Learning Paths

> **Description:** Structured progression from beginner to expert ADK development

🎯 **Purpose**: Structured learning progression from ADK fundamentals to production mastery.

📚 **Source of Truth**: do (https://github.com/raphaelmansuy/adk_training/tree/main/do) + tutorial_implementation/ (https://github.com/raphaelmansuy/adk_training/tree/main/tutorial_implementation/) + research/ (https://github.com/raphaelmansuy/adk_training/tree/main/research/) (ADK 1.15)

🏁 Beginner Path (1-2 weeks)

Phase 1: Core Concepts (Days 1-3)

📖 **Tutorials**: 01_hello_world_agent.md (01_hello_world_agent.md), 02_function_tools.md (02_function_tools.md)

🎯 **Goals**:

- Understand agent lifecycle
- Create basic LLM agents
- Implement function tools
- Run agents locally

💡 **Key Concepts**:

- Agent class structure
- Tool function patterns
- State management basics
- Local development setup

Phase 2: Workflow Patterns (Days 4-7)

📖 **Tutorials**: 03_openapi_tools.md (03_openapi_tools.md), 04_sequential_workflows.md (04_sequential_workflows.md), 05_parallel_processing.md (05_parallel_processing.md)

🎯 **Goals**:

- Integrate external APIs
- Build sequential pipelines
- Implement parallel processing
- Handle complex workflows

💡 **Key Concepts**:

- OpenAPI tool generation
- SequentialAgent composition
- ParallelAgent optimization
- Error handling patterns

🚀 Intermediate Path (2-4 weeks)

Phase 3: Advanced Patterns (Days 8-14)

📖 **Tutorials**: 06_multi_agent_systems.md (06_multi_agent_systems.md), 07_loop_agents.md (07_loop_agents.md), 08_state_memory.md (08_state_memory.md)

🎯 **Goals**:

- Design multi-agent systems
- Implement iterative refinement
- Master state management
- Build complex agent hierarchies

💡 **Key Concepts**:

- Agent communication patterns
- LoopAgent convergence criteria

- State scoping (session/user/app/temp)
- Memory persistence strategies

Phase 4: Production Foundations (Days 15-21)

📖 **Tutorials**: 09_callbacks_guardrails.md (09_callbacks_guardrails.md), 10_evaluation_testing.md (10_evaluation_testing.md), 11_built_in_tools_grounding.md (11_built_in_tools_grounding.md)

🎯 **Goals**:

- Implement safety guardrails
- Set up comprehensive testing
- Use built-in grounding tools
- Prepare for production deployment

💡 **Key Concepts**:

- Callback integration
- Automated testing frameworks
- Grounding with web/data/location
- Quality assurance patterns

🎯 Advanced Path (4-8 weeks)

Phase 5: Real-Time & Streaming (Days 22-28)

📖 **Tutorials**: 12_planners_thinking.md (12_planners_thinking.md), 13_code_execution.md (13_code_execution.md), 14_streaming_sse.md (14_streaming_sse.md), 15_live_api_audio.md (15_live_api_audio.md)

🎯 **Goals**:

- Master advanced reasoning
- Enable code execution
- Implement real-time streaming

- Handle multimodal inputs

💡 **Key Concepts**:

- Custom planner strategies
- Code execution environments
- SSE and BIDI streaming
- Audio/video processing

Phase 6: Enterprise Integration (Days 29-42)

📖 **Tutorials**: 16_mcp_integration.md (16_mcp_integration.md), 17_agent_to_agent.md (17_agent_to_agent.md), 18_events_observability.md (18_events_observability.md), 19_artifacts_files.md (19_artifacts_files.md)

🎯 **Goals**:

- Integrate MCP protocol
- Build distributed agent systems
- Implement comprehensive observability
- Handle file artifacts

💡 **Key Concepts**:

- MCP tool standardization
- A2A communication protocols
- Event-driven architectures
- File system integration

🏭 Expert Path (8+ weeks)

Phase 7: Production Mastery (Days 43-56)

📖 **Tutorials**: 20_yaml_configuration.md (20_yaml_configuration.md), 21_multimodal_image.md (21_multimodal_image.md), 22_model_selection.md (22_model_selection.md), 23_production_deployment.md (23_production_deployment.md)

