

# Getting started with GitHub Copilot

## In this article

About GitHub Copilot and JetBrains IDEs

Prerequisites

Installing the GitHub Copilot extension in your JetBrains IDE

Seeing your first suggestion

Seeing alternative suggestions

Seeing multiple suggestions in a new tab

Generating code suggestions from comments

Enabling and disabling GitHub Copilot

About GitHub Copilot and Visual Studio

Prerequisites

Installing the GitHub Copilot extension in Visual Studio

Seeing your first suggestion

Seeing alternative suggestions

Generating code suggestions from comments

Enabling or disabling GitHub Copilot

About GitHub Copilot and Visual Studio Code

Prerequisites

Installing the GitHub Copilot extension in Visual Studio Code

Seeing your first suggestion

Seeing alternative suggestions

Seeing multiple suggestions in a new tab

Generating code suggestions from comments

Using a framework

Enabling or disabling GitHub Copilot

About GitHub Copilot and Vim/Neovim

Prerequisites

Installing the GitHub Copilot extension in Vim/Neovim on macOS

Installing the GitHub Copilot extension in Vim/Neovim on Windows

Installing the GitHub Copilot extension in Vim/Neovim on Linux

Learning to use GitHub Copilot in Vim/Neovim

About GitHub Copilot and Azure Data Studio

Prerequisites

Installing the GitHub Copilot extension in Azure Data Studio

Seeing your first suggestion

Seeing alternative suggestions

Generating code suggestions from comments

Enabling or disabling GitHub Copilot

Next steps

Further reading

---

You can start using GitHub Copilot by installing the extension in

your preferred environment.

Azure Data Studio    JetBrains IDEs    Vim/Neovim    Visual Studio    Visual Studio Code

GitHub Copilot can be managed through personal accounts with GitHub Copilot for Individuals or through organization accounts with GitHub Copilot for Business.

GitHub Copilot is free to use for verified students, teachers, and maintainers of popular open source projects. If you are not a student, teacher, or maintainer of a popular open source project, you can try GitHub Copilot for free with a one-time 30-day trial. After the free trial, you will need a paid subscription for continued use. For more information, see "[About billing for GitHub Copilot](#)."

Try GitHub Copilot ↗

## About GitHub Copilot and JetBrains IDEs

GitHub Copilot provides autocomplete-style suggestions from an AI pair programmer as you code. For more information, see "[About GitHub Copilot for Individuals](#)."

If you use a JetBrains IDE, you can view and incorporate suggestions from GitHub Copilot directly within the editor. This guide demonstrates how to use GitHub Copilot within a JetBrains IDE for macOS, Windows, or Linux.

## Prerequisites

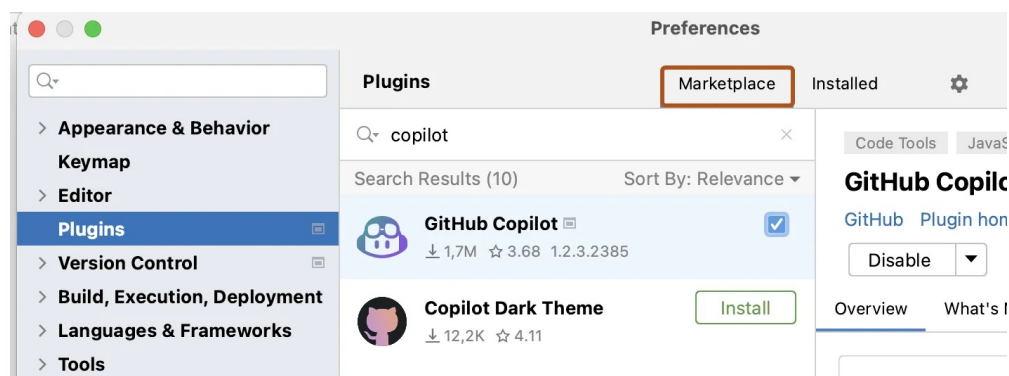
- To use GitHub Copilot you must have an active GitHub Copilot subscription. For more information, see "[About billing for GitHub Copilot](#)."
- To use GitHub Copilot in JetBrains, you must have a compatible JetBrains IDE installed. GitHub Copilot is compatible with the following IDEs:
  - IntelliJ IDEA (Ultimate, Community, Educational)
  - Android Studio
  - AppCode
  - CLion
  - Code With Me Guest
  - DataGrip
  - DataSpell
  - GoLand
  - JetBrains Client
  - MPS
  - PhpStorm
  - PyCharm (Professional, Community, Educational)
  - Rider
  - RubyMine
  - RustRover
  - WebStorm

For more information, see the [JetBrains IDEs](#) tool finder.

