

MS-AZ2007: ACCELERATE APP DEVELOPMENT BY USING GITHUB COPILOT



DURATION	LEVEL	TECHNOLOGY	DELIVERY METHOD	TRAINING CREDITS
1 Day	Intermediate	Microsoft 365	Instructor-led	NA

INTRODUCTION

Learn how to accelerate app development by using GitHub Copilot and GitHub Copilot Chat in a Visual Studio Code environment.

AUDIENCE PROFILE

This course is designed for experienced developers who want to enhance their productivity and coding efficiency using AI-powered tools.

PREREQUISITES

Before attending this course, delegates must have:

- An active subscription for GitHub Copilot is required for either a personal GitHub account or a GitHub account managed by an organization or enterprise.
- One or more years of code develop experience is recommended.
- Some experience developing C# applications in Visual Studio Code with the C# Dev Kit extension is recommended.

COURSE OBJECTIVES

After completing this course, you will be able to:

- Understand the features and advantages of GitHub Copilot: Learn about the capabilities and benefits of using GitHub Copilot in your development workflow.
- Install and configure GitHub Copilot extensions for Visual Studio Code: Get hands-on experience with setting up GitHub Copilot in your coding environment.
- Generate code and project documentation: Use GitHub Copilot to create code snippets, project documentation, and inline code comments.
- Develop code features and manage suggestions: Leverage GitHub Copilot to develop new code features and manage code suggestions effectively.
- Create unit tests: Utilize GitHub Copilot to generate unit tests for various scenarios, ensuring code quality and reliability.
- Implement code improvements: Apply GitHub Copilot tools to enhance code quality, performance, and security.

COURSE CONTENT

Module 1: Get started with GitHub Copilot

This module introduces developers to the GitHub Copilot products, the benefits that GitHub Copilot provides to developers, the GitHub Copilot and GitHub Copilot Chat product features, and the GitHub Copilot extensions for Visual Studio Code.

Lessons

- Introduction
- Examine AI tools from GitHub, OpenAI, and Microsoft
- Examine GitHub Copilot tools, benefits, and limitations
- Exercise - Install GitHub Copilot extensions for Visual Studio Code
- Examine the autocomplete features of the GitHub Copilot extension

- Examine the AI assistance features of the GitHub Copilot Chat extension
 - Exercise - Configure GitHub Copilot extensions for Visual Studio Code
 - Knowledge check
 - Summary
- By the end of this module, you'll be able to:
- Explain what GitHub Copilot is and the advantages it provides.
 - Install the GitHub Copilot extensions for Visual Studio Code.
 - Explain the features of the GitHub Copilot extensions for Visual Studio Code and how to use them.

- Configure the GitHub Copilot extensions for Visual Studio Code.

Module 2: Generate documentation using GitHub Copilot tools

This module explores the generation of code explanations, project documentation, and inline code comment documentation using the GitHub Copilot Chat extension for Visual Studio Code.

Lessons

- Introduction
- Examine the "document" and "explain" features of GitHub Copilot Chat
- Exercise - Generate code explanations using GitHub Copilot Chat

- Exercise - Generate project documentation by using GitHub Copilot Chat
- Exercise - Generate inline code documentation by using GitHub Copilot Chat
- Exercise - Complete the "code documentation" challenge
- Review the "code documentation" solution
- Knowledge check
- Summary

By the end of this module, you'll be able to:

- Generate explanations of code blocks, code files, and entire code workspaces using the GitHub Copilot Chat extension for Visual Studio Code.
- Generate code project documentation using the GitHub Copilot Chat extension for Visual Studio Code.
- Generate inline code documentation using the GitHub Copilot Chat extension for Visual Studio Code.

Module 3: Develop code features using GitHub Copilot tools

This module explores using GitHub Copilot and GitHub Copilot Chat suggestions to create new code. Autocompletion and code update suggestions are generated, managed, and implemented using the GitHub Copilot extensions for Visual Studio Code.

