

VERSION IT

Since 2001



Generative AI & Agentic AI with Python



Best Training And Placements Institute



#No -46, SAP street, Behind Mytrivanam,
Ameerpet, Hyderabad-500038



+91 9848015399
+91 9391237284

www.versionit.co.in

About Version IT

Version IT is not a mere software training institute, a team of IT professionals developed it as the best knowledge centre for hundreds of career-building conscious young people. Our training academy is the best training institute in Hyderabad offering various software courses with aptly placement orientation. We proudly announce that we achieved 100% placements in every batch we have taken up in the past two decades. Version IT Academy's strength is our academic excellence with which we have been placed in the top position among the software training institute in Hyderabad.



Corporate Training



Class Room Training

Online Training



Why Choose Us!

Training By Certified Instructors



Mock Interviews



**Weekly
Assignments**



Project Training



**Interview
Cracking tips**



**Resume
Preparation**

Generative AI & Agentic AI with Python

Module 1: Python

1. Introduction to Python

- Getting Started With Python
- Python Basics-Syntax
- Variables In Python
- Basics Data Types In Python
- Operators In Python

2. Python Control Flow

- Conditional Statements (if, elif, else)
- Loops

3. Data Structures Using Python

- Lists and List Comprehension
- Tuples
- Dictionaries
- Sets

4. Functions in Python

- Getting Started With Functions
- Lambda Function In Python
- Map Function In Python
- Filter Functions In Python

5. Importing, Creating Modules and Package

- Import Modules And Packages
- Standard Libraries Overview

6. File Handling

- File Operations With Python
- Working with File Paths

7. Exception Handling

- Exception Handling With try except else and finally blocks

8. OOPs

- Classes & Objects
- Single And Multiple Inheritance
- Polymorphism
- Encapsulation
- Abstraction

9. Introduction to Artificial Intelligence

- What is Artificial Intelligence (AI) in simple terms
- Why AI is important in today's industry
- Evolution of AI: from rule-based systems to intelligent systems
- Real-world examples of AI (daily life & industry use cases)
- Difference between human intelligence and machine intelligence
- Myths and facts about Artificial Intelligence

10. High-Level View of Data Science, Artificial Intelligence & Machine Learning

- What is Data Science and what problems it solves
- Role of data in AI and ML systems
- Relationship between:
 - Data Science
 - Artificial Intelligence

- **Machine Learning**
- **Deep Learning (high-level only)**
- **Where AI fits in the overall technologies system**
- **Simple industry examples mapping DS→ML→AI**
- **Use cases comparison (when to use DS vs ML vs AI)**

11. Terms & Terminologies of Artificial Intelligence

- **Common AI terms explained in simple language:**
- **Dataset**
- **Features**
- **Model**
- **Training & Testing**
- **Algorithm**
- **Supervised vs Unsupervised learning (conceptual level)**
- **Model accuracy and performance (basic understanding)**
- **Bias, fairness, and limitations of AI (intro level)**
- **Difference between automation and intelligence**

12. Understanding and-to-End Generative AI Pipeline & Implementation Cycle

- **What is Generative AI (simple explanation)**
- **Examples of Generative AI applications (text, image, code-concept only)**
- **High-level flow of a Generative AI system:**
- **Data collection**
- **Data preparation**
- **Model selection**
- **Training process**
- **Evaluation**
- **Deployment**
- **Understanding prompt → response flow**
- **Basic idea of implementation cycle in real projects**
- **Challenges in real - world AI implementation (high level)**

Module 2: Python for Artificial Intelligence

1. Numerical Operations with NumPy

- Introduction to NumPy and its role in AI
- Creating arrays(1D,2D,multi-dimensional)
- Difference between Python lists and NumPy arrays
- Basic numerical operations:
- Addition, subtraction, multiplication, division
- Array indexing and slicing
- Shape, size, and data types of arrays
- Broad casting concept (high-level understanding)
- Performance benefits of NumPy in AI applications

2. Statistical Operations using NumPy

- Understanding why statistics is important in AI
- Measures of central tendency:
- Mean
- Median
- Mode(conceptual)
- Measures of dispersion:
- Variance
- Standard deviation
- Min, max, sum, cumulative operations
- Working with axis-based operations
- Handling missing or in valid values(intro level)
- Real-world data interpretation using statistics

3. Loading Data with Pandas

- Introduction to Pandas library
- Why Pandas is used in data-driven AI systems
- Loading data from different sources:
- CSV files
- Excel files (conceptual)

