COURSE TITLE	**PYTHON PROGRAMMING AND PROMPT ENGINEERING**
COURSE CODE	**09CE0304**
COURSE CREDITS	**2**

Objective:

1 In this course students will understand fundamentals of Python programming along with Generative AI and Prompt Engineering concepts. Students will learn basic programming principles and AI-assisted development techniques to meet modern computing requirements.

Course Outcomes: After completion of this course, student will be able to:

1 Apply fundamental concepts of Python programming and basic AI concepts to solve simple computational problems.

2 Apply control structures and data structures in Python to develop logical programs.

3 Apply functions, modules, and file handling techniques to build structured Python applications.

4 Apply prompt engineering techniques to generate, refine, and debug Python code using AI tools.

5 Apply AI-assisted development methods and evaluation techniques to create reliable and ethical Python-based solutions.

Pre-requisite of course: Basic Programming Knowledge

Teaching and Examination Scheme

Theory Hours	Tutorial Hours	Practical Hours	ESE	IA	CSE	Viva	Term Work
0	0	4	0	30	20	25	25

Contents : Unit	Topics	Contact Hours
1	**INTRODUCTION TO PYTHON & AI FUNDAMENTALS** Introduction to Programming, History and Features of Python, Structure of Python Program, Variables and Data Types, Operators, Input and Output Functions, Introduction to Artificial Intelligence, Machine Learning, Generative AI and Large Language Models (Concepts only), Overview of AI Tools for Programming Assistance	0
2	**CONTROL FLOW AND DATA STRUCTURES IN PYTHON** Decision Making Statements – if, if-else, elif ladder, Looping Statements – for loop, while loop, Nested loops, Lists – creation and operations, Tuples, Sets, Dictionaries – creation and access, String handling and built-in string functions.	0

MR. SMIT MAHENDRABHAI THACKER
Digitally signed by (Name of HOD)

MR. MITTAL KIRITBHAI JOISER
Digitally signed by (Name of Dean/ Principal)

Contents : Unit	Topics	Contact Hours
3	**FUNCTIONS, MODULES AND FILE HANDLING** User-defined Functions, Function Arguments and Return Values, Recursion, Built-in Functions, Modules and Libraries, File Handling – Read and Write operations, Exception Handling – try, except, finally.	0
4	**PROMPT ENGINEERING FUNDAMENTALS** Introduction to Prompt Engineering, Structure of a Prompt – Instruction, Context, Constraints, Output Format, Role-based Prompts, Instruction Prompts, Few-shot Prompting, Prompt Templates for Code Generation and Debugging, Evaluating AI-generated Code – Accuracy, Clarity and Correctness	0
5	**AI-ASSISTED PYTHON DEVELOPMENT & ETHICS** Using AI Tools for Code Generation, Code Debugging and Optimization, Prompt Chaining for Step-by-step Development, Detecting Hallucinations in AI-generated Code, Responsible AI Usage, Data Privacy and Security Considerations, Mini Project using Python and Prompt Engineering.	0
Total Hours		**0**

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
1	**Practical-1** Write a Python program that accepts student name, roll number and marks, and displays formatted output.	2
2	**Practical-2** Write a program to perform addition, subtraction, multiplication and division using user input.	2
3	**Practical-3** Accept different data types from user, perform type conversion and display results.	2
4	**Practical-4** Use an AI tool to generate a basic Python program, refine the prompt, and compare output with manual code.	2
5	**Practical-5** Write a Python program using if-else ladder to find the largest number among three inputs.	2
6	**Practical-6** Develop a program that calculates student grade based on percentage using conditional statements.	2
7	**Practical-7** Write a program to generate Fibonacci series up to N terms using loop.	2
8	**Practical-8** Write a program to print prime numbers between 1 and N using loop logic.	2

MR. SMIT MAHENDRABHAI THACKER
Digitally signed by (Name of HOD)

MR. MITTAL KIRITBHAI JOISER
Digitally signed by (Name of Dean/ Principal)

