🧠 Advanced Course on Agentic AI: Single-Agent & Multi-Agent Systems

📌 **Module 1: Introduction to Agentic AI**

- **What is Agentic AI?** From Traditional AI to Autonomous AI
- **Single-Agent vs Multi-Agent Systems (MAS)**
- **Agentic AI in Large Language Models (LLMs)**
- **Applications of Agentic AI in Automation, Research, and Business**

Hands-on:

✅ Running a basic autonomous agent using OpenAI API

✅ Setting up a local agent with Python

📌 **Module 2: Building Single-Agent Systems**

- **Single-Agent AI: Concepts & Architectures**
- **Decision-Making in Single-Agent AI**
- **Memory & Long-Term Planning**
- **Integrating LLMs with Agents (LangChain, OpenAI, Hugging Face)**
- **Handling User Inputs & Actions with Tool-Use Capabilities**

Hands-on:

✅ Building a goal-driven single AI agent with LangChain

✅ Implementing memory-based reasoning with FAISS

📌 **Module 3: Multi-Agent Systems (MAS)**

- **What are Multi-Agent Systems (MAS)?**
- **Agent Communication & Coordination**
- **Role Assignment in Multi-Agent Environments**
- **Task-Oriented AI vs Autonomous Decision-Making**
- **Swarm Intelligence & Distributed Agents**

Hands-on:

✅ Creating a multi-agent research assistant

✅ Implementing a task-based workflow with CrewAI

📌 **Module 4: AI Agents with CrewAI**

- **Overview of CrewAI: AI Agents Working in Teams**
- **Role-Based Agent Assignments**
- **Orchestrating Task Execution Between Agents**
- **Implementing Workflow Pipelines with CrewAI**

Hands-on:

✅ Creating an AI-powered content generation team

✅ Using CrewAI for multi-step research tasks

📌 **Module 5: SmolAgent – Lightweight AI Agents**

- **Introduction to SmolAgent: Minimalistic AI Agents**
- **When to Use SmolAgent vs Heavyweight AI Agents**

- **Optimizing AI Agents for Cost and Performance**
- **Combining SmolAgent with LLMs for Fast Execution**

Hands-on:

✅ Deploying a SmolAgent-based chatbot

✅ Running SmolAgent on an edge device

📌 Module 6: Phi Data – Memory & Context Optimization

- **Introduction to Phi Data: AI Agent Memory & Learning**
- **Vector Database Integration (FAISS, Pinecone, ChromaDB)**
- **Retrieval-Augmented Generation (RAG) for AI Agents**
- **Personalized AI Assistants with Memory**

Hands-on:

✅ Implementing Phi Data for long-term memory

✅ Storing and retrieving context for AI-powered Q&A

📌 Module 7: Building & Deploying Agentic AI Applications

- **Deploying AI Agents on Cloud (Hugging Face, AWS, Azure)**
- **Integrating AI Agents into Web Applications (FastAPI, Streamlit)**
- **Security Considerations in Autonomous AI**

🎯 Mastering Prompt Engineering for LLMs

📌 Module 1: Introduction to Prompt Engineering

- **What is Prompt Engineering?**
  - The role of prompts in LLMs
  - How LLMs process and interpret prompts
- **Why Learn Prompt Engineering?**
  - Optimizing LLM performance
  - Reducing hallucinations & improving accuracy
  - Enhancing AI-driven applications

Hands-on:

✅ Experimenting with OpenAI's API for basic text generation

📌 Module 2: LLM Model Settings & Configurations

- **Understanding LLM Settings:**
  - Temperature, Top-k, Top-p (Nucleus Sampling)
  - Stop Tokens, Context Length & Tokenization
- **Fine-tuning vs Prompt Optimization**
- **Customizing Model Behavior with System Prompts**

Hands-on:

✅ Experimenting with different model settings in OpenAI Playground

---

📌 Module 3: Prompt Elements & Structuring

- **Key Components of a Well-Designed Prompt:**

    - Instructions
    - Context
    - Input Data
    - Output Constraints
- **Optimizing Prompts for Accuracy & Consistency**

