INDEX

1. Introduction to Python
2. Data Structures and Algorithms
3. File Handling and Data Processing
4. Object-Oriented Programming (OOP)
5. Modules and Packages
6. Exception Handling and Debugging
7. Advanced Python Concepts
8. Working with Libraries & Testing
9. Generative AI

Our Mission & Vision

Our Mission

Our mission is simple — to help you upgrade your knowledge and stay relevant in the fast-changing IT industry. As technology evolves, staying updated is not just important, it's necessary to survive and grow. With our program, you'll build the right skillsets to sustain your career confidently. A simple rule we follow: The more you learn, the faster you earn.

Our Vision

Our vision is to UPSKILL Professionals and prepare them for real-world industry needs. We want to build a strong, skilled workforce that not only grows in their careers but also contributes to the growth of the country. Our goal is to create a future-ready, job-focused, practical training that makes a real difference in people's lives.

Module 1: Introduction to Python

Module 1: Basics of Python Programming

1.1-Installation and Setup:
1.2-Installing Python and setting up a development environment (IDEs like PyCharm, VSCode, Jupyter Notebooks)
1.3-Syntax and Basic Constructs:
1.4-Variables and data types (integers, floats, strings, booleans)
1.5-Basic input and output
1.6- Comments and documentation

Control Structures

1.7- Conditional Statements:
1.8- If, elif, else Loops:
1.9- For, while
2.0- Loop control statements (break, continue, pass)

Functions

2.1- Defining Functions:
2.2- Parameters and return values Scope and Lifetime:
2.3- Local and global variables
2.4- Lambda Functions:
2.5- Anonymous functions

Module 2: Data Structures and Algorithms

Core Data Structures

0.2- Lists:
2.1-Creating, accessing, modifying, and iterating over lists
2.2-List comprehensions Tuples:
2.3-Creating and using tuples
2.4- Unpacking tuples Sets:
2.5- Creating and using sets
2.6- Set operations (union, intersection, difference) Dictionaries:
2.7- Creating and using dictionaries

Advanced Data Structures

2.8-Collections Module
2.9-defaultdict, Counter, OrderedDict, deque

Module 3: File Handling and Data Processing

File Operations

3.1- Reading and Writing Files:
3.2- Opening, reading, writing, and closing files Working with different file modes (r, w, a, rb, wb)
3.3- Working with CSV and JSON:
3.4-Reading from and writing to CSV and JSON files using csv and json modules

Module 4: Object-Oriented Programming (OOP)

OOP Basics

4.1- Classes and Objects:
4.2-Defining classes and creating objects
4.3- Instance variables and methods Class Variables and Methods:
4.4-Using class variables and class methods Inheritance:
4.5-Single and multiple inheritance
4.6-Polymorphism and Encapsulation:
4.7-Method overriding
4.8-Private variables and name mangling

Module 5: Modules and Packages

Modular Programming

5.1-Creating and Importing Modules:
5.2-Defining and using modules Packages:
5.3-Organizing code into packages
5.4-Importing from packages

Module 6: Exception Handling and Debuggin

Error Handling

6.1- Exception Types:
6.2-Common exceptions (ValueError, TypeError, etc.)
6.3-Try, Except Blocks:
6.4-Using try, except, else, and finally
6.5-Custom Exceptions:
6.7-Creating and raising custom exceptions

Debugging

6.8-Debugging Techniques:
6.9-Using print statements and logging
6.10-Using debuggers (pdb)

Module 7: Advanced Python Concepts

Decorators:

7.1- Function decorators
7.2- Class decorators Context Managers:
7.3- Using with statement
7.4- Creating custom context managers with __enter__ and __exit__
7.5- Iterators and Generators

Iterators:

7.6- Creating iterators using __iter__ and __next__ Generators:
7.7- Creating generators with yield
7.8- Generator expressions

Module 8: Working with Libraries

Scientific Computing

NumPy:

8.1 Arrays and matrix operations Pandas:
8.2-DataFrames for data manipulation
8.3-Reading and writing data (CSV, Excel)
8.4- Data Visualization

Matplotlib:

8.5 Plotting graphs and charts Seaborn:
8.6 Statistical data visualization

Module 9 : Testing & Generative AI

Unit Testing

9.1- Unittest Framework:
9.2- Writing and running tests Test fixtures and test suites

GenAI and It's Industry Applications

Introduction to Generative AI
AI vs ML vs DL vs NLP vs Generative AI
Generative AI principles
What is the role of ML in Gen-AI
Different ML techniques (Supervised, Unsupervised, Reinforcement & Reinforcement Learning)
Applications in various domains
Ethical considerations

NLP & Deep Learning

NLP essentials
Basic NLP tasks
Different text classification approaches
Frequency based - Bag of words,TF-IDF, N-gram.
Distribution Models - CBOW, Skipgram(Traditional approaches) and word2vec, Glove.
ML Methods (Random Forest, Gradient Boosting, AdaBoost) & Traditional Machine Learning Models- Naive Bayes, Support Vector Machine (SVM), Decision Trees, Logistic Regression.
Deep learning techniques - CNNs, RNNs, LSTMs, GRU and Transformers.

Generative AI Models

Autoencoder
VAE's and applications
GAN's and it's applications
Different types of GAN's and applications

Language Models & Transformer Models

Different types of Language models
Applications of Language models
Transformers and its architecture
BERT, RoBERTa, SPT variations
Applications of transformer models

Prompt Engineering

What is Prompt Engineering
What are the different principles of Prompt Engineering
Types of Different Prompt Engineering Techniques
How to Craft effective prompts to the LLMs
Priming Prompt
Prompt Decomposition

Large Language Models

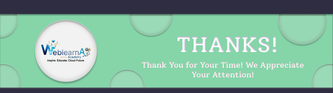
Generative AI lifecycle
What is BLMF
LLM pre-training and scaling
Different Fine-Tuning techniques

Different Cloud Metrics

What is Chunking
What is the use of chunking the document
What are the traditional effective chunking techniques
What are the problems and limitations with traditional chunking techniques?
How to overcome the limitations of Traditional chunking
Advanced Chunking Techniques:
1. Character Splitting
2. Recursive Character Splitting
3. Document based Chunking
4. Semantic Chunking
5. Agentic Chunking

Vector Databases

LlamaIndex
What are Vector Databases
Why do we prefer Vector Databases over Traditional Databases
Different Types of Vector Databases: OpenSource and Close Source
OpenSource: Chroma DB, Weaviate,Faiss,Qdrant
Different Types of Vector Databases: OpenSource and Close Source
Close-Source Vector Databases:Pinecone,AzureDB,Cloud-Based Solutions

RAG and Advanced RA with Langchain

What is RAG
What are the main components of RAG
High level architecture of RAG
How to Build RAG using external data sources
Advanced RA