

A. Course Handout

Institute/School Name	Chitkara University Institute of Engineering and Technology		
Department Name	Department of Computer Science & Engineering		
Programme Name	Bachelor of Engineering (B.E.), Computer Science & Engineering		
Course Name	Problem Solving using Python Programming	Session	2023-2024
Course Code	23CS001	Semester/Batch	1 st /2023
L-T-P (Per Week)	4-0-4	Course Credits	06
Course Coordinator	Dr. Renu Popli		

1. Objectives of the Course

Python is an open-source, high-level, dynamically-typed, portable, expressive, easy to learn, and well known programming language. Python is available as an open-source offering and is associated with large global community support. Top companies such as Google, Yahoo!, Pinterest, Disney, Nokia, IBM have python as one of their main programming language. This course provides a wide scope of learning & understanding of python programming. The main objectives of the course are:

- To understand the basics of python (operations, control structures, data types, etc.)
- Apply various data structures (list, tuple, string, dict.) for problem solving.
- Design and implementation of logic building programs using python constructs.
- Understand the object-oriented program design and development.
- Understand class inheritance and polymorphism.
- Design and develop real time Applications using python GUI programming.

2. Course Learning Outcomes

Student should be able to:

	Course Outcome	POs	CL	KC	Sessions
CLO01	Understand python lexical features, structures and flow control for logic building.	PO1, PO3, PO10	K2	Factual Conceptual	4
CLO02	Apply decision statements and loops in python to solve complex problems.	PO1, PO2, PO3, PO4	K3	Fundamental Conceptual	8
CLO03	Implement Python functions to facilitate code reuse and manipulate strings.	PO3, PO5, PO12	K3	Conceptual Procedural	9
CLO04	Create structured data using Lists and various techniques to access elements within the Lists.	PO2, PO3, PO5, PO9, PO11, PO12	K3	Conceptual Procedural	6
CLO05	Apply features of sets, tuples, and dictionaries to create solutions for real world problems.	PO5, PO11, PO12	K3	Conceptual Procedural	10
CLO06	Develop real world applications using Object Oriented Approach.	PO3, PO5, PO7, PO10, PO11, PO12	K3	Procedural Fundamental Design Principles	8

CLO07	Demonstrate the use of built-in functions to navigate the file system.	PO3,PO4,PO5	K3	Conceptual Procedural	5
CLO08	Develop projects using python GUI programming.	PO3,PO5,PO7,PO10,PO11,PO12	K3	Procedural Fundamental Design Principles	10
Total Contact Hours					90

Revised Bloom's Taxonomy Terminology

*Cognitive Level =CL

*Knowledge Categories = KC

Course Learning Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CLO01	H		H							L		
CLO02	H	H	H	M								
CLO03			H		L							M
CLO04		H	H		M				M		L	L
CLO05					M						H	M
CLO06			H		H		M			H	M	L
CLO07			H	M	H							
CLO08			M		H		H			H	M	L

H=High, M=Medium, L=Low

3. ERISE Grid Mapping

Feature Enablement	Level(1-5, 5 being highest)
Entrepreneurship	2
Research	4
Innovation	3
Skills	5
Employability	4

4. Recommended Books:

B01: Thareja, Reema, "Python Programming: Using Problem Solving Approach". Oxford University Press c2017.

B02: Wesley J Chun, "Core Python Applications Programming", 3rd Edition, Pearson Education

India, 2015. ISBN-13: 978-9332555365.

B03: Budd, Timothy A., "Exploring Python", MacGraw Hill Education c2017.

B04: David Beazley & Brian K. Jones, "Python Cookbook: Recipes for Mastering Python 3", O'Reilly Media, 3rd Edition.

B05: Roberto Tamassia, Michael H Goldwasser, Michael T Goodrich, "Data Structures and Algorithms in Python", 1st Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126562176

B06: Ravichandran, R.Saravanan, "Learn to Master Python", Staredu Solution, c2018.

5. Other readings and relevant websites:

Serial No	Link of Journals, Magazines, websites and Research Papers
1.	https://www.python.org/
2.	https://learnpython.org/
3.	https://www.codecademy.com/learn/python
4.	http://www.pyschools.com/
5.	https://www.codementor.io/learn-python-online