Hands-on:

✅ Structuring prompts for summarization, classification, and question-answering

---

📌 Module 4: Prompt Engineering Techniques (Shot-Based Prompting)

- **Zero-Shot Prompting**
- **One-Shot Prompting**
- **Few-Shot Prompting**
- **Comparing Shot-Based Techniques for Different Use Cases**

Hands-on:

✅ Designing and testing different shot-based prompts on GPT models

---

📌 Module 5: Chain of Thought (CoT) Prompting

- **What is CoT Prompting?**
- **Step-by-Step Reasoning in LLMs**
- **Implementing CoT in Math, Logic & Coding Tasks**

Hands-on:

✅ Using CoT to improve reasoning-based problem-solving

📌 Module 6: Self-Consistency in Prompt Engineering

- **What is Self-Consistency?**
- **Generating Multiple Answers & Selecting the Best**
- **Improving Output Reliability with Self-Consistency**

Hands-on:

✅ Implementing Self-Consistency for multi-answer tasks

📌 Module 7: Out-of-Date Learning in Prompt Engineering

- **How LLMs Handle Outdated Information**
- **Strategies to Overcome Out-of-Date Learning:**
    - Prompting with External Data
    - Fine-Tuning vs Retrieval-Augmented Generation (RAG)
- **When to Use Updated APIs & Tools**

Hands-on:

✅ Experimenting with model responses on time-sensitive queries

---

📌 Module 8: Role-Playing in Prompt Engineering

- **What is Role-Playing in Prompting?**
- **Creating AI Personas for Specialized Tasks**
- **Enhancing Response Accuracy with Role-Based Prompts**

Hands-on:

✅ Designing AI assistants with different personas (e.g., Doctor, Lawyer, Coder)

📌 **Module 9: RAG (Retrieval-Augmented Generation) in Prompt Engineering**

- **What is RAG & Why is it Important?**
- **Integrating Vector Databases (FAISS, Pinecone, ChromaDB)**
- **Enhancing AI Responses with External Knowledge**

Hands-on:

✅ Implementing a RAG-based chatbot using FAISS & OpenAI

📌 **Module 10: ReAct (Reasoning + Acting) in Prompt Engineering**

- **What is ReAct Framework?**
- **Combining CoT + Tool Use for Autonomous Agents**
- **Building AI Agents that Reason & Execute Actions**

Hands-on:

✅ Implementing a ReAct-based agent using LangChain

📌 **Module 11: DSP (Dynamic Structured Prompting)**

- **What is Dynamic Structured Prompting (DSP)?**
- **Generating Structured & Dynamic Prompts Based on Context**
- **Using DSP for Adaptive AI Interactions**

Hands-on:

✅ Creating dynamically structured prompts for personalized AI responses

Advanced Course on Generative AI:

📌 Module 1: Introduction to Generative AI

- What is Generative AI?
- Types of Generative AI Models:
  - Text-based (GPT, LLaMA, Claude)
  - Multimodal (CLIP, DALL·E, Stable Diffusion)
- Use Cases in NLP, Image Generation, and Code Generation

Hands-on:

✅ Running a simple text-based generative model using OpenAI API

📌 Module 2: Text-Based Generative Models

- How Text-Based Models Work
- Training LLMs (Large Language Models) with Transformers
- Pretrained Models vs Fine-Tuned Models
- Popular LLMs: GPT-4, LLaMA, Mistral, Falcon

Hands-on:

✅ Generating text using Hugging Face Transformers

📌 Module 3: Multimodal Models (Text + Image + Audio)

- What are Multimodal Models?
- Combining Text & Vision Models for AI Applications
- Examples of Multimodal Models: CLIP, DALL·E, Gemini, GPT-4 Turbo

Hands-on:

✅ Running OpenAI's CLIP model for text-to-image retrieval

📌 Module 4: CLIP (Contrastive Language-Image Pretraining) Architecture

- How CLIP Works: Text-Image Pairing
- Applications of CLIP in Image Search & Generation
- Fine-tuning CLIP for Custom Tasks

