Generative and Agentic AI

Duration: 4 months | **Monday to Friday** | **Every day 1.5 hour**

Month 1: Prerequisites – Building Strong Foundations (Week 1-6)

1.1 Python Programming (Week 1-3)

Python Fundamentals

- Basic Data Types: int, float, string, list, tuple, dictionary, set
- Control Flow: Conditional Statements (if-else), Loops (for, while)
- Functions, Scope, Lambda Functions, Recursion
- Lists, Strings,Dictionary,Tuples,sets
- Object-Oriented Programming (OOP): Classes, Objects, Inheritance, Polymorphism

Data Handling & Libraries

- File Handling: Reading/Writing Files, Working with JSON & APIs
- NumPy Basics: Arrays, Indexing, Broadcasting, Operations
- Pandas for Data Manipulation: DataFrames, Series, Merging, GroupBy
- Matplotlib & Seaborn: Data Visualization Techniques

1.2 Basics of Machine Learning (Week 4-5)

- Fundamental ML concepts: Regression, Classification, Clustering
- Model evaluation techniques: Cross-validation, Metrics (Precision, Recall, F1-score)
- Feature Engineering and Feature Selection
- Hyperparameter tuning and Model Optimization

1.3 Neural Networks (Week 6)

- Basics of Neural Networks: Perceptron, MLP
- Activation Functions and Backpropagation
- Training Deep Learning Models with TensorFlow & PyTorch
- Implementing Neural Networks for Classification & Regression

---

Month 2: NLP, RAG, and Generative AI (Week 7-10)

2.1 Natural Language Processing (Week 7-8)

- Text processing techniques: One-hot Encoding, Bag of Words, TF-IDF

- Word Embeddings: Word2Vec, GloVe, AvgWord2Vec
- Sentiment analysis and text classification
- Named Entity Recognition (NER) and POS Tagging

2.2 Retrieval-Augmented Generation (RAG) (Week 9)

- Understanding RAG and its importance in AI applications
- Implementing RAG with vector databases (ChromaDB, Pinecone)
- Fine-tuning LLMs with retrieved knowledge
- Hands-on project: Building a Knowledge-Augmented Chatbot

2.3 Generative AI Basics (Week 10)

- Introduction to LLMs (GPT, Llama, Gemini, Claude)
- Fine-tuning generative models for text, image, and code generation
- Hands-on projects with Hugging Face, OpenAI API, and Gemini

---

Month 3: Mastering Agentic AI Frameworks (Week 11-12) & AI Deployments (Week 13)

3.1 Agentic AI - Core Concepts (Week 11)

- Definition and importance of Agentic AI
- How AI Agents work and interact with environments
- Role of LLMs in autonomous decision-making
- Overview of Autonomous AI Agents in business applications

3.2 LangChain & LangGraph (Week 12)

- Understanding Chains, Memory, and Tools in LangChain
- Implementing conversational AI with LangChain
- Workflow automation with LangGraph

3.3 Cloud Deployments (Week 13)

- Deploying AI models on Azure ML Studio
- Using GCP Vertex AI for training and inference
- Implementing Azure OpenAI solutions for business applications

---

Month 4: Real-World Agentic AI Projects (Week 14-16)

4.1 End-to-End AI Agent Projects (Week 14-15)

1. **Autonomous AI Customer Support Agent**

- Develop an AI chatbot that interacts with users, answers queries, and escalates unresolved issues.
- Integrate Retrieval-Augmented Generation (RAG) for knowledge-based responses.
- Deploy using Flask/FastAPI and host on AWS/GCP.

2. **AI-Powered Code Generator (AutoGen + LangChain)**
   - Build an AI agent that generates Python/JavaScript code based on user prompts.
   - Use LangChain and OpenAI's Codex API for intelligent code suggestions.
   - Implement debugging and code correction features.
3. **AI Financial Advisor for Investment Recommendations**
   - Train an AI model to analyze stock market trends and suggest investments.
   - Use real-time market data APIs for dynamic decision-making.
   - Build an interactive UI to display portfolio recommendations.
4. **Multi-Agent Collaboration System for Research Papers**
   - Create AI agents that research academic papers, summarize findings, and provide citations.
   - Use LLMs (GPT-4, Gemini) for information retrieval and summarization.
   - Implement an interface where users can query research topics.
5. **Personalized AI Tutor for Exam Preparation**
   - Build an AI tutor that provides custom study plans based on user strengths/weaknesses.
   - Integrate NLP for answering questions and generating quizzes.
   - Use retrieval-based learning to adapt content dynamically.

4.2 Practical AI Deployment & Scaling (Week 16)

1. **Deploying AI Agents on Kubernetes & Docker**
   - Containerize an AI application for scalable deployments.
   - Use Kubernetes to manage multiple AI agent instances.
2. **Building AI Agents for IoT Smart Home Automation**
   - Create AI agents that control smart home devices via voice/text commands.
   - Implement real-time decision-making based on environmental data.
3. **AI-Driven Automated Resume Screening System**
   - Develop an AI agent that screens job applicants based on skill matching.

- Use NLP for keyword extraction and ranking candidates.

4. **AI-Powered Fake News Detector**
    - Train a deep learning model to identify fake news from news articles.
    - Implement web scraping and real-time news verification.
5. **AI-Powered Personalized Shopping Assistant**
    - Build an AI agent that recommends products based on user preferences.
    - Integrate sentiment analysis for better recommendations.