Lessons

- Introduction
- Examine the code development features of GitHub Copilot
- Examine GitHub Copilot best practices
- Exercise - Create code by using code line completions
- Exercise - Create code by using GitHub Copilot Inline Chat
- Exercise - Complete the create new code challenge
- Review the create new code solution
- Exercise - Complete the code logic challenge
- Review the code logic solution
- Exercise - Convert code from one programming language to another
- Knowledge check
- Summary

By the end of this module, you'll be able to:

- Generate autocompletion suggestions based on code and code comments using the GitHub Copilot extension for Visual Studio Code.
- Manage autocompletion suggestions by accepting, partially accepting, and dismissing suggestions.

- Create chat prompts/questions that communicate your intent by using a combination of chat participants, slash commands, chat variables, and natural language text.
- Create new code using the Chat View, Inline Chat, Quick Chat, and Smart Actions features provided by the GitHub Copilot Chat extension for Visual Studio Code.
- Manage code update suggestions by accepting, partially accepting, editing, and discarding the suggestions generated by GitHub Copilot Chat.

Module 4: Develop unit tests using GitHub Copilot tools

This module explores using GitHub Copilot and GitHub Copilot Chat to create unit tests. Exercises provide practical experience creating unit test projects and running unit tests in Visual Studio Code.

Lessons

- Introduction
- Examine the unit testing tools and environment
- Exercise - Create unit tests by using GitHub Copilot Chat
- Exercise - Create unit tests for specific conditions by using GitHub Copilot
- Exercise - Complete the "create unit tests" challenge
- Review the "create unit tests" solution
- Knowledge check
- Summary

By the end of this module, you'll be able to:

- Create unit tests using the GitHub Copilot and GitHub Copilot Chat extensions for Visual Studio Code.
- Create unit tests that target edge cases and specific conditions using the GitHub Copilot and GitHub Copilot Chat extensions for Visual Studio Code.
- Use Visual Studio Code, the .NET SDK, and the C# Dev Kit extension to create a test project and verify that your unit tests build and run successfully.

Module 5: Implement code improvements using GitHub Copilot tools

This module explores using GitHub Copilot Chat to develop improvements for an existing codebase. Exercises provide practical experience implementing GitHub Copilot suggestions that improve code quality, reliability, performance, and security.

Lessons

- Introduction
- Examine GitHub Copilot support for code improvements
- Exercise - Improve code quality by using GitHub Copilot Chat
- Exercise - Improve code reliability and performance by using GitHub Copilot Chat
- Exercise - Improve code security by using GitHub Copilot Chat
- Exercise - Complete the "app improvement" challenge
- Review the "app improvement" solution
- Knowledge check
- Summary

By the end of this module, you'll be able to:

- Develop prompts/questions for GitHub Copilot Chat that help you to improve the code in your existing code projects.
- Implement code update suggestions from GitHub Copilot Chat that improve existing code in the areas of code quality, reliability, performance, and security.

Module 6: Guided project - Accelerate app development using GitHub Copilot tools

This Module explores using GitHub Copilot to accelerate development of an end-to-end project. GitHub Copilot is used to explain unfamiliar code, generate project documentation, develop a new app feature, develop unit tests, and implement improvements.

Lessons

- Introduction
- Prepare the development environment
- Exercise - Analyze and document code using GitHub Copilot tools
- Exercise - Develop code features using GitHub Copilot tools
- Exercise - Develop unit tests using GitHub Copilot tools
- Exercise - Refactor and improve code sections using GitHub Copilot tools
- Knowledge check
- Summary

By the end of this module, you'll be able to:

- Explain an unfamiliar codebase using GitHub Copilot tools.
- Generate README project documentation using GitHub Copilot tools.
- Develop a new application feature using GitHub Copilot tools.
- Create unit tests using GitHub Copilot tools.

- | | |
|---|---|
| – Implement code quality, reliability, performance, and | security improvements using GitHub Copilot tools. |
|---|---|

ASSOCIATED CERTIFICATIONS & EXAM

There is no Associated Certification or Exam for this course.