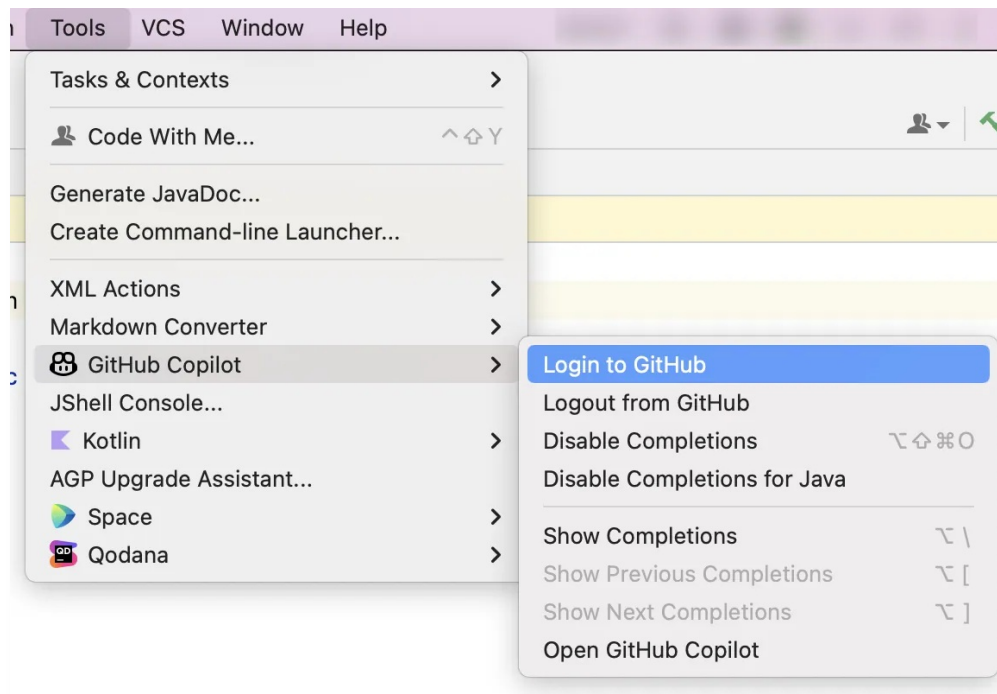
## Installing the GitHub Copilot extension in your JetBrains IDE

To use GitHub Copilot in a JetBrains IDE, you must install the GitHub Copilot extension. The following procedure will guide you through installation of the GitHub Copilot plugin in IntelliJ IDEA. Steps to install the plugin in another supported IDE may differ.

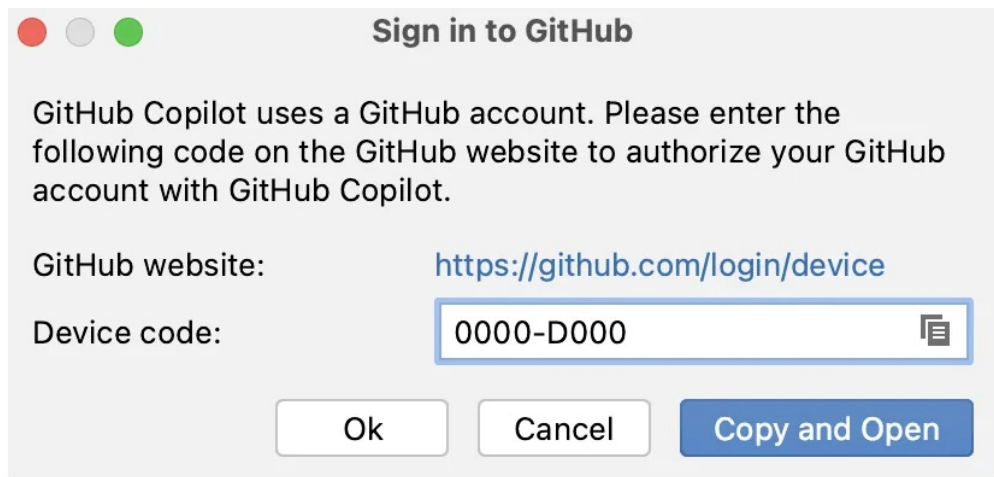
- 1 In your JetBrains IDE, under the **File** menu for Windows or under the name of your IDE for Mac (for example, **PyCharm** or **IntelliJ**), click **Settings** for Windows or **Preferences** for Mac.
- 2 In the left-side menu of the **Settings/Preferences** dialog box, click **Plugins**.
- 3 At the top of the **Settings/Preferences** dialog box, click **Marketplace**. In the search bar, search for **GitHub Copilot**, then click **Install**.



- 4 After GitHub Copilot is installed, click **Restart IDE**.
- 5 After your JetBrains IDE has restarted, click the **Tools** menu. Click **GitHub Copilot**, then click **Login to GitHub**.



- 6 In the "Sign in to GitHub" dialog box, to copy the device code and open the device activation window, click **Copy and Open**.



- 7 A device activation window will open in your browser. Paste the device code, then click **Continue**.
- 8 GitHub will request the necessary permissions for GitHub Copilot. To approve these permissions, click **Authorize GitHub Copilot Plugin**.
- 9 After the permissions have been approved, your JetBrains IDE will show a confirmation. To begin using GitHub Copilot, click **OK**.

## Seeing your first suggestion [🔗](#)

**Note:** If you have duplication detection enabled for GitHub Copilot, you may receive limited suggestions, or no suggestions, when using the code examples provided. As an alternative, you can start by typing your own code to see suggestions from GitHub Copilot. For more information on duplication detection, see "[Configuring GitHub Copilot settings on GitHub.com](#)."

GitHub Copilot provides suggestions for numerous languages and a wide variety of frameworks, but works especially well for Python, JavaScript, TypeScript, Ruby, Go, C# and C++. GitHub Copilot can also assist in query generation for databases. The following samples are in Java, but other languages will work similarly.

- 1 In your JetBrains IDE, create a new Java (\*.java) file.
- 2 In the Java file, create a class by typing `class Test`. GitHub Copilot will automatically suggest a class body in grayed text. The exact suggestion may vary.
- 3 To accept the suggestion, press `Tab`.
- 4 To prompt GitHub Copilot to suggest a function body, type the following line below the bracket of the `main` function. The exact suggestion may vary.

```
Java
```

```
int calculateDaysBetweenDates(
```

- 5 To accept the suggestion, press `Tab`.

GitHub Copilot will attempt to match the context and style of your code. You can always edit the suggested code.