🎯 **Goals**:

- Master configuration management
- Handle multimodal content
- Optimize model selection
- Deploy production systems

💡 **Key Concepts**:

- YAML-based configuration
- Image/video/document processing
- Model performance optimization
- Cloud deployment strategies

Phase 8: Advanced Topics (Days 57+)

📖 **Tutorials**: 24_advanced_observability.md (24_advanced_observability.md) through 34_pubsub_adk_integration.md (34_pubsub_adk_integration.md)

🎯 **Goals**:

- Master Pub/Sub patterns
- Build event-driven systems
- Implement advanced integrations
- Create enterprise-scale solutions

💡 **Key Concepts**:

- Event-driven agent communication
- Scalable system architecture
- Advanced integration patterns
- Enterprise deployment strategies

🎯 Specialization Tracks

API Integration Specialist

Focus: External service integration, API design, authentication

Key Tutorials:

- 03_openapi_tools.md
- 16_mcp_integration.md
- 24-34_pubsub_integration.md

Skills: REST API design, OAuth flows, webhook handling

Performance Optimization Expert

Focus: Speed, cost, and quality optimization

Key Tutorials:

- 05_parallel_processing.md
- 12_planners_thinking.md
- 22_model_selection.md

Skills: Parallel processing, model tuning, cost management

Enterprise Architect

Focus: Large-scale systems, observability, security

Key Tutorials:

- 17_agent_to_agent.md
- 18_events_observability.md
- 23_production_deployment.md

Skills: Distributed systems, monitoring, compliance

AI Product Builder

Focus: User experience, multimodal, real-time interaction

Key Tutorials:

- 14_streaming_sse.md
- 15_live_api_audio.md
- 21_multimodal_image.md

Skills: UX design, real-time systems, multimodal AI

📚 Learning Resources

Documentation

- **Mental Models**: Core concepts and architectural patterns
- **Tutorial Series**: 34 comprehensive implementation guides
- **Research**: ADK source code analysis and examples
- **ADK Cheat Sheet**: Quick reference guide (adk-cheat-sheet.md) for commands, patterns, and troubleshooting

Practice Projects

Beginner Projects:

- Q&A chatbot with function tools
- Data processing pipeline
- Simple API integration

Intermediate Projects:

- Multi-agent content creation system
- Real-time data analysis dashboard
- E-commerce recommendation engine

Advanced Projects:

- Enterprise document processing system
- Real-time collaborative coding assistant
- Multimodal content analysis platform

Community & Support

- **GitHub Issues**: Bug reports and feature requests
- **Stack Overflow**: Technical Q&A with `google-adk` tag
- **Discord/Slack**: Community discussions and help
- **Official Docs**: Comprehensive API reference

🎯 Progress Tracking

Skill Assessment Checklist

Core Fundamentals ☐

- [ ] Agent lifecycle understanding
- [ ] Basic tool implementation
- [ ] Local development setup
- [ ] Simple workflow patterns

Intermediate Skills ☐

- [ ] Multi-agent system design
- [ ] State management mastery
- [ ] Production testing patterns
- [ ] API integration expertise

Advanced Capabilities ☐

- [ ] Real-time streaming implementation
- [ ] Enterprise observability

- [ ] Performance optimization
- [ ] Distributed system architecture

Expert Level □

- [ ] Custom planner development
- [ ] Multimodal processing
- [ ] Production deployment mastery
- [ ] Enterprise integration patterns

🎯 Key Takeaways

1. **Structured Progression**: Follow the 8-phase learning path for comprehensive mastery
2. **Hands-on Practice**: Complete tutorial implementations alongside theoretical learning
3. **Specialization Options**: Choose focus areas based on career goals and interests
4. **Continuous Learning**: ADK evolves rapidly - stay updated with latest patterns
5. **Community Engagement**: Join discussions, contribute to open source, share knowledge

🔗 **Next**: Use the Reference Guide (reference-guide.md) for quick lookups and configuration examples. Check the Glossary (glossary.md) for definitions of key terms.

Generated on 2025-10-21 09:03:23 from learning-paths.md

Source: Google ADK Training Hub