+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated on 12/09/2025

Sign up

AI Agent Training with Model Context Protocol

4 days (28 hours)

Overview

Our AI Agents with Model Context Protocol training course will enable you to understand how MCP works and create your own connected AI agents. You'll learn how to deploy a client, develop MCP servers in Python, orchestrate multiple tool calls, and integrate these agents into data or application workflows.

You'll be able to design wizards that summarize PDF files, answer business questions via SQL, or call external services, all without writing a line of code in the agent itself.

MCP enables secure, modular and transparent interaction between LLM and your internal systems, without exposing your data or keys.

At the end of this course, you'll be able to create, connect and supervise interoperable AI agents in your business environments. Like all our training courses, this one is designed around the latest stable MCP protocol specification, with up-to-date tools and SDKs.

As with all our training courses, this one will include the latest advances in AI Agents with Model Context Protocol.

Objectives

- Understand the role and architecture of AI agents connected via MCP
- Deploy a client and connect MCP servers
- Create your own tools accessible via MCP (files, databases, APIs)
- Orchestrate multi-tool AI workflows with autonomous reasoning
- Integrate an agent into a data pipeline or secure workstation

Target audience

- Data engineers
- AI / Pyhton developers
- Innovation or IT managers
- Technical analysts

Prerequisites

- Basic knowledge of Python and file/API manipulation
- Notions in generative AI or use of LLM recommended
- Knowledge of data tools (files, SQL, JSON, REST) appreciated
- No MCP protocol prerequisites: everything is introduced in the course

Program of our AI Agent Training with Model Context Protocol

Introduction to modern AI agents

- What is an AI agent: definition, components, capabilities
- Evolution from LLM assistants to autonomous agents
- Limitations of isolated vs. connected LLMs
- Agent model = LLM + tools + reasoning + context
- Use cases in data, business, research

Presentation of the Model Context Protocol (MCP)

- Protocol objectives: standardize access to tools
- MCP clients (Claude, ChatGPT, etc.) vs. MCP servers
- Technical operation: exchange of structured messages
- Client/server architecture + dynamic discovery
- Advantages over proprietary APIs (interoperability, security)

Installing and configuring your MCP environment

- SDK and tools: mcp, cmcp, Claude Desktop, OpenAI + tools
- MCP project structure (manifest, services, log)
- Launching a local MCP client (IA agent + server)
- Connecting to Claude or ChatGPT with customized tools
- Workshop: Launching a local IA agent with an MCP file server to summarize a PDF

Creating your first MCP server

- Anatomy of an MCP server
- Defining exposed methods, input/output scheme

- Configuring permissions and local security
- Log calls + structure responses
- Simple examples: file reading, text transformation

Exposing tabular and analytical data

- Connecting to a SQLite or PostgreSQL database via MCP
- Create a natural-language query-db server
- SQL error handling + query validation
- Access control to columns or schemas
- Workshop: Creating an agent that answers business questions by querying a database

Interaction with external APIs

- Calling a REST API from an MCP server
- Mapping API parameters / documentation ? MCP schema
- Managing API keys for server-side security
- Use agent to orchestrate complex calls
- Example: weather API, OpenAI, internal REST tool

Enriching a multi-tool agent

- Action chaining: reasoning + MCP call
- Step-by-step planning with Claude or ChatGPT
- Reuse of intermediate results
- Multiple MCP servers for a single task
- Workshop: Building an agent that reads a PDF file, extracts a piece of information, then queries a second tool

Advanced design: tools, chains and memory

- Structuring an enterprise toolkit
- Temporary storage of context data
- Conversational memory integration + tool calls
- Example: "Customer brief chain ? dashboard ? PDF summary".
- Managing formatting, tokens and summaries

Security and governance in MCP

- Server permissions: scoping, authorized functions
- Local vs. cloud data isolation
- Access logging and auditability
- Partitioning by user or service
- Trust model vs. LLM black box

Integration into a data/IA workflow

- Using an MCP agent as a tool in a pipeline
- Automated calls (cron, Airflow, Bash, FastAPI)
- Exporting results (CSV, database, API, e-mail, etc.)
- Workshop: Creating an automated mini-workflow orchestrated by an AI agent (reading ? query ? synthesis)

Comparison with LangChain and other frameworks

- ReAct, OpenAI agents, LangChain: key differences
- MCP vs tool + wrapper (plug-and-play vs orchestration)
- Strengths: simplicity, interoperability, security
- Current limitations of MCP (asynchronous, streaming, GPT-4o...)
- Choosing the right approach for each use case

Deployment, maintenance and prospects

- Packaging and redeploying an MCP server
- Local, cloud, container or serverless hosting
- Pooling tools in a team workspace
- MCP protocol roadmap (remote authentication, tool market)
- Full-autonomous agents + supervision

Companies concerned

This course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the forthcoming course, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical training: 60% hands-on, 40% theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, with brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire is used to check that skills have been correctly acquired.

Certification

A certificate will be awarded to each trainee who completes the training course.