



INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad -500 043

INFORMATION TECHNOLOGY

COURSE DESCRIPTOR

Course Title	OBJECT ORIENTED PROGRAMMING THROUGH PYTHON				
Course Code	AITB01				
Programme	B.Tech				
Semester	III	CSE IT			
Course Type	Core				
Regulation	IARE - R18				
Course Structure	Theory			Practical	
	Lectures	Tutorials	Credits	Laboratory	Credits
	3		3	-	-
Chief Coordinator	Dr.M Purushotham Reddy, Associate Professor				
Course Faculty	Dr. R Obulakonda Reddy, Associate Professor Ms. A Lakshmi, Assistant Professor Ms. B Padmaja, Associate Professor Ms. B Tejaswi, Assistant Professor Mr. P Ravinder, Assistant Professor				

I. COURSE OVERVIEW:

This course explains the fundamental ideas behind the object oriented approach to programming. Knowledge of python helps to create the latest innovations in programming. Like the successful computer languages that came before, python is the blend of the best elements of its rich heritage combined with the innovative concepts required by its unique environment. This course involves OOP concepts, python basics, inheritance, polymorphism, interfaces, packages, Exception handling. This course is presented to students by power point projections, course handouts, lecture notes, assignments, objective and subjective tests.

II. COURSE PRE-REQUISITES:

Level	Course Code	Semester	Prerequisites	Credits
UG	ACSB01	II	Programming for problem solving	3

III. MARKS DISTRIBUTION:

Subject	SEE Examination	CIA Examination	Total Marks
Object Oriented Programming Through Python	70 Marks	30 Marks	100

IV. DELIVERY / INSTRUCTIONAL METHODOLOGIES:

✗	Chalk & Talk	✓	Quiz	✓	Assignments	✗	MOOCs
✓	LCD / PPT	✓	Seminars	✗	Mini Project	✓	Videos
✗	Open Ended Experiments						

V. EVALUATION METHODOLOGY:

The course will be evaluated for a total of 100 marks, with 30 marks for Continuous Internal Assessment (CIA) and 70 marks for Semester End Examination (SEE). Out of 30 marks allotted for CIA during the semester, marks are awarded by taking average of two CIA examinations or the marks scored in the make-up examination.

Semester End Examination (SEE): The SEE is conducted for 70 marks of 3 hours duration. The syllabus for the theory courses is divided into FIVE modules and each module carries equal weightage in terms of marks distribution. The question paper pattern is as follows. Two full questions with “either” or “choice” will be drawn from each module. Each question carries 14 marks. There could be a maximum of two sub divisions in a question.

The emphasis on the questions is broadly based on the following criteria:

50 %	To test the objectiveness of the concept.
50 %	To test the analytical skill of the concept OR to test the application skill of the concept.

Continuous Internal Assessment (CIA):

CIA is conducted for a total of 30 marks (Table 1), with 20 marks for Continuous Internal Examination (CIE), 05 marks for Quiz and 05 marks for Alternative Assessment Tool (AAT).

Table 1: Assessment pattern for CIA

Component	Theory			Total Marks
	CIE Exam	Quiz	AAT	
CIA Marks	20	05	05	30

Continuous Internal Examination (CIE):

Two CIE exams shall be conducted at the end of the 8th and 16th week of the semester respectively. The CIE exam is conducted for 20 marks of 2 hours duration consisting of five descriptive type questions out of which four questions have to be answered where, each question carries 5 marks. Marks are awarded by taking average of marks scored in two CIE exams.

Quiz - Online Examination

Two Quiz exams shall be online examination consisting of 25 multiple choice questions and are to be answered by choosing the correct answer from a given set of choices (commonly four). Such a question paper shall be useful in testing of knowledge, skills, application, analysis, evaluation and understanding of the students. Marks shall be awarded considering the average of two quiz examinations for every course.

Alternative Assessment Tool (AAT)

This AAT enables faculty to design own assessment patterns during the CIA. The AAT converts the classroom into an effective learning centre. The AAT may include tutorial hours/classes, seminars, assignments, term paper, open ended experiments, METE (Modeling and Experimental Tools in Engineering), five minutes video, MOOCs etc.