- Understanding rows, columns, and indexes
- Previewing data using head(),tail(),info()
- Basic data in section and validation

4. Data Frames and Operations on DataFrames

- What is a Data Frame and why it is powerful
- Selecting rows and columns
- Filtering data using conditions
- Updating and modifying Data Fram evaluates
- Adding and removing columns
- Handling missing data(basic techniques)
- Sorting and grouping data (intro level)
- Understanding real-world tabular data sets

5. Merging, Joining, and Concatenating Data Frames

- Why combining data sets is required in AI projects
- Understanding relationships between data sets
- Types of joins:
 - Inner join
 - Left join
 - Right join
 - Outer join
- Concatenation of Data Frames (row-wise & column-wise)
- Real-world examples like student, marks, and attendance data
- Common mistakes while merging data sets

6. Data Visualization using Matplotlib

- Importance of visualization in AI and analytics
- Introduction to Matplotliblibrary
- Creating basic plots:
 - Lineplots

- **Bar charts**
- **Histograms**
- **Scatterplots**
- **Customizing plots (titles, labels, legends)**
- **Understanding trends and patterns visually**
- **Interpreting visual insights for decision-making**

7. Data Visualization using Seaborn

- **Difference between Matplotlib and Seaborn**
- **Why Seaborn is preferred for statistical visualization**
- **Creating advanced plots:**
 - **Box plots**
 - **Violin plots**
 - **Heatmaps**
 - **Visualizing distributions and relationships**
 - **Styling and aesthetics for professional graphs**
 - **Comparing multiple variables visually**

Module 3: Machine Learning

1. Introduction to Machine Learning

- **What is Machine Learning in simple terms**
- **Difference between traditional programming and ML**
- **Why Machine Learning is important in AI systems**
- **Types of Machine Learning (high-level overview)**
- **Real-world examples of ML in daily life**
- **ML work flow overview (data → model → prediction)**

2. Applications of Machine Learning

- **Machine Learning in:**
 - **Healthcare**
 - **Finance**
 - **E-commerce**
 - **Education**

- **Transportation**
- **Recommendation systems (conceptual understanding)**
- **Fraud detection and risk analysis**
- **Image, speech, and text-based applications**
- **How companies use ML for decision-making**

3. Supervised Machine Learning

a) Classification

- **What is classification and where it is used**
- **Binary classification vs multi-class classification**
- **Examples:**
- **Spamvsnon-spam**
- **Passvsfail**
- **Understanding decision boundaries (conceptual)**
- **Real-world classification use cases**

b) Regression

- **What is regression and when to use it**
- **Predicting continuous values**
- **Examples:**
- **House price prediction**
- **Salary prediction**
- **Difference between classification and regression**
- **Understanding prediction error(basic idea)**

4. Unsupervised Machine Learning

- **Understanding unlabeled data**
- **Why unsupervised learning is required**
- **Concept of clustering**
- **Pattern discovery in data**
- **Dimensionality reduction (intro level)**
- **Real-world examples like customer segmentation**

5. Reinforcement Learning

- Learning through interaction with environment
- Concept of agent, environment, action, reward
- Trial-and-error learning approach
- Real-world examples:
 - Games
 - Robotics
- Difference between supervised and reinforcement learning

6. Latest Advances in Machine Learning

- Overview of recent trends in ML (conceptual)
- Role of ML in modern AI applications
- Integration of ML with big data and cloud platforms
- Industry demand and future scope of ML
- Ethical considerations and responsible ML (intro level)

7. Model Representation

- What is a model in Machine Learning
- Mathematical vs logical representation (high-level)
- Understanding features, weights, and parameters
- How models learn patterns from data
- Visualizing model behavior conceptually

8. Model Evaluation

- Training accuracy vs testing accuracy
- Why model evaluation is important
- Concept of overfitting and underfitting
- Common evaluation metrics (intro):
 - Accuracy
 - Precision
 - Recall
- Choosing the right metric for the problem

9. Hyperparameter Tuning of Machine Learning Models

- Difference between parameters and hyper parameters
- Why tuning is required
- Effect of hyper parameters on model performance
- Manual tuning vs automated tuning (conceptual)
- Trade-off between performance and complexity

10. Evaluation of ML Models

- Cross-validation concept (high-level)
- Comparing multiple models
- Bias-variance trade off(intro level)
- Selecting the best-performing model
- Selecting the best-performing model
- Interpreting evaluation results

