Agents - Agent Development Kit

Source URL: https://google.github.io/adk-docs/agents/

Agents¶

In the Agent Development Kit (ADK), an **Agent** is a self-contained execution unit designed to act autonomously to achieve specific goals. Agents can perform tasks, interact with users, utilize external tools, and coordinate with other agents.

The foundation for all agents in ADK is the `BaseAgent` class. It serves as the fundamental blueprint. To create functional agents, you typically extend `BaseAgent` in one of three main ways, catering to different needs – from intelligent reasoning to structured process control.

Types of agents in ADK

Core Agent Categories¶

ADK provides distinct agent categories to build sophisticated applications:

1. **LLM Agents (`LlmAgent`, `Agent`)**: These agents utilize Large Language Models (LLMs) as their core engine to understand natural language, reason, plan, generate responses, and dynamically decide how to proceed or which tools to use, making them ideal for flexible, language-centric tasks. Learn more about LLM Agents...
2. **Workflow Agents (`SequentialAgent`, `ParallelAgent`, `LoopAgent`)**: These specialized agents control the execution flow of other agents in predefined, deterministic patterns (sequence, parallel, or loop) without using an LLM for the flow control itself, perfect for structured processes needing predictable execution. Explore Workflow Agents...
3. **Custom Agents**: Created by extending `BaseAgent` directly, these agents allow you to implement unique operational logic, specific control flows, or specialized integrations not covered by the standard types,

catering to highly tailored application requirements. Discover how to build Custom Agents...

Choosing the Right Agent Type¶

The following table provides a high-level comparison to help distinguish between the agent types. As you explore each type in more detail in the subsequent sections, these distinctions will become clearer.

Feature	LLM Agent (`LlmAgent`)	Workflow Agent	Custom Agent (`BaseAgent` subclass)
Primary Function	Reasoning, Generation, Tool Use	Controlling Agent Execution Flow	Implementing Unique Logic/ Integrations
Core Engine	Large Language Model (LLM)	Predefined Logic (Sequence, Parallel, Loop)	Custom Code
Determinism	Non-deterministic (Flexible)	Deterministic (Predictable)	Can be either, based on implementation
Primary Use	Language tasks, Dynamic decisions	Structured processes, Orchestration	Tailored requirements, Specific workflows

Agents Working Together: Multi-Agent Systems¶

While each agent type serves a distinct purpose, the true power often comes from combining them. Complex applications frequently employ multi-agent architectures where:

- **LLM Agents** handle intelligent, language-based task execution.
- **Workflow Agents** manage the overall process flow using standard patterns.
- **Custom Agents** provide specialized capabilities or rules needed for unique integrations.

Understanding these core types is the first step toward building sophisticated, capable AI applications with ADK.

What's Next?¶

Now that you have an overview of the different agent types available in ADK, dive deeper into how they work and how to use them effectively:

- **LLM Agents:** Explore how to configure agents powered by large language models, including setting instructions, providing tools, and enabling advanced features like planning and code execution.
- **Workflow Agents:** Learn how to orchestrate tasks using `SequentialAgent`, `ParallelAgent`, and `LoopAgent` for structured and predictable processes.
- **Custom Agents:** Discover the principles of extending `BaseAgent` to build agents with unique logic and integrations tailored to your specific needs.
- **Multi-Agents:** Understand how to combine different agent types to create sophisticated, collaborative systems capable of tackling complex problems.
- **Models:** Learn about the different LLM integrations available and how to select the right model for your agents.