## Seeing alternative suggestions [↗](#)

For any given input, GitHub Copilot may offer multiple suggestions. You can select which suggestion to use, or reject all suggestions.

- 1 In your JetBrains IDE, create a new Java (\*.java) file.
- 2 To prompt GitHub Copilot to show you a suggestion, type the following line in the Java file.

```
Java
```

```
int calculateDaysBetweenDates(
```

- 3 Optionally, you can see alternative suggestions, if any are available.

OS	See next suggestion	See previous suggestion
macOS	<code>Option + ]</code>	<code>Option + [</code>
Windows	<code>Alt + ]</code>	<code>Alt + [</code>
Linux	<code>Alt + ]</code>	<code>Alt + [</code>

- 4 To accept a suggestion, press `Tab`. To reject all suggestions, press `Esc`.

## Seeing multiple suggestions in a new tab [↗](#)

You may not want any of the initial suggestions GitHub Copilot offers. You can use a keyboard shortcut to prompt GitHub Copilot to show you multiple suggestions in a new tab.

- 1 In your JetBrains IDE, create a new Java (\*.java) file.
- 2 To prompt GitHub Copilot to show you a suggestion, type the following line in the Java file.

```
Java
```

```
int calculateDaysBetweenDates(
```

- 3 Open a new tab with multiple additional suggestions.
  - On macOS, press `Command + Shift + A`, then click **Open GitHub Copilot**, or press `Command + Shift + </kbd>` to open the new tab immediately.
  - On Windows or Linux, press `Ctrl + Enter`, then click **Open GitHub Copilot**.
- 4 To accept a suggestion, above the suggestion, click **Accept Solution**. To reject all suggestions, close the tab.

## Generating code suggestions from comments [↗](#)

You can describe something you want to do using natural language within a comment, and GitHub Copilot will suggest the code to accomplish your goal.

- 1 In your JetBrains IDE, create a new Java (\*.java) file.
- 2 To prompt GitHub Copilot to suggest an implementation of a function in the Java file, type the following lines.

```
Java

// find all images without alternate text
// and give them a red border
void process () {
```

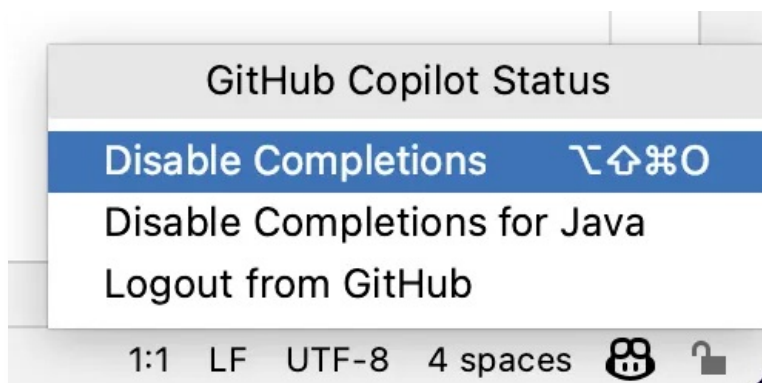
## Enabling and disabling GitHub Copilot [↗](#)

You can enable or disable GitHub Copilot for all languages, or for individual languages. The GitHub Copilot status icon in the bottom panel of your JetBrains IDE window indicates whether GitHub Copilot is enabled or disabled. When enabled, the icon is highlighted. When disabled, the icon is grayed out.

- 1 To enable or disable GitHub Copilot, click the status icon in the bottom panel of the JetBrains window.



- 2 If you are disabling GitHub Copilot, you will be asked whether you want to disable it globally, or for the language of the file you are currently editing.
  - To disable suggestions from GitHub Copilot globally, click **Disable Completions**.
  - To disable suggestions from GitHub Copilot for the specified language, click **Disable Completions for LANGUAGE**.



## About GitHub Copilot and Visual Studio [↗](#)

GitHub Copilot provides autocomplete-style suggestions from an AI pair programmer as you code. For more information, see "[About GitHub Copilot for Individuals](#)."

If you use Visual Studio, you can view and incorporate suggestions from GitHub Copilot directly within the editor. This guide demonstrates how to use GitHub Copilot within

## Prerequisites

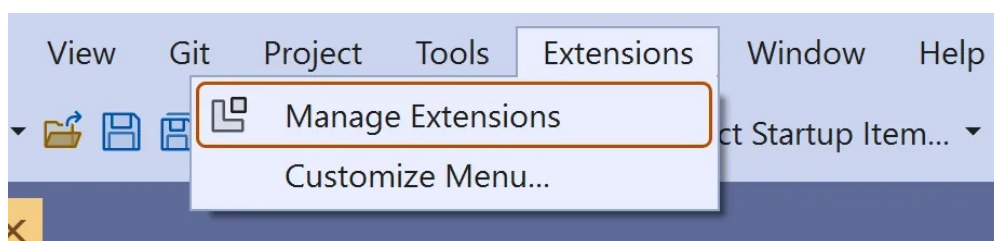
- To use GitHub Copilot you must have an active GitHub Copilot subscription. For more information, see "[About billing for GitHub Copilot](#)."
- To use GitHub Copilot in Visual Studio, you must have Visual Studio 2022 17.6 or later installed. For more information, see the [Visual Studio IDE](#) documentation.

