

SES's L.S. RAHEJA COLLEGE OF ARTS AND COMMERCE (AUTONOMOUS)



BOARD OF STUDIES: Information Technology and Data Science

PROGRAMME: Bachelor of Science (Information Technology)

SEMESTER: III

NOMENCLATURE OF THE COURSE: PYTHON PROGRAMMING

NEP Vertical: MAJOR

Credit: 04

(As Per Choice Based Credit System (under NEP 2020) with effect from the academic year 2025-26)



Programme:	Bachelor of Science (Information Technology)
Nomenclature of the Course	Python Programming
Total Marks	100
Semester:	III
Academic year	2025-26

LEARNING OBJECTIVES:

1. Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.
2. Express proficiency in the handling of strings and functions.
3. Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.
4. Identify the commonly used operations involving file systems and regular expressions.
5. Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python.
6. Understand python programming fundamentals for data analysis

COURSE OUTCOMES:

1. Aware of the variables, expressions, looping and conditions used in Python programming.
2. Implement functions, strings, lists, tuples and directories.
3. Create GUI forms and add widgets.
4. Implement database connectivity in MySQL to store data.
5. Apply the programming skillset learnt here into perform data analysis with Python.

Unit	Course Content	Andragogy	No of Lectures
I	<p>1.1 Fundamentals concepts of Python Programming: Features, Installing Python, Running Python program, Debugging : Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages, The Difference Between Brackets, Braces, and Parentheses, Variables and Expressions Values and Types, Variables, Variable Names and Keywords, Type conversion, Operators and Operands, Expressions, Interactive Mode and Script Mode, Order of Operations. Conditional Statements: if, if-else, nested if –else Looping: for, while, nested loops Control statements: Terminating loops, skipping specific conditions</p> <p>1.2 Functions: Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution, Parameters and Arguments, Variables and Parameters Are Local, Stack Diagrams, Fruitful Functions and Void Functions, Why Functions? Importing with</p>	<ul style="list-style-type: none"> • Use real-life examples: Incorporate real-life examples into lessons. • Give students problems: Provide problems for students to solve independently or in groups. • Focus on practical applications: Present knowledge and abilities in terms of their practical uses. 	15

	from, Return Values, Incremental Development, Composition, Boolean Functions, More Recursion, Leap of Faith, Checking Types		
II	<p>2.1 Strings: A String Is a Sequence, Traversal with a for Loop, String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, The in Operator, String Comparison, String Operations.</p> <p>2.2 Lists: Values and Accessing Elements, Lists are mutable, traversing a List, Deleting elements from List, Built-in List Operators, Concatenation, Repetition, In Operator, Built-in List functions and methods</p> <p>2.3 Tuples and Dictionaries: Tuples, Accessing values in Tuples, Tuple Assignment, Tuples as return values, Variable-length argument tuples, Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in Tuple Functions Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods</p> <p>2.4 Files: Text Files, The File Object Attributes, Directories</p> <p>2.5 Exceptions: Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions</p>	<ul style="list-style-type: none"> • Use real-life examples: Incorporate real-life examples into lessons. • Give students problems: Provide problems for students to solve independently or in groups. • Focus on practical applications: Present knowledge and abilities in terms of their practical uses. 	15
III	<p>3.1: Regular Expressions – Concept of regular expression, various types of regular expressions, using match function.</p> <p>3.2: Classes and Objects: Overview of OOP (Object Oriented Programming), Class Definition, Creating Objects, Instances as Arguments, Instances as return values, Built-in Class Attributes, Inheritance, Method Overriding, Data Encapsulation, Data Hiding</p> <p>3.3: Multithreaded Programming: Thread Module, creating a thread, synchronizing threads, multithreaded priority queue</p> <p>3.4: Modules: Importing module, Creating and exploring modules, Math module, Random module, Time module</p> <p>3.5: Creating the GUI Form and Adding Widgets: Widgets- Button, Canvas, Checkbutton, Entry, Frame, Label, Listbox,</p>	<ul style="list-style-type: none"> • Use real-life examples: Incorporate real-life examples into lessons. • Give students problems: Provide problems for students to solve independently or in groups. • Focus on practical applications: Present knowledge and abilities in terms of their practical uses. 	15

	Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel, Spinbox, PanedWindow, LabelFrame, tkMessageBox. Handling Standard attributes and Properties of Widgets.		
IV	<p>4.1: Layout Management: Designing GUI applications with proper Layout Management features, UI Customization Enhancing Look and Feel of GUI using different appearances of widgets.</p> <p>4.2: Python with MySQL Database: Connecting to a MySQL database from Python, Configuring the MySQL connection, Designing the Python GUI database, Using the INSERT command, Using the UPDATE command, Using the DELETE command, Storing and retrieving data from MySQL database.</p> <p>4.3: Python for data analysis: Introduction to Numpy, Numerical Operations on Numpy Arrays, perform mathematical operations in NumPy, perform elements selection (conditional), understand pandas fundamentals, pandas with csv and html data, pandas operations, perform sorting and ordering in pandas</p> <p>4.4: Introduction to LLMs and LangChain: Basic concepts of LLMs, Introduction to popular LLMs (e.g., GPT models, BERT), Simple applications of LLMs in Python, LangChain Fundamentals, Overview of LangChain framework, Key components: LLMs, Prompt Templates, Basic LangChain setup in Python</p>	<ul style="list-style-type: none"> • Use real-life examples: Incorporate real-life examples into lessons. • Give students problems: Provide problems for students to solve independently or in groups. • Focus on practical applications: Present knowledge and abilities in terms of their practical uses. 	15

SUGGESTED READINGS

1. Allen Downey, Think Python, O'Reilly
2. Jason Montojo, Jennifer Campbell, Paul Gries, An Introduction to Computer Science using Python 3, SPD
3. Balagurusamy, Introduction to Problem Solving with Python E, TMH
4. Bernd Klein (2021), Data Analysis Numpy, Matplotlib and Pandas, bodenseo

QUESTION PAPER PATTERN

(A) FOR CONTINUOUS EVALUATION

Particulars	Marks
Class Test	20
Practical Implementation	10
Assignment/Project	10
Total	40

(B) FOR SEMESTER END EXAMINATION

Maximum Marks: 60

Duration: 2 Hours

Question No.	Description	Total Marks
Q. 1	Attempt the following Unit I	15
A	Remembering	
B	Analysing	
C	Applying	
	OR	
P	Remembering	
Q	Analysing	
R	Applying	
Q. 2	Attempt the following Unit II	15
A	Understand	
B	Analysing	
C	Evaluating	
	OR	
P	Understand	
Q	Analysing	
R	Evaluating	
Q. 3	Attempt the following Unit III	15
A	Understand	
B	Apply	
C	Evaluating	
	OR	
P	Understand	
Q	Apply	
R	Evaluating	
Q. 4	Attempt the following Unit IV	15
A	Understand	
B	Evaluating	
C	Creating	
	OR	
P	Understand	
Q	Evaluating	
R	Creating	