

**Uka Tarsadia University**



**B.Tech.  
Semester II**

**PROGRAMMING WITH PYTHON  
IT3008**

**EFFECTIVE FROM July-2021**

**Syllabus version: 1.00**

Subject Code	Subject Title	Teaching Scheme			
		Hours		Credits	
		Theory	Practical	Theory	Practical
IT3008	Programming with Python	3	4	3	2

Subject Code	Subject Title	Theory Examination Marks		Practical Examination Marks	Total Marks
		Internal	External	CIE	
IT3008	Programming with Python	40	60	100	200

### Objectives of the course:

- To educate students about programming by developing an understanding about concepts of python programming.
- To unfold concepts of object oriented programming paradigms.

### Course Outcomes:

Upon completion of the course, the student shall be able to,

- C01: Understand foundational concepts of python programming.  
 C02: Understand and use primary data structures used in python programming.  
 C03: Understand and use several control structures and functions.  
 C04: Explore object oriented programming along with its significance.  
 C05: Understand and apply exception handling methods.  
 C06: Apply file operations and understand its need in real world applications.

Sr. No.	Topics	Hours
<b>Unit – I</b>		
<b>1</b>	<b>Introduction to Python</b> Using the Python interpreter, Variables, Identifiers and Keywords, Numbers and Expressions.	5
<b>Unit – II</b>		
<b>2</b>	<b>Data Structures</b> Common sequence operations: Indexing, Slicing, Adding Sequences, Multiplication, Membership, Length, Minimum, and Maximum. Using lists as Stacks, Using lists as Queues, List Comprehensions, Nested List comprehensions, The del statement, Tuples and Sequences, Sets, Dictionaries, Comparing Sequences and Basic string operations.	8

<b>Unit – III</b>		
<b>3</b>	<b>Control Structures</b> Conditional branching: if Statements, break and continue Statements, and else Clauses on loops, pass Statements, Loops: while Loops, for Loops. <b>Functions</b> Defining functions, More on Defining Functions: Default argument values, Keyword arguments, Arbitrary argument lists, Unpacking argument lists, lambda Expressions, Documentation strings, Function annotations. Executing modules as scripts, The module search path, Compiled Python files, Packages: Importing * from a package, Intra-package references, Packages in multiple directories.	<b>9</b>
<b>Unit – IV</b>		
<b>4</b>	<b>Object Oriented programming</b> The object oriented paradigm, Scopes and Namespaces, Class definition, Class objects, Instance objects, Method objects, Class and Instance variables, Inheritance, Private variables, Polymorphism - Method overloading and method overriding, Data hiding, Decorators, Metaclass, Multithreading, Using properties to control attribute access, Creating complete fully integrated data types.	<b>10</b>
<b>Unit – V</b>		
<b>5</b>	<b>Exception Handling and Regular Expressions</b> The try statement, with statement and context manager, Exception propagation, The raise statement, Exception object, Custom exception class, Error checking strategies. Regular expressions with special characters, Regular expressions and raw strings, Extracting matched text from strings, Substituting text with regular expressions.	<b>8</b>
<b>Unit – VI</b>		
<b>6</b>	<b>File Handling</b> Writing and reading binary data, Writing and Parsing text files, The io module, The os module, File system operations, Text input and output, Richer text I/O.	<b>5</b>

<b>Sr. No.</b>	<b>Programming with Python (Practical)</b>	<b>Hours</b>
1	Write a python program to print welcome messages on screen.	2
2	Write a python program to demonstrate the use of input() method with implicit type-casting.	2
3	Write a python program that performs basic arithmetic operations on user entered numbers.	2
4	Write a python program to show the use of primary data structures in python with associated methods.	4

5	Write a python program to demonstrate the use of for() loop and while() loop.	4
6	Write a python program to demonstrate the use of user-defined functions with single, multiple and arbitrary arguments.	4
7	Modify the Program-3, in order to design user-defined functions for each arithmetic operation and provide a choice to the user which operation he/she wants to perform on given numbers.	2
8	Demonstrate the file operations in python with in-built mechanisms.	4
9	Write a python program that scans two lists with arbitrary elements from the user and returns the total repetitions of elements in both the lists.	4
10	Write a python program that scans user provided metrics and perform following operations: Matrix Addition, Matrix Subtraction, Matrix Multiplication, Inverse of Matrix.	4
11	Develop a python program that scans content of a particular file from the user. Divide content of files into chunks and return frequency of occurrence of each chunk.	4
12	Write a python program to demonstrate the use of class and object in python.	4
13	Write a python program to show the need of inheritance and encapsulation.	4
14	Write a python program to demonstrate polymorphism.	4
15	Write a python program to demonstrate the use of exception handling in python.	4
16	Write a python program to demonstrate various regular expressions.	2
17	Project	6

#### **Text book:**

1. Alex Martelli, Anna Ravenscroft and Steve Holden, "Python in Nutshell", 3rd Edition, O'Reilly Publication.

#### **Reference books:**

1. Magnus Lie Hetland, "Beginning Python From Novice to Professional", Third Edition, Apress, 2017.
2. David Beazley, Brian K. Jones, "Python Cookbook", 3rd edition, O'Reilly Publication, 2016.
3. Brett Slatkin, "Effective Python: 59 Specific Ways to Write Better Python", Novatec, 2016.
4. Mark Lutz "Learning Python", 4th Edition, O'Reilly Publication, 2016.

**Course objectives and Course outcomes mapping:**

- To educate students about programming by developing an understanding about concepts of python programming.: CO1, CO2, CO3, and CO6
- To unfold concepts of object oriented programming paradigms: CO4 and CO5

**Course units and Course outcomes mapping:**

Unit No.	Unit Name	Course Outcomes					
		CO1	CO2	CO3	CO4	CO5	CO6
1	Introduction to Python	✓					
2	Data Structures		✓				
3	Control Structure and Functions			✓			
4	Object Oriented Programming				✓		
5	Exception Handling and Regular Expression					✓	
6	File Handling						✓

**Programme outcomes:**

PO 1:	Engineering knowledge: An ability to apply knowledge of mathematics, science, and engineering.
PO 2:	Problem analysis: An ability to identify, formulates, and solves engineering problems.
PO 3:	Design/development of solutions: An ability to design a system, component, or process to meet desired needs within realistic constraints.
PO 4:	Conduct investigations of complex problems: An ability to use the techniques, skills, and modern engineering tools necessary for solving engineering problems.
PO 5:	Modern tool usage: The broad education and understanding of new engineering techniques necessary to solve engineering problems.
PO 6:	The engineer and society: Achieve professional success with an understanding and appreciation of ethical behavior, social responsibility, and diversity, both as individuals and in team environments.
PO 7:	Environment and sustainability: Articulate a comprehensive world view that integrates diverse approaches to sustainability.
PO 8:	Ethics: Identify and demonstrate knowledge of ethical values in non-classroom activities, such as service learning, internships, and field work.
PO 9:	Individual and team work: An ability to function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO 10:	Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give/receive clear instructions.

PO 11:	Project management and finance: An ability to demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO 12:	Life-long learning: A recognition of the need for, and an ability to engage in life-long learning.

**Programme outcomes and Course outcomes mapping:**

Programme Outcomes	Course Outcomes					
	C01	C02	C03	C04	C05	C06
P01	✓	✓	✓	✓	✓	✓
P02		✓	✓	✓		✓
P03		✓		✓		
P04						
P05			✓		✓	
P06						
P07						
P08						
P09						
P010						
P011				✓	✓	
P012						