**Debug with GitHub Copilot**

GitHub Copilot can help improve your debugging workflow in Visual Studio Code. Copilot can assist with the setup of the debug configuration for your project and provide suggestions for fixing issues discovered during debugging. This article gives an overview of how to use Copilot for debugging applications in VS Code.

Copilot can help with the following debugging tasks:

* **Configure debug settings**: generate and customize launch configurations for your project.
* **Start a debugging session**: use copilot-debug to start a debugging session from the terminal.
* **Fix issues**: receive suggestions for fixing issues discovered during debugging.

**Set up debug configuration with Copilot**

VS Code uses the launch.json file to store debug configuration. Copilot can help you create and customize this file to set up debugging for your project.

1. Open the Chat view (Ctrl+Alt+I).
2. Enter the /startDebugging command.
3. Follow Copilot's guidance to set up debugging for your project.

Alternatively, you can use a natural language prompt like:

* "Create a debug configuration for a Django app"
* "Set up debugging for a React Native app"
* "Configure debugging for a Flask application"

**Start debugging with Copilot**

The copilot-debug terminal command simplifies the process of configuring and starting a debugging session. Prefix the command you'd use for starting your application with copilot-debug to have Copilot automatically configure and start a debugging session.

1. Open the integrated terminal (Ctrl+`).
2. Enter copilot-debug followed by your application's start command. For example:

Bash

copilot-debug node app.js

or

Bash

copilot-debug python manage.py

1. Copilot launches a debugging session for your application. You can now use the built-in debugging features in VS Code.

**Fix coding issues with Copilot**

You can use Copilot Chat to help you fix coding issues or improve your code.

**Use chat prompts**

1. Open your application code file.
2. Open one of these views:
   * Chat view (Ctrl+Alt+I)
   * Inline Chat (Ctrl+I)
3. Enter a prompt like:
   * "/fix"
   * "Fix this #selection"
   * "Validate input for this function"
   * "Refactor this code"
   * "Improve the performance of this code"

**Use editor smart actions**

To fix coding issues for your application code without writing a prompt, you can use the editor smart actions.

1. Open your application code file.
2. Select the code you want to fix.
3. Right-click and select **Generate Code** > **Fix**.

VS Code provides a code suggestion to fix the code.

1. Optionally, refine the generated code by providing additional context in the chat prompt.

**Next steps**

* Explore [general debugging features in VS Code](https://code.visualstudio.com/docs/debugtest/debugging).
* Learn more about [Copilot in VS Code](https://code.visualstudio.com/docs/copilot/overview).