GH-300: GitHub Copilot Fundamentals

Duration: 3 Days (24 hours)

Course description

GH-300: GitHub Copilot Fundamentals is designed to provide developers with a comprehensive understanding of GitHub Copilot, an AI-powered tool that enhances coding efficiency. The course begins with an exploration of responsible AI usage, emphasizing the importance of ethical standards, transparency, and accountability in AI systems. Participants will learn how to apply these principles to ensure that AI-generated code aligns with project-specific requirements and mitigates potential risks.

As the course progresses, participants will delve into the various features of GitHub Copilot, including its autocomplete-style suggestions, chat interface, and integration with different development environments. The course covers practical aspects such as setting up and configuring GitHub Copilot, troubleshooting common issues, and using advanced features to generate useful code suggestions. Through hands-on exercises, participants will gain experience in using GitHub Copilot to enhance their coding workflows and improve productivity.

The course also introduces the concept of prompt engineering, teaching participants how to craft effective prompts to optimize GitHub Copilot's performance. By understanding the data flow and the role of large language models in generating context-aware responses, participants will be equipped to leverage GitHub Copilot's capabilities to their fullest potential. The course concludes with an overview of GitHub Copilot's applications across different programming languages and environments, providing a well-rounded foundation for developers to integrate AI assistance into their daily coding practices.

The course is designed as a blended learning experience that combines instructor-led training with online materials on the Microsoft Learn platform (https://docs.microsoft.com/learn). Students are encouraged to use the content on Learn as reference materials to reinforce what they learn in class and to explore topics in more depth.

IMPORTANT: This course is designed to be delivered in one full day. The activities are approximately 70% instructional led and 30% student interactive exercises and/or instructor demos.

Learning objectives

After completing this course, students will be able to:

- Understand and apply the principles of responsible AI usage.
- Configure and troubleshoot GitHub Copilot in various development environments.
- Utilize GitHub Copilot's features for code suggestions and completions.
- Craft effective prompts to optimize GitHub Copilot's performance.
- Integrate GitHub Copilot into different programming languages and workflows.
- Enhance coding efficiency and productivity with advanced GitHub Copilot techniques.

Audience profile

Audience profile for this course is the following:

- Developers looking to enhance their coding efficiency with AI-powered tools.
- Programmers interested in learning about responsible AI usage and ethical standards.
- Software engineers seeking to integrate GitHub Copilot into their development workflows.
- Coders wanting to improve their prompt engineering skills for better AI-generated code suggestions.

Audience prerequisites

The audience for this 3-day course consists of developers, programmers, software engineers, and coders who want to enhance their coding efficiency, learn about responsible AI usage, and integrate GitHub Copilot into their development workflows.

NOTE: The exercise activities in this 1-day class are *Advanced* and require intermediate knowledge of Git and GitHub functions and features.

Candidates should have the following:

- Basic understanding of programming concepts and experience with at least one programming language.
- Familiarity with integrated development environments (IDEs) and version control systems like GitHub.
- Foundational knowledge of AI and machine learning principles.

Learning Paths and modules

Learning Path: GitHub Copilot Fundamentals Part 1 of 2

Module 1: Responsible AI with GitHub Copilot

- Introduction
- Mitigate AI risks
- Microsoft and GitHub's six principles of responsible AI

Module 2: Introduction to GitHub Copilot

- Introduction
- GitHub Copilot, your AI pair programmer
- Interact with Copilot
- Set up, configure, and troubleshoot GitHub Copilot
- Exercise - Develop with AI-powered code suggestions by using GitHub Copilot and VS Code

Module 3: Introduction to prompt engineering with GitHub Copilot

- Introduction
- Prompt engineering foundations and best practices
- GitHub Copilot user prompt process flow
- GitHub Copilot data
- GitHub Copilot Large Language Models (LLMs)

Module 4: Using advanced GitHub Copilot features

- Introduction
- Advanced GitHub Copilot features
- Exercise - Set up GitHub Copilot to work with Visual Studio Code
- Applied GitHub Copilot techniques
- Exercise - Update a web API with GitHub Copilot

Module 5: GitHub Copilot Across Environments: IDE, Chat, and Command Line Techniques

