# Introduction

3 minutes

Unit testing is a crucial aspect of software development that ensures the functionality of individual components within a system.

This module introduces how to generate unit tests with GitHub Copilot and GitHub Copilot Chat in Visual Studio Code. The module focuses on using the Chat view, inline chat, and code completions to create unit tests for the xUnit testing framework. Visual Studio Code and the C# Dev Kit extension are used to create a unit test project and manage unit tests.

Imagine you're a software developer working on a large codebase. Your team is tasked with ensuring code reliability. You determine that unit tests are needed for most of the codebase. However, creating unit tests manually can be time-consuming and error-prone. You need a tool that helps you develop unit tests quickly and accurately. The tool should also help identify edge cases and boundary conditions. You hear that GitHub Copilot can accelerate the development of unit tests and help to identify edge cases. You're looking forward to developing unit tests more quickly and accurately using GitHub Copilot.

The topics covered in this module include:

- Using GitHub Copilot and Visual Studio Code for unit testing.
- Generate unit tests with GitHub Copilot's Generate Tests smart action.
- Creating unit tests with GitHub Copilot Inline Chat.
- Creating unit tests with GitHub Copilot Chat Modes.
- Developing unit tests with GitHub Copilot in C#.

After completing this module, you'll be able to:

- Explain how GitHub Copilot Chat supports various testing tasks, including unit, integration, and end-to-end tests.
- Use the Generate Tests smart action to create unit tests for specific code blocks or entire files.
- Leverage the Inline Chat feature to generate and refine unit tests directly within the code editor.
- Utilize the Chat view modes—Ask, Edit, and Agent—to create, update, and automate unit test workflows.
- Apply GitHub Copilot's capabilities to streamline the development of unit tests for a C# application in Visual Studio Code.

> **Important**
>
> To complete this GitHub Copilot training, you must have an active subscription for GitHub Copilot in your personal GitHub account (includes the GitHub Copilot Free plan), or you must be assigned to a subscription managed by an organization or enterprise. Module activities may include GitHub Copilot suggestions that match public code. If you're a member of an organization on GitHub Enterprise Cloud who has been assigned a GitHub Copilot subscription through your organization, the setting for suggestions matching public code may be inherited from your organization or enterprise. If your account blocks suggestions that match public code, module activities may not work as expected.

## Next unit: Examine the unit testing tools and environment

Next >