

LangGraph 1.0 - Quick Start Guide

Installation (5 minutes)

```
# 1. Navigate to AI service directory
cd /Users/morlock/fun/psscript/src/ai

# 2. Install dependencies
pip install -r requirements.txt

# 3. Set API key
export OPENAI_API_KEY=sk-your-key-here

# 4. Test the setup
python test_langgraph_setup.py
```

Start the Service

```
# Start FastAPI server
cd /Users/morlock/fun/psscript/src/ai
python main.py

# Server runs on: http://localhost:8001
```

Test Endpoints

```
# Health check
curl http://localhost:8001/langgraph/health

# Service info
curl http://localhost:8001/langgraph/info

# Test analysis
curl -X POST http://localhost:8001/langgraph/test

# Analyze your script
curl -X POST http://localhost:8001/langgraph/analyze \
  -H "Content-Type: application/json" \
  -d '{
    "script_content": "Get-Process | Select-Object Name, CPU"
  }'
```

Python Usage

```
from agents.langgraph_production import LangGraphProductionOrches
import asyncio

async def main():
    orchestrator = LangGraphProductionOrchestrator()

    result = await orchestrator.analyze_script(
        script_content="Get-Service | Where-Object Status -eq 'Run
    )

    print("Status:", result["status"])
    print("Response:", result["final_response"])

asyncio.run(main())
```

Key Files

File	Purpose	Location
Production Orchestrator	Main LangGraph implementation	src/ai/agents/langgraph_production.py
API Endpoints	REST API routes	src/ai/langgraph_endpoints.py
Main Integration	FastAPI app	src/ai/main.py
Test Script	Verification	src/ai/test_langgraph_setup.py
Requirements	Dependencies	src/ai/requirements.txt

Documentation

- **Setup Summary:** [LANGGRAPH-SETUP-SUMMARY.md](#)
- **Implementation Guide:** [LANGGRAPH-IMPLEMENTATION.md](#)
- **Migration Plan:** [LANGGRAPH-MIGRATION-PLAN.md](#)

API Endpoints

Method	Endpoint	Purpose
POST	/langgraph/analyze	Analyze PowerShell script
POST	/langgraph/feedback	Provide human feedback
GET	/langgraph/health	Health check
GET	/langgraph/info	Service information
POST	/langgraph/batch-analyze	Batch analysis
POST	/langgraph/test	Test endpoint









Example Analysis

```
curl -X POST http://localhost:8001/langgraph/analyze \
  -H "Content-Type: application/json" \
  -d '{
    "script_content": "param([string]$Path)\nGet-ChildItem -Path $Path
  }' | jq '.'
```

Response:

```
{
  "workflow_id": "analysis_1704649200.123",
  "status": "completed",
  "final_response": "This script lists all files and folders...",
  "analysis_results": {
    "analyze_powershell_script": {...},
    "security_scan": {...},
    "quality_analysis": {...}
  }
}
```

Features

-  Type-safe state management
-  Production checkpointing
-  Human-in-the-loop support
-  Streaming responses
-  Error recovery
-  Security scanning
-  Quality analysis
-  Optimization recommendations

Troubleshooting

Import errors?

```
pip install --upgrade langgraph langchain langchain-openai
```

API key not working?

```
echo $OPENAI_API_KEY # Should show sk-...  
export OPENAI_API_KEY=sk-your-key-here
```

Service won't start?

```
# Check if port is in use  
lsof -i :8001  
  
# Use different port  
uvicorn main:app --port 8002
```

Next Steps

1. ☒ Run `test_langgraph_setup.py`
 2. ☐ Read [LANGGRAPH-IMPLEMENTATION.md](#)
 3. ☐ Review [LANGGRAPH-MIGRATION-PLAN.md](#)
 4. ☐ Test with your PowerShell scripts
 5. ☐ Deploy to staging
-

Need Help? - Documentation: `/docs/LANGGRAPH-*.md` - Code:
`src/ai/agents/langgraph_production.py` - Tests:
`src/ai/test_langgraph_setup.py`

Generated 2026-01-13 06:26 UTC