Hands-on:

✅ Using CLIP to find relevant images based on text prompts

📌 Module 5: VQGAN & Taming Transformers

- Introduction to VQGAN (Vector Quantized GAN)
- How Taming Transformers Improve Image Quality
- Combining VQGAN + CLIP for AI Art

Hands-on:

✅ Generating AI Art using VQGAN + CLIP

📌 Module 6: Autoencoders & VAEs (Variational Autoencoders)

- What is an Autoencoder?
- Difference Between Autoencoders & VAEs
- Generating High-Resolution Images with VAEs

Hands-on:

✅ Implementing a simple Variational Autoencoder (VAE) in PyTorch

---

📌 Module 7: Retrieval-Augmented Generation (RAG)

- What is RAG & Why It Matters for AI?
- Enhancing LLMs with External Knowledge
- Vector Databases for RAG (FAISS, Pinecone, ChromaDB)

Hands-on:

✅ Implementing a RAG-based chatbot using LlamaIndex & FAISS

📌 Module 8: Hugging Face Ecosystem

- Overview of Hugging Face Transformers
- Fine-tuning LLMs with Hugging Face
- Deploying Models Using Hugging Face Spaces

Hands-on:

✅ Fine-tuning a text generation model on Hugging Face

📌 Module 9: CrewAI for Multi-Agent AI Systems

- What is CrewAI?
- Building Teams of AI Agents
- Role-Based Task Assignment in CrewAI

Hands-on:

✅ Setting up an AI research team using CrewAI

📌 Module 10: Groq – High-Speed AI Inference

- What is Groq?

- Running AI Models at Lightning Speed
- Optimizing Large Models for Low Latency

Hands-on:

✅ Deploying a transformer model with Groq hardware

📌 Module 11: Stable Diffusion for Image Generation

- Understanding Stable Diffusion Architecture
- Text-to-Image Generation with Diffusion Models
- Fine-Tuning & Customizing Stable Diffusion

Hands-on:

✅ Running Stable Diffusion on a local machine

📌 Module 12: GitHub Copilot for AI-Powered Coding

- How GitHub Copilot Uses AI for Code Generation
- Best Practices for Using AI in Software Development
- Comparing Copilot with Other AI Coding Tools

Hands-on:

✅ Writing AI-assisted Python scripts using GitHub Copilot

📌 Module 13: LlamaIndex – AI-Powered Document Processing

- What is LlamaIndex?
- Connecting LLMs to Private Data
- Using LlamaIndex for Enterprise AI Applications

Hands-on:

✅ Implementing LlamaIndex for a document-based AI assistant

📌 Module 14: FastAPI for AI Model Deployment

- Introduction to FastAPI for AI
- Building a REST API for LLMs
- Deploying AI Models as Web Services

Hands-on:

✅ Deploying a text-based LLM using FastAPI

Advanced Course on : LLMs

📌 Module 1: Introduction to Generative AI & LLMs

- What is Generative AI?

- Types of Generative AI Models (Text, Image, Multimodal, Speech)
- Comparison of Leading AI Models (GPT, Gemini, LLaMA, Claude, Mixtral, DeepSeek, Grok)
- Foundation Models vs Fine-Tuned Models

Hands-on:

✅ Running a basic LLM-powered chatbot using OpenAI API

📌 Module 2: OpenAI's AI Ecosystem

🟢 LangChain Framework

- Introduction to LangChain
- Building AI Agents with LangChain
- Memory & Context Handling in LangChain
- Connecting LLMs with External Data Sources

Hands-on:

✅ Implementing a chatbot using OpenAI + LangChain

🟢 OpenAI Whisper (Speech-to-Text AI)

- What is OpenAI Whisper?
- Multilingual Speech Recognition
- Building Real-World Applications with Whisper

Hands-on:

✅ Transcribing audio into text using OpenAI Whisper

📌 Module 3: Google's Generative AI Ecosystem

🟠 Gemini AI: Google's Multimodal LLM

- Introduction to Gemini AI
- Comparison with OpenAI & Meta Models
- Using Gemini API for Text & Image Generation

