Faysal Aziz

Mastering Gemini AI Tools

From CLI to Production-Ready AI Apps

FA-0306 · one-day

Agentic & Generative AI · DevOps, Cloud & Infrastructure

Introduction

This one-day intensive takes you through the essential workflows and tooling of the Gemini-powered ecosystem by Google: using the Gemini CLI as a terminal-based AI assistant, harnessing Gemini Code Assist for developer productivity in IDEs, and scaling up with Vertex AI for deploying and managing production-ready generative AI apps.

You'll get hands-on exposure to how these tools integrate, when to use which, and how to architect a pipeline from prototyping to deployment — all grounded in real industry practices, not academic abstractions.

The course is led by an instructor with over 30 years of industry experience, ensuring the content is practical, relevant, and directly oriented toward real-world demands.

Learning outcomes

By the end of this workshop, participants will be able to:

- Use Gemini CLI to interact with Gemini models directly from the terminal for coding, content generation, debugging, and automation tasks.
- Integrate Gemini Code Assist in a modern IDE (e.g. VS Code, IntelliJ) to speed up coding with AI-powered completions, test generation, documentation, and refactoring.
- Understand how Gemini models (e.g. Gemini 2.5 or newer) work under the hood and what capabilities they bring (multimodal inputs, reasoning, tool use).
- Use the unified Google Gen AI SDK to access Gemini via the CLI or via Vertex AI, enabling portability between local prototyping and cloud-based deployment.
- Build and deploy a simple generative AI application on Vertex AI, taking it from prompt-based experimentation to a production-ready endpoint, while applying good practices for authentication, project configuration, and resource management.

- Make informed decisions about which tool (CLI, Code Assist, or Vertex AI) is right for which stage of development or deployment — balancing speed, control, scalability, and compliance needs.

Prerequisites

- Comfortable with at least one general-purpose programming language (e.g. Python or JavaScript).
- Familiarity with basic command-line / terminal usage, and optionally with an IDE like VS Code or IntelliJ.
- Basic understanding of cloud concepts and being able to set up or access a project in Google Cloud.
- Willingness to perform setup operations (install CLI / SDK, configure environment variables, authenticate).

Training outline

1. Understanding the Landscape — Models & Platforms

- Overview of the Gemini family of models: Pro, Flash, etc.; what "multimodal" means (text, code, images, maybe future media) and why it matters.
- Distinction between "Developer-first" vs "Enterprise / Cloud-native" workflows: when to use the simpler Developer API / CLI vs when to go for a managed platform with full lifecycle support.
- Introduction to the unified SDK (google-genai) that allows code to run on either the standalone Gemini API or via Vertex AI; why this abstraction matters for portability.

2. Getting Hands-On: Gemini CLI

- Installing Gemini CLI (global npm package or OS-specific), initial setup and authentication (OAuth or API-key based, or using cloud credentials).
- Understanding how Gemini CLI works: the "reason-and-act (ReAct)" loop, how it can use local or remote tools, shell commands, file operations within your project, enabling tasks beyond just code generation.

  Common use-cases via CLI:

  - Generating code skeletons or boilerplate from prompts
  - Refactoring or improving existing code (bug fixing, optimizations)
  - Running shell commands, automating repetitive tasks, project scaffolding
  - Content generation / documentation / research tasks using natural-language requests
- Practical limitations & quotas (for free/personal tiers): understanding usage limits and implications for heavy workloads.

3. Using Gemini Code Assist in IDEs

- Setting up Gemini Code Assist in supported IDEs (e.g. VS Code, IntelliJ, JetBrains), configuration, authentication, linking to a cloud project if needed.

  Core features in Code Assist:

  - Code completions and suggestions while you type
  - Generating code blocks from comments or prompts
  - Smart actions: /generate, /fix, /doc, /simplify, etc.
  - Unit-test generation, docstring or documentation generation, code refactoring, and debugging assistance.
- When to prefer Code Assist vs CLI: e.g. for interactive development, real-time feedback, integration with code editor and workflow; vs CLI for scripting, heavy automation, or external integration.

4. Bridging to Cloud & Scaling: Vertex AI + Gemini API

- Overview of Vertex AI as a unified platform for training, deploying, and scaling machine-learning / generative-AI models — both custom and from model garden.
- How Gemini models on Vertex AI work: using the Gemini API via Vertex, enabling the same models used in CLI/Code Assist to be deployed at scale.
- Quickstart: using google-genai SDK to call Gemini via Vertex AI — setup including environment variables, project & location config.
- Example deployment scenarios: prompt-based content generation services, chatbot or conversational agents, integrations with web/mobile back-end, automation workflows that require scalability & cloud management.
- MLOps considerations: versioning models, environ management, using Vertex AI's deployment, monitoring, resource controls, security/compliance for enterprise usage.

5. Putting It All Together: Workflow Patterns & Best Practices

- Local prototyping → rapid iteration with Gemini CLI or Code Assist (fast feedback, no deployment overhead)
- When to transition to Vertex AI (scaling, multi-user, production deployment, integration with cloud infra)
- Hybrid usage: e.g. team uses CLI for quick dev tasks + Code Assist for daily coding, but deploy finished models on Vertex for production / shared services
- Authentication / project configuration / environment hygiene (separating dev vs prod, credential management)

- Cost, quota management, model versioning, security considerations especially when dealing with private codebases or sensitive data — choosing between CLI/Code Assist free tier vs enterprise-grade deployment on Vertex

---

© 2026 Faysal Aziz · Kuala Lumpur