VI. HOW PROGRAM OUTCOMES ARE ASSESSED:

Program Outcomes (POs)		Strength	Proficiency assessed by
PO1	Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.	3	Assignments
PO2	Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences	2	Assignments
PO3	Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.	3	Guest Lectures
PO 4	Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.	2	5 minutes Video/ Seminars
PO 5	Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.	3	Seminars / Term Paper / 5 minutes video

3 = High; 2 = Medium; 1 = Low

VII. HOW PROGRAM SPECIFIC OUTCOMES ARE ASSESSED:

Program Specific Outcomes (PSOs)		Strength	Proficiency assessed by
PSO 1	Professional Skills: To produce engineering professional capable of synthesizing and analyzing mechanical systems including allied engineering streams.	2	Seminar/SEE
PSO 2	Software Engineering Practices: An ability to adopt and integrate current technologies in the design and manufacturing domain to enhance the employability.	3	Quiz/AAT
PSO 3	Successful Career and Entrepreneurship: To build the nation, by imparting technological inputs and managerial skills to become technocrats.	2	Guest Lectures

3 = High; 2 = Medium; 1 = Low

VIII. COURSE OBJECTIVES :

The course should enable the students to:	
I	Understand the fundamentals of Python programming concepts and its applications.
II	Understand the object-oriented concepts using Python in problem solving.
III	Apply string handling and function basics to solve real-time problems.
IV	Illustrate the method of solving errors using exception handling.
V	Design and implement programs using multi threading concepts

IX. COURSE OUTCOMES (COs):

COs	Course Outcome	CLOs	Course Learning Outcome
CO 1	Describe Features of Python, Data types, Operators, Input and output, Control Statements, Features of Object oriented programming system (OOPS).Classes and Objects, Encapsulation, Abstraction, Inheritance, Polymorphism	CLO 1	Describe the Features of Python, Data types.
		CLO 2	Summarize the concept of Operators, Input and output, Control Statements.
		CLO 3	Identify the features of Object Oriented Programming System (OOPS),
		CLO 4	Use the concept of Classes and Objects, Encapsulation.
		CLO 5	Describe Abstraction, Inheritance, and Polymorphism.
CO 2	Determine Creating a class, The Self variable, Constructor, Types of Variable, Namespaces, Types of Methods,	CLO 6	Determine Creating a class, The Self variable.
		CLO 7	Understand types of variable, Namespaces.
		CLO 8	Determine types of Methods, Inheritance and Polymorphism.

COs	Course Outcome	CLOs	Course Learning Outcome
	Inheritance and Polymorphism Constructors in inheritance, The super() method, Types of inheritance, Polymorphism, Abstract classes and Interfaces	CLO 9	Use Constructors in inheritance, the super() method.
		CLO 10	Illustrate types of inheritance, Polymorphism, Abstract classes and Interfaces.
CO 3	Understand Creating strings and basic operations on strings, String testing methods, Defining a function, Calling a function, Returning multiple values from a function, Functions are first class objects, Formal and actual arguments, Positional arguments, Recursive functions.	CLO 11	Understand Creating strings and basic operations on strings.
		CLO 12	Analyze the concept of String testing methods, Defining a function.
		CLO 13	Illustrate Calling a function, Returning multiple values from a function.
		CLO 14	Contrast the Usage of Functions are first class objects, Formal and actual arguments,
		CLO 15	Define Positional arguments, Recursive functions.
CO 4	Explore the concept of Errors in a Python program, Exceptions, Exception handling, Types of exceptions, The Except block, The assert statement, user-defined exceptions.	CLO 16	Discuss the concept of Errors in a Python program.
		CLO 17	Understand Exceptions, Exception handling.
		CLO 18	Summarize the concept of types of exceptions.
		CLO 19	Discuss the Except block, the assert statement.
		CLO 20	Understand the concept of user-defined exceptions.
CO 5	Knowledge The Root window, Fonts and colors, Working with containers, Canvas, Frames, Widgets ,Button widget, Label Widget, Message widget, Text widget, Radio button Widget, Entry widget.	CLO 21	Knowledge about the Root window, Fonts and colors.
		CLO 22	Apply Working with containers, Canvas.
		CLO 23	Understand Widgets, Button widget, Label Widget.
		CLO 24	Implement Message widget, Text widget.
		CLO 25	Illustrate Radio button Widget, Entry widget.

X. COURSE LEARNING OUTCOMES (CLOs):

CLO Code	CLO's	At the end of the course, the student will have the ability to:	PO's Mapped	Strength of Mapping
AITB01.01	CLO 1	Describe the Features of Python, Data types.	PO1	3
AITB01.02	CLO 2	Summarize the concept of Operators, Input and output, Control Statements.	PO2	2
AITB01.03	CLO 3	Identify the features of Object Oriented Programming System (OOPS).	PO1	3

