

# Configure GitHub Copilot settings in Visual Studio Code

6 minutes

GitHub Copilot natively integrates with Visual Studio Code, Visual Studio, and other developer environments.

GitHub Copilot uses two extensions for Visual Studio Code:

- GitHub Copilot: The GitHub Copilot extension supports code completions, including code suggestions generated using code comments.
- GitHub Copilot Chat: The GitHub Copilot Chat extension supports the AI peer programming tools, including chat conversations between the user and AI models.

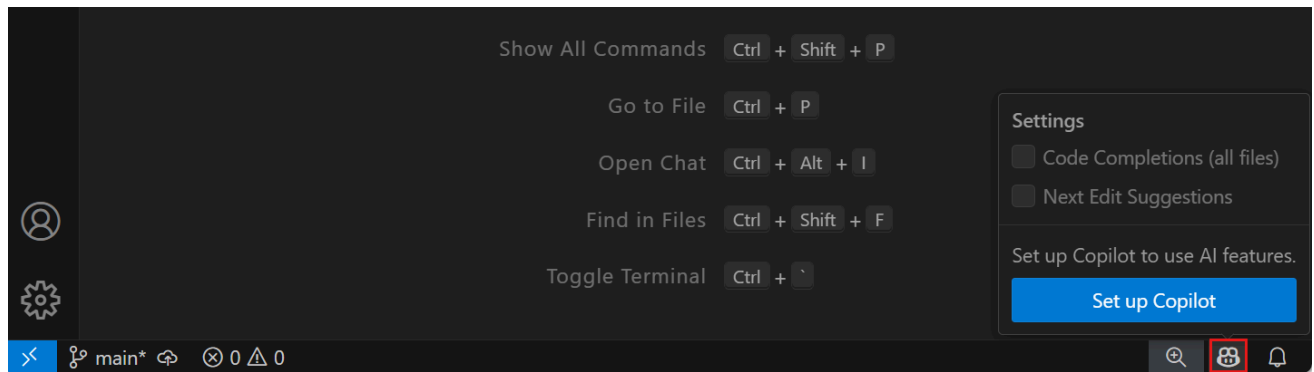
## Configure GitHub Copilot extensions in Visual Studio Code

When you use Visual Studio Code to set up GitHub Copilot, the two extensions are automatically installed with default settings. You don't need to download or install the extensions manually.

## Install the GitHub Copilot extensions in Visual Studio Code

Installing the GitHub Copilot extensions in Visual Studio Code is a straightforward process.

1. Open Visual Studio Code and ensure that you have the latest version installed.
2. Hover the mouse pointer over the GitHub Copilot icon on the Status Bar and then select **Set up Copilot**.



3. On the **Sign in to use Copilot** dialog, use the **Sign in** button to open the GitHub login page.

If you've already authenticated your GitHub account in Visual Studio Code, you may see **Use Copilot** rather than **Sign in**.

