

Roll No.

Total No. of Pages : 02

Total No. of Questions : 09

**B.Tech. (AI&DS/CSE/CS/DS) (Sem.-5)**

**PROGRAMMING IN PYTHON**

**Subject Code :BTAIML501-20**

**M.Code :93939**

**Date of Examination : 02-12-2025**

**Time : 3 Hrs.**

**Max. Marks : 60**

**INSTRUCTIONS TO CANDIDATES :**

1. SECTION-A is COMPULSORY consisting of TEN questions carrying TWO marks each.
2. SECTION-B contains FIVE questions carrying FIVE marks each and students have to attempt any FOUR questions.
3. SECTION-C contains THREE questions carrying TEN marks each and students have to attempt any TWO questions.

**SECTION-A**

**1. Write briefly:**

- a. What is the primary purpose of Python's interactive help feature?
- b. Write the role of indentation in Python programming?
- c. Which function checks conditions in Python and raises an error if the condition is false?
- d. How do you create an infinite loop in Python?
- e. What is an anonymous function in Python, and what keyword is used to define it?
- f. Which module in Python provides access to mathematical functions?
- g. Name two common built-in exceptions in Python.
- h. What is garbage collection in the context of Python objects?
- i. Name a collection in Python that maintains the order of elements.
- j. Which module in Python offers tools for working with dates and times?

## SECTION - B

2. Describe the different data types available in Python with examples.
3. Explain the properties and operations of Python lists, with examples of common list methods.
4. What is the difference between local and global scope in Python functions, with examples.
5. How classes and objects are created in Python, including examples of defining attributes and methods?
6. Explain how to create and use a generator in Python, with a simple example?

## SECTION - C

7. Describe the purpose and functionalities of Python's collections module, covering namedtuple, deque, ChainMap, Counter, OrderedDict and DefaultDict. Include practical examples for each and illustrate their structure and use cases with diagrams.
8. Explain the concept of Object-Oriented Programming(OOP) in Python, including classes, objects, attributes and methods. Discuss access to and modification of class attributes, garbage collection and object destruction. Use diagrams to illustrate the relationships and flow between classes, objects and methods.
9. Write about Python functions in detail, covering built-in functions, user-defined functions, anonymous functions (lambda), recursion and the scope and lifetime of variables. Use diagrams to explain pass-by-value vs. pass-by-reference and illustrate recursion with a flowchart.

**NOTE : Disclosure of Identity by writing Mobile No. or Making of passing request on any page of Answer Sheet will lead to UMC against the Student.**