Government Polytechnic, Pune

'180OB' – Scheme

Programme Name	**Diploma in Computer Engineering, Diploma in Information Technology**
Programme Code	01/02/03/04/05/**06**/**07**/08/16/17/21/22/23/24/**26**
Course Title	**Programming with PYTHON**
Course Code	**CM5101**
Prerequisite course code and name	**NA**
Class Declaration	**YES**

1. TEACHING AND EXAMINATIONSCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P)		Examination Scheme				
					Theory Marks		Practical Marks		Total Marks
L	**T**	**P**	**C**		**#ESE**	**PA**	***ESE**	**PA**	
02	00	04	06	**Marks**	40	10	50	50	150
				Exam Duration	2 Hrs.	1/2Hr.			

Legends: L- Lecture, P- Practical, T- Tutorial, C- Credit, ESE-End Semester Examination, PA- Progressive Assessment (Test I, II/Term Work), *- Practical Exam, $- Oral Exam, #- Online Examination each Lecture/Practical period is of one clock hour.

2. RATIONALE

Python is powerful programming language. It has efficient high level data structures and a simple but effective approach to object-oriented programming. Python code is simple, short, readable, intuitive and powerful and thus it is effective for introducing computing and problem solving for beginners. Its elegant syntax and dynamic typing together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

3. COMPETENCY

The aim of this course is to attend following industry identified competency through various teaching learning experiences:

- **Develop applications using Python programming to solve given problems.**

4. COURSE OUTCOMES(COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry-oriented COs associated with the above-mentioned competency**:**

1. Develop simple Python programs using Python IDE.
2. Execute programs using operators and control flow statements.
3. Perform Operations using Python Data structures.
4. Develop applications using Functions, Modules and Packages.
5. Develop applications using object-oriented concepts in python.
6. Write Python code for File and Exception Handling.

5. SUGGESTED PRACTICALS/ EXERCISES

Sr. No	Unit No.	Practical Exercises (Outcomes in Psychomotor Domain)	Relevant CO	Approximate Hours Required.
1.	1	Install and configure python IDE.	1	01
2.	1	Write simple Python Program to display message on screen.	1	01
3.	2	Write simple Python Program using operators: • Arithmetic Operators • Logical Operators • Bitwise Operators	2	02
4.	2	Write simple Python Program to demonstrate use of conditional statements: • 'if' Statement • 'if...else' Statement • Nested 'if' Statement	2	02
5.	2	Write Python Program to demonstrate use of looping statements: • 'while' loop • 'for' loop • Nested loops	2	04
6.	2	Write Python Program to demonstrate use of looping statements: • continue • pass • break	2	04
7.	3	Write Python Program to perform following operations on Lists: • Create List • Access List • Update List (Add Item, Remove Item) • Delete List	3	04
8.	3	Write Python Program to perform following operations on Tuples: • Create Tuple • Access Tuple • Update Tuple • Delete Tuple	3	04
9.	3	Write Python Program to perform following operations on Set: • Create Set • Access Set elements • Update Set • Delete Set	3	04

Sr. No	Unit No.	Practical Exercises (Outcomes in Psychomotor Domain)	Relevant CO	Approximate Hours Required.
10.	3	Write Python Program to perform following operations on Dictionaries: • Create Dictionary • Access Dictionary elements • Update Dictionary • Delete Dictionary • Looping through Dictionary	3	04
11.	4	i. Write Python Program to demonstrate math built-in functions (Any 2Programs) ii. Write Python Program to demonstrate string built-in functions (Any 2 Programs)	4	04
12.	4	Develop user defined python function for given problem: • Function with minimum 2arguments • Function returning values	4	04
13.	4	Write Python Program to demonstrate use of: • Built-in module (e.g., Keyword, math, number, operator) • User defined module	4	04
14.	4	Write Python Program to demonstrate use of: • Built-in packages (e.g., NumPy, Pandas) • User defined packages	4	04
15.	5	Write Python Program to demonstrate following operations: • Method overloading • Method overriding	5	02
16.	5	Write Python Program to demonstrate following operations: • Simple Inheritance • Multiple Inheritance	5	04
17.		Write Python Program to demonstrate File Handling through: • Opening file in different modes • Accessing file • Reading and Writing file • Closing file • Renaming and Deleting file	6	04
18.	6	Write Python Program to handle user defined exception for given problem.	6	04
19	All	Micro-project (Refer point 11 for micro project list)	All COs	04
		Total Hours		**64**

Sr. No.	Performance Indicators	Weightage in %
a.	Use of Appropriate tool to solve the problem (Process)	40
b.	Quality of output achieved (Product)	30
c.	Complete the practical in stipulated time	10
d.	Observations and Recording	10
e.	Answer to sample questions	10
Total		**100**

6. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of practical, as well as aid to procure equipment by authorities concerned**.**

Sr. No.	Equipment Name / Instruments required	Experiment Sr. No.
1	Hardware: Personal computer Pentium IV,2 GHz minimum (i3-i5 preferable), RAM minimum 2 GB.	For all experiments
2	Any open-source tool (SPYDER / Eclipse IDE), Python Interpreter	

7. THEORYCOMPONENTS

Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
SECTION I	
UNIT 1. Introduction to Python Programming (Weightage-04, Hrs.- 04)	
1a. Explain features of Python. 1b. Identify the given variables, keywords and constants in python.	1.1 Features of Python-Interactive, Object Oriented, Interpreted, Platform independent. 1.2 Python Building blocks- Identifiers, Keywords, Indention, variables, comments. 1.3 Python Environment Setup- Installation and working of IDE.
1c. Use Indention, Comments in the given program. 1d. Install the Python IDE and editor. 1e. Write the python program to display the given text.	1.4 Running Simple Python scripts to display message. 1.5 Python Data Types: Numbers, Strings,Tuples, Lists, Dictionary, Declaration and use of data types.
UNIT 2. Python Operators and Control Flow (Weightage-06, Hrs.- 04)	

Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
2a. Write simple Python program for the given arithmetic expressions. 2b. Write a Python program using decision making structure for two- way/multi-way branching to solve the given problem.	2.1 Basic Operators: Arithmetic, Comparison/ Relational, Assignment, Logical, Bitwise, Membership, Identity Operators. Python Operator precedence. 2.2 Control Flow. 2.3 Conditional Statements (if, if...else, nested if). 2.4 Looping in Python (While loop, for loop, nested loops). 2.5 Loop manipulation using continue, pass, break, else.
UNIT 3. Data Structures in Python (Weightage-10, Hrs.- 08)	
3a. Write python program to use and manipulate lists for the given problem. 3b. Write python program to use and manipulate Tuples for the given problem. 3c. Write python program to use and manipulate Sets for the given problem. 3d. Write python program to use and manipulate Dictionaries for the given problem.	3.1 Lists: Defining Lists, accessing values in list, deleting values from list, updating lists. Basic List Operations, Built-in List Functions. 3.2 Tuples: Accessing values in Tuples, deleting values from. Tuple and updating Tuples. Basic Tuple operations, Built- in Tuple Functions. 3.3 Sets: Accessing values in Set, deleting values from Set and updating Sets. Basic Set operations, Built-in Set Functions. 3.4 Dictionaries: Accessing values in Dictionary, deleting Values from Dictionary and updating Dictionary. Basic Dictionary operations, Built-in Dictionary Functions.
SECTION II	
UNIT 4. Python Functions, Modules and Packages (Weightage-08, Hrs.- 06)	
4a. Use the Python standard functions for the given problem. 4b. Develop relevant user defined functions for the given problem. 4c. Write Python module for the given problem. 4d. Write Python Package for the given problem.	4.1 Use of Python built-in functions (e.g., type/data conversion functions, math function setc.). 4.2 User defined functions: Function definition, Function calling, function arguments and parameter passing, return statement, scope of variable: Global variable and Local variable. 4.3 Modules: Writing modules, importing modules, importing objects from modules, python built-in modules, (e.g. Numeric and mathematical module, Functional programming module), Namespace and Scoping. 4.4 Python Packages: Introduction, Writing Python Packages, using standard (e.g., math, scipy, Numpy, matplotlib, pantalets.) and user defined Packages.
UNIT 5. Object Oriented Programming in Python (Weightage-06, Hrs.- 04)	
5a. Create Classes and Objects to solve the given problem. 5b. Write Python code for data hiding for the given problem.	5.1 Creating Classes and Objects. 5.2 Method Overloading and Overriding. 5.3 Data Hiding. 5.4 Data Abstraction.

Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
5c. Write Python code using data abstraction for the given problem. 5d. Write Python program using inheritance for the given problem.	5.5 Inheritance and Composition Classes. 5.6 Customization vi inheritance specializing inherited methods.
UNIT 6. File and Exception Handling (Weightage-06, Hrs.- 06)	
6a. Write Python code for the given reading values from keyboard. 6b. Read data from the given file. 6c. Write the given data to a file. 6d. Handle the given exceptions through python program.	6.1 I/O operations: Reading keyboard input, printing to screen. 6.2 File Handling: Opening file in different modes, accessing file contents using standard library functions, reading and writing files, closing files renaming and deleting files. 6.3 Exception Handling: Introduction, 'try: except:' statement, 'raise' statement, user defined exceptions.

