Resolution No: AC/II(23-24).2.RUS6

S. P. Mandali's

Ramnarain Ruia Autonomous College

(Affiliated to University of Mumbai)

Syllabus for

Program: F.Y.B.Sc.

Program Code: RUSCS

(As per the guidelines of National Education Policy 2020-Academic year 2023-24)

(Choice based Credit System)

PROGRAM OUTLINE
B.Sc. Computer Science
Open Electives

YEAR	SEM	COURSE CODE	TYPE OF COURSE	COURSE TITLE	CREDITS
FY B.Sc.	I	RUSCS.O101 01	Open Elective I	Python Programming	3
		RUSCSP.O 101	Practical based on Open Elective I	Practicals of Python Programming	1
	II	RUSCS.O101 02	Open Elective II	Advanced Python Programming	3
		RUSCSP.O 102	Practical based on Open Elective II	Practicals Of Advanced Python Programming	1

Computer science: F.Y.B.Sc. (OE):

Theory Course: Python Programming RUSOECS.O101(2024-25)

COURSE OUTCOMES

A student completing successfully completing this course will be able to:

CO#	Description
CO1	Describe core syntax and semantics of python
CO2	Explain data storing and processing mechanism on String, List, Dictionary, Tuples
CO3	Summarize File handling techniques
CO4	Understand concepts of functions and modules

RUSCS.O101	Python Programming	Credits 3/ 45 Hours
Unit I	**Why Python?** Reasons for Python as the learner are first programming language. Introduction to the IDLE interpreter (shell) and its documentation. **Building Blocks of Program:**	15 Hrs

	Data, Data Types, Data Binding, Variables, Constants, Declaration, Operations on Data such as assignment, arithmetic, relational, logical operations, dry run, and variables used. **Develop Code using Python:** Features, basic syntax, Writing and executing simple program, Basic Data Types such as numbers, strings, etc Declaring variables, Performing assignments, arithmetic operations, Simple input-output	
Unit II	**Sequence Control:** Precedence of operators, Type conversion Conditional Statements: if, if-else, nested if –else Looping: for, while, nested loops **Control statements:** Terminating loops, skipping specific conditions Collection Manipulation: declaring strings, string functions, Lists, Tuples, Maps	15 Hrs
Unit III	**Functions And Modules:** Defining a function, calling a function, Advantages of functions, types of functions, function parameters, Formal parameters, Actual parameters, global and local variables, Anonymous functions, List comprehension Importing module, Creating & exploring modules **Python File Input-Output:** Opening and closing files, various types of file modes, reading and writing to files, manipulating directories. Iterables, iterators and their problem solving applications.	15 Hrs

References:

1. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress,2nd edition.
2. Practical Programming: An Introduction to Computer Science Using Python, Paul Gries, et al., Pragmatic Bookshelf, 2nd Edition 2014.

Additional References:

- Introduction to Computer Science using Python, Charles Dierbach, Wiley, 2013.
- Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries , Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2nd Edition 2014
- Programming Languages - Principles and Paradigms, Adesh Pandey, Naros

CODE: RUSCSP.E102
Practicals of Python Programming

COURSE OUTCOMES

A student completing successfully completing this course will be able to:

CO	Description
CO1	Apply basics I/O operation
CO2	Demonstrate Data storage and handling techniques
CO3	Explain function and File handling techniques

COURSE CODE: RUSCSP.E102	
Sr. No.	**PRACTICAL TITLE**
1	I/O concepts
2	Conditional Statements
2	Control Statement.
3	Strings
4	Tuples
5	List
6	Dictionaries.
7	Function and anonymous function
8	Modules
9	File Handling

Computer Science: F.Y.B.Sc. (OE):

Theory Course: Advanced Python Programming RUCS.E112(2024-25)

COURSE OUTCOMES

A student completing successfully completing this course will be able to:

CO#	Description
CO1	Articulate OOPS concept such as Inheritance, polymorphism in python
CO2	Explain exception handling in python
CO3	Identify commonly used functions in Regular expression for efficiency
CO4	Develop GUI application with database

RUSCS.O101	Advanced Python Programming	Credits 3/ 45 Hours
Unit I	**Fundamental ideas of OOP**: encapsulation, inheritance, abstraction, polymorphism, Classes, Objects in python **Exception Handling:** What is an exception, various keywords to handle exceptions such try, catch, except, else, finally, raise. **Regular Expressions:** Concept of regular expression, various types of regular expressions, using match function	15 Hrs
Unit II	**GUI Programming in Python :** What is GUI, Advantages of GUI, Introduction to GUI library. Layout management, events and bindings, fonts, colours, drawing on canvas (line, oval, rectangle, etc.) **Widgets:** frame, label, button, checkbutton, entry, listbox, message, radiobutton, text, spinbox etc	15 Hrs
Unit III	**Database Connectivity In Python:** Installing mysql connector, accessing connector module module, using connect, cursor, execute & close functions, reading single & multiple results of query execution, executing different types of statements, executing transactions, understanding exceptions in database connectivity. **Network Connectivity:** Socket module, creating server-client programs, sending email, reading from URL	15 Hrs

References:

1. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries , Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2nd Edition 2014

Additional References:

- Beginning Python: Using Python 2.6 and Python 3, James Payne , Wiley India, 2010.
- MySQL for Python: Database Access Made Easy, A. Lukaszewski, Pact Publisher, 2010.

CODE: RUSCSP.E112
Practicals of Advanced Python Programming

COURSE OUTCOMES

A student completing successfully completing this course will be able to:

CO	Description
CO1	Implement Exception handling techniques
CO2	Design GUI Applications
CO2	Apply Object-oriented Programming concepts

COURSE CODE: RUSCSP.E112	
Sr. No.	**PRACTICAL TITLE**
1	OOPS concepts
2	Exception handling
3	Regular expression
4	Database connectivity
5	Basic GUI programming
6	Advanced GUI programming
7	Networking
8	Email communication