Agentic Development with AWS Kiro | AI Training

Module 1 – Introduction to Agentic Development and AWS Kiro

- Overview of AWS Kiro
- Evolution of AI-Assisted Software Development
- The Concept of an Agentic IDE
- Traditional Development vs. Agentic Development
- Kiro Architecture and Components
- Use Cases and Benefits
- AI-Assisted Development Lifecycle

Module 2 – Getting Started with the AWS Kiro Environment

- Installing and Configuring Kiro
- Overview of the Interface
- Project Management
- Code Navigation
- Development Context Management
- Using Agentic Chat
- Interacting with AI Agents

Module 3 – Spec-Driven Development

- Introduction to Specification-Based Development
- The Requirements → Design → Tasks Approach
- Generating Specifications from Prompts
- Structure of Requirements Files
- Structure of Design Files
- Structure of Tasks Files
- Introduction to EARS Syntax
- Best Practices for Writing Specifications

Module 4 – Conversational Development and Code Generation

- Using AI Chat to Develop Features
- AI-Assisted Code Generation
- Understanding Project Context
- AI-Powered Code Analysis
- AI-Assisted Refactoring
- Intelligent Debugging
- AI-Powered Code Optimization

Module 5 – Automation with Agent Hooks

- Introduction to Agent Hooks
- Event-Driven Automation
- Triggers and Actions
- Test Automation
- Automated Documentation Generation
- Automating Quality Checks
- Automating Security Processes

Module 6 – Steering and AI Governance

- Overview of the Steering Mechanism
- Managing .kiro/steering Files
- Defining Development Standards
- Coding Conventions
- Controlling AI Behavior
- Managing Persistent Context
- Governance Best Practices

Module 7 – Integrating External Tools with MCP

- Introduction to the Model Context Protocol (MCP)
- MCP Server Architecture
- Connecting to External Services
- Integrating Third-Party APIs
- Automating External Interactions
- Practical Use Cases
- Security and Best Practices

Module 8 – Agentic Development Capstone Project

- Analyzing Project Requirements
- Writing Functional Specifications
- Automatically Generating Tasks
- AI-Assisted Development
- Workflow Automation
- Integrating External Tools Through MCP
- Project Validation and Optimization
- Presenting the Final Solution