8. SUGGESTED SPECIFICATION TABLE FORQUESTION PAPERDESIGN

Unit No	Unit Title	Teaching Hrs.	Distribution of Theory Marks			
			R Level	U Level	A Levels	Total Marks
Section I						
I	Introduction to Python Programming	04	02	02	00	**04**
II	Python Operators and Control Flow	04	00	02	04	**06**
III	Data Structures in Python	08	02	04	04	**10**
	Total	**16**	**04**	**08**	**08**	**20**
Section II						
IV	Python Functions, Modules and Packages	06	02	02	04	**08**
V	Object Oriented Programming in Python	04	00	02	04	**06**
VI	File and Exception Handling	06	00	02	04	**06**
Total		**16**	**02**	**06**	**12**	**20**
Grand Total		**32**	**06**	**14**	**20**	**40**

9. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related ***co-curricular*** activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

a. Prepare journal based on practical performed in laboratory.
b. Follow Coding Standards.
c. Undertake micro-projects.
d. Develop variety of programs to improve logical skills.
e. Develop Application oriented real world programs.

10. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

a. Massive open online courses (***MOOCs***) may be used to teach various topics/sub topics.
b. About ***15-20% of the topics/sub-topics*** which is relatively simpler or descriptive in nature is to be given to the students for ***self-directed learning*** and assess the development of the COs through classroom presentations (see implementation guideline for details).
c. With respect to item No.9, teachers need to ensure to create opportunities and provisions for ***co-curricular activities***.
d. Use different Audio-Visual media for Concept understanding.
e. Guide student(s) in undertaking micro-projects.
f. Demonstrate students thoroughly before they start doing the practice.
g. Observe continuously and monitor the performance of students in Lab.

11. SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her. In special situations where groups have to be formed for micro-projects, the number of students in the group should ***not exceed three.*** The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. (Affective Domain Outcomes). Each student will have to maintain activity chart consisting of individual contribution in the project work and give a seminar presentation of it before submission. The student ought to submit micro-project by the end of the semester to develop the industry-oriented COs.

A suggestive list of micro-projects is given here. Similar micro-projects could be added by the concerned faculty:

a. Create an English Dictionary which is able to perform following function
    - Add a word and its meaning.
    - Delete a word and its meaning.
    - Update a word and its meaning.
    - Print list of word and its meaning.
b. Create Finance Currency calculator using classes and objects.
c. Develop a game (Hangman, Tick Toe, Snake etc.) using Python data structure,

functions and packages.

d. Develop Calculator.

e. Develop Alarm clock.

f. Develop Music player.

12. SUGGESTED LEARNING RESOURCES

Sr. No.	Title of Book	Author	Publisher, Edition, Year of publication, ISBN Number
1	Python Programming	K. Nageswara Rao, Shaikh Akbar	Scitech Publications (India) Pvt. Ltd. • ISBN:9789385983450
2	Learning Python	Mark Lutz	O'Reilly Publication, 5th Edition • ISBN13:9781449355739
3	Python Essential Reference	David Beazley	Addison-Wesley Professional 4th Edition • ISBN:9780672329784
4	Head First Python	Paul, Barry	O'Reilly Publication, 2nd Edition • ISBN: 1491919531

13. SOFTWARE/LEARNINGWEBSITES

a. https://www.tutorialspoint.com/python/index.htm
b. nptel.ac.in/courses/117106113/34
c. https://www.w3schools.com/python/default.asp
d. https://www.programiz.com/python-programming
e. http://spoken-tutorial.org/
f. https://docs.python.org/3/tutorial/
g. https://www.w3resource.com/python-exercises/
h. https://anandology.com/python-practice-book/

14. PO – COMPETENCY CO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	1	1	2	-	-	1
CO2	2	2	2	3	-	-	1
CO3	2	2	3	3	-	-	2
CO4	2	2	3	3	-	1	3
CO5	2	2	3	3	-	2	2
CO6	2	2	3	2	-	2	3

	PSO1	PSO2
CO1	-	3
CO2	-	3
CO3	-	3
CO4	-	3
CO5	-	3
CO6	-	3

Sign: Name: 1. Smt S.P.Panchakshari 2. Smt H F Khan 3. Smt A M Galshetwar 4. Smt A B Bhusagare 5. Smt S.A.Ade (CourseExperts)	Sign: Name: Mr. U.V. Kokate Dr. S.B.Nikam (Head of Department) (Department of Computer Engineering)
Sign: Name: Mr. U.V. Kokate Dr. S.B.Nikam (Programme Head) (Department of Computer Engineering)	Sign: Name: Mr. A.S. Zanpure (CDC In-charge)