**Note:** GitHub Copilot is not currently available for use with Visual Studio for Mac.

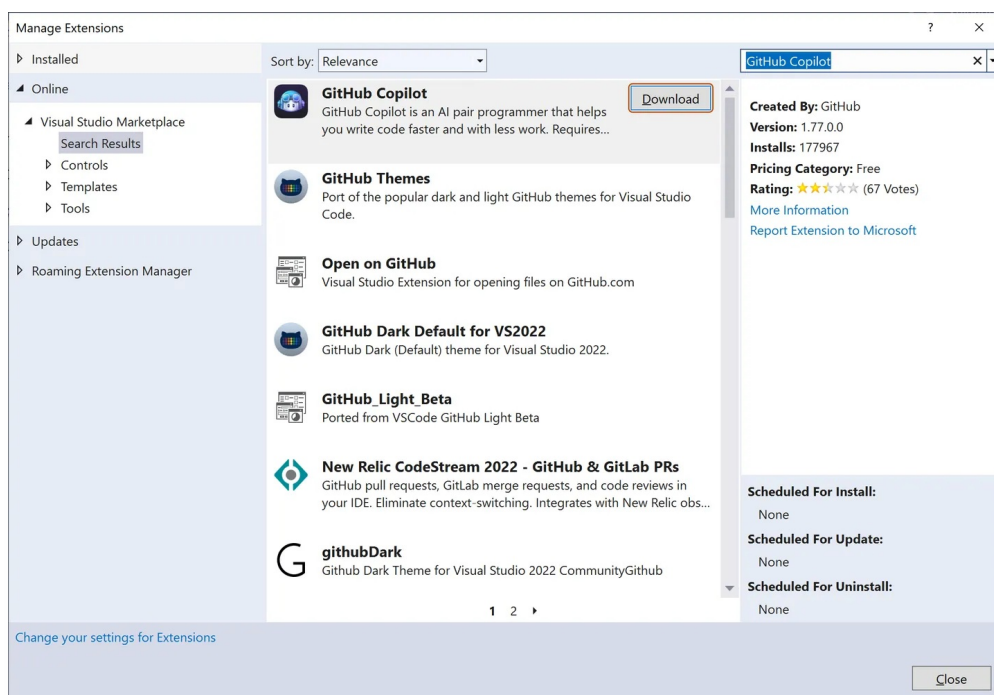
## Installing the GitHub Copilot extension in Visual Studio

To use GitHub Copilot, you must first install the GitHub Copilot extension.

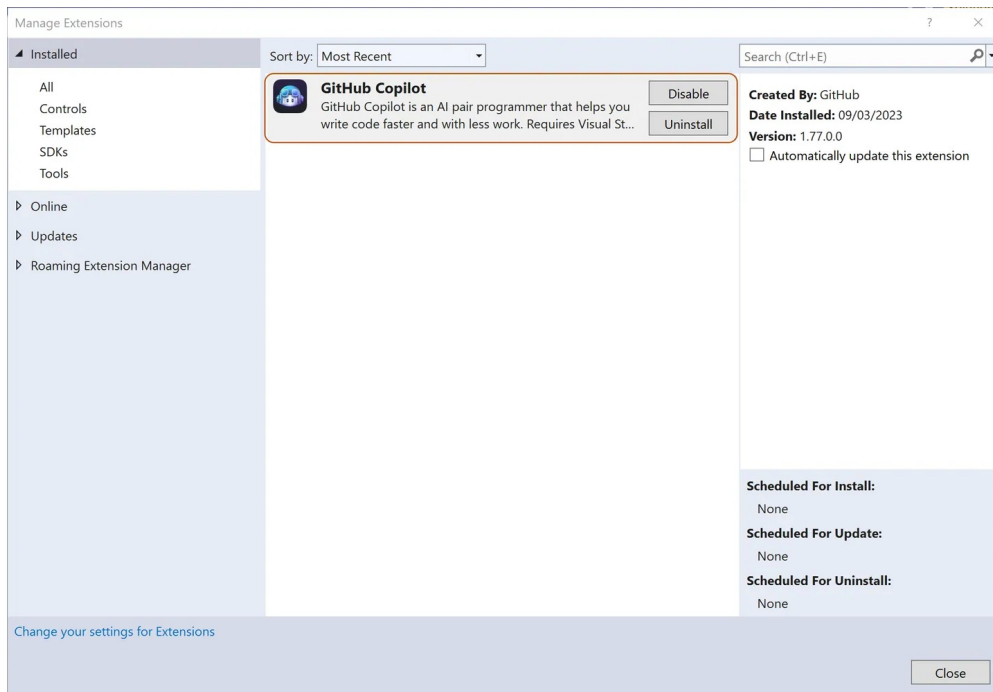
- 1 In the Visual Studio menu bar, click **Extensions**, then click **Manage Extensions**.



- 2 In the "Manage Extensions" window, click **Visual Studio Marketplace**, search for the GitHub Copilot extension, then click **Download**.



- 3 Close the "Manage Extensions" window, then exit and relaunch Visual Studio.
- 4 Optionally, to check that GitHub Copilot is installed and enabled, go back to **Manage Extensions**, click **Installed** to view your currently installed extensions, then click **GitHub Copilot** to see status information.



- 5 Open or create a new project in Visual Studio.
- 6 To enable GitHub Copilot, ensure you have added your GitHub account to Visual Studio. For more information, see [Work with GitHub accounts in Visual Studio](#) in the Microsoft documentation.

## Seeing your first suggestion [🔗](#)

**Note:** If you have duplication detection enabled for GitHub Copilot, you may receive limited suggestions, or no suggestions, when using the code examples provided. As an alternative, you can start by typing your own code to see suggestions from GitHub Copilot. For more information on duplication detection, see "[Configuring GitHub Copilot settings on GitHub.com](#)."

