Cloud Contraptions
TRAINING & CONSULTING

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

Building AI Agents with the Claude Agent SDK, OpenAI Agents SDK, and Google ADK

Class Duration

21 hours of live training delivered over 3 days.

Student Prerequisites

- Professional development experience in Python or TypeScript
- Familiarity with LLM APIs: prompts, tool calling, and structured outputs (see *LLM Application Development with TypeScript and Python*)
- Comfort with async programming, Git, and containers
- Prior agent framework experience is helpful but not required

Target Audience

Software engineers, AI engineers, and architects building agents on vendor SDKs, including teams comparing SDKs or connecting agents built on different ones. Examples are in Python and TypeScript. For AWS and .NET stacks, see *Building AI Agents with Strands Agents and Amazon Bedrock AgentCore* and *Building AI Agents in .NET with Microsoft Agent Framework*.

Description

This course teaches agent development with three vendor SDKs: the Claude Agent SDK, the OpenAI Agents SDK, and Google's Agent Development Kit (ADK). The Claude Agent SDK packages the agent harness behind Claude Code. The OpenAI Agents SDK is a light orchestration library, and ADK is a full framework with graph workflows. Students build agents with each SDK, then connect tools over the Model Context Protocol (MCP) and agents over Agent2Agent (A2A). Later modules cover human approvals, durable execution, tracing, evaluation, security, deployment, and choosing an SDK.

Learning Outcomes

- Explain the agent loop and what an agent SDK adds to a model API.

Copyright 2026 Cloud Contraptions LLC – All Rights Reserved

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

- Build Claude Agent SDK agents with tools, subagents, hooks, and permission modes.
- Build OpenAI Agents SDK agents with tools, handoffs, guardrails, and sandbox agents.
- Build Google ADK agents with graph workflows, sessions, state, and callbacks.
- Connect agents in each SDK to MCP servers.
- Connect agents across frameworks with the A2A protocol.
- Add human approvals and durable execution to long-running agents.
- Trace agent runs with OpenTelemetry and evaluate them with test suites.
- Defend agents against prompt injection with least privilege and sandboxing.
- Choose a hosting pattern and an SDK for a given system.

Training Materials

Comprehensive courseware is distributed online at the start of class. All students receive a downloadable MP4 recording of the training.

Software Requirements

Python 3.12+ and Node.js 22+, API keys for Anthropic, OpenAI, and Google Gemini, Docker or Podman, VS Code or an editor of choice, and Git. A Google Cloud project is optional for the managed deployment topics.

Training Topics

Anatomy of an Agent Loop

- The agent loop step by step
- Stop conditions and turn limits
- What an SDK adds to a model API
- Harness, library, and framework designs
- Language support across the SDKs
- Budgets and runaway prevention

Claude Agent SDK: The Agent Harness

- The Python and TypeScript packages
- The bundled Claude Code CLI
- `query()` and `ClaudeSDKClient`

Copyright 2026 Cloud Contraptions LLC – All Rights Reserved

- Message types and `ResultMessage` subtypes
- Built-in file, shell, and web tools
- Custom tools on in-process MCP servers
- System prompts and tool allow lists
- `max_turns`, `max_budget_usd`, and effort levels
- Resuming and forking sessions
- Automatic compaction
- Persisting transcripts with `SessionStore`
- Structured outputs with Zod or Pydantic

Claude Agent SDK: Subagents, Skills, and Control

- Subagents with `AgentDefinition`
- Scoped tools and models per subagent
- Background subagents and context isolation
- Capping subagent depth, concurrency, and spend
- Skills, commands, and `CLAUDE.md` via `settingSources`
- Loading plugins with the `plugins` option
- Hooks such as `PreToolUse` and `Stop`
- Blocking, rewriting, or auditing tool calls
- Permission modes such as `acceptEdits` and `plan`
- Permission rule evaluation order
- File checkpointing and rewind

OpenAI Agents SDK: Agents, Tools, and the Runner

- `Agent` instructions, tools, and model settings
- Standard, synchronous, and streamed runs
- Function tools with generated schemas
- Hosted tools such as web search
- Code interpreter and image generation
- Tool search for large tool sets
- Shell and apply-patch tools
- Sessions with SQLite, Redis, or SQLAlchemy
- Encrypted sessions
- `previous_response_id` and server-side conversations
- Streaming run events to a UI
- The Python and TypeScript packages