11. Estimating and Prediction of Machine Learning Models

- Difference between estimation and prediction
- Making predictions on unseen data
- Confidence in model outputs
- Understanding prediction errors
- Practical use of predictions in real systems

12. Deployment Strategy of ML Models

- What does it mean to deploy an ML model
- Offline vs online deployment (conceptual)
- Model lifecycle after deployment
- Monitoring model performance
- Updating and retraining models
- Real-world ML deployment challenges

Module 4: Natural Language Processing

1. Text Analytics

- What is text data and why it is different from numeric data
- Importance of text analytics in AI applications
- Common sources of text data:
 - Emails
 - Reviews
 - Social media posts
- Basic text processing workflow
- Understanding unstructured data
- Real-world use cases of text analytics

2. Stemming, Lemmatization and Stop Word Removal

- Stemming:
 - Concept of word reduction
 - Example-based understanding
- Lemmatization:
 - Difference between stemming and lemmatization
 - Importance of root words
- Stopwords:
 - What are stop words
 - Why they are removed
 - Impact of preprocessing on model performance

3. POSTagging and Named Entity Recognition

- What is Part-of-Speech (POS) tagging
- Identifying nouns, verbs, adjectives, etc.
- Importance of grammatical structure in NLP
- Named Entity Recognition (NER):

- Identifying names, locations, organizations
- Practical use cases:
 - Resume parsing
 - News analysis
- High-level understanding of how machines understand language

4. Bigrams, Ngrams and Collocations

- Why single words are sometimes not enough
- Understanding words sequences
- Bigrams:
 - Two-word combinations
- Ngrams:
 - Multiple word combinations
- Collocations:
 - Frequently occurring word pairs
- Improving context understanding using ngrams
- Real-world examples like search queries and chatbots

5. Term Document Matrix

- What is a document and what is a term
- Converting text into numerical format
- Structure of a term-document matrix
- Rows as documents and columns as terms
- Importance of matrix representation in NLP
- How ML models consume text data

6. Count Vectorizer

- Purpose of Count Vectorizer
- Converting text into word counts
- Vocabulary creation concept
- Sparse matrix understanding (high level)
- Advantages and limitations of count-based approaches
- Simple examples for clear understanding

7. Term Frequency and TF-IDF

- Understanding importance of words in a document
- Limitations of simple word counts
- What is TF-IDF and why it is used
- Difference between TF and TF-IDF
- How TF-IDF improves text representation
- Use of TF-IDF in real-world NLP models

Module 5: Deep Learning

A. Deep Learning – Core Concepts

1. TensorFlow & Keras Installation

- Introduction to deep learning frameworks
- Why TensorFlow and Keras are used in industry
- Underst and in genvironment setup requirements
- Verifying installation and basic configuration
- Understanding decision boundaries (conceptual)
- Overview of TensorFlow-Keras workflow

2. Cost Function (Loss Function) – Detailed Understanding

- What is a cost(loss)function
- Why cost function is critical in deep learning
- Relationship between prediction and error
- Common intuition behind loss minimization
- How cost function guides model learning

3. Measuring Accuracy of Hypothes is Function

- Measuring correctness of predictions
- What is a hypothes is in deeplearning
- Difference between loss and accuracy
- Understanding model performance indicators
- Interpreting accuracy values in real projects

4. Role of Gradient Function in Minimizing Cost Function

- Concept of gradients in simple terms
- How gradients show direction of improvement
- Gradient descent intuition
- Role of learning rate (high-level idea)
- How models improve step-by-step

5. Explicit Discussion of Bayes Models

- Introduction to probabilistic thinking
- Concept of prior and posterior probability
- Why Bayes models are important in AI
- Use of Bayes models in prediction problems
- Relationship between probability and learning

6. Hidden Markov Models (HMM)

- What is a Hidden Markov Model
- States, observations, and transitions
- Sequential data understanding
- Where HMMs are used (speech, text, sequences)
- High-level intuition behind hidden states

7. Optimization Basics

- Why optimization is required in deep learning
- Objective of optimization algorithms
- Local minima vs global minima (conceptual)
- Trade-off between speed and accuracy
- Overview of optimization challenges

8. Sales Prediction of a Gaming Company using Neural Networks

- Understanding business problem formulation
- Input features and output prediction
- Mapping business data to neural networks
- Training and evaluating the prediction model

- Interpreting results for decision-making

9. Build an Image Similarity Engine

- What is image similarity
- Feature extraction from images
- Comparing images using learned representations
- Practical applications(search, recommendations)
- End-to-end understanding of similarity systems