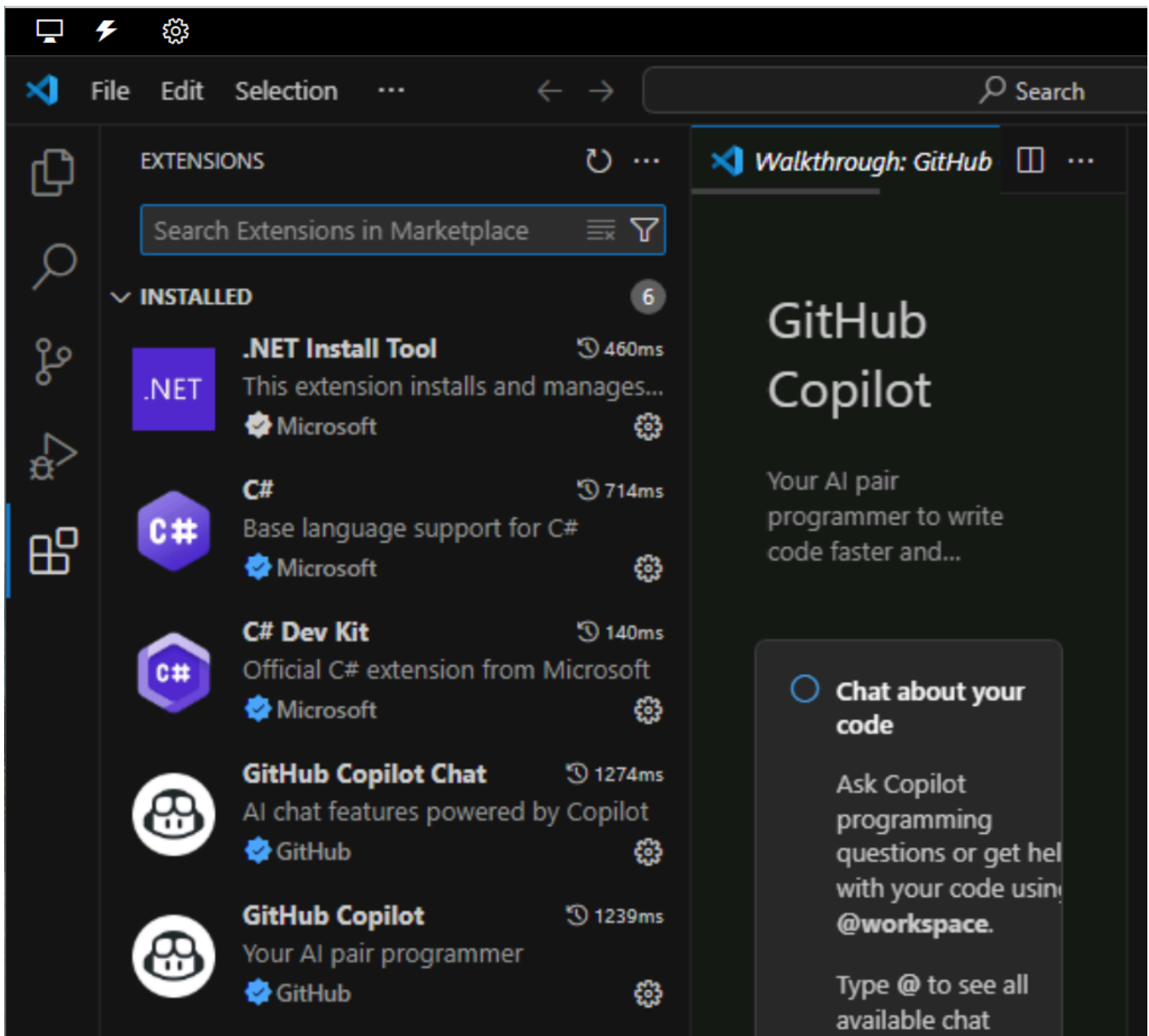
4. Follow the online instructions to authenticate your account and authorize access in Visual Studio Code.

#### ⓘ Note

If you don't already have a GitHub Copilot subscription, you'll be signed up for the GitHub Copilot Free plan.

The setup process uses a background operation to install the GitHub Copilot and GitHub Copilot Chat extensions in Visual Studio Code.

You can access the GitHub Copilot extensions from the Extensions view in Visual Studio Code.

