V Semester

Course 15 B : Application Development using Python

Credits -3

Learning Objectives:

To enable students to develop IoT solutions for real-world problems

Learning Outcomes: On successful completion of the course, students will be able to

1. Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
2. Demonstrate proficiency in handling Strings and File Systems.
3. Create, run and manipulate Python Programs using core data structures like Lists,Dictionaries and use Regular Expressions.
4. Interpret the concepts of Web Programming and GUI in Python
5. Apply concepts of Python programming in various fields related to IOT, Web Services and Databases in Python.

UNIT-I

Python basics, Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types

Numbers - Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules

Sequences - Strings, Lists, and Tuples, Dictionaries and Set Types

Control Flow, Truthiness, Sorting, List Comprehensions, Generators and Iterators

UNIT-II

Files: File Objects, File Built-in Function [open()], File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution

Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management, Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, Creating Exceptions, Why Exceptions (Now)?, Why Exceptions at All?, Exceptions and the sys Module, Related Modules

Modules: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules

UNIT-III

Regular Expressions: Introduction, Special Symbols and Characters, Res and Python**Multithreaded Programming**: Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module, Related Modules

UNIT-IV

GUI Programming: Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs

Web Programming: Introduction, Wed Surfing with Python, Creating Simple Web Clients, Advanced Web Clients, CGI-Helping Servers Process Client Data, Building CGI Application, Advanced CGI, Web (HTTP) Servers

UNIT-V

Database Programming: Introduction, Python Database Application Programmer's Interface (DBAPI), Object Relational Managers (ORMs), Related Modules

Text Book(s)

1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.
2. Think Python, Allen Downey, Green Tea Press.

Reference Books

1. Introduction to Python, Kenneth A. Lambert, Cengage.
2. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.
3. Learning Python, Mark Lutz, O' Really.

SUGGESTED CO-CURRICULAR ACTIVITIES & EVALUATION METHODS:

Unit 1: Activity: Hands-on Lab exercise on Python Control Statements
Evaluation Method: Lab Performance and Correctness of solution Implementation

Unit 2: Activity: Assignment of Files in Python
Evaluation Method: Problem Solving, Accuracy

Unit 3: Activity: Exercises on Regular expressions
Evaluation Method: Solutions, Accuracy of Validation

Unit 4: Activity: Poster Presentation on various GUI components in Python
Evaluation Method: Content knowledge, organization, clarity, presentation skills, visualaids.

Unit 5: Activity: Group Project
Evaluation Method: Project effectiveness, User interface, Solution to the Problem

V Semester

Course 15 B: Application Development using Python

Credits -1

List of Experiments:

1. Write a menu driven program to convert the given temperature from Fahrenheit to Celsiusand vice versa depending upon user's choice.
2. Write a python program to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades accordingto the following criteria :

   Grade A: Percentage >=80 Grade B: Percentage>=70 and 80

   Grade C: Percentage>=60 and <70 Grade D: Percentage>=40 and <60 Grade E: Percentage<40

3. Demonstrate various methods of Sequence Data Types
4. Write a python program to display the first n terms of Fibonacci series.
5. Write a python program to calculate the sum and product of two compatible matrices.
6. Write a function that takes a character and returns True if it is a vowel and False otherwise.
7. Write a program to implement exception handling.
8. Write a program to implement Multithreading
9. Develop a Python GUI calculator using Tkinter
10. Write a Python program to read last 5 lines of a file.
11. Design a simple database application that stores the records and retrieve the same
12. Design a database application to search the specified record from the database.
13. Design a database application to that allows the user to add, delete and modify the records.