🟥 What is Agentic AI?

---

1. 🪫 Definition of Agentic AI

- **Formal Definition:**

"Agentic AI is a type of AI that can take up a task or goal from a user and then work towards completing it on its own with minimal human guidance. It plans, takes actions, adapts to changes, and seeks help only when necessary."

- **Simplified:**
  - A **software paradigm** where you provide a **goal**, and the system **autonomously plans & executes** steps to achieve it.
  - Human involvement is **minimal**.
  - Very different from **reactive systems** like standard chatbots.

---

3. 🏝️ Example: Planning a Goa Trip

- **Generative AI (Reactive):**
  - You ask: "Best way to go to Goa on 15th?" → Gives one direct answer.
  - You ask separately about hotels, sightseeing, food → chatbot replies to each query.
  - Always **reactive**, step-by-step.
- **Agentic AI (Autonomous):**
  - You just say: *"Plan my Goa trip from 10th–15th."*
  - The system:
    - Finds best travel option.
    - Suggests hotels.
    - Plans itinerary (where to visit, eat, etc.).
    - Keeps track of everything → fully **proactive** & **autonomous**.

---

4. 🏢 Real-World Example: HR Recruiter

- Task: **Hire a backend engineer (2–4 yrs experience, remote).**
- Using an **Agentic AI chatbot**:
  1. **Understand goal** → "Hire backend engineer".
  2. **Plan** → Draft JD → Post on job portals → Monitor applicants → Shortlist → Schedule interviews → Send offers → Onboard.
  3. **Execute autonomously** → Uses tools like LinkedIn API, resume parser, calendar API, email API, HR software.
  4. **Adapt** if applicants are low → modify JD (backend → full-stack), run ads, etc.
  5. **Notify human only when needed** → e.g., approvals, risks.
  6. **Proactive monitoring** → Reports progress, triggers next steps automatically.
- Key Observation:
  - System is **autonomous, proactive, adaptive, goal-driven, and efficient**.

---

5. 🔑 Six Key Characteristics of Agentic AI Systems

1. **Autonomy**
   - Makes decisions and takes actions **without step-by-step instructions**.
   - Multiple levels:
     - Execution autonomy (runs steps sequentially).
     - Decision-making autonomy (who to shortlist).
     - Tool-use autonomy (chooses which API/tool to use).
   - Must be **controlled** via:
     - Permission scope.
     - Human-in-the-loop checkpoints.
     - Override controls (pause/stop agent).
     - Guardrails & policies (e.g., no weekend interviews, no paid ads without approval).
2. **Goal-Oriented**
   - Works persistently towards a **given objective**, not isolated queries.

- Goals can have **constraints** (e.g., remote only, budget limits).
- Stored in memory as structured JSON with:
  - Goal, constraints, status, progress, timestamps.
- Goals are **flexible** → can be altered midway (e.g., switch from hiring full-time engineer → freelancer).

3. **Planning**
   - Agentic systems work in **two iterative steps**:
     - **Plan** → Break goal into structured sub-goals & actions.
     - **Execute** → Carry out plan step-by-step.
   - Planning involves:
     - Generating multiple candidate plans.
     - Evaluating based on efficiency, tool availability, cost, risk, alignment with constraints.
     - Selecting the best plan (via human approval, policies, or automatic selection).
   - Example: Hire via job portals vs via referrals/agency.
4. **Reasoning**
   - Cognitive process: interpret info → draw conclusions → make decisions.
   - Needed in both **planning & execution**.
   - Examples:
     - Planning stage → goal decomposition, tool selection, resource estimation.
     - Execution stage → decision-making, error handling (e.g., LinkedIn API down), human-in-the-loop triggers.
5. **Adaptability**
   - Ability to modify plans & strategies in response to:
     - **Failures** (API down, tool not available).
     - **External feedback** (low job applications).
     - **Changing goals** (switch from hiring employee → freelancer).
   - Always finds alternate paths while staying aligned with goal.
6. **Context Awareness**
   - Retains and uses relevant info across steps, sessions, and environments.
   - Context types:
     - Original goal.
     - Current progress.
     - Past interactions with user.
     - Environmental state (e.g., number of applicants).
     - Tool responses (resume parser, calendar availability).
     - User preferences (format, constraints).
     - Policies/guardrails.
   - Implemented via **memory**:
     - Short-term memory → current session info.
     - Long-term memory → goals, constraints, past sessions, preferences.

---

6. 🧩 Five Core Components of Agentic AI Systems

1. **Brain (LLM)**
   - Interprets goals, performs planning, reasoning, tool selection, natural language communication.
   - Backbone of the system.
2. **Orchestrator**
   - Executes plans step-by-step.
   - Handles task sequencing, conditional routing, retries, looping, delegation (LLM ↔ human).
   - Works like the **project manager / nervous system**.
3. **Tools**
   - Interfaces with external world: APIs, databases, email, calendars, HR software, RAG knowledge base.
   - Equivalent to the **hands & legs** of the system.
4. **Memory**

    - Stores context: goals, constraints, past chats, tool outputs, progress.
    - Supports short-term & long-term storage.
    - Enables continuity across sessions.
5. **Supervisor**
    - Implements **human-in-the-loop**.
    - Handles approvals (high-risk actions).
    - Enforces guardrails & policies.
    - Escalates exceptions (e.g., special candidate outside rules).

---

7. 📊 Summary

- **Agentic AI = Autonomous, goal-driven AI systems** that plan, act, adapt, reason, and use context.
- **Six key traits:** Autonomy, Goal-Oriented, Planning, Reasoning, Adaptability, Context Awareness.
- **Five core components:** Brain (LLM), Orchestrator, Tools, Memory, Supervisor.
- This video lays the **foundation** for understanding Agentic AI before diving into **LangGraph implementations**.