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
9	**Practical-9** Create a list and perform append, remove, reverse and display operations.	2
10	**Practical-10** Write a program to count occurrence of specific element in a list.	2
11	**Practical-11** Create a dictionary, access values, update and delete elements.	2
12	**Practical-12** Generate a looping-based program using AI, intentionally insert error, debug using improved prompt.	2
13	**Practical-13** Write a program using user-defined function to perform arithmetic operation.	2
14	**Practical-14** Develop a recursive function to calculate factorial of a number.	2
15	**Practical-15** Write a function to check whether given string is palindrome or not.	2
16	**Practical-16** Create a text file, write data into it and read content from file using Python.	2
17	**Practical-17** Demonstrate use of try, except and finally block with division-by-zero example.	2
18	**Practical-18** Use AI to generate modular Python code and prepare brief documentation using structured prompt.	2
19	**Practical-19** Design a structured prompt template for generating Python programs.	2
20	**Practical-20** Create a prompt format specifically for debugging incorrect Python code.	2
21	**Practical-21** Provide example input-output pairs and generate optimized Python solution using AI.	2
22	**Practical-22** Generate same program using different prompting strategies and compare outputs.	2
23	**Practical-23** Identify incorrect AI-generated code, verify manually and correct logic.	2
24	**Practical-24** Develop a small console-based Python application using AI guidance.	2

MR. SMIT MAHENDRABHAI THACKER
Digitally signed by (Name of HOD)

MR. MITTAL KIRITBHAI JOISER
Digitally signed by (Name of Dean/ Principal)

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
25	**Practical-25** Break a problem into steps and generate code step-by-step using prompt chaining.	2
26	**Practical-26** Improve an existing Python program using AI suggestions and test efficiency.	2
27	**Practical-27** Mini Project Development like Result Analyzer / Expense Tracker / Quiz Generator.	2
28	**Practical-28** Prepare short report explaining AI usage, limitations, improvements and ethical considerations like Case Study.	2
Total Hours		**56**

Textbook :

1 Fluent Python: Clear, Concise, and Effective Programming, 2nd Edition, Luciano Ramalho, O'Reilly Media, 2022

References:

1 Python Crash Course: A Hands-On, Project-Based Introduction to Programming, 2nd Edition, Python Crash Course: A Hands-On, Project-Based Introduction to Programming, 2nd Edition, Eric Matthes, No Starch Press, 2019

2 Learning Python, 5th Edition, Learning Python, 5th Edition, Mark Lutz, O'Reilly Media, 2013

3 Automate the Boring Stuff with Python: Practical Programming for Total Beginners, 2nd Edition, Automate the Boring Stuff with Python: Practical Programming for Total Beginners, 2nd Edition, Al Sweigart, No Starch Press, 2019

4 Core Python Programming, Core Python Programming, Wesley J. Chun, Prentice Hall, 2006

Suggested Theory Distribution:

The suggested theory distribution as per Bloom's taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery and evaluation					
Remember / Knowledge	**Understand**	**Apply**	**Analyze**	**Evaluate**	**Higher order Thinking / Creative**
0.00	0.00	35.00	35.00	30.00	0.00

MR. SMIT MAHENDRABHAI THACKER
Digitally signed by (Name of HOD)

MR. MITTAL KIRITBHAI JOISER
Digitally signed by (Name of Dean/ Principal)

Instructional Method:

1. The course delivery method will depend upon the requirement of the content and need of students. The teacher, in addition to conventional teaching method by blackboard, may also use tools such as demonstration, coding practice, brainstorming sessions, case studies, AI-assisted learning activities and MOOCs.
2. The internal evaluation will be done on the basis of continuous evaluation of students in the laboratory and classroom through assignments, prompt design activities, coding exercises and mini projects.
3. Practical examination will be conducted at the end of semester for evaluation of performance of students in laboratory based on programming skills and prompt engineering applications.
4. Students will use supplementary resources such as online videos, NPTEL lectures, e-courses, official Python documentation, AI tool user guides and virtual laboratory platforms.

Supplementary Resources:

1. https://docs.python.org
2. https://www.w3schools.com
3. https://www.learnpython.org
4. https://www.tutorialspoint.com

MR. SMIT MAHENDRABHAI THACKER
Digitally signed by (Name of HOD)

MR. MITTAL KIRITBHAI JOISER
Digitally signed by (Name of Dean/ Principal)