LITC
Computer Education

GET YOURSELF IN AI-AGENTIC WORLD

your journey To AI Mastery Starts Here

BASIC

Build strong AI fundamentals

(Perfect for Beginners)

INTERMEDIATE

hands-on projects &real -world applications

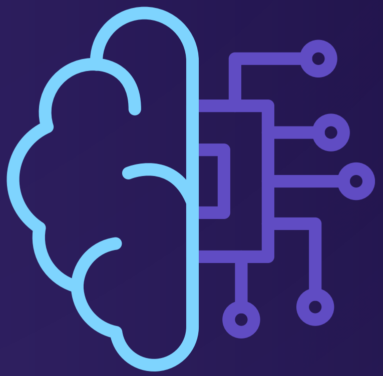
ADVANCE

Deep learning , advance models &indusrtey skills

Why join this Course ?

- Expert-led sessions
- Step-by-step learing path
- Practical projects &portfiolio building
- Future-ready skills for the AI-driven world

Limited seats Available-don't Miss Out!



☎ +91-8878855568 / +91-9522220892

✉ info@litcindore.com

join now

course content

- **Agentic AI Basic Course.**

- **Agentic AI Intermediate Course**

- **Agentic AI Advanced Course.**

Basic Course

Learn Python. Leverage AI. Build Smarter.

Course Overview -

The Agentic AI Basic Course is a hands-on, beginner-friendly program designed to teach Python programming fundamentals while integrating AI tools like ChatGPT to enhance learning, debugging, and development. Learners will gain practical coding skills, understand how to use AI responsibly, and build real-world projects.

Why This Course?

Whether you're a curious beginner or a future developer, this course blends Python fundamentals with the power of AI tools like ChatGPT. Learn to code, debug, and build smarter—with hands-on projects and real-world applications.

What You'll Learn

Setup & Environment

- Install Python across platforms
- Choose between JupyterLab vs. Notebook
- Use pip and virtual environments
- Explore the Jupyter interface

AI-Assisted Coding

- What is ChatGPT and how it works
- Prompt engineering for coding tasks
- Templates for effective prompts
- Brainstorm, debug, and refactor with AI

Python Fundamentals

- Data types: int, float, str, list, dict
- Loops, conditionals, and operators
- String manipulation and list/dict operations
- Writing and running scripts

Functions & Modularity

- Define and use functions
- Import libraries and custom modules
- Organize code into reusable components

Tools You'll Use

Tool / Platform	Purpose
Python (3.9–3.11)	Core programming language
Jupyter Notebook	Interactive coding and documentation
VS Code / PyCharm	Development environment
pip	Package management
AI assistant for coding help and debugging	AI assistant for coding help and debugging
Virtualenv / venv	Isolated Python environment

- **Hands-On Learning**

Activity	What You'll Do
Live Setup Lab	Install Python, set up IDEs, run your first script
ChatGPT Prompt Workshop	Practice writing prompts for coding help
Debugging Challenge	Use ChatGPT to fix broken code
Build a Chatbot	Create a decision-making chatbot using Python logic
Loop Tracing Exercise	Track variables through nested loops
Function Builder	Write and test reusable functions
Modular Refactor	Break a script into clean, maintainable modules
Final Project	Build a mini app using Python and ChatGPT

Outcomes

By the end of this course, you'll be able to:

✅ Set up and manage a Python development environment
✅ Write and debug Python code using ChatGPT
✅ Understand and apply core programming concepts
✅ Build modular, maintainable Python scripts
✅ Use AI responsibly in coding workflows
✅ Complete a hands-on project that showcases your skills

Bonus: Real-World Case Studies

Explore how developers use ChatGPT to:

- Refactor legacy code
- Debug collaboratively
- Accelerate prototyping and brainstorming

Ready to code smarter with AI?

Let's build something brilliant together.
Absolutely! Here's a brochure-style version of your Agentic AI Intermediate Course, designed to be visually engaging, modular, and ideal for learners ready to dive deeper into AI development concepts like embeddings, prompt engineering, and Retrieval-Augmented Generation (RAG).

Intermediate Course

Build Smarter Agents with Prompt Engineering, Embeddings & RAG

Who Is This For?

This course is designed for developers, data scientists, and AI enthusiasts who already understand Python and basic LLM usage—and are ready to level up. Learn how to build intelligent, context-aware agents using advanced prompt techniques, embeddings, and retrieval-augmented generation.

Tools You'll Use

Tool / Platform	Purpose
OpenAI Embeddings	Generate vector representations of text
FAISS / ChromaDB / Pinecone	Perform fast similarity search
LangChain / Flowise	Build modular LLM pipelines
Hugging Face Transformers	Access open-source LLMs and embedding models
t-SNE / PCA	Visualize high-dimensional embeddings

What You'll Learn

LLM Internals & Context Management

- Context windows and memory limits
- Tokenization demo: breaking inputs into tokens
- Visualization: how attention works in Transformers
- Live exploration: prompt responses across LLMs

Prompt Engineering Mastery

- Prompt tuning challenge: generate summaries with constraints
- Prompt types: zero-shot, one-shot, few-shot
- Chain-of-thought prompting
- Instruction tuning basics
- Dynamic prompt construction patterns
- Prompt engineering lab: refine prompts for clarity, tone, reasoning
- Hands-on: modular prompt assembly using LangChain or Flowise