GitHub Copilot provides suggestions for numerous languages and a wide variety of frameworks, but works especially well for Python, JavaScript, TypeScript, Ruby, Go, C# and C++. GitHub Copilot can also assist in query generation for databases. The following samples are in C#, but other languages will work similarly.

- 1 In Visual Studio, create a new C# (\*.cs) file.
- 2 In the C# file, type the following function signature. GitHub Copilot will automatically suggest an entire function body in grayed text, as shown below. The exact suggestion may vary.

```
C#  
  
int CalculateDaysBetweenDates(  
    
```

- 3 To accept the suggestion, press `Tab`.

## Seeing alternative suggestions [🔗](#)

For any given input, GitHub Copilot may offer multiple suggestions. You can select which



suggestion to use, or reject all suggestions.

- 1 In Visual Studio, create a new C# (\*.cs) file.
- 2 In the C# file, type the following function signature. GitHub Copilot will show you a suggestion.

C#

```
int CalculateDaysBetweenDates(
```

- 3 If alternative suggestions are available, you can see these alternatives by pressing `Alt + .` (or `Alt + ,`).
- 4 Optionally, you can hover over the suggestion to see the GitHub Copilot command palette for choosing suggestions.
- 5 To accept a suggestion, press `Tab`. To reject all suggestions, press `Esc`.

## Generating code suggestions from comments [↗](#)

You can describe something you want to do using natural language within a comment, and GitHub Copilot will suggest the code to accomplish your goal.

- 1 In Visual Studio, create a new C# (\*.cs) file.
- 2 In the C# file, type the following comment. GitHub Copilot will suggest an implementation of the function.

C#

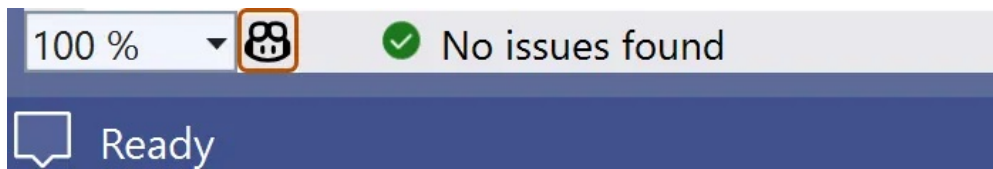
```
using System.Xml.Linq;  
  
var doc = XDocument.Load("index.xml");  
  
// find all images
```

- 3 To accept the suggestion, press `Tab`.

## Enabling or disabling GitHub Copilot [↗](#)

The GitHub Copilot status icon in the bottom panel of the Visual Studio window indicates whether GitHub Copilot is enabled or disabled. When enabled, the background color of the icon will match the color of the status bar. When disabled, it will have a diagonal line through it.

- 1 To enable or disable GitHub Copilot, click the GitHub Copilot icon in the bottom panel of the Visual Studio window.



- 2 If you are disabling GitHub Copilot, you will be asked whether you want to disable suggestions globally, or for the language of the file you are currently editing.
  - To disable suggestions from GitHub Copilot globally, click **Enable Globally**.
  - To disable suggestions from GitHub Copilot for the specified language, click **Enable for LANGUAGE**.

## About GitHub Copilot and Visual Studio Code [↗](#)

GitHub Copilot provides autocomplete-style suggestions from an AI pair programmer as you code. For more information, see "[About GitHub Copilot for Individuals](#)."

If you use Visual Studio Code, you can view and incorporate suggestions from GitHub Copilot directly within the editor. This guide demonstrates how to use GitHub Copilot within Visual Studio Code for macOS, Windows, or Linux.

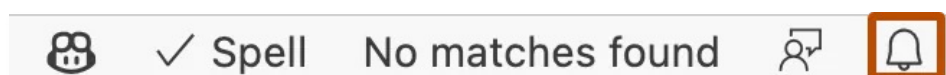
## Prerequisites [↗](#)

- To use GitHub Copilot you must have an active GitHub Copilot subscription. For more information, see "[About billing for GitHub Copilot](#)."
- To use GitHub Copilot in Visual Studio Code, you must have Visual Studio Code installed. For more information, see the [Visual Studio Code download page](#).

## Installing the GitHub Copilot extension in Visual Studio Code [↗](#)

To use GitHub Copilot, you must first install the GitHub Copilot extension.

- 1 In the Visual Studio Code Marketplace, go to the [GitHub Copilot extension](#) page and click **Install**.
- 2 A popup will appear, asking to open Visual Studio Code. Click **Open Visual Studio Code**.
- 3 In the "Extension: GitHub Copilot" tab in Visual Studio Code, click **Install**.
- 4 If you have not previously authorized Visual Studio Code in your GitHub account, you will be prompted to sign in to GitHub in Visual Studio Code.
  - If you have previously authorized Visual Studio Code for your account on GitHub, GitHub Copilot will be automatically authorized.
  - If you don't get the prompt to authorize, click the bell icon in the bottom panel of the Visual Studio Code window.



- 5 In your browser, GitHub will request the necessary permissions for GitHub Copilot. To approve these permissions, click **Authorize Visual Studio Code**.
- 6 To confirm the authentication, in Visual Studio Code, in the "Visual Studio Code" dialog box, click **Open**.

## Seeing your first suggestion

**Note:** If you have duplication detection enabled for GitHub Copilot, you may receive limited suggestions, or no suggestions, when using the code examples provided. As an alternative, you can start by typing your own code to see suggestions from GitHub Copilot. For more information on duplication detection, see "[Configuring GitHub Copilot settings on GitHub.com](#)."