CLO Code	CLO's	At the end of the course, the student will have the ability to:	PO's Mapped	Strength of Mapping
AITB01.04	CLO 4	Use the concept of Classes and Objects, Encapsulation.	PO2	2
AITB01.05	CLO 5	Describe Abstraction, Inheritance, and Polymorphism.	PO1	3
AITB01.06	CLO 6	Determine Creating a class, The Self variable.	PO2	2
AITB01.07	CLO 7	Understand types of variable, Namespaces.	PO2, PO3	3
AITB01.08	CLO 8	Determine types of Methods, Inheritance and Polymorphism.	PO2	2
AITB01.09	CLO 9	Use Constructors in inheritance, the super() method.	PO3	3
AITB01.10	CLO 10	Illustrate types of inheritance, Polymorphism, Abstract classes and Interfaces.	PO2, PO3	3
AITB01.11	CLO 11	Understand Creating strings and basic operations on strings.	PO2	2
AITB01.12	CLO 12	Analyze the concept of String testing methods, Defining a function.	PO4	2
AITB01.13	CLO 13	Illustrate Calling a function, Returning multiple values from a function.	PO2, PO4	2
AITB01.14	CLO 14	Contrast the Usage of Functions are first class objects, Formal and actual arguments,	PO2	2
AITB01.15	CLO 15	Define Positional arguments, Recursive functions.	PO2, PO4	2
AITB01.16	CLO 16	Discuss the concept of Errors in a Python program.	PO1, PO2	3
AITB01.17	CLO 17	Understand Exceptions, Exception handling.	PO1, PO2	3
AITB01.18	CLO 18	Summarize the concept of types of exceptions.	PO2	2
AITB01.19	CLO 19	Discuss the Except block, the assert statement.	PO2	2
AITB01.20	CLO 20	Understand the concept of user-defined exceptions.	PO1, PO2	3
AITB01.21	CLO 21	Knowledge about the Root window, Fonts and colors.	PO1, PO4	3
AITB01.22	CLO 22	Apply Working with containers, Canvas.	PO4	2
AITB01.23	CLO 23	Understand Widgets, Button widget, Label Widget.	PO1	3
AITB01.24	CLO 24	Implement Message widget, Text widget.	PO4	2
AITB01.25	CLO 25	Illustrate Radio button Widget, Entry widget.	PO1	3

3= High; 2 = Medium; 1 = Low

XI. MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES

Course Outcomes (COs)	Program Outcomes (POs)						
	PO1	PO2	PO3	PO4	PSO1	PSO2	PSO3
CO1	3	2			2		
CO2		2	3			3	
CO3		2		2		3	
CO4	3	2				3	
CO5	3			2			2

3= High; 2 = Medium; 1 = Low

XII. MAPPING COURSE LEARNING OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Learning Outcomes (CLOs)	Program Outcomes (POs)												Program Specific Outcomes (PSOs)		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CLO 1	3												2		
CLO 2		2													
CLO 3	3												2		
CLO 4		2											2		
CLO 5	3														
CLO 6		2													
CLO 7		2	3											3	
CLO 8		2													
CLO 9			3											3	
CLO 10		2	3												
CLO 11		2												3	
CLO 12				2											
CLO 13		2		2										3	
CLO 14		2												3	
CLO 15		2		2											
CLO 16	3	2												3	

Course Learning Outcomes (CLOs)	Program Outcomes (POs)												Program Specific Outcomes (PSOs)		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CLO 17	3	2												3	
CLO 18		2													
CLO 19		2												3	
CLO 20	3	2													
CLO 21	3			2											2
CLO 22				2											
CLO 23	3														2
CLO 24				2											2
CLO 25	3			2											

3 = High; 2 = Medium; 1 = Low

XIII. ASSESSMENT METHODOLOGIES – DIRECT

CIE Exams	PO1, PO2, PO3, PO4, PO5, PSO1, PSO2, PSO3	SEE Exams	PO1, PO2, PO3, PO4, PO5, PSO1, PSO2, PSO3	Assignments	PO1, PO2, PSO2	Seminars	PO4, PO5, PSO1
Laboratory Practices	-	Student Viva	-	Mini Project	-	Certification	-
Term Paper	PO5						

XIV. ASSESSMENT METHODOLOGIES – INDIRECT

✓	Early Semester Feedback	✓	End Semester OBE Feedback
✗	Assessment of Mini Projects by Experts		

XV. SYLLABUS

Module-I	INTRODUCTION TO PYTHON AND OBJECT ORIENTED CONCEPTS
Introduction to Python: Features of Python, Data types, Operators, Input and output, Control Statements.	
Introduction to Object Oriented Concepts: Features of Object oriented programming system (OOPS) - Classes and Objects, Encapsulation, Abstraction, Inheritance, Polymorphism.	
Module-II	PYTHON CLASSES AND OBJECTS
Classes and Objects: Creating a class, The Self variable, Constructor, Types of Variable, Namespaces, Types of Methods, Inheritance and Polymorphism – Constructors in inheritance, The super() method,	