B. Deep Learning with Convolutional Neural Networks (CNNs)

1. Architecture of CNN

- Why CNNs are used for images
- Flow of data through a CNN
- Understanding feature learning visually
- Difference between traditional NN and CNN

2. Types of Layers in CNN

- Convolution layers
- Pooling layers
- Activation layers
- Fully connected layers
- Role of each layer in feature extraction

3. Different Filters and Kernels

- What are filters and kernels
- How kernels detect patterns(edges, shapes)
- Kernel size and its effect
- Learning filters automatically
- Visual intuition of convolution operation

4. Building an Image Classifier With and With out CNN

- Traditional image classification approach
- Limitations of non-CNN methods
- Advantages of CNN-based classifiers
- Comparing performance and accuracy
- Real-world image classification examples

C. Recurrent Neural Networks (RNNs)

1. Fundamental Notions & Ideas

- Why sequence data is different
- Concept of memory in neural networks
- How RNNs process sequential information

a) Recurrent Neurons

- Feedback connections
- Hidden state concept
- Passing information through time

b) Handling Variable Length Sequences

- Why sequence length varies
- Challenges in processing sequences
- Conceptual handling of padding and truncation

2. Training a Sequence Classifier

- Classification of sequential data
- Examples : text, signals, events
- Training process overview
- Evaluating sequence models

3. Training to Predict Time Series

- What is time series data
- Forecasting future values
- Understanding trends and patterns
- Practical applications(sales, demand, signals)

Module 6: Generative AI

1. Introduction to Generative AI

- What Generative AI means in simple words
- Difference between predictive AI and generative AI
- How Generative AI creates new content in stead of just predictions
- Types of content generated:
 - Text
 - Images
 - Code
- Real-world applications of Generative AI
- Business impact of Generative AI
- Limitations and responsible use(high-level understanding)

2. Getting Started with Lang Chain

- Why Lang Chain is used in Generative AI projects
- Understanding LLM-based application flow
- Role of Lang Chain in:
 - Prompt handling
 - Model orchestration
 - Applicationlogic
- Setting up development environment:
 - Python environment
 - Required libraries
- Understanding basic Lang Chain components
- How Lang Chain simplifies Gen AI development

3. Hugging Face Integration

- **What Hugging Face provides in the Alecosystem**
- **Using pre-trained models for faster development**
- **Integrating Hugging Face models in to Lang Chain workflows**
- **Understanding model selection based on use case**
- **Customizing models for domain-specific tasks**
- **Concept of fine -tuning (high-level understanding)**
- **Performance considerations when using pre-trained models**

4. Building Generative AI Applications

- **End-to-end flow of a Generative AI application**
- **Prompt design and prompt improvement concepts**
- **Handling user input and model output**
- **Building applications such as:**
 - **Chatbots**
 - **Content generators**
 - **Question-answering systems**
- **Data augmentation using generative models**
- **Error handling and response validation**
- **Making applications user-friendly and reliable**

5. Deployment Strategies

- **Why deployment is critical for Generative AI systems**
- **Different deployment approaches:**
 - **Local deployment**
 - **Cloud deployment**
 - **On-premise deployment**
- **Scalability challenges in Gen AI applications**
- **Managing latency and performance**
- **Monitoring deployed models**
- **Updating and improving models after deployment**
- **Basic cost considerations in production**

Module 7: Retrieval-Augmented Generation

1. RAG Pipelines

- What is Retrieval-Augmented Generation in simple terms
- Why pure LLMs are not enough for real-world applications
- How RAG improves accuracy and reduces hallucinations
- Core components of a RAG pipeline:
 - User query
 - Retriever
 - Knowledge source (documents/ data)
 - Generator (LLM)
- Understanding the flow: Query → Retrieve → Augment → Generate response
- Difference between traditional search and RAG
- Use cases of RAG:
 - Chatbots with company data
 - Document-based Q&A systems
 - Knowledge assistants
- Benefits of combining retrieval with generation

2. Combining Generative Models with Retrieval Systems

- Why external knowledge is required for AI systems
- Role of embeddings in retrieval (conceptual understanding)
- Storing and searching information efficiently
- How retrieved data improves response quality
- Handling dynamic and frequently changing data
- High-level understanding of vector-based search
- Practical challenges in RAG systems

3. Optimizing AI Models

- Why optimization is critical after deployment
- Monitoring AI model performance in production
- Understanding latency, accuracy, and cost trade-offs
- Identifying performance bottlenecks
- Improving response quality over time

