

**Introduction to GitHub Copilot**

**Course Number:** AI-108WA
**Duration:** 1 day

**Overview**

Develop code faster in Visual Studio Code and other leading development environments using GitHub Copilot as your AI-powered pair programmer. GitHub Copilot is powered by OpenAI’s language model, which has been trained on a massive dataset of public code repositories. It helps developers write code by providing suggestions for lines and entire functions, making the coding process faster and more efficient.

Accelebrate's GitHub Copilot training course teaches attendees how to use GitHub Autopilot and its AI capabilities to optimize their coding process properly. Participants not only grasp the theoretical aspects of Generative AI and GitHub Copilot but also gain the ability to apply these technologies to streamline their coding processes, enhance productivity, and innovate their approach to software development.

**Prerequisites**

All students must have experience with programming languages like Python.

**Materials**

All Generative AI with GitHub CoPilot training students receive comprehensive courseware.

**Software Needed on Each Student PC**

* A modern web browser and an Internet connection
* Visual Studio Code
* GitHub Copilot extension for Visual Studio Code
* A GitHub Copilot subscription

**Objectives**

* Gain foundational knowledge in generative AI technologies, focusing on LLMs, exploring their significance and applications in today's tech landscape
* Learn to effectively utilize GitHub Copilot as an AI-powered programming assistant, from basic introductions to advanced functionalities, enabling efficient coding practices
* Acquire essential skills in prompt engineering to communicate effectively with AI, optimizing the generation of code snippets, explanations, and solutions
* Explore the mechanisms and strategies for prompting GitHub Copilot, mastering its use for code completion, debugging, and working with large codebases
* Understand how GitHub Copilot adapts to different programming languages, enhancing versatility in coding tasks and projects
* Integrate GitHub Copilot into software development workflows, adopting best practices and design patterns for cleaner, more efficient, and reliable code
* Discover the potential of GitHub Copilot Chat for collaborative coding, troubleshooting, and ideation, fostering a more interactive and productive development environment

**Outline**

* Introduction to Generative AI and LLMs
	+ Power of Generative AI
	+ Technical Foundation of Generative AI
	+ Opportunities Created by Generative AI
	+ Challenges and Key Concerns
	+ Generative AI and LLM
	+ Common Generative AI Applications
	+ ChatGPT vs GitHub Copilot
* GitHub Copilot Introduction
	+ Define GitHub Copilot
	+ Explore Common Features
	+ GitHub Copilot Subscription Types
	+ How Copilot Works
* Working with GitHub Copilot
	+ Recap the importance of understanding core programming concepts, algorithms, and data structures.
	+ Copilot complements these skills, not replaces them.
	+ Must understand crucial role of critical thinking, problem-solving, and debugging skills in effective coding.
	+ Copilot is a tool, not a solution, and independent judgment is paramount.
	+ continuous learning and adaptability in the ever-evolving world of technology.
	+ Copilot can assist in navigating change and adopting new skills.
	+ We will create projects that showcase skills and how to utilize Copilot responsibly.
* Mastering Basic Completion
	+ Variable Names and Structure
	+ Context-Aware Completion
	+ Accepting Suggestions
	+ Power of Comments
	+ Completion for Documentation
	+ Code Snippet magic
	+ Fine Tuning Control
* Prompt Engineering Essentials
	+ Prompt Engineering Introduction
	+ Key Elements of Prompts
	+ Prompting Techniques
	+ Zero-Shot
	+ One-Shot
	+ Few Shot
	+ Chain of Thoughts
* Copilot Prompting Mechanism
	+ Prompt Principles
	+ Best Practises to follow
	+ Prompt Process Flow in GitHub Copilot
	+ Example: Zero Shot
	+ Example: One Shot
	+ Example: Few Shot
* Working with GitHub Copilot Chat
	+ Introduction to GitHub Copilot Chat
	+ GitHub Copilot Chat Process Flow
	+ Critical Use cases
	+ Limitations of Copilot Chat
	+ Using Commands
	+ Keyboard Shortcuts
* Advance Completion and managing Large Codebases
	+ Regular Expressions
	+ Code Formatting and Style Consistency
	+ Code Refactoring
	+ Navigating Large Codebases
	+ Coding with Question
	+ Debugging Assistance
	+ Testing code with Prompts
	+ Tool Integration
* GitHub Copilot and Programming Languages
	+ Python-specific Features
	+ JavaScript and TypeScript
	+ Secure Code with Copilot
	+ Adding Accessibility
	+ Game Development with Copilot
* GitHub Copilot Design Patterns and Best Practices
	+ What are Patterns
	+ GitHub Copilot Pattern Categories
	+ Design Pattern at Work
	+ Practically Viable Patterns
	+ Pattern for Test Engineers
	+ Best Practices
* Conclusion