

Days: 3

Prerequisites: While no prior Claude Code experience is required, comfort working in a terminal, basic Git knowledge, and an active Claude Pro or Max subscription may be beneficial.

Audience: This class is intended for: Software developers and engineers; DevOps and platform engineers; technical leads and engineering managers; teams building AI-assisted development workflows; developers evaluating or adopting AI coding tools.

Description: Claude Code is not just a code suggestion tool—it's a full agentic development environment that can read your codebase, write and edit files, run tests, commit to Git, and iterate until the task is complete.

This immersive 3-day hands-on course teaches developers how to use Claude Code the same way Anthropic engineers do: spec-driven, context-aware, and production-ready.

Participants leave with a working system built on their own project, including custom skills, automation hooks, multi-agent workflows, and CI/CD integration.

Course Objectives: In this course, you will build:

- A project configured with **CLAUDE.md** and AI development conventions
- Custom reusable AI skills and workflows
- Security, audit, and automation hooks
- Custom slash commands and plugins
- A custom MCP server connected to external tools
- Multi-agent and parallel sub-agent workflows
- A GitHub Actions CI/CD pipeline using Claude in headless mode
- Production-ready security, permission, and cost controls

OUTLINE:

CORE WORKFLOWS, CONTEXT MANAGEMENT, AND GIT INTEGRATION

- Installing and configuring Claude Code
- Understanding the agentic development loop
- Creating and managing CLAUDE.md project memory
- Context and session hygiene best practices
- Spec-driven development workflows
- Secure Git and GitHub workflows using natural language
- Hands-On Labs: Build and ship a working application using Claude Code

SKILLS, HOOKS, AND DISTRIBUTION: REUSABLE WORKFLOWS, GUARDRAILS, AND PACKAGING

- Designing and building reusable AI skills
- Applying Anthropic's skill design frameworks
- Creating deterministic hooks for security and automation
- Building custom slash commands
- Testing, benchmarking, and optimizing AI workflows
- Packaging skills, hooks, and commands into distributable plugins

Claude Code Foundations

- Hands-On Labs: Build skills, enforce guardrails, and package a reusable plugin

MCP, SUBAGENTS, AND PRODUCTION DEPLOYMENT: ORCHESTRATION, INTEGRATIONS, AND ENTERPRISE READINESS

- Understanding the Model Context Protocol (MCP)
- Building and connecting custom MCP servers
- Orchestrating subagents and multi-agent teams
- Running Claude Code in CI/CD pipelines
- Configuring enterprise security and permissions
- Scaling, optimizing costs, and monitoring production usage
- Hands-On Labs: Deploy Claude Code in CI/CD and scale multi-agent workflows

KEY LEARNING OUTCOMES

- Use Claude Code as a full agentic development system
- Create effective project memory and AI conventions
- Build reusable AI skills and workflows
- Enforce security, auditing, and automation with hooks
- Orchestrate parallel AI agents for complex tasks
- Integrate Claude Code into CI/CD pipelines
- Configure production-grade security and cost controls