Hands-on:

✅ Generating text & images using Gemini API

🟠 Google Vision: AI for Image Analysis & Recognition

- What is Google Vision?
- AI-Powered Image Processing & OCR
- Building AI-powered Image Search Systems

Hands-on:

✅ Using Google Vision API for image classification

📌 Module 4: DeepSeek, Mistral, Mixtral, Grok & Claude

🟡 DeepSeek AI

- Overview of DeepSeek Language Model
- Optimizing Search and AI Retrieval with DeepSeek
- Applications in AI-Assisted Knowledge Systems

Hands-on:

✅ Running DeepSeek for document-based AI search

🟡 Mistral & Mixtral: High-Performance Open-Source AI

- What is Mistral & Mixtral?
- Dense vs Sparse Transformer Models
- Optimizing Mixtral for Multi-Task AI

Hands-on:

✅ Running Mixtral models on Hugging Face

🟡 Grok (X AI by Elon Musk)

- How Grok is Designed for Real-Time AI Processing
- Comparison with GPT, Gemini & Claude
- Use Cases of Grok in AI Chatbots & Assistants

Hands-on:

✅ Running Grok on X (formerly Twitter) API

🟡 Claude (Anthropic AI)

- What is Claude & How It Differs from Other LLMs?
- Claude's Approach to AI Safety & Constitutional AI
- Fine-Tuning Claude for Enterprise Applications

Hands-on:

✅ Building an AI-powered assistant with Claude API

📌 Module 5: Meta's Generative AI Ecosystem

🔵 LLaMA 3: Open-Source LLM by Meta

- Introduction to LLaMA 3
- Comparison with GPT-4, Gemini, Mixtral, & Claude
- Fine-Tuning & Customizing LLaMA for Specific Tasks

Hands-on:

✅ Running LLaMA 3 on a local machine using Hugging Face

🔵 Building Generative AI on Cloud

- Cloud Platforms for AI (AWS, GCP, Azure, Meta Cloud)
- Deploying LLMs on Cloud for Scalability
- Building AI-Driven Web Apps with Cloud-Based LLMs

Hands-on:

✅ Deploying an LLM-powered chatbot on Cloud

🔵 Meta's Foundation Models

- Understanding Meta's AI Foundation Models
- Pre-Trained Models vs Custom Models
- Adapting Foundation Models for Industry Use Cases

Hands-on:

✅ Using Meta's AI Models for custom NLP tasks

📌 Module 6: Fine-Tuning LLMs with Quantization, LoRA & QLoRA

🔷 Fine-Tuning Large Language Models (LLMs)

- Why Fine-Tune an LLM?
- Datasets & Preprocessing for LLM Fine-Tuning
- Fine-Tuning vs Prompt Engineering

🔷 LoRA (Low-Rank Adaptation) Fine-Tuning

- What is LoRA?
- Reducing Computation for LLM Training
- Implementing LoRA with Hugging Face & PyTorch

🔷 QLoRA (Quantized LoRA) for Efficient Model Fine-Tuning

- What is QLoRA?

- Memory Optimization for Large LLMs
- Running Fine-Tuned Models on Low-End Hardware

Hands-on:
✅ Fine-tuning a LLaMA 3 model using LoRA & QLoRA

📌 Module 7: End-to-End AI Model Deployment & Optimization

- Deploying AI Models with APIs (FastAPI, Flask)
- Optimizing AI Models for Performance & Cost
- Best Practices for AI Model Security & Governance

Hands-on:
✅ Deploying a fine-tuned LLM as a FastAPI web service

Vector database

📌 Module 1: Introduction to Vector Databases
What is a Vector Database?

Difference Between Traditional & Vector Databases

Why Use Vector Databases in Generative AI & LLMs?

How Vector Embeddings Work in AI Search & Retrieval

Hands-on:
✅ Generating vector embeddings from text using OpenAI's text-embedding-ada-002

📌 Module 2: Understanding Vector Embeddings
What Are Embeddings in AI?

