OFFICE OF THE REGISTRAR :: DIBRUGARH UNIVERSITY :: DIBRUGARH

No. DU/DR-A/CCSA/Syllabus-SEC(FYIPGP-1ˢᵗ Sem.)/2024/1519   Date: 09.09.2024

NOTIFICATION

This is for information of all concerned that the following Three (03) Credit syllabus of Skill Enhancement Course (SEC) for *Semester – I* has been approved by the Hon'ble Vice-Chancellor, Dibrugarh University as per the Dibrugarh University Regulations (Revised) for Five Year Integrated Post Graduate Programme (FYIPGP) in Choice Based Credit System (CBCS), 2024 with effect from the academic session 2024-2025 mentioned as below -

➢ **SEC – Programming using Python**

This is notified under report to the next meetings of the Post Graduate Board and Academic Council, Dibrugarh University.

Syllabus is attached herewith.

Issued with due approval.

Deputy Registrar (Academic)
Dibrugarh University

Copy forwarded for kind information and necessary action to:

1. The Hon'ble Vice-Chancellor, Dibrugarh University.
2. The Deans, Dibrugarh University.
3. The Registrar, Dibrugarh University.
4. The Heads of the Teaching Departments offering FYIPGP, Dibrugarh University.
5. The Chairperson, Centre for Computer Science and Applications, Dibrugarh University.
6. The Controller of Examinations i/c, Dibrugarh University.
7. The Joint / Deputy Controller of Examinations 'B', 'C' & 'A', Dibrugarh University.
8. The Programmer, Dibrugarh University, with a request to upload the notification on the Dibrugarh University Website.
9. File.

Deputy Registrar (Academic)
Dibrugarh University

Title of the Course	:	**Programming using Python**
Course Code	:	**SEC**
Nature of the Course	:	**SKILL ENHANCEMENT COURSE**
Total Credits	:	**03**
Distribution of Marks	:	**End Sem : 50 TH + 10 PR, In-Sem: 30 TH + 10 PR**

COURSE OBJECTIVES:
- Comprehend the Python programming environment and its basic syntax.
- Write and execute simple Python programs.
- Apply control structures like loops and conditionals in Python.
- Working with Python data types, lists, and dictionaries.
- Utilize functions, modules in Python programming.
- Working with Python files and exception handling.
- Debug and test Python code effectively.

UNITS	CONTENTS	L	T	P	Total Hours
1 **(Marks)** **6 TH + 2 PR**	**Introduction to Python Programming:** Overview of Python, Installing Python, Python IDEs, Structure of a Python program, Writing and Running Simple Python Programs.	03	01	04	08
2 **(Marks)** **12 TH + 2 PR**	**Basic Python Syntax and Control Structures:** Identifiers and keywords, Literals, numbers, and strings, Operators, Expressions, Input/output statements, Control structures (conditional statements, loop control statements, break, continue and pass, exit function).	05	01	04	10
3 **(Marks)** **12 TH + 2 PR**	**Working with Python Data Structures:** List: Introduction to list, Accessing list, list operations, Working with lists. Tuples: Introduction to tuple, Accessing tuples, Operations, Working with Tuples. Dictionary: Introduction to dictionaries, Accessing values in dictionaries, Working with dictionaries.	06	01	10	17
4 **(Marks)** **8 TH + 2 PR**	**Functions and Modules:** Defining Functions, Arguments and Return Values, Importing Modules, Working with Python Libraries	05	01	04	10
5 **(Marks)** **12 TH + 2 PR**	**File and Exception Handling:** Reading and Writing Files, File handling through libraries, Introduction to Exception, Exception Handling, Except clause, Try ? finally clause, User Defined Exceptions.	06	01	08	15
	Total (in Hrs)	25	05	30	60

Where,   L: Lectures   T: Tutorials   P: Practicals

MODES OF IN-SEMESTER ASSESSMENT: **(40 Marks)**
- One Internal(TH) Examination -   **10 Marks**
- One Internal(PR) Examination -   **10 Marks**
- Others   -   **20 Marks**
    - Quiz
    - Seminar presentation
    - Assignment

COURSE OUTCOMES:

CO1: Explain the basic structure of a Python program and the process of writing and running simple Python scripts.

CO2: Apply Python syntax and control structures, such as conditional statements and loops, to solve basic programming problems.

CO3: Analyze different Python data structures, including lists, tuples, and dictionaries, to determine their appropriate usage in various contexts.

CO4: Create functions and modules in Python to develop modular, reusable code for more complex programming tasks.

CO5: Evaluate file handling methods and exception handling mechanisms to ensure robust and error-free Python applications.

SUGGESTED READINGS:
1. Taneja, S., Kumar, N. "Python Programming- A modular Approach", 1st edition, Pearson Education India, 2018.
2. T. Budd, Exploring Python, TMH, 1st Ed, 2019.
3. A. B. Downey, Think Python, 2e: How to Think Like a Computer Scientist, O′Reilly, 2015.
4. C. Morris, "https://www.kaggle.com/learn/python," [Online].
5. "https://docs.python.org/3/tutorial/index.html," [Online].