Lok Jagruti Kendra University
University with a Difference

Diploma in Mechanical Engineering

Course Code:025060301

Python Programming

Programme / Branch Name			Diploma in Mechanical Engineering			
Course Name	Python Programming				**Course Code**	025060301
Course Type	HSSC	BSC	ESC	PCC	OEC	PEC

Legends: HSSC: Humanities and Social Sciences Courses BSC: Basic Science Courses
ESC: Engineering Science Courses PCC: Program Core Courses
OEC: Open Elective Courses PEC: Program Elective Courses

1. Teaching and Evaluation Scheme

Teaching Hours / Week				**Evaluation Scheme**			
L	**T**	**P**	**Total Credit**	**CCE**	**SEE (Th)**	**SEE (Pr)**	**TOTAL**
2	0	4	4	50	50	50	150

Legends:
L: Lectures T: Tutorial P: Practical
CCE: Continuous & Comprehensive Evaluation
SEE (Th): Semester End Evaluation (Theory)
SEE (Pr): Semester End Evaluation (Practical)

2. Prerequisites

- ✓ Basic knowledge of programming and its constructs.

3. Rationale

Python is a scripting language that has become the most widely used language today. Its vast range of applications includes Artificial Intelligence including Machine Learning, Natural Language Processing, Robotics, Facial Identification & many others. It can also be used for web application development. While learning this subject, students would get a solid knowledge base of Python language to write programs for Artificial Intelligence applications or web applications.

4. Objectives

- ✓ Develop a Python program to display a message and demonstrate the use of operators.
- ✓ Perform operations on data structures in Python.
- ✓ Develop functions for a given problem.
- ✓ Design classes for a given problem.

5. Contents

Unit No.	Topics	Sub-Topics	Learning Outcome	% Weightage	Hours
1.	**Python Introduction & Basic Syntax**	1.1. What is Python Python interpreter, extension & implementations 1.2. Python installation 1.3. Variables, strings, formatted strings & string methods 1.4. Escape sequence 1.5. Working with numbers 1.6. Type conversion 1.7. Basic operators	• To understand what is Python and its applications are, Installation of Python programming language and its interpreter, IDE. • To understand the primitive data types available in Python and how to use methods associated with them.	15%	4
2.	**Control flow and Functions in Python**	2.1. Conditional statements 2.2. Looping statements 2.3. Iterables 2.4. Defining functions 2.5. Types of functions 2.6. Keyword arguments, Default arguments	• To understand the syntax of conditional and looping statements in Python and their use along with the syntax of user defined function with its type. Different types of arguments like keyword arguments and default arguments.	15%	4
3.	**Data Structures in Python**	3.1. List 3.2. Tuple 3.3. Dictionary 3.4. Set	• To understand different types of data structures available in Python and their built-in methods.	30%	8
4.	**Classes & Modules in Python**	4.1. Class 4.2. Attribute and method 4.3. OOP concept implementations in Python 4.4. Creating module 4.5. Compiled Python files 4.6. Packages and subpackages	• To understand the basic concepts of object-oriented programming language and their implementation, modules, packages, and commands like Pypi and Pip to import libraries.	20%	6

		4.7. Pypi and Pip			
5.	**Python Standard Libraries**	5.1. Working with path, files, and directories 5.2. Working with CSV 5.3. Working with time, datetimes and timestamps. 5.4. Sending emails	• To learn the use of some of the standard libraries available in Python	20%	6
				Total Hours	**28**

6. List of Practicals / Exercises

The practicals/exercises have been properly designed and implemented in an attempt to develop different types of skills so that students can acquire the competencies/programme outcomes. Following is the list of practicals/exercises.

Sr. No	Practical / Exercises	Key Competency	Hours
1.	Install and configure Python IDE	To be able to install Python & its IDE	2
2.	Write a Python program to display a message on the screen	To be able to display a message using Python	2
3.	Write simple Python program using operators: A) arithmetic operators B) logical operators	To be able to use the basic operators in Python	2
4.	Write a Python program to demonstrate different string methods	To be able to use string methods to manipulate them	2
5.	Write a Python program to demonstrate formatted strings	To be able to use formatted string concept in message displaying	2
6.	Write a Python program to demonstrate type conversion	To be able to use type conversion when required	2
7.	Write a Python program to identify a digit as even or odd	To be able to use conditional statements	2
8.	Write a Python program to calculate the sum of all numbers till 100 which are divisible by 3 and 5	To be able to use looping statements	2
9.	Write a Python program to print the sum of all digits of a given number using function	To be able to use user defined function	2
10.	Write a Python program to demonstrate the use of keyword argument	To be able to use keyword arguments in user defined function	2
11.	Write a Python program to declare a list and display it	To be able to prepare a list and access it	2
12.	Write a Python program to perform the following operations on the list:	To be able to perform basic operations on a list	4

	a) Insert b) Update c) Delete d) Search		
13.	Write a Python program to declare a tuple and display it	To be able to prepare a tuple and access it	2
14.	Write a Python program to perform the following operations on a tuple: a) Search b) Merge	To be able to perform basic operations on a tuple	4
15.	Write a Python program to declare a dictionary and display it	To be able to prepare a dictionary and access it	2
16.	Write a Python program to perform the following operations on the dictionary: a) Insert b) Update c) Delete d) Search	To be able to perform basic operations on a dictionary	4
17.	Write a Python program to declare a set and display it	To be able to prepare a set and access it	2
18.	Write a Python program to perform different operations using two sets	To be able to perform basic operations between two sets	2
19.	Write a Python program to demonstrate the concept of modules and packages	To be able to prepare modules, packages and call them	2
20.	Write a Python program to install an external Python library and use it	To be able to import an external Python library and use it	2
21.	Write a Python program to demonstrate the use of pathlib module	To be able to use different functionalities of the pathlib module	2
22.	Write a Python program to demonstrate the use of the csv module	To be able to use different functionalities of the CSV module	2
23.	Write a Python program to demonstrate the use of time and datetime module	To be able to use different functionalities of time and datetime modules	2
24.	Write a Python program to demonstrate the use of the email module	To be able to use different functionalities of the email module	2
		Total Hours	**56**

7. Suggested Specification Table for Evaluation Scheme

Unit No.	Chapter Name	Distribution of Topics According to Bloom's Taxonomy					
		R %	U %	App %	C %	E %	An %
1.	Python Introduction & Basic Syntax	40	30	20	-	5	5
2.	Control Flow & Function in Python	30	30	20	-	10	10
3.	Data Structures in Python	10	30	20	20	10	10
4.	Classes & Modules in Python	20	20	20	10	15	15
5.	Python Standard Library	20	20	30	10	10	10

Legends: R: Remembering U: Understanding
App: Applying C: Creating
E: Evaluating An: Analyzing

8. Text Books

1) Learning Python by Lutz, Mark, 5th Edition, O'Reilly Publication ISBN-13: 978-1449355739.

9. Reference Books

1) Python Programming by Rao, K. Nageswara, Shaikh Akbar, Scitech Publications (India) Pvt. Ltd.
2) Python Essential Reference by Beazley, David, 4th Edition, Addison-Wesley Professional, ISBN: 9780672329784
3) Head First Python, 2nd Edition by Paul, Barry, O'Reilly Publication, 2nd Edition, ISBN: 1491919531

10. Open Sources (Website, Video, Movie)

1) www.python.org/doc
2) www.learnpython.org
3) www.tutorialspoint.com/python/index.htm
4) www.w3schools.com/python
5) www.programiz.com/python-programming
6) www.w3resource.com/python-exercises
7) pynative.com/python-exercises-with-solutions