Embeddings & Semantic Search

- Token vectors vs. sentence embeddings
- Cosine vs. Euclidean distance metrics
- Visualize embeddings using PCA/t-SNE
- Explore cosine similarity between queries and documents

Build a Vector Search Engine

- Create a basic vector search using OpenAI embeddings + FAISS
- Tools: LangChain Embedding Functions, Hugging Face, ChromaDB
- Project: Build a mini "document similarity finder"

Retrieval-Augmented Generation (RAG)

- Contextual embeddings and relevance ranking
- Customizing RAG for Q&A and summarization
- Demo: Build a document Q&A bot with RAG
- Hands-on: connect embedding model + vector store + LLM chain
- RAG tuning exercise: improve relevance, reduce hallucination
- Group debugging: why is this RAG agent failing?
- Project: Build a custom RAG-powered assistant

Hands-On Activities

Activity	What You'll Build or Explore
Tokenization Lab	Break down inputs and visualize token flow
Prompt Refinement Workshop	Tune prompts for clarity, tone, and logic
Embedding Explorer	Visualize and compare sentence embeddings
Vector Search Builder	Create a semantic search engine using FAISS
RAG Agent Debugging	Diagnose and fix hallucinations in a RAG pipeline
Final Project	Build a knowledge-aware assistant using RAG

Outcomes

By the end of this course, you'll be able to:

✅ Understand how LLMs process and respond to input
✅ Engineer prompts for precision, reasoning, and tone
✅ Use embeddings for semantic search and similarity matching
✅ Build modular LLM pipelines using LangChain or Flowise
✅ Implement and tune RAG architectures for real-world tasks
✅ Complete a hands-on project that showcases your skills

Bonus: Real-World Case Studies

Explore how teams use:

- Embeddings for document clustering and search
- Prompt engineering for customer support bots
- RAG pipelines for enterprise Q&A systems

Absolutely! Here's a brochure-style version of the Agentic AI Advanced Course, designed to showcase the depth and sophistication of the curriculum while keeping it visually engaging and easy to digest.

Advanced Course

Design Autonomous Agents. Deploy Intelligence.
Drive Impact.

Who Is This For?

This course is built for experienced developers, AI engineers, and technical product teams ready to master autonomous agent design. Learn how to architect multi-agent systems, embed memory, apply reinforcement learning, and deploy intelligent agents responsibly

Tools You'll Use

Tool / Platform	Purpose
CrewAI / AutoGen	Multi-agent orchestration
LangChain Agents	Modular agent workflows
ChromaDB / Pinecone / FAISS	Memory and vector search
OpenAI API / Hugging Face	LLMs and embeddings
LangSmith / PromptLayer	Agent tracing and evaluation
Guardrails AI / Ethics Cards Toolkit	Responsible AI design

What You’ll Learn

Agent Architectures & Collaboration

- Reactive, deliberative, and hybrid agents
- Task-oriented vs goal-oriented agents
- Multi-agent system fundamentals
- Agent communication and reasoning frameworks
- Activities:
  - Design your own agent architecture
  - Simulate agent collaboration (e.g., writer + editor)
  - Role-play agent responsibilities in a team
- Tools: CrewAI, AutoGen, LangChain Agents, AgentVerse

Goal Setting & Planning

- Hierarchical vs reactive planning
- Multi-agent strategies and reinforcement loops
- Modify plans based on changing goals
- Activities:
  - Live demo: CrewAI or LangChain PlannerAgent
  - Group scenario: Travel planner with constraints
- Tools: CrewAI, AutoGen Planning Loops, LangChain Agents

Agent Memory Systems

- Short-term vs long-term memory
- Semantic chunking and similarity
- Retrieval-Augmented Memory (RAG) loop
- Activities:
  - Build an agent with memory
  - Document chunking + vector search
  - Workshop: Improve responses with memory access
- Tools: ChromaDB, Pinecone, LangChain Memory, FAISS

Reinforcement Learning & RLHF

- Reward systems and adaptive behavior
- Feedback loops in task agents
- Hardcoded vs adaptive agent comparison
- Activities:
  - Design a reward system
  - Demo: Adaptive agent behavior

Tools: OpenAI API, LangChain + feedback loops, RL environments

Hosting & Deployment

- Cloud, serverless, and embedded options
- Latency optimization and monitoring
- Live deployment walkthrough
- Activities:
  - Deploy agent on Hugging Face or Streamlit
  - Add observability with LangSmith or OpenTelemetry
- Tools: LangChain + FastAPI, Render, Replit

Agent Evaluation & Metrics

- Task success, coherence, correctness, coverage, latency
- Logging and tracing fundamentals
- Manual vs automated evaluation
- Activities:
  - Design a custom evaluation rubric
  - Diagnose and fix broken agents
- Tools: LangSmith, PromptLayer, LangChain Tracing, Gradio logs

Responsible AI & Ethics

- Risks: bias, hallucinations, misuse
- Data privacy and consent
- Designing safety layers (rate limiting, content filtering)
- Activities:
  - Group debate: Should agents act independently?
  - Review real-world failures and ethical breaches
- Tools: OpenAI Moderation API, Guardrails AI, AI Fairness Checklist

Hands-On Project

Build a fully autonomous agent system that:

- Collaborates across roles
- Plans and adapts to changing goals
- Embeds memory and retrieves relevant context
- Responds ethically and reliably in real-world scenarios

Case Studies

Explore how advanced agentic systems are used in:

- Customer support automation
- Research assistants
- Workflow orchestration in enterprise AI