

ARTIFICIAL INTELLIGENCE

ADVANCED

4 DAYS

Building AI Agents with Claude: API, Agent SDK and MCP

The full engineering path from first API call to production agent. Covers the Claude API, tool use, retrieval, the Agent SDK, MCP server development, evaluation, and the cost and performance work that decides whether an agent survives contact with production.

Why Build AI Agents with Claude?

End-to-end coverage

API, retrieval, agents, MCP and evaluation taught as one connected system.

Own your integrations

Build MCP servers that connect Claude to your real internal systems.

Production-grade, not demo-grade

Focus on the reliability, evaluation and cost work that demos skip.

Measurable quality

Leave with an evaluation harness, not just a working prototype.

What you'll be able to do

- ✓ Build applications on the Claude API using the Messages API and SDKs
- ✓ Apply structured prompting, tool use and extended thinking effectively
- ✓ Implement retrieval-augmented generation with embeddings and contextual retrieval
- ✓ Design agent architectures using chaining, routing and parallelisation
- ✓ Build production agents with the Claude Agent SDK, sessions and subagents
- ✓ Develop MCP servers and clients exposing tools, resources and prompts
- ✓ Create evaluation suites that measure and regress agent quality
- ✓ Optimise latency and cost through prompt caching, batching and model routing

Who this is for

- AI and machine learning engineers
- Backend and full-stack developers
- Solution architects and technical leads
- Data engineers moving into AI application development

Curriculum

01 Claude Platform and Messages API Foundations

- The Messages API, official SDKs and authentication patterns
- Model selection criteria (capability, latency, cost and context limits)
- Request and response shapes, streaming and error handling
- Building a thin service layer around Claude for your application
- **Hands-on:** Lab: Build a working Claude-powered service that answers a domain-specific query over the Messages API

02 Structured Prompting and Reliable Output Control

- Structured prompting with XML tags, examples and clear directives
- Constraining format for downstream parsers and UI contracts
- Few-shot patterns that improve consistency without overfitting
- Failure modes of unstructured prompts and how to diagnose them
- **Hands-on:** Lab: Refactor a brittle prompt into a structured template with validated JSON output

03 Extended Thinking and Multimodal Document Processing

- Extended thinking and when deeper reasoning improves results
- Trade-offs between thinking depth, latency and token cost
- Multimodal input for images, PDFs and long documents
- Extracting structured facts from messy enterprise artefacts
- **Hands-on:** Lab: Process a multi-page PDF into structured fields with optional extended thinking enabled

04 Tool Use: Custom and Built-In Tools

- Tool use fundamentals and custom tool definitions
- Built-in tools (code execution, text editor and web search)
- Tool schemas, argument validation and safe side-effect design
- Multi-tool workflows and recovering from tool failures
- **Hands-on:** Lab: Define two custom tools and wire them into an agent loop that solves a multi-step task

05 Embeddings, Chunking and RAG Pipelines

- Chunking strategies for technical and business corpora
- Embedding pipelines and vector store basics
- Retrieval quality issues (noise, duplication and stale content)
- Grounding answers with citations and source provenance
- **Hands-on:** Lab: Build a RAG pipeline over an internal corpus and serve grounded answers with sources

06 Contextual Retrieval and Hybrid Search at Scale

- Contextual retrieval patterns for higher recall and precision
- Hybrid search combining embeddings with BM25 / keyword signals
- Re-ranking, metadata filters and access-aware retrieval
- Evaluating retrieval quality separately from generation quality
- **Hands-on:** Lab: Upgrade a baseline RAG system with hybrid search and measure retrieval lift on a fixed eval set

07 Agent Architecture Patterns and Workflow Design

- Agent architectures: chaining, routing and parallelisation
- Choosing single-agent versus multi-step workflow designs
- State, memory and hand-off patterns between steps
- Designing for observability from the first prototype
- **Hands-on:** Lab: Design and implement a routed multi-step workflow for a realistic business process

08 Claude Agent SDK: Sessions, Subagents and Orchestration

- The Claude Agent SDK, sessions and lifecycle management
- Subagents and multi-agent orchestration patterns
- Persisting session state across user and tool interactions
- Guardrails for production agents (timeouts, retries and abort paths)
- **Hands-on:** Lab: Build a production-shaped agent with sessions and a specialised subagent

09 MCP Server and Client Development

- MCP primitives: tools, resources and prompts
- Building an MCP server against an internal API
- Client integration patterns and capability discovery
- Security, auth and least-privilege exposure of enterprise systems
- **Hands-on:** Lab: Build and deploy an MCP server that exposes a real internal capability to Claude

10 Evaluation, Cost, Observability and Production Readiness

- Designing evaluation sets and grading rubrics for agents
- Automated evals, regression testing and release gates
- Prompt caching, batching, model routing and cost-performance trade-offs
- Observability, guardrails and failure handling in production
- **Hands-on:** Lab: Build an evaluation harness and run a costed regression suite against your agent