Copyright 2026 Cloud Contraptions LLC – All Rights Reserved

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

OpenAI Agents SDK: Orchestration, Guardrails, and Sandbox Agents

- Handoffs versus agents as tools
- Routing with structured outputs
- Chaining, evaluator loops, and parallel runs
- Input, output, and tool guardrails
- Tripwires and parallel guardrail execution
- `SandboxAgent`, manifests, and capabilities
- Resumable sandbox workspaces
- Local Unix and Docker sandbox clients
- Hosted sandboxes such as E2B and Modal
- Realtime and voice agents

Google ADK: Agents, Tools, and Workflows

- The ADK workflow runtime
- LLM agents with `Agent` and `LlmAgent`
- Managed agents with server-side tools
- Function and OpenAPI tools
- Built-in Google Search and code execution
- Tool authentication
- Graph workflows with nodes, edges, and routes
- Fan-out and joins
- Mixing agents with deterministic code
- Coordinator agents and subagent modes
- Dynamic workflows
- Moving off deprecated template workflows

Google ADK: Sessions, State, Callbacks, and Plugins

- Sessions, events, and state
- Memory for cross-session knowledge
- Artifacts for files and binary data
- Rewinding sessions
- Context compaction and caching
- Agent, model, and tool callbacks
- Plugins such as Model Armor
- Local development with `adk web`
- The ADK API server

Copyright 2026 Cloud Contraptions LLC – All Rights Reserved

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

MCP in Every SDK

- Tools, resources, prompts, and elicitation
- stdio and Streamable HTTP transports
- `mcpServers` in the Claude Agent SDK
- MCP tool naming and permission rules
- Tool search for large catalogs
- `HostedMCPTool` and local MCP servers
- Tool filters and tool list caching
- `McpToolset` and `tool_filter` in ADK
- Exposing an ADK agent as an MCP server
- One MCP server shared across SDKs
- The Tasks and MCP Apps extensions

A2A for Cross-Framework Agents

- Agent Cards and task states
- Messages, parts, and artifacts
- JSON-RPC, gRPC, and HTTP+JSON bindings
- Streaming and push notifications
- Native A2A support in ADK
- Wrapping agents as A2A servers
- The official A2A SDKs
- Security schemes and signed Agent Cards
- A2A extensions
- MCP versus A2A

Human-in-the-Loop and Durable Execution

- `canUseTool` approval callbacks
- Clarifying questions with `AskUserQuestion`
- Hooks that ask about or defer a tool call
- `needs_approval` tools and run interruptions
- Resuming from a serialized `RunState`
- Tool confirmations in ADK
- Approval design and escalation paths
- ADK resumable apps
- Temporal, Restate, DBOS, and Dapr integrations
- Idempotent tools and safe re-execution

Copyright 2026 Cloud Contraptions LLC – All Rights Reserved

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

Tracing, Observability, and Evaluation

- OpenTelemetry export from the Claude Agent SDK
- OpenAI Agents SDK traces and spans
- Custom trace processors
- ADK tracing to Cloud Trace or OTLP
- OpenTelemetry GenAI semantic conventions
- Keeping prompts out of trace exports
- ADK evalsets and `adk eval`
- User simulation
- Golden tasks and CI regression suites
- LLM-as-judge transcript review

Security for Tool-Using Agents

- Prompt injection through tool results
- The lethal trifecta
- Scoped tools and deny rules
- Read-only mounts and per-tool credentials
- Anthropic’s sandbox runtime
- Hardened containers, gVisor, and microVMs
- Sandboxed code execution in ADK
- Credential proxies and egress allowlists
- Guardrails as defense in depth

Deployment Patterns

- Claude Agent SDK container patterns
- Claude Managed Agents as a hosted option
- OpenAI Agents SDK agents in your services
- ADK on Agent Runtime, Cloud Run, or GKE
- Multi-tenant isolation and session pinning
- Scaling within provider rate limits
- Token cost, budgets, and prompt caching

Choosing an SDK

- Comparison criteria
- When the Claude Agent SDK fits
- When the OpenAI Agents SDK fits
- When Google ADK fits

Copyright 2026 Cloud Contraptions LLC – All Rights Reserved



To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

- Combining SDKs through MCP and A2A
- Portable tools, prompts, and evals

Copyright 2026 Cloud Contraptions LLC – All Rights Reserved