Generative AI Course Curriculum (Intermediate Level)

Duration: 10 Weeks | **Mode:** Weekend | **Total Hours:** ~60
Target Audience: Working professionals, final-year students, early-career ML engineers
Prerequisites:

- Python basics (functions, loops, OOP)
- Numpy, Pandas, Matplotlib/Plotly
- Basics of Machine Learning (regression, classification, metrics)
- Basic understanding of neural networks

📌 Week 1 – Introduction to Generative AI

Objective: Build strong conceptual foundations of GenAI and LLMs.

Topics

- Overview of Generative AI & its applications
- Difference between traditional ML, DL, and Generative AI
- Types of Generative AI:
  - Text → Text (LLMs)
  - Text → Image (Stable Diffusion, DALL·E)
  - Text → Audio & Video
- Transformer architecture: high-level understanding
- Intro to OpenAI, Hugging Face, Google Gemini, Meta LLaMA
- Setting up the development environment

Hands-On

- Installing essential libraries: `transformers`, `datasets`, `langchain`, `openai`
- First LLM API call using OpenAI GPT & Hugging Face

📌 Week 2 – Foundations of LLMs & Transformers

Objective: Deep dive into how LLMs work internally.

Topics

- Tokenization & embeddings
- Transformer architecture step-by-step:
    - Self-attention mechanism
    - Positional encodings
    - Multi-head attention
    - Feed-forward networks
- Pre-training vs fine-tuning vs instruction tuning
- Understanding LLM evaluation metrics: Perplexity, BLEU, ROUGE, METEOR

Hands-On

- Visualizing tokenization & embeddings using Hugging Face
- Inspecting Transformer internals using PyTorch hooks

📌 Week 3 – Working with Hugging Face Transformers

Objective: Build LLM pipelines & generate text.

Topics

- Hugging Face Hub overview
- Using pre-trained models (`GPT-2`, `BERT`, `Falcon`, `Mistral`)
- Zero-shot, few-shot, and fine-tuned inference
- Prompt engineering basics:
    - Zero-shot prompting
    - Few-shot prompting
    - Chain-of-thought prompting

Hands-On

- Build a Q&A chatbot using Hugging Face pipelines
- Experiment with prompt engineering for different tasks

📌 Week 4 – Fine-Tuning & LoRA / QLoRA

Objective: Learn parameter-efficient fine-tuning of LLMs.

Topics

- Why fine-tune? Use cases & challenges
- Full fine-tuning vs LoRA vs QLoRA
- Quantization for efficiency
- PEFT (Parameter Efficient Fine-Tuning) with Hugging Face
- RLHF basics (Reinforcement Learning from Human Feedback)

Hands-On

- Fine-tune `GPT-2` on a custom dataset using LoRA
- Evaluate model performance & save artifacts

📌 Week 5 – RAG (Retrieval-Augmented Generation)

Objective: Enhance LLMs with private knowledge sources.

Topics

- Why RAG is needed for enterprise GenAI apps
- RAG architecture explained
- Vector embeddings & vector databases (FAISS, Pinecone, ChromaDB)
- LangChain & LlamaIndex basics

Hands-On

- Build a PDF-based RAG chatbot using LangChain
- Connect Hugging Face + FAISS + OpenAI API

📌 Week 6 – Building Agentic AI Systems

Objective: Make LLMs perform reasoning and multi-step workflows.

Topics

- What are AI Agents? How they differ from RAG
- LangChain agents vs CrewAI vs LangGraph
- Tools, memory, and planning in agents
- Autonomous vs semi-autonomous agents

Hands-On

- Build an AI Agent that:
  - Searches Google
  - Extracts real-time stock data
  - Summarizes insights into a report

📌 Week 7 – Multimodal Generative AI

Objective: Extend LLMs beyond text.

Topics

- Multimodal models: CLIP, Flamingo, Gemini, LLaVA
- Text → Image generation (Stable Diffusion, DALL·E, MidJourney)
- Text → Speech & Speech → Text (Whisper, Bark, TTS)
- Text → Video generation (Sora, Pika, Runway)

Hands-On

- Build an image captioning app using CLIP + Stable Diffusion
- Generate AI images with custom prompts

📌 Week 8 – MLOps for Generative AI

Objective: Learn deployment and monitoring.

Topics

- Challenges of deploying LLMs

- Model serving frameworks: FastAPI, Streamlit, Gradio
- Using AWS/Azure/GCP for LLM deployment
- Monitoring LLM performance (latency, cost, accuracy)
- Prompt injection & jailbreak defenses

Hands-On

- Deploy a RAG chatbot on AWS using FastAPI + Streamlit
- Integrate OpenAI API keys securely

📌 Week 9 – Enterprise Use Cases & Projects

Objective: Work on real-world Generative AI applications.

Capstone Project Options

1. **AI Resume Analyzer & Generator** (LLM + RAG)
2. **Financial Insights Agent** (LangChain + OpenAI + Yahoo Finance API)
3. **Multimodal Medical Assistant** (LLaVA + RAG)
4. **AI-Powered Marketing Copywriter** (GPT + Custom Dataset)

📌 Week 10 – Future of Generative AI + Final Project Demo

Objective: Wrap-up + prepare students for advanced research & careers.

Topics

- Latest research in LLMs (Mistral, Claude, Gemini, etc.)
- Open-source vs closed-source LLMs
- Ethical, legal, and societal implications
- Building a GenAI portfolio for jobs & startups

Hands-On

- Showcase capstone projects
- Personalized roadmap for further learning

📌 Deliverables & Extras

- **Assignments:** Weekly coding exercises
- **Quizzes:** To test conceptual clarity
- **Cheat Sheets:** For Transformers, LoRA, RAG, LangChain
- **Mini Projects:** 1–2 hours hands-on every weekend
- **Capstone Project:** End-to-end GenAI app
- **Certification:** On successful completion

📌 Tools & Tech Stack

- **LLMs:** GPT-4, LLaMA 3, Mistral, Falcon, Gemini
- **Libraries:** `transformers`, `datasets`, `peft`, `langchain`, `llama-index`
- **Vector DBs:** FAISS, Pinecone, Chroma
- **Frameworks:** FastAPI, Streamlit, Gradio
- **Deployment:** AWS, Azure, Hugging Face Spaces
- **Version Control & Tracking:** GitHub, DVC, MLflow