# 20CS1201 - PYTHON PROGRAMMING

(Common to CSE, IT, AI&DS, ECE & EEE Branches)

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Category:** | Professional Core | **Credits:** | 3 |
| **Course Type:** | Theory | **Lecture - Tutorial - Practical:** | 3-0-0 |
| **Prerequisite:** | Basic mathematical knowledge to solve problems and programming. | **Sessional Evaluation:**  **Univ. Exam Evaluation:**  **Total Marks:** | 40  60  100 |
| **Objectives** | * To learn the fundamentals of Python constructs. * To develop various simple programs using Python. * To define Python functions, exceptions and various other features. * To explore features of object-oriented concepts. | | |

|  |  |  |
| --- | --- | --- |
| **Course Outcomes** | Upon successful completion of this course students will be able to: | |
| CO1 | Learn the basic building blocks of Python. |
| CO2 | Understand the flow of execution, exception handling mechanism and functions for application development. |
| CO3 | Study Strings, Lists and their applications. |
| CO4 | Acquire knowledge in the concepts of Dictionaries, Tuples, and Sets. |
| CO5 | Comprehend the rules to construct regular expressions, and apply them to text to search for patterns and make changes. |
| CO6 | Understand Object-oriented programming paradigm in controlling the access of data and reducing the duplication of code by employing code reusability techniques. |
| **Course Content** | UNIT - I  **Why Python:** Thrust areas of Python, Open Source Software.  **Python Basics**: Identifiers, Keyword, Statements and Expressions, variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input and Writing Output, Type Conversions, type () function and “is” operator, Dynamic and Strongly Typed Language  UNIT - II  **Control Flow Statements**: if and nested if, for, while Continue and Break statements, Catching Exceptions.  **Functions:** Built-in Functions, Commonly Used Modules, Function Definition and Calling the function, The return statement and void function, scope and lifetime of variables, Default Parameters, Keyword Arguments, Variable number of arguments with \*args and \*\*kwargs, command line arguments  UNIT - III  **Strings:** Creating and Storing Strings, Basic String Operations, Access characters by Index, Slicing and Joining of Strings, String Methods and Formatting Strings.  **Lists:** Creating Lists, List operations, indexing and Slicing, Built-in Functions, List Methods, del() vs pop()  UNIT - IV  **Dictionaries:** Creation, Accessing and modifying key-value pairs, Built-in functions used on dictionaries, Dictionary methods, del statement.  **Tuples and Sets**: Creation of Tuples, Basic Tuple Operations, Indexing and Slicing in Tuples, Built-in functions, Relationship among Tuples, Lists and Dictionaries, Tuple Methods, aggregation with zip(), Sets, Set Methods and Frozen sets.  UNIT - V  **Files**: Types, Creating, Reading Text data and methods used for it, Manipulating Binary and CSV files, pickling (serialization of objects), os and os.path modules.  **Regular Expression Operations**: Using Special Characters, Regular Expression Methods, Named Groups in Python Regular Expression and Regular Expression with glob Module.  UNIT - VI  **Object-Oriented Programming:** Classes and Objects and Creating them, The Constructor Method, Classes with Multiple Objects, Class Attributes versus Data Attributes, Encapsulation, Inheritance, Polymorphism. | |
| **Text Books and References** | Text Books:   1. Gowrishankar. S, Veena.A, “Introduction to Python Programming”, CRC Press, Taylor and Francis group,2019. | |
| Reference Books:   1. Brian Heinold, A Practical Introduction to Python Programming. 2. April Speigh, Bite-Size Python: An Introduction to Python Programming. Kenneth A. Lambert, Fundamentals of python - Data structures. 3. Mark Summerfield, Programming in python 3. 4. Yaswanth Kanetkar, Aditya Kanetkar, Let Us Python, BPB Publications, 2020 | |
| **E-Resources** | 1. <https://nptel.ac.in/courses> 2. <https://freevideolectures.com/university/iitm> 3. <https://wiki.python.org/moin/PythonBooks> | |

**CO-PO Mapping:** 3-High Mapping, 2-Moderate Mapping, 1-Low Mapping, - -Not Mapping

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
| **CO1** | 3 | - | - | - | - | - | - | - | - | - | - | - |
| **CO2** | 2 | 3 | - | - | - | - | - | - | - | - | - | - |
| **CO3** | 2 | 3 | - | - | 3 | - | - | - | - | - | - | - |
| **CO4** | - | 2 | - | 3 | 3 | - | - | - | - | - | - | - |
| **CO5** | - | - | 3 | - | 3 | - | - | - | - | - | - | - |
| **CO6** | - | 2 | 3 | 2 | 3 | - | - | - | - | 3 | - | 2 |