GitHub Copilot provides suggestions for numerous languages and a wide variety of frameworks, but works especially well for Python, JavaScript, TypeScript, Ruby, Go, C# and C++. GitHub Copilot can also assist in query generation for databases. The following samples are in JavaScript, but other languages will work similarly.

- 1 In Visual Studio Code, create a new JavaScript (\*.js) file.
- 2 In the JavaScript file, type the following function header. GitHub Copilot will automatically suggest an entire function body in grayed text, as shown below. The exact suggestion may vary.

JavaScript



```
function calculateDaysBetweenDates(begin, end) {
```

- 3 To accept the suggestion, press `Tab`.

## Seeing alternative suggestions

For any given input, GitHub Copilot may offer multiple suggestions. You can select which suggestion to use, or reject all suggestions.

- 1 In Visual Studio Code, create a new JavaScript (\*.js) file.
- 2 In the JavaScript file, type the following function header. GitHub Copilot will show you a suggestion.

JavaScript



```
function calculateDaysBetweenDates(begin, end) {
```

- 3 Optionally, you can see alternative suggestions, if any are available.


OS	See next suggestion	See previous suggestion
macOS	<code>Option (⌘) or Alt + ]</code>	<code>Option (⌘) or Alt + [</code>
Windows	<code>Alt + ]</code>	<code>Alt + [</code>
Linux	<code>Alt + ]</code>	<code>Alt + [</code>

- Alternatively, you can hover over the suggestion to see the GitHub Copilot command palette for choosing suggestions.
- To accept a suggestion, press `Tab`. To reject all suggestions, press `Esc`.

## Seeing multiple suggestions in a new tab [↗](#)

You may not want any of the initial suggestions GitHub Copilot offers. You can use a keyboard shortcut to prompt GitHub Copilot to show you multiple suggestions in a new tab.

- In Visual Studio Code, create a new JavaScript (`*.js`) file.
- In the JavaScript file, type the following function header. GitHub Copilot will show you a suggestion.

JavaScript 


```
function calculateDaysBetweenDates(begin, end) {
```

- To open a new tab with multiple additional options, press `Ctrl + Enter`.
- To accept a suggestion, above the suggestion, click **Accept Solution**. To reject all suggestions, close the tab.

## Generating code suggestions from comments [↗](#)

You can describe something you want to do using natural language within a comment, and GitHub Copilot will suggest the code to accomplish your goal.

- In Visual Studio Code, create a new JavaScript (`*.js`) file.
- In the JavaScript file, type the following comment. GitHub Copilot will suggest an implementation of the function.

JavaScript 

```
// find all images without alternate text  
// and give them a red border  
function process() {
```

## Using a framework [↗](#)

You can also use GitHub Copilot to generate suggestions for APIs and frameworks. The following example uses GitHub Copilot to create a simple Express server that returns the current time.

- In Visual Studio Code, create a new JavaScript (`*.js`) file.
- In the JavaScript file, type the following comment and then press `Enter`. GitHub

Copilot will suggest an implementation of the Express app.

```
JavaScript

// Express server on port 3000
```

3 To accept each line, press `Tab`, then `Enter`.

4 Type the following comment and then press `Enter`. GitHub Copilot will suggest an implementation for the default handler.

```
JavaScript

// Return the current time
```

5 To accept each line, press `Tab`.

## Enabling or disabling GitHub Copilot [↗](#)

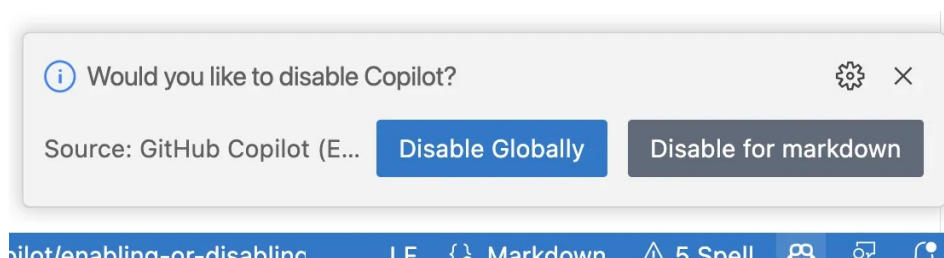
You can enable or disable GitHub Copilot from within Visual Studio Code. The GitHub Copilot status icon in the bottom panel of the Visual Studio Code window indicates whether GitHub Copilot is enabled or disabled. When enabled, the background color of the icon will match the color of the status bar. When disabled, the background color of the icon will contrast with the color of the status bar.

1 To enable or disable GitHub Copilot, click the status icon in the bottom panel of the Visual Studio Code window.



2 If you are disabling GitHub Copilot, you will be asked whether you want to disable suggestions globally, or for the language of the file you are currently editing.

- To disable suggestions from GitHub Copilot globally, click **Disable Globally**.
- To disable suggestions from GitHub Copilot for the specified language, click **Disable for LANGUAGE**.



## About GitHub Copilot and Vim/Neovim [↗](#)

GitHub Copilot provides autocomplete-style suggestions from an AI pair programmer as you code. For more information, see "[About GitHub Copilot for Individuals](#)."

If you use Vim/Neovim, you can view and incorporate suggestions from GitHub Copilot directly within the editor.

## Prerequisites

- To use GitHub Copilot you must have an active GitHub Copilot subscription. For more information, see "[About billing for GitHub Copilot](#)."
- To use GitHub Copilot in Vim/Neovim you must have Vim version 9.0.0185 / Neovim version 0.6 or above and Node.js version 17 or below installed. For more information, see the [Vim](#) / [Neovim documentation](#) and the [Node.js website](#).

