# Pre-requisite

* Basic understanding of Python and Java
* Basic understanding of Data Engineering

**Total Duration** - 48 Hours. 8 Hours per day

# Day 1 - Python

1. Understanding of Python
   1. What is python?
   2. Python documentation and help
2. Python Environment Set-up and Installation
3. Jupyter Notebook Overview
4. Python Basics
   1. Data Types
      1. Numbers
      2. Strings
      3. Print Formatting
      4. Lists
      5. Dictionaries
      6. Booleans
      7. Tuples and Sets
5. Python Operators
6. If, elif and else Statements
7. Loops in Python
8. Errors & Exceptions
   1. Try – except
   2. Assert, Raise
   3. Finally
9. File handling
   1. Syntax
   2. How to Open File
   3. Read Lines
   4. Write to an Existing File
   5. Create a New File
   6. Delete a File
10. Using NumPy Package in Python
    1. Why use NumPy?
    2. Numpy Arrays
    3. Numpy Array Indexing
    4. Numpy Array Manipulation
    5. Numpy Operations
    6. Various useful Numpy functions
    7. Broadcasting
    8. Numpy Statistical Functions
    9. Numpy Mathematical Functions

# Day 2 - Python

1. Using Pandas Package in Python
   1. Series
   2. DataFrames
   3. Missing Data Treatment
   4. Groupby
   5. Merging Joining and Concatenating
   6. Read Excel, JSON, XML files
   7. Data Input and Output
2. Python Performance Tuning and Best Practices
3. Python Regex
   1. What is Regex
   2. How to use Regex in Python
   3. Examples of Regex in Python
4. Python - MySQL Database Access
   1. Introduction
   2. Steps for DB API
   3. What is MySQLdb?
   4. How do I Install MySQLdb?
   5. Install MySQLdb
   6. Database Connection
   7. Creating Database Table
   8. INSERT/Read/Update/Delete Operations
5. Debugging in Python
6. Logging in Python
   1. Why Use the logging Module
   2. Creating a Simple Logger
   3. Logging Exceptions
   4. Adding additional LogRecord attributes
   5. Formatted Log Stamp
   6. Table of Logging Levels
   7. Default Logging Level

# Day 3 - Python

1. Python Packaging and Virtual Environment
   1. Background of Packaging
   2. Why we need to create Virtual Environment?
   3. Managing Packages with PIP
   4. Installing Packaging in Virtual Environment
      1. Installing PIP on Windows
      2. Installing Virtualenv
      3. Creating Virtual Environment
      4. Activating Virtual Environment
      5. Leaving/Deactivate Virtual Environment
      6. Installing Packages
   5. Using Requirement Files
   6. Freezing Dependencies
   7. Installing packages from Requirements file
2. Deployment of Python Project
   1. Deployment tools
   2. Deployment resources
   3. Deployment learning checklist
   4. Create The Python Application
   5. Push Code to Repository
   6. Create Pipelines Application
   7. Deploy Steps
   8. PreInstall
   9. PostInstall
   10. Re-Build Application
   11. Deploy Application

# Day 4 - Scala

1. Understanding of Scala
   1. What is Scala?
   2. Scala documentation and help
2. Scala Environment Set-up and Installation
3. Working with Data
   1. Values
   2. Variables
   3. Naming
   4. Types
      1. Numeric Data Types
      2. Strings
      3. An Overview of Scala Types
      4. Tuples
4. Expressions and Conditionals
   1. Expressions
      1. Defining Values and Variables with Expressions
      2. Expression Blocks
      3. Statements
   2. If..Else Expression Blocks
      1. If Expressions
      2. If-Else Expressions
   3. Match Expressions
      1. Matching with Wildcard Patterns
      2. Matching with Pattern Guards
      3. Matching Types with Pattern Variables
   4. Loops
      1. Iterator Guards
      2. Nested Iterators
      3. Value Binding
      4. While and Do/While Loops
5. Functions
   1. Procedures
   2. Functions with Empty Parentheses
   3. Function Invocation with Expression Blocks
   4. Calling Functions with Named Parameters
   5. Parameters with Default Values

# Day 5 - Apache Spark with Python and Scala

1. What is Apache Spark
2. Running Spark
   1. Downloading Spark Locally
   2. Launching Spark’s Interactive Consoles
3. An Introduction to Apache Spark
   1. Spark’s Basic Architecture
   2. Spark’s Language APIs
   3. Starting Spark
   4. The SparkSession
   5. DataFrames
      1. Partitions
   6. Transformations
      1. Lazy Evaluation
   7. Actions
4. Spark UI
5. Spark’s Toolset
6. Structured API
   1. DataFrames and Datasets
   2. Schemas
   3. Overview of Structured Spark Types
      1. DataFrames Versus Datasets
      2. Columns
      3. Rows
      4. Spark Types
   4. Overview of Structured API Execution
      1. Logical Planning
      2. Physical Planning
      3. Execution
7. Basic Structured Operations
   1. Schemas
   2. Columns and Expressions
   3. Records and Rows
   4. DataFrame Transformations
8. Aggregations
   1. Aggregation Functions
      1. count
      2. min and max
      3. sum
      4. avg
   2. Grouping

# Day 6 - Apache Spark with Python and Scala

1. Data Sources
   1. The Structure of the Data Sources API
   2. CSV Files
   3. JSON Files
   4. Advanced I/O Concepts
      1. Splittable File Types and Compression
      2. Reading Data in Parallel
      3. Writing Data in Parallel
      4. Managing File Size
2. Resilient Distributed Datasets (RDDs)
   1. What Are the Low-Level APIs?
   2. About RDDs
      1. Types of RDDs
      2. When to Use RDDs?
   3. Creating RDDs
   4. Manipulating RDDs
   5. Transformations
      1. distinct
      2. filter
      3. map
      4. sort
      5. Random Splits
   6. Actions
      1. reduce
      2. count
      3. first
      4. max and min
      5. take
   7. Saving Files
      1. saveAsTextFile
      2. SequenceFiles
      3. Hadoop Files
3. Performance Tuning
   1. Caching - Concepts, Storage Type, Guidelines
   2. What is Check-pointing?
   3. Check-pointing vs caching
   4. Minimizing Shuffling for Increased Performance
   5. Using Broadcast Variables and Accumulators
   6. General Performance Guidelines
4. Spark Streaming
   1. Introduction and Streaming Basics
   2. Structured Streaming
      1. Continuous Applications
      2. Table Paradigm, Result Table
      3. Steps for Structured Streaming
      4. Sources and Sinks
   3. Consuming Kafka Data
      1. Kafka Overview
      2. Structured Streaming - "Kafka" format
      3. Processing the Stream