Artificial Intelligence Agentic AI

---

1. Introduction

Agentic AI represents the **next evolution** of AI systems.
Earlier AI models (like chatbots, LLMs) could only respond to prompts.
Agentic AI goes beyond this — it can:

- **Take actions**
- **Make decisions**
- **Plan tasks**
- **Use tools**
- **Learn from feedback**
- **Self-correct**
- **Work autonomously**

Agentic AI is basically AI that acts like a **digital agent** rather than just a predictor.

---

2. What is Agentic AI?

Definition:
Agentic AI refers to AI systems that operate as **autonomous agents** capable of performing multi-step tasks by planning, reasoning, taking actions, and interacting with external systems or tools.

Simple meaning:
Agentic AI =
LLM + Memory + Reasoning + Tools + Autonomy

Example:
Instead of just answering your question, an Agentic AI can:

- Search the internet
- Run code

- Read a document
- Update a database
- Book appointments
- Deploy servers
- Fix its own errors

---

3. Need for Agentic AI

1 To automate complex workflows

For example:

- Build an app
- Fix code and deploy
- Do research
- Run experiments
- Process documents

2 Reduces human effort

Agents can:

- Monitor systems
- Execute tasks
- Generate reports
- Take decisions
- Trigger workflows

3 Handles long tasks end-to-end

Old chatbots could only answer short prompts.
Agents can perform tasks that take **minutes or hours** with intermediate thinking.

4 Works like a digital employee

- Research agent
- Coding agent
- Marketing agent
- Data analysis agent
- QA agent
- Automation agent

5 Scales business processes

Multiple agents working together create a fully automated workflow.

---

4. Real-World Applications of Agentic AI

A. Coding & Development

- AI writes code
- Tests it
- Fixes errors
- Deploys to cloud
- Monitors services

Tools:

- Devin AI
- Cursor IDE
- GitHub Copilot Agents

B. Customer Support

An AI agent that:

- Reads policies
- Searches knowledge base
- Responds to queries
- Escalates if needed
- Creates tickets

C. Research & Data Analysis

Agents can:

- Crawl websites
- Extract data
- Clean data
- Create visualizations
- Summarize insights

D. Business Automation

- Generate weekly reports
- Email automation
- CRM data updates
- Lead qualification

E. Personal Automation

- Manage calendar

- Book tickets
- Write emails
- Track tasks

F. Product/QA

Agent can:

- Test API endpoints
- Test UI flows
- Generate bug reports
- Verify deployments

G. Education

- AI tutor
- Agent that creates custom schedules
- Prepares practice questions

---

5. Architecture of Agentic AI

1 LLM (Core Brain)

Provides:

- Language understanding
- Reasoning
- Planning
- Problem-solving

Examples: GPT-4/5, Claude, Gemini, LLaMA.

---

2 Memory

Short-term (conversation context)
Long-term (cross-session memory)

Agent uses memory to:

- Store goals
- Track progress
- Retrieve past knowledge

---

3 Planning Engine

Break large tasks into:

- Subtasks
- Steps
- Actions
- Execution order

This creates **multi-step reasoning**.

---

4 Tools / Skills

Agents can use external tools:

- Python interpreter
- Web browser / Search
- Database
- APIs

- Code editor
- File system

---

5 Execution Layer

Agent executes:

- Code
- Commands
- Searches
- Actions in real world (IoT, robotics)

---

6 Feedback Loop

Agents evaluate:

- Did the result meet the goal?
- Is correction needed?
- Should it retry?

This loop makes them **self-correcting**.

---

6. Types of Agentic AI

1. Reactive Agents

Respond only to current input.
Old chatbots (simple rule-based).

2. Proactive Agents

Plan ahead and take action.

3. Tool-Using Agents

Use APIs, web search, code execution.

4. Multi-Agent Systems

Multiple agents collaborate.

Example:

- Research agent
- Writer agent
- Editor agent
- Fact-checking agent
  Working as a team.

---

7. Why Agentic AI Is Important (vs Traditional AI)

Feature	Traditional AI	Agentic AI
Works	Only on prompts	Autonomously
Thinking	No planning	Multi-step reasoning
Actions	None	Uses tools & APIs
Memory	Short	Long-term memory
Responsibility	Reactive	Goal-driven
Complexity	Simple	Can handle end-to-end workflows

---

8. Challenges in Agentic AI

1 Safety

Agents can take harmful actions if not controlled.

2 Hallucinations

Wrong assumptions → wrong actions.

3 Tool Misuse

Agents must not perform dangerous operations.

4 Coordination

Multi-agent systems can conflict.

5 Unpredictable behavior

Too much autonomy can cause chaos.

6 Privacy

Agents may access sensitive data.

---

9. Agentic AI Frameworks & Tools

Popular frameworks

- LangChain
- LangGraph
- Microsoft Autogen
- OpenAI Swarm
- Google Agents Framework
- CrewAI
- ReAct pattern (Reason + Act)

Industry tools

- Devin
- Cursor
- GPT Agents
- Automata
- Adept
- Fixie
- Cognition Labs

---

10. Examples of Agentic AI Workflows

1. Code Development Agent

1. Understand requirement
2. Write code
3. Run tests
4. Fix errors
5. Deploy
6. Document output

---

2. Research Agent

1. Web search
2. Extract pages
3. Summarize data
4. Compare results
5. Create a final report

---

3. Business Email Agent

1. Read incoming email
2. Determine intent
3. Open CRM
4. Update records
5. Write personalized reply

---

4. QA Agent

1. Read test plan
2. Execute test cases
3. Capture screenshots
4. Log bug report in Jira

---

11. Assignment (Hands-on)

Part A — Build a Simple Agent (Python + LangChain)

Agent should:

- Search Wikipedia
- Summarize results
- Save summary in a text file

Part B — Tool-Using Agent

Agent can:

- Accept a math question
- Write code
- Execute code
- Return final answer

Part C — Multi-Agent Workflow

Agents:

- Research agent finds data
- Writer agent converts to article
- Editor agent fixes grammar

Part D — Theory (Short Answers)

Explain in 4–5 lines:

1. What is Agentic AI?
2. What is the ReAct pattern?
3. Difference between LLM vs Agentic AI
4. Importance of tool usage
5. What is an autonomous agent?