## Installing the GitHub Copilot extension in Vim/Neovim on macOS

- 1 GitHub recommends that you install the GitHub Copilot plugin with Vim/Neovim's built-in plugin manager. Alternatively, you can use a plugin manager of your choice to install `github/copilot.vim`.

- To install GitHub Copilot with Vim's built-in plugin manager, enter the following command in Terminal:

```
git clone https://github.com/github/copilot.vim \  
~/.vim/pack/github/start/copilot.vim
```

- To install GitHub Copilot with Neovim's built-in plugin manager, enter the following command in Terminal:

```
git clone https://github.com/github/copilot.vim \  
~/.config/nvim/pack/github/start/copilot.vim
```

- 2 To configure GitHub Copilot, open Vim/Neovim and enter the following command.

```
:Copilot setup
```

- 3 Enable GitHub Copilot in your Vim/Neovim configuration, or with the Vim/Neovim command.

```
:Copilot enable
```

## Installing the GitHub Copilot extension in Vim/Neovim on Windows

- 1 GitHub recommends that you install the GitHub Copilot plugin with Vim/Neovim's built-in plugin manager. Alternatively, you can use a plugin manager of your choice to install `github/copilot.vim`.

- To install GitHub Copilot with Vim's built-in plugin manager, enter the following command in Git Bash:

```
git clone https://github.com/github/copilot.vim.git \  
$HOME/vimfiles/pack/github/start/copilot.vim
```

- To install GitHub Copilot with Neovim's built-in plugin manager, enter the following command in Git Bash:

```
git clone https://github.com/github/copilot.vim.git \  
$HOME/AppData/Local/nvim/pack/github/start/copilot.vim
```

- 2 To configure GitHub Copilot, open Vim/Neovim and enter the following command.

```
:Copilot setup
```

- 3 Enable GitHub Copilot in your Vim/Neovim configuration, or with the Vim/Neovim command.

```
:Copilot enable
```

## Installing the GitHub Copilot extension in Vim/Neovim on Linux [↗](#)

- 1 GitHub recommends that you install the GitHub Copilot plugin with Vim/Neovim's built-in plugin manager. Alternatively, you can use a plugin manager of your choice to install `github/copilot.vim`.

- To install GitHub Copilot with Vim's built-in plugin manager, enter the following command:

```
git clone https://github.com/github/copilot.vim \  
~/.vim/pack/github/start/copilot.vim
```

- To install GitHub Copilot with Neovim's built-in plugin manager, enter the following command:

```
git clone https://github.com/github/copilot.vim \  
~/.config/nvim/pack/github/start/copilot.vim
```

- 2 To configure GitHub Copilot, open Vim/Neovim and enter the following command.

```
:Copilot setup
```

- 3 Enable GitHub Copilot in your Vim/Neovim configuration, or with the Vim/Neovim command.

```
:Copilot enable
```

## Learning to use GitHub Copilot in Vim/Neovim [↗](#)

For guidance on using GitHub Copilot in Vim/Neovim, you can view the plugin documentation. To see the documentation, open Vim/Neovim and run the following command:

## About GitHub Copilot and Azure Data Studio [↗](#)

GitHub Copilot provides autocomplete-style suggestions from an AI pair programmer as you code. For more information, see "[About GitHub Copilot for Individuals](#)."

If you use Azure Data Studio, you can view and incorporate suggestions from GitHub Copilot directly within the editor. This guide demonstrates how to use GitHub Copilot within Azure Data Studio for macOS, Windows, or Linux.

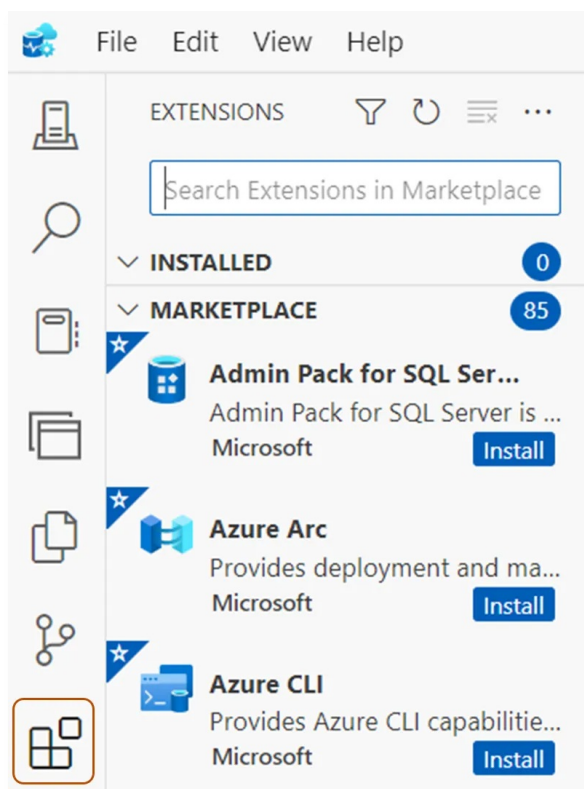
## Prerequisites [↗](#)

- To use GitHub Copilot you must have an active GitHub Copilot subscription. For more information, see "[About billing for GitHub Copilot](#)."
- To use GitHub Copilot in Azure Data Studio, you must have Azure Data Studio version 1.44.0 or later installed. For more information, see the [Azure Data Studio download page](#) in the Azure Data Studio documentation.

## Installing the GitHub Copilot extension in Azure Data Studio [↗](#)

To use GitHub Copilot, you must first install the GitHub Copilot extension.