- Introduction
- Code completion with GitHub Copilot
- GitHub Copilot Chat
- GitHub Copilot for the Command Line

Module 6: Management and customization considerations with GitHub Copilot

- Introduction
- Explore GitHub Copilot plans and their associated management and customization features
- Explore contractual protections in GitHub Copilot and disabling matching public code
- Manage content exclusions
- Troubleshoot common problems with GitHub Copilot

Learning Path: GitHub Copilot Fundamentals Part 2 of 2

Module 7: Developer use cases for AI with GitHub Copilot

- Introduction
- Boost developer productivity with AI
- Align with developer preferences
- AI in the Software Development Lifecycle (SDLC)
- Understand limitations and measure impact

Module 8: Develop unit tests using GitHub Copilot tools

- 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

Module 9: Introduction to GitHub Copilot Business

- Introduction
- About GitHub Copilot for Business
- GitHub Copilot for Business use cases and customer stories
- How to get started with GitHub Copilot for Business

Module 10: Introduction to GitHub Copilot Enterprise

- Introduction
- About GitHub Copilot Enterprise
- How to get started

Module 11: Using GitHub Copilot with JavaScript

- Introduction
- What is GitHub Copilot
- Exercise - Set up GitHub Copilot to work with Visual Studio Code
- Use GitHub Copilot with JavaScript
- Exercise - Update a JavaScript portfolio with GitHub Copilot

Module 12: Using GitHub Copilot with Python

- Introduction
- What is GitHub Copilot?
- Exercise - Set up GitHub Copilot to work with Visual Studio Code
- Use GitHub Copilot with Python
- Exercise - Update a Python web API with GitHub Copilot

Exercises and Demos (10 exercises, 2 hours)

Exercises are to be used as hands-on activities for individual students which are led by the instructor, or demonstrations led by the instructor. The decision to lead hands-on activities or perform demonstrations is the instructor's responsibility.

Module 2: Introduction to GitHub Copilot

- Exercise - Develop with AI-powered code suggestions by using GitHub Copilot and VS Code **Module 4: Using advanced GitHub Copilot features**
- Exercise - Set up GitHub Copilot to work with Visual Studio Code
- Exercise - Update a web API with GitHub Copilot

Module 8: Develop unit tests using GitHub Copilot tools

- 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 **Module 11: Using GitHub Copilot with JavaScript**
- Exercise - Set up GitHub Copilot to work with Visual Studio Code
- Exercise - Update a JavaScript portfolio with GitHub Copilot **Module 12: Using GitHub Copilot with Python**
- Exercise - Set up GitHub Copilot to work with Visual Studio Code
- Exercise - Update a Python web API with GitHub Copilot

Course timing

The following agenda provides estimated times to complete each classroom activity. However, the estimated times may vary depending on the background of your students, which may affect whether you can move faster or slower through the course material.

Estimated times for each Module include the time to complete as part of a 3-day course:

- The module's PowerPoint slide deck presentation (70% of course timing)
- Exercises (30% of course timing)

- Optional: Pre-defined product demonstrations, determined by instructor
- Optional: Knowledge Check questions (see the section on Additional Timing Notes below)

You should adjust the agenda accordingly based on any classroom activities that you personally created or plan to deliver that are not included in the slides for this course. For example, if you plan to present:

- ad-hoc demonstrations
- review activities
- classroom games
- and so on…

Note: Each module activity for the following agenda is the slide deck presentation for that module.

Portal, Cloud Shell, PowerShell, and the CLI (when necessary for the courseware)

The exercise instructions are written to use the Cloud Shell. The Cloud Shell automatically connects to Azure and provides access to PowerShell and the CLI.

If you would rather have students use PowerShell or the CLI locally, you can use these links.

- Install Azure PowerShell on Windows with PowerShellGet
- Install Azure CLI on Windows

Azure subscriptions (when necessary for the courseware)

To complete the exercises and any additional demonstration exercises in this course, students will need an Azure Subscription.

The Azure pass effectively functions in the same way as the publicly available Microsoft Azure Trial Subscription. This means there are limitations on what you can do with the pass.