- Handling real-world user behavior and queries

4. Maintaining and Updating AI Systems

- Model life cycle management
- Updating knowledge sources without retraining models
- Handling data drift and model drift(intro level)
- Versioning models and prompts
- Ensuring system reliability and availability
- Best practices for long-term AI system maintenance
- Ethical and responsible AI considerations(high level)

Module 8: Agentic AI

Introduction to Agentic AI

- What is Agentic AI in simple terms
- Difference between:
 - Generative AI
 - RAG systems
 - Agentic AI
- What makes an AI system an agent
- Characteristics of AI agents :
 - Autonomy
 - Goal-oriented behavior
 - Decision-making ability
- Real-world motivation for Agentic AI
- Examples of agent-based systems in industry

Core Concepts of AI Agents

- Understanding the concept of an Agent
- Agent components:
 - Perception(input)
 - Reasoning
 - Action

- **Environment and interaction**
- **Agent goals and constraints**
- **Stateless vs stateful agents**
- **Reactive vs deliberative agents**

Agent Architecture

- **High-level architecture of an AI agent**
- **Planning, reasoning, and execution loop**
- **Memory in agents:**
 - **Short-term memory**
 - **Long-term memory**
- **Role of tools and APIs in agents**
- **Error handling and recovery strategies**

Agent Reasoning and Decision Making

- **How agents decide “what to do next”**
- **Chain-of-Thought reasoning (conceptual)**
- **Task decomposition:**
 - **Breaking big tasks into smaller tasks**
- **Planning vs execution**
- **Handling uncertainty and incomplete information**
- **Feedback-based improvement**

Tool-Using Agents

- **Why agents need tools**
- **Types of tools agent scan use:**
 - **APIs**
 - **Databases**
 - **Filesystems**
 - **Search engines**
- **Tools election and execution flow**
- **Tool failure handling**
- **Combining multiple tools in one agent**

Multi-Step and Multi-Task Agents

- **Single-step vs multi-step agents**
- **Task sequencing and dependency handling**
- **Managing long-running tasks**
- **Agent loops and stopping conditions**
- **Avoiding infinite loops and unsafe actions**

Multi-Agent Systems

- **What are multi-agent systems**
- **Agent collaboration and coordination**
- **Role-based agents (planner, executor, verifier)**
- **Communication between agents**
- **Conflict resolution between agents**
- **Use cases of multi-agent architectures**

Agentic AI with Lang Chain/Agent Frameworks

- **Role of Lang Chain in agent development**
- **Understanding agent executors**
- **Prompt-driven vs rule-driven agents**
- **Memory integration with agents**
- **Using tools with Lang Chain agents**
- **High-level overview of popular agent frameworks**

Agentic AI with RAG

- **Why RAG is important for agents**
- **Agents querying knowledge bases**
- **Decision-making using retrieved context**
- **Updating knowledge dynamically**
- **Agent+RAG workflow understanding**

Building Agentic AI Applications

- **Designing an agent-based solution**
- **Defining goals, tools, and constraints**

- **End-to-end agent work flow**
- **Example applications:**
 - **AI personal assistant**
 - **Autonomous research agent**
 - **Customer support agent**
 - **Task automation agent**
- **Evaluating agent performance**

Deployment and Monitoring of AI Agents

- **Challenges in deploying autonomous agents**
- **Monitoring agent actions**
- **Logging and observability**
- **Safety and guardrails**
- **Cost control and rate limiting**
- **Updating agents without breaking workflows**

Ethics, Safety, and Responsible Agentic AI

- **Risks of autonomous agents**
- **Preventing harmful actions**
- **Human-in-the-loop concepts**
- **Permission boundaries for agents**
- **Responsible AI practices**

Module 9: End-to-End AI Project with MLOps & CI/CD

End-to-End AI Project Overview

- **Understanding real-world AI project lifecycle**
- **Problem statement definition and requirement analysis**
- **Data collection and data preparation**
- **Model selection and training workflow**
- **Evaluation and validation of the model**

Module 10: Prompt Engineering

TOPICS

- Prompt design fundamentals
- Zero-shot vs Few-shot prompting
- Chain-of-Thought (CoT)
- Role-based prompting
- Output formatting techniques
- Prompt optimization strategies

Module 11 : LangChain Framework (Core Development)

TOPICS

- Lang Chain architecture
- Components:
- LLMs, Prompts, Chains
- Memory:
- Buffer, summary, window memory
- Output parsers
- Runnable & pipelines
- Tool integration