How LLMs Convert Text, Images & Audio to Vectors

Similarity Metrics: Cosine Similarity, Euclidean Distance, Dot Product

Choosing the Right Embedding Model (OpenAI, Hugging Face, SentenceTransformers, BERT, etc.)

Hands-on:
✅ Generating embeddings with OpenAI, Hugging Face, and BERT models

📌 Module 3: Implementing FAISS (Facebook AI Similarity Search)
What is FAISS & How It Works?

Indexing & Searching Large-Scale Vectors with FAISS

Optimizing FAISS for Fast Retrieval

Hands-on:
✅ Implementing a FAISS-based search engine for document retrieval

📌 Module 4: Using Pinecone for Scalable AI Search
Introduction to Pinecone: A Managed Vector Database

Building Real-Time AI Search Applications with Pinecone

Comparing FAISS vs Pinecone vs Milvus

Hands-on:
✅ Creating a question-answering chatbot using OpenAI + Pinecone

📌 Module 5: ChromaDB for LLMs & AI Applications
What is ChromaDB?

How ChromaDB Works with LangChain

Building RAG (Retrieval-Augmented Generation) Pipelines with ChromaDB

Hands-on:
✅ Integrating ChromaDB with OpenAI's GPT for AI-powered search

📌 Module 6: Exploring Weaviate & Its AI Capabilities
Overview of Weaviate as a Hybrid Search Engine

Using Weaviate for Semantic Search & Knowledge Graphs

Deploying Weaviate on Cloud & Local Environments

Hands-on:
✅ Implementing a semantic search engine with Weaviate

📌 Module 7: Milvus & Qdrant for Large-Scale AI Applications
🟢 Milvus (Distributed & Cloud-Based Vector Search)
Introduction to Milvus for AI Applications

Optimizing Milvus for Large-Scale Data Processing

🟡 Qdrant (High-Performance Open-Source Vector DB)
What is Qdrant & How It Works?

Fine-Tuning Qdrant for AI Search & Recommendation Systems

Hands-on:
✅ Deploying Milvus & Qdrant for AI-driven search and recommendations

📌 Module 8: Building AI-Powered Search & RAG Applications
What is Retrieval-Augmented Generation (RAG)?

Integrating Vector Databases with LLMs for Intelligent Search

Building Enterprise AI Assistants Using Vector Databases

Hands-on:
✅ Building an RAG pipeline using LangChain, Pinecone, and OpenAI

🚀 MLOps & CI/CD Pipeline for AI & Machine Learning

📌 **Module 1: Introduction to MLOps & CI/CD for ML Models**

- **What is MLOps & Why is it Important?**
- **DevOps vs MLOps: Key Differences**
- **MLOps Lifecycle & Stages**
- **Understanding CI/CD Pipelines in AI/ML Projects**
- **Challenges in Deploying ML Models at Scale**

Hands-on:
✅ Setting up a basic **CI/CD pipeline for ML models** using GitHub Actions

---

📌 **Module 2: Containerization & Model Packaging with Docker**

- **Why Containerize ML Models?**
- **Building & Running Docker Containers for ML Applications**
- **Deploying ML Models inside Docker Containers**
- **Optimizing Containers for AI Workloads**

Hands-on:
✅ **Containerizing an ML Model with Docker & Running it Locally**

📌 **Module 3: CI/CD Pipeline for ML with GitHub Actions & Jenkins**

- **What is CI/CD & How Does It Work in ML?**
- **Setting up a CI/CD Pipeline for ML Models**
- **Automating Model Testing, Validation & Deployment**
- **CI/CD with GitHub Actions,**

Hands-on:
✅ **Implementing a CI/CD Pipeline for an AI Model Deployment Using GitHub Actions**

📌 **Module 4: MLOps with MLflow**

- **Introduction to Kubeflow for AI & ML**
- **Integrating with MLflow for Experiment Tracking**

Hands-on:

✅ **Building an ML Workflow with Kubeflow Pipelines**