- 1 In Azure Data Studio, click the **Extensions** icon in the left-side menu.



- 2 In the "Extensions" tab, search for **GitHub Copilot** and then click **Install**.
- 3 If a popup window in Azure Data Studio prompts you to sign in to use GitHub Copilot, click **Sign in to GitHub** and follow the instructions on screen.
  - If you have previously authorized Azure Data Studio for your account on GitHub,



GitHub Copilot will be automatically authorized.

- If you don't get the prompt to authorize, you can view notifications by clicking the bell icon in the bottom panel of the Azure Data Studio window.

- 4 If you are following the authorization steps, in your browser, GitHub will request the necessary permissions for GitHub Copilot. To approve these permissions, click **Authorize Azure Data Studio**.

## Seeing your first suggestion

**Note:** If you have duplication detection enabled for GitHub Copilot, you may receive limited suggestions, or no suggestions, when using the code examples provided. As an alternative, you can start by typing your own code to see suggestions from GitHub Copilot. For more information on duplication detection, see "[Configuring GitHub Copilot settings on GitHub.com](#)."

GitHub Copilot can provide you with inline suggestions as you create SQL databases in Azure Data Studio. For example, if you're writing a query that joins two tables, Copilot may suggest the join condition from columns in the open editor, other files in the workspace, and common syntax patterns.

- 1 In Azure Data Studio, create a new SQL file.
- 2 In the SQL file, type the following query. GitHub Copilot will automatically suggest a join condition in grayed text. The exact suggestion may vary.

SQL 


```
SELECT [UserId], [Red], [Orange], [Yellow], [Green], [Blue], [Purple],  
[Rainbow]  
FROM [Tag].[Scoreboard]  
INNER JOIN
```

- 3 To accept the suggestion, press `Tab`.

## Seeing alternative suggestions

For some suggestions, GitHub Copilot may provide multiple alternatives. You can select which suggestion you want to use, or reject all suggestions.

- 1 In Azure Data Studio, create a new SQL file.
- 2 In the SQL file, type the following query. GitHub Copilot will show you a suggestion.

SQL 

```
SELECT [UserId], [Red], [Orange], [Yellow], [Green], [Blue], [Purple],  
[Rainbow]  
FROM [Tag].[Scoreboard]  
INNER JOIN
```

- 3 Optionally, you can see alternative suggestions, if any are available.

OS

See next suggestion


See previous suggestion

Windows	Alt + [	Alt + ]
Linux	Alt + [	Alt + ]
macOS	Option + [	Option + ]

## Generating code suggestions from comments [↗](#)

You can describe something you want to do using natural language within a comment, and GitHub Copilot will suggest the code to accomplish your goal.

- 1 In Azure Data Studio, create a new SQL file.
- 2 In the SQL file, type the following query and comment. GitHub Copilot will suggest an implementation of the query.

SQL 

```

SELECT TokenColor, COUNT(UserID) AS UserCount
FROM Tag.Users
GROUP BY TokenColor
-- pivot that query on tokencolor for Purple, Blue, Green, Yellow, Orange,
Red
-- and rename the columns to match the colors
SELECT [Purple], [Blue], [Green], [Yellow], [Orange], [Red]
```

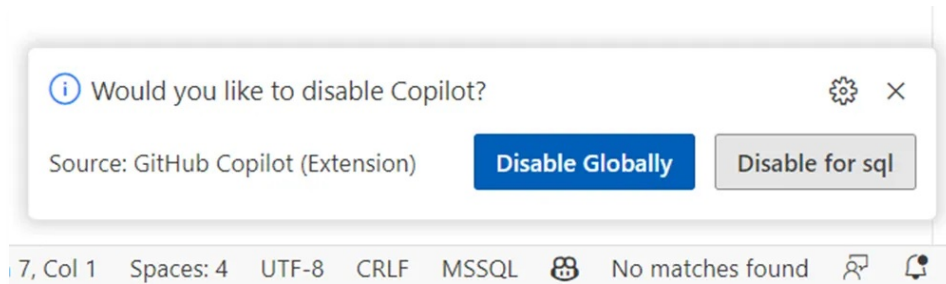
## Enabling or disabling GitHub Copilot [↗](#)

You can enable or disable GitHub Copilot from within Azure Data Studio. The GitHub Copilot status icon in the bottom panel of the Azure Data Studio window indicates whether GitHub Copilot is enabled or disabled. When enabled, the background color of the icon will match the color of the status bar. When disabled, the background color of the icon will contrast with the color of the status bar.

- 1 To enable or disable GitHub Copilot, click the status icon in the bottom panel of the Azure Data Studio window.



- 2 If you are disabling GitHub Copilot, you will be asked whether you want to disable suggestions globally, or for the language of the file you are currently editing.
  - To disable suggestions from GitHub Copilot globally, click **Disable Globally**.
  - To disable suggestions from GitHub Copilot for the specified language, click **Disable for LANGUAGE**.



## Next steps [↗](#)

You successfully installed GitHub Copilot and received your first suggestion, but that's just the beginning! Here are some helpful resources for taking your next steps with GitHub Copilot.

- [GitHub Copilot](#): See practical examples of how GitHub Copilot can help you work.
- [Configuring GitHub Copilot](#): These guides provide details on how to configure GitHub Copilot to your personal preferences.
- [Managing billing for GitHub Copilot](#): Learn more about billing for GitHub Copilot.
- [Troubleshooting GitHub Copilot](#): These guides provide information for troubleshooting GitHub Copilot.

## Further reading [↗](#)

- [The GitHub Copilot website](#)
- [About GitHub Copilot for Individuals](#)

### Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)