CTR-809 GitHub Copilot

About This Course

This hands-on course provides an in-depth understanding of how to use GitHub Copilot to assist in building full-stack applications. Designed for developers with a foundational understanding of web development, the course demonstrates how to integrate Copilot into day-to-day workflows to boost productivity, reduce boilerplate coding, and improve software quality. Students will learn how to use Copilot to write, debug, and refactor code efficiently while building a full-stack project using Flask for the backend and React for the frontend.

At Course Completion

By the end of the course, participants will:

- Understand how GitHub Copilot works and how to configure it
- Use Copilot to accelerate coding in Python, JavaScript, HTML/CSS, and React
- Prompt Copilot effectively for code generation and auto-completion
- Identify Copilot's limitations and how to handle incorrect suggestions
- Build and document a full-stack application using Copilot as an assistant
- Apply best practices around privacy, ethics, and security when using AI-assisted development

Course Details

Module 1: Introduction to GitHub Copilot

- The origin and purpose of GitHub Copilot
- Overview of AI-assisted development
- Licensing, pricing models, and Copilot for business
- Installing Copilot in VS Code
- Overview of Copilot Labs

Module 2: Understanding How Copilot Works

- Architecture and model foundations (OpenAI Codex)
- Inline completions, multiline suggestions, and ghost text
- Prompting Copilot with comments and function names
- What Copilot can and cannot do

Module 3: Copilot for Python (Backend Development)

- Using Copilot to scaffold Flask routes and logic
- Autogenerating models, input validation, and serializers
- Working with databases (e.g., SQLAlchemy)
- Writing unit tests and automated scripts with Copilot
- API documentation and OpenAPI generation

Module 4: Copilot for JavaScript and React (Frontend Development)

- Autogenerating React components
- Using Copilot for form handling, hooks, and state management
- Connecting to Flask REST APIs with Axios
- Building and styling components with Copilot (JSX and CSS)
- Writing tests with Jest and React Testing Library

Module 5: Real-Time Copilot Usage: Building a Full-Stack App (Part 1)

- Project Setup: React + Flask folder structure
- GitHub repository initialization
- Using Copilot for project boilerplate generation
- Creating models, API endpoints, and test data

Module 6: Real-Time Copilot Usage: Building a Full-Stack App (Part 2)

- Frontend: Using Copilot to design the UI
- Backend: Securing routes and managing sessions or tokens
- Integrating Copilot in debugging and refactoring tasks
- Connecting the frontend with the backend

Module 7: Copilot in Team Environments

- Using Copilot with GitHub Codespaces
- Copilot in pair programming settings
- Customizing Copilot behavior per developer
- Limitations when working in teams

Module 8: Ethical Use, Privacy, and Limitations

- Understanding training data and IP concerns
- Avoiding code leaks and secure coding with Copilot
- GDPR and corporate policy compliance
- When not to use Copilot

Module 9: Final Project and Presentation

- Full-stack mini-project developed with Copilot
- Documentation written partially with Copilot
- Presentation on where Copilot helped vs. where manual input was essential
- Peer review and Q&A session

Prerequisites

Participants should have:

- Proficiency in Python (especially Flask basics)
- Basic understanding of JavaScript and React

- Experience using Git and GitHub
- Familiarity with Visual Studio Code (VS Code)
- Ability to work with REST APIs