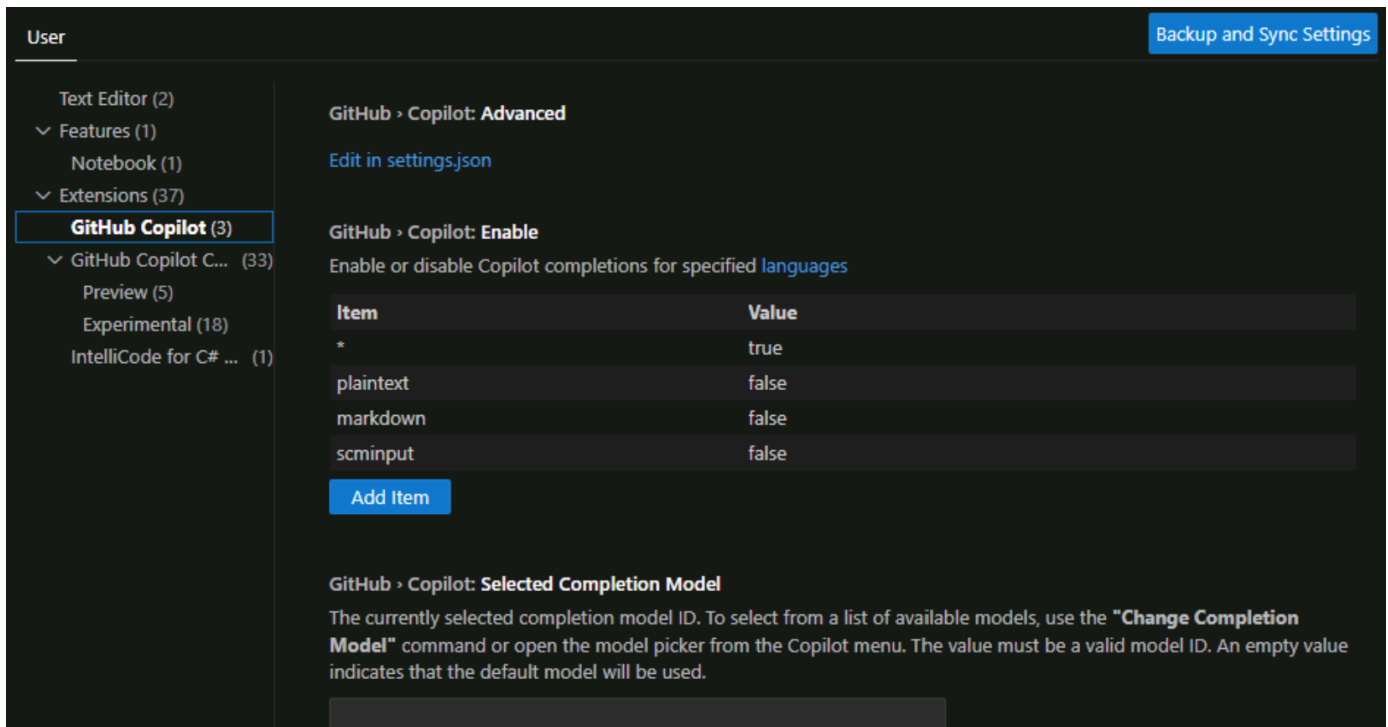


## Configure GitHub Copilot settings in Visual Studio Code

You can configure GitHub Copilot settings in Visual Studio Code to customize the behavior of the GitHub Copilot extensions. Each extension has its own settings.

You can access the Visual Studio Code settings by selecting **Manage** (the gear icon) in the lower left corner of the Visual Studio Code window, and then selecting **Settings**.

The GitHub Copilot extension settings are used to manage primary features, such as the supported programming languages and AI models.

