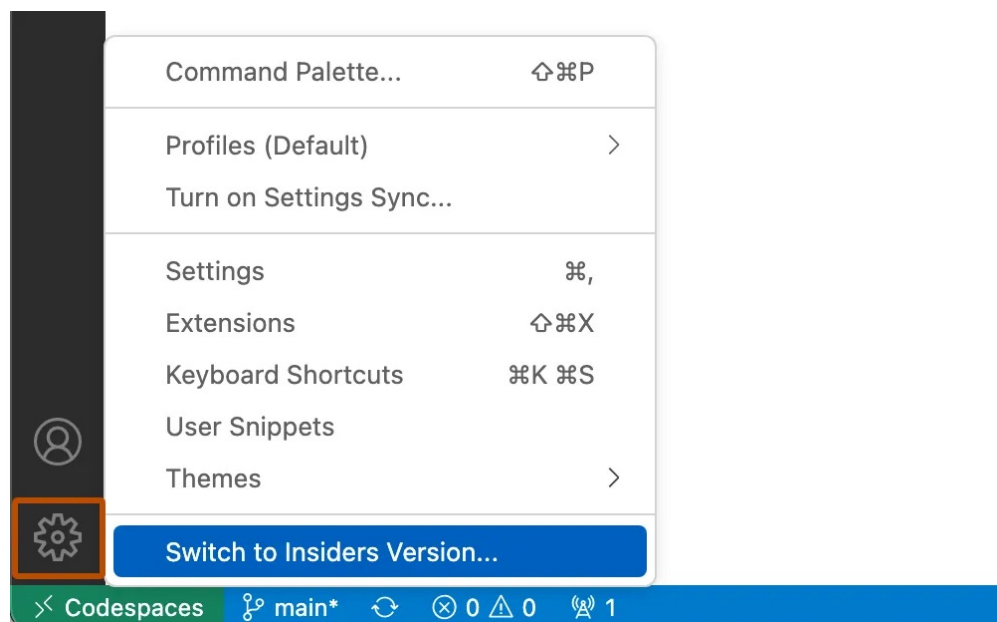**2** Under "GitHub Codespaces," right-click the codespace you want to delete.

**3** Click **Delete Codespace**.

## Switching to VS Code Insiders in the web client 🔗

If you are using the VS Code web client, you can switch to the Insiders version of the application. For more information about this version of VS Code, see Introducing the Insiders Build in the VS Code blog.

After you switch versions in a codespace, the web client will continue to use the Insiders version if you stop and restart the codespace. New codespaces that you create and open in the VS Code web client will also use the Insiders version.

1. In bottom left of the browser window that's displaying a codespace, click ⚙.

2. In the menu, select "Switch to Insiders Version."



3. Click **Reload**.

To switch back to the Stable version of VS Code, repeat the process but choose **Switch to Stable Version**. After you switch back, the codespace will continue to use the Stable version if you stop and restart the codespace. New codespaces that you create and open in the VS Code web client will also use the Stable version.

## Using the Insiders desktop application for Codespaces ⚭

To use GitHub Codespaces in the Insiders version of the VS Code desktop application, start or create your codespaces from within the VS Code Insiders application. For more information see "Creating a codespace in VS Code" and "Opening a codespace in VS Code" earlier in this article.

## Further reading ⚭

- "Using the Visual Studio Code Command Palette in GitHub Codespaces"
- "Using GitHub Copilot in GitHub Codespaces"