Custom Python - Course Outline

Training Details:

Course Name	Custom Python
Level	Beginner/Intermediate
Duration	Days
Participant background (Prerequisites)	As specified below

Topics	**Number of Days**
Python Core	5
Total	**5 Days**

www.cloudthat.com
+91-888-000-2200
102, 4th B cross, 5th Block, Koramangala, Industrial Area, Bengaluru, Karnataka 560095.

Custom Python Training:

Prerequisite:

- Basic IT Fundamentals

PCEP & Data Handling (5 Days/ 40 hours)

Python Fundamentals for Data Handling

- **Programming Concepts**
  - Compilation vs. Interpretation
  - Lexis, Syntax, Semantics
- **Environment Setup**
  - Installing Python and IDEs (VS Code, Jupyter)
  - Writing your first script
- **Data Types & Literals**
  - Boolean, Integers, Floats, Scientific notation
  - Binary, Octal, Decimal, Hexadecimal systems
- **Variables & Naming**
  - PEP-8 conventions for data clarity
- **Console I/O**
  - `print()`, `input()`, `len()`, type casting
- **Hands-on Labs**
  - Data type conversions
  - Input/output formatting for structured data
  - Simple unit converter for real-world data

Control Flow & Data-Driven Logic

- **Operators**
  - Arithmetic, Assignment, Bitwise, Boolean, Relational
- **Control Structures**
  - Conditional logic (`if`, `elif`, `else`)
  - Loops (`for`, `while`, `range()`)
  - Loop control (`break`, `continue`, `pass`)
- **Hands-on Labs**
  - Data filtering and transformation
  - Decision-making scripts (e.g., grading system)
  - Iterative data processing (e.g., batch calculations)

Data Collections & Structured Data Manipulation

- **Strings**

www.cloudthat.com
+91-888-000-2200
102, 4th B cross, 5th Block, Koramangala, Industrial Area, Bengaluru, Karnataka 560095.

- Indexing, slicing, formatting (`f-strings`)
- String methods for data cleaning

- **Lists & Tuples**
  - Creation, slicing, comprehension
  - Mutability and use cases in data pipelines
- **Dictionaries & Sets**
  - Key-value data storage
  - Iteration and retrieval for structured datasets
- **Hands-on Labs**
  - Data cleaning and transformation
  - Mini address book or inventory system
  - Membership checks and slicing for data subsets

Functions, Exceptions & File-Based Data Handling

- **Functions**
  - Defining reusable logic
  - Parameters, return values, scope
  - `*args` for flexible data input
- **Exception Handling**
  - Try-except blocks for robust data operations
  - Handling conversion and file errors
- **File I/O**
  - Reading/writing structured data (CSV, TXT)
  - Using `with open()` for safe access
- **Hands-on Labs**
  - Aggregating data using functions
  - Error-proof data parsing
  - File-based data logging and retrieval

Pandas & MySQL Integration

- **Pandas Module**
  - Series and DataFrames
  - Reading/writing CSV and Excel files
  - Data cleaning, filtering, aggregation
- **MySQL Integration**
  - Connecting using `mysql.connector` or `SQLAlchemy`
  - Executing queries and fetching results
  - Data transfer between MySQL and Pandas
- **Hands-on Labs**
  - CSV-based data analysis
  - MySQL query automation
  - Pandas-driven report generation

www.cloudthat.com
+91-888-000-2200
102, 4th B cross, 5th Block, Koramangala, Industrial Area, Bengaluru, Karnataka 560095.

PCEP Capstone: IPv4 Address Validator

Problem Statement: IPv4 Address Validator

Objective: Design and implement a robust Python function, validate_ipv4(ip_string), that leverages PCEP-level Python fundamentals—including **control flow**, **data collections**, and **exception handling**—to determine whether a given string is a valid IPv4 dotted-decimal address.

Core PCEP Concepts to be Tested

The solution *must* demonstrate proficiency in the following:

1. **Functions:** Defining and calling a user-defined function with a parameter and a clear Boolean return value.
2. **Strings and Lists:** Using string methods (.split()) and list processing to break down and iterate through the address parts.
3. **Control Flow (Loops & Conditionals):** Using a for loop to iterate through the octets and if/else statements for validation checks.
4. **Exception Handling (Mandatory):** Using a **try-except block** to catch errors that occur during the conversion of a string octet to an integer (e.g., if the octet contains non-numeric characters).

Input and Output Requirements

- **Input:** A single string, ip_string (e.g., "192.168.1.1", "256.0.0.1", "10.0.0").
- **Output:** The function must return a Boolean value (True for valid, False otherwise).

www.cloudthat.com
+91-888-000-2200
102, 4th B cross, 5th Block, Koramangala, Industrial Area, Bengaluru, Karnataka 560095.

Capstone Project: CSV-Based Student Performance Analyzer

Problem Statement:

Design and implement a Python program that reads a CSV file containing student performance data, validates the entries, and generates a summary report. The solution should demonstrate proficiency in **functions**, **control flow**, **data collections**, **exception handling**, and **file I/O**—all core PCEP concepts.

Objective:

Build a function-driven Python script that:

- Reads student data from a CSV file (name, subject, marks)
- Validates each row (e.g., marks must be numeric and between 0–100)
- Calculates average marks per student
- Flags students with missing or invalid data
- Writes a cleaned and summarized report to a new CSV file

Core PCEP Concepts to be Tested:

- **Functions**: Modular logic for reading, validating, and processing data
- **Strings & Lists**: Parsing CSV rows, handling string operations
- **Control Flow**: Conditional checks and loops for data validation
- **Exception Handling**: Try-except blocks for file access and data conversion
- **File I/O**: Reading from and writing to CSV files using `with open()`

Input Format:

CSV file with rows like:

```
Name,Subject,Marks
Alice,Math,85
Bob,Science,abc
Charlie,English,92
```


www.cloudthat.com
+91-888-000-2200
102, 4th B cross, 5th Block, Koramangala, Industrial Area, Bengaluru, Karnataka 560095.

Output Format:

Cleaned CSV report:

NAME,AVERAGE MARKS,STATUS

ALICE,85,VALID

BOB,N/A,INVALID MARKS

CHARLIE,92,VALID

www.cloudthat.com
+91-888-000-2200
102, 4th B cross, 5th Block, Koramangala, Industrial Area, Bengaluru, Karnataka 560095.