6. Recommended Tools and Platforms

Python 3 IDLE, Anaconda 3 (Jupyter notebook 3.6.0), Google collab

7. Course Plan: Theory+Lab Plan

Theory Plan

Session No.	Topic(s)
1	Introduction to Python programming. Python environment setup, identifiers, reserved words ,lines and indentation, multiline statements, comments, quotation, input/output, output formatting(Integers and string). Fundamentals: Variables(simple, assigning multiple values, output variables, global/local variables)
2	Python decision making- If else and Nested, If else and elif. Match statement, Introduction to while and for loop.
3	Concept of nested looping. For loop and range method, Break, Continue and Pass, Pattern Design.
4	Defining and calling Functions, Type of arguments (required, keyword, default, variable length), Scope Rules(Global/local Statements), Passing Arguments(call by value/call by reference), Recursive Functions, Lambda Expressions.
5	Lists: Introduction to Lists, Operations on Lists, Storage Structure, Resizing and Looping on Lists, Indexing and Slicing, Passing variables through Functions, passing Lists through Functions.
ST1	
6	List Comprehension, map, filter & reduce functions, Two Dimensional Lists: Introduction and Storage, Jagged Lists.
7	String- Introduction, methods, comprehension and its relative properties.
8 to 9	Tuples, Set and Dictionaries- Introduction, methods, comprehension and its relative properties.
10	Modules: Creating Modules, Variables in Modules, Imports and Attributes, Namespaces , Reloading, Built in modules, Generating Random values.
11	Files and Directories, File I/O, File positioning, File operators.

12	Exception Handling: Introduction to Exception, Exception Handling, Except clause, Try, finally clause, User Defined Exceptions.
ST2	
13-15	Exploring Object-Oriented Programming In Python-Introduction, Creating classes, objects, attributes. Constructors., Idea of inheritance, Superclass And SubClass. Overriding methods, Object class , Instance function.
End Term Exam	

Lab Plan

Session no	Sub-session as per Theory plan	Topic(s)
1	1.1	Anaconda Installation, Running Python Programs , Writing Python Scripts with Jupyter Notebook.
2	1.2	Fundamentals of Python as per lecture no. 1
3	1.3	Data types(text type, numeric, sequence, mapping, set), Python basic operators-arithmetic, comparison, assignment operators, logical operators. , mix type conversion, integer to float and float to integer conversion.
4	2.1	Coding problems related to conditional statements(if, elif, nested if else) and match statement.
5	2.2	Coding problems related to conditional statements(if, elif, nested if else) and match statement.
6-7	3.1-3.2	Coding problems related to for/while loops, For loop and range method, Break, continue and pass.
8-9	3.3-3.4	Nested loops and pattern designs
10-12	4.1-4.3	Implementation related to Functions: User Defined and Built in functions(Python mathematical functions, random number functions), Passing Arguments(call by value/call by reference), Recursive Functions, Lambda Expressions.
13-14	5.1-5.2	Implementation of lists. Creating empty list, Initializing list, list indexing and slicing operations, input method, Concatenating multiple lists, generating range based lists. list based methods like append(), extend(), insert(), index(), count(), sort(), reverse(), pop() and nested lists.
15	6.1	Implementation of 2-D list with basic operations.
16-18	7.1-7.3	Python strings- accessing values in strings, updating strings, string special characters, string formatting operators, triple quotes, built in string methods. Capitalize(), count(), endswith, startswith, find, index, isalpha, isnum, isdigit, islower, isnumeric, isupper, lower, max, min, replace, splitwith, strip.
19-20	8.1-8.2	Implementation of Sets and relative methods.
21	8.3	Implementation of Tuples and relative methods.
22-23	Continuous Evaluation 1(15 marks)	
24-26	9.1-9.3	Creating empty dictionaries, initializing dictionaries, Accessing dictionary items, merging, and deletion. Understanding dictionary specific methods, keys (), values (), items (), copy (), update (), pop () and dictionary comprehension, Python date and time functions.
27	10.1	Python modules-import, from... import statements, dir function, Generating Random values (random module)
28-29	11.1-11.2	File handling- Writing to a file, Reading from a file, Resetting the file current position, fseek
30	13.1	Exploring Object-Oriented Programming In Python:, Creating classes, objects, attributes (implementation with Examples)

31-32	14.1-14.2	Constructors and idea of inheritance., Superclass And SubClass (implementation with Examples)
33-34	15.1-15.2	Overriding methods, Object class , Isinstance function, Exploring Operator Overloading- Indexing And Slicing, Iterable Objects (implementation with Examples)
35	NA	Introduction to GUI Programming (Tkinter), Creating user defined modules, importing existing modules, Creating root windows and calling various dialog boxes on button click response.
36-37	NA	Adding a Widget, Working with widgets(Displaying Text and Images With Label Widgets, Displaying Clickable Buttons With Button Widgets, Getting User Input With Entry Widgets, Getting Multiline User Input With Text Widgets, Assigning Widgets to Frames With Frame Widgets, Naming Conventions)
38-39	NA	Controlling Layout With Geometry Managers(.pack(), .place(), .grid()), Using event and event handlers(.bind())
40	NA	Dealing with text-boxes, list-boxes, menus and dealing with basic events.
41	NA	Building a simple App using Tkinter
42-43	NA	Connecting Python With Oracle 11G- Performing insertion, Updation and deletion operation using python GUI.
44-45	NA	Continuous Evaluation 2(15 Marks)