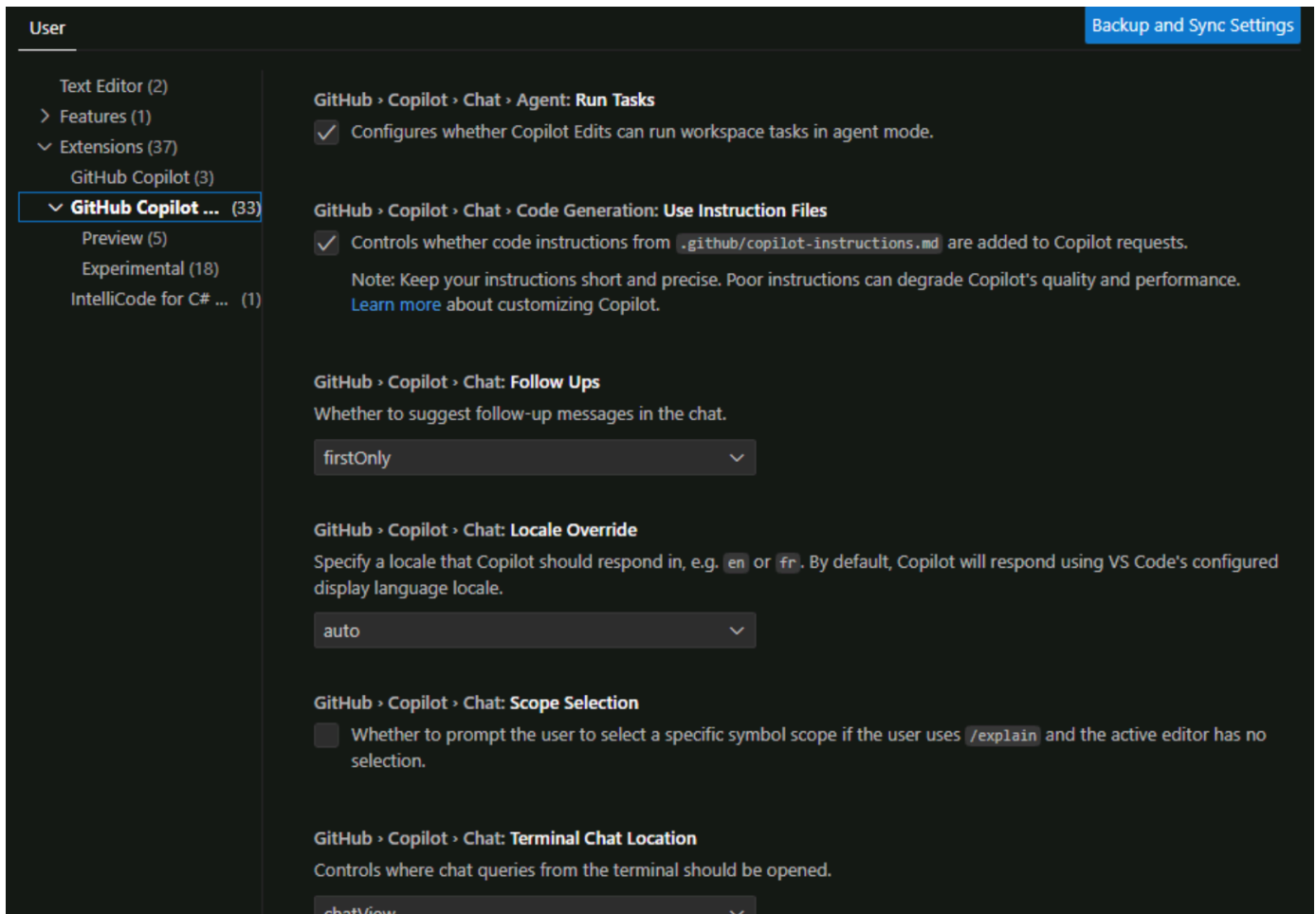


The screenshot shows the 'User' settings page for the GitHub Copilot extension. The left sidebar lists various settings categories, with 'GitHub Copilot (3)' selected. The main panel displays the 'GitHub > Copilot: Advanced' settings. At the top, there is a link to 'Edit in settings.json'. Below this, the 'GitHub > Copilot: Enable' section allows users to enable or disable Copilot completions for specified languages. A table lists the following settings:

Item	Value
*	true
plaintext	false
markdown	false
scminput	false

An 'Add Item' button is located below the table. The 'GitHub > Copilot: Selected Completion Model' section explains that the currently selected completion model ID is shown, and users can select from a list of available models using the 'Change Completion Model' command or the Copilot menu. The value must be a valid model ID, and an empty value indicates the default model will be used.

The GitHub Copilot Chat extension settings are used to manage chat interactions.



The screenshot shows the 'User' settings page for the GitHub Copilot Chat extension. The left sidebar lists various settings categories, with 'GitHub Copilot ... (33)' selected. The main panel displays the 'GitHub > Copilot > Chat' settings. The 'Agent: Run Tasks' section has a checked checkbox for 'Configures whether Copilot Edits can run workspace tasks in agent mode.' The 'Code Generation: Use Instruction Files' section has a checked checkbox for 'Controls whether code instructions from .github/copilot-instructions.md are added to Copilot requests.' A note below this section states: 'Note: Keep your instructions short and precise. Poor instructions can degrade Copilot's quality and performance. Learn more about customizing Copilot.' The 'Follow Ups' section has a dropdown menu set to 'firstOnly' with the description 'Whether to suggest follow-up messages in the chat.' The 'Locale Override' section has a dropdown menu set to 'auto' with the description 'Specify a locale that Copilot should respond in, e.g. en or fr. By default, Copilot will respond using VS Code's configured display language locale.' The 'Scope Selection' section has an unchecked checkbox for 'Whether to prompt the user to select a specific symbol scope if the user uses /explain and the active editor has no selection.' The 'Terminal Chat Location' section has a dropdown menu set to 'chatView' with the description 'Controls where chat queries from the terminal should be opened.'

# Summary

GitHub Copilot integrates with Visual Studio Code using two extensions: GitHub Copilot and GitHub Copilot Chat. You can configure the settings for each extension to customize the behavior of GitHub Copilot in Visual Studio Code.

---

**Next unit: Manage GitHub Copilot policies, features, and extensions for individuals**

[< Previous](#)[Next >](#)