Types of inheritance, Polymorphism, Abstract classes and Interfaces.	
Module-III	STRINGS AND FUNCTIONS
Strings: Creating strings and basic operations on strings, String testing methods.	
Functions: Defining a function, Calling a function, Returning multiple values from a function, Functions are first class objects, Formal and actual arguments, Positional arguments, Recursive functions.	
Module-IV	EXCEPTION HANDLING
Exception: Errors in a Python program, Exceptions, Exception handling, Types of exceptions, The Except block, The assert statement, user-defined exceptions.	
Module-V	GRAPHICAL USER INTERFACE
GUI in Python: The Root window, Fonts and colors, Working with containers, Canvas, Frames, Widgets Button widget, Label Widget, Message widget, Text widget, Radio button Widget, Entry widget.	
Text Books:	
1. R Nageswara Rao, Core Python Programming, Dreamtech press, 2017 Edition. 2. Dusty Philips, Python 3 Object Oriented Programming, PACKT Publishing, 2 nd Edition 2015.	
Reference Books:	
1. Michael H.Goldwasser, David Letscher, Object Oriented Programming in Python, Prentice Hall; 1 st Edition, 2007.	

XVI. COURSE PLAN:

The course plan is meant as a guideline. Probably there may be changes.

Lecture No	Topics to be covered	Course Learning Outcomes (CLOs)	Reference
1	Describe the Features of Python, Data types.	CLO 1	T1:1.2
2-3	Summarize the concept of Operators, Input and output, Control Statements.	CLO 2	T1:4,5,6
4-5	Identify the features of Object Oriented Programming System (OOPS),	CLO 3	T12.3
6-7	Use the concept of Classes and Objects, Encapsulation.	CLO 4	T1:12.4,12.5
8-9	Describe Abstraction, Inheritance, and Polymorphism.	CLO 5	T1:12.6-12.8
10-11	Determine Creating a class, The Self variable.	CLO 6	T1:13.1,13.2
12-13	Understand types of variable, Namespaces.	CLO 7	T1:13.4,13.5
14-15	Determine types of Methods, Inheritance and Polymorphism.	CLO 8	T1:13.6,14
16-18	Use Constructors in inheritance, the super() method.	CLO 9	T1:14.1,14.3
19-20	Illustrate types of inheritance, Polymorphism, Abstract classes and Interfaces.	CLO 10	T1:14.4,14.6
21-22	Understand Creating strings and basic operations on strings.	CLO 11	T1:8.1
23	Analyze the concept of String testing methods,	CLO 12	T1:8.17

Lecture No	Topics to be covered	Course Learning Outcomes (CLOs)	Reference
24-25	Defining a function.	CLO 12	T1:9.2
26-27	Illustrate Calling a function.	CLO 13	T1:9.3
28	Illustrate Returning multiple values from a function.	CLO 13	T1:9.5
29	Contrast the Usage of Functions is first class objects.	CLO 14	T1:9.6
30	Contrast the Usage of Formal and actual arguments.	CLO 14	T1:9.8
31	Define Positional arguments, Recursive functions.	CLO 15	T1:9.9,9.16
32-34	Discuss the concept of Errors in a Python program.	CLO 16	T1:16.1
35	Understand Exceptions, Exception handling.	CLO 17	T1:16.2,16.3
36	Summarize the concept of types of exceptions.	CLO 18	T1:16.4
37	Discuss the Except block, the assert statement.	CLO 19	T1:16.5,16.6
38	Understand the concept of user-defined exceptions.	CLO 20	T1:16.7
39	Knowledge about the Root window, Fonts and colors.	CLO 21	T1:22.2,22.3
40-41	Apply Working with containers, Canvas.	CLO 22	T1:22.4,22.5
42	Understand Widgets, Button widget, Label Widget.	CLO 23	T1:22.7
43	Implement Message widget, Text widget.	CLO 24	T1:22.11
44-45	Illustrate Radio button Widget, Entry widget.	CLO 25	T1:22.8

XVII. GAPS IN THE SYLLABUS-TO MEET INDUSTRY / PROFESSION REQUIREMENTS:

S no	Description	Proposed actions	Relevance with POs	Relevance with PSOs
1	To improve standards and analyze the concepts.	Seminars	PO1	PSO1
2	Implementation of Threads and many problems with threads	Seminars / NPTEL	PO4	PSO2
3	Encourage students to solve real time applications and prepare towards competitive examinations.	NPTEL	PO 2	PSO1

Prepared by:
Dr.M Purushotham Reddy, Associate Professor, IT

HOD, IT