8. Delivery/Instructional Resources

Theory Plan:

Lec tur e No.	Topics	Web References	Audio-Video
1	Introduction to Python programming. Python environment setup, identifiers, reserved words ,lines and indentation, multiline statements, comments, quotation, input/output, output formatting(Integers and string). Fundamentals: Variables(simple, assigning multiple values, output variables, global/local variables)	https://www.anaconda.com/ https://www.learnpython.org/ https://www.geeksforgeeks.org/python-programming-language/	https://nptel.ac.in/courses/106106182
2	Python decision making- If else and Nested, If else and elif. Match statement, Introduction to while and for loop.	https://docs.python.org/3/tutorial/controlflow.html	https://nptel.ac.in/courses/106106182 https://nptel.ac.in/courses/106106182 https://nptel.ac.in/courses/106106182

3	Concept of nested looping. For loop and range method, Break, Continue and Pass, Pattern Design.	https://docs.python.org/3/tutorial/controlflow.html https://docs.python.org/3/tutorial/controlflow.html#for-statements	https://nptel.ac.in/courses/106106182 https://nptel.ac.in/courses/106106182
4	Defining and calling Functions, Type of arguments (required, keyword, default, variable length), Scope Rules(Global/local Statements), Passing Arguments(call by value/call by reference), Recursive Functions, Lambda Expressions.	https://docs.python.org/3/tutorial/controlflow.html#more-on-defining-functions	https://www.coursera.org/lecture/python-genomics/lecture-5-1-functions-part-1-5-54-PGXxZ https://www.coursera.org/lecture/python-genomics/lecture-5-2-functions-part-2-8-20-ylbSK
5	Lists: Introduction to Lists, Operations on Lists, Storage Structure, Resizing and Looping on Lists, Indexing and Slicing, Passing variables through Functions, passing Lists through Functions.	https://docs.python.org/3/tutorial/datastructures.html	https://nptel.ac.in/courses/106106182
6	List Comprehension, map, filter & reduce functions, Two Dimensional Lists: Introduction and Storage, Jagged Lists.	https://docs.python.org/3/tutorial/datastructures.html#list-comprehensions	https://nptel.ac.in/courses/106106182
7	String- Introduction, methods, comprehension and its relative properties.	https://docs.python.org/3/tutorial/controlflow.html#documentation-strings	https://www.coursera.org/lecture/python-representation/python-strings-bbBTs https://www.coursera.org/lecture/python-data-analysis/python-more-on-strings-
8-9	Tuples, Set and Dictionaries- Introduction, methods, comprehension and its relative properties.	https://docs.python.org/3/tutorial/datastructures.html#tuples-and-sequences	https://nptel.ac.in/courses/106106182
10	Modules: Creating Modules, Variables in Modules, Imports and Attributes, Namespaces , Reloading, Built in modules, Generating Random values.	https://docs.python.org/3/tutorial/modules.html	https://www.coursera.org/lecture/python-genomics/lecture-6-modules-and-packages-10-32-rejwK

11	Files and Directories, File I/O, File positioning, File operators.	https://python-course.eu/python-tutorial/file-management.php https://docs.python.org/3/tutorial/inputoutput.html#reading-and-writing-files	https://www.coursera.org/lecture/data-analytics-accountancy-1/python-file-i-o-cSSim https://www.coursera.org/lecture/accouting-data-analytics-python/5-1-python-file-io-R6L3h
12	Exception Handling: Introduction to Exception, Exception Handling, Except clause, Try, finally clause, User Defined Exceptions.	https://docs.python.org/3/tutorial/errors.html https://ehmatthes.github.io/pcc/cheatsheets/README.html#chapter-10-files-and-exceptions	https://www.coursera.org/projects/exception-handling-in-python
13-15	Exploring Object-Oriented Programming In Python- Introduction, Creating classes, objects, attributes. Constructors., Idea of inheritance, Superclass And SubClass. Overriding methods, Object class , Instance function.	https://docs.python.org/3/tutorial/classes.html?highlight=object%20oriented	https://www.coursera.org/learn/object-oriented-python

Lab Plan:

Session No.	Topics	Web References	Audio-Video
1.1	Anaconda Installation, Running Python Programs , Writing Python Scripts with Jupyter Notebook.	https://docs.anaconda.com/anaconda/navigator/tutorials/ https://www.anaconda.com/	https://nptel.ac.in/courses/106106182
1.2	Fundamentals of Python as per lecture no. 1	https://www.anaconda.com/ https://www.learnpython.org/	https://nptel.ac.in/courses/106106182
1.3	Data types(text type, numeric, sequence, mapping, set), Python basic operators- arithmetic, comparison, assignment operators, logical operators. , mix type conversion, integer to float and float to	https://www.learnpython.org/ https://www.geeksforgeeks.org/python-programming-language/	https://nptel.ac.in/courses/106106182
2.1-2.2	Coding problems related to conditional statements(if, elif, nested if else) and match statement.	https://www.learnpython.org/ https://docs.python.org/3/tutorial/controlflow.html	https://nptel.ac.in/courses/106106182
3.1-3.2	Coding problems related to for/while loops, For loop and range method, Break, continue and pass.	https://docs.python.org/3/tutorial/controlflow.html https://docs.python.org/3/tutorial/controlflow.html#for-	https://nptel.ac.in/courses/106106182

3.3-3.4	Nested loops and pattern designs	https://docs.python.org/3/tutorial/controlflow.html https://docs.python.org/3/tutorial/controlflow.html#for-statements	https://nptel.ac.in/courses/106106182
4.1-4.3	Implementation related to Functions: User Defined and Built in functions(Python mathematical functions, random number functions), Passing Arguments(call by value/call by reference), Recursive Functions, Lambda Expressions.	https://docs.python.org/3/tutorial/controlflow.html#more-on-defining-functions	https://www.coursera.org/lecture/python-genomics/lecture-5-1-functions-part-1-5-54-PGXxZ https://www.coursera.org/lecture/python-genomics/lecture-5-2-functions-part-2-8-20-ylbSK
5.1-5.3	Implementation of lists. Creating empty list, Initializing list, list indexing and slicing operations, input method, Concatenating multiple lists, generating range based lists. list based methods like append(), extend(), insert(), index(), count(), sort(), reverse(), pop() and nested lists.	https://docs.python.org/3/tutorial/datastructures.html	https://nptel.ac.in/courses/106106182
6.1	Implementation of 2-D list with basic operations.	https://docs.python.org/3/tutorial/datastructures.html	https://nptel.ac.in/courses/106106182
7.1-7.3	Python strings- accessing values in strings, updating strings, string special characters, string formatting operators, triple quotes, built in string methods. Capitalize(), count(), endswith, startswith, find, index, isalpha, isnum, isdigit, islower, isnumeric, isupper, lower, max, min, replace, splitwith, strip.	https://docs.python.org/3/tutorial/controlflow.html#documentation-strings	https://www.coursera.org/lecture/python-representation/python-on-strings-bbBTs https://www.coursera.org/lecture/python
8.1-8.2	Implementation of Sets and relative methods.	https://docs.python.org/3/tutorial/datastructures.html#tuples-and-sequences	https://nptel.ac.in/courses/106106182
8.3	Implementation of Tuples and relative methods.	https://docs.python.org/3/tutorial/datastructures.html#tuples-and-sequences	https://nptel.ac.in/courses/106106182
9.1-9.3	Creating empty dictionaries, initializing dictionaries, Accessing dictionary items, merging, and deletion. Understanding dictionary specific methods, keys (), values (), items (), copy (), update (), pop () and dictionary comprehension, Python date and time functions	https://docs.python.org/3/tutorial/datastructures.html#tuples-and-sequences	https://nptel.ac.in/courses/106106182

10.1	Python modules-import, from... import statements, dir function, Generating Random values (random module)	https://docs.python.org/3/tutorial/modules.html	https://www.coursera.org/lecture/python-genomics/lecture-6-modules-and-
11.1-11.2	File handling- Writing to a file, Reading from a file, Resetting the file current position, fseek	https://python-course.eu/python-tutorial/file-management.php https://docs.python.org/3/tutorial/inputoutput.html#reading-and-writing-files	https://www.coursera.org/lecture/data-analytics-accountancy-1/python-file-i-o-cSSim https://www.coursera.org/lecture/accounting-data-analytics-python/5-1-python-file-io-R6L3h
13.1	Exploring Object-Oriented Programming In Python:, Creating classes, objects, attributes (implementation with Examples)	https://docs.python.org/3/tutorial/classes.html?highlight=object%20oriented	https://www.coursera.org/learn/object-oriented-python
14.1-14.2	Constructors and idea of inheritance., Superclass And SubClass (implementation with Examples)	https://docs.python.org/3/tutorial/classes.html?highlight=object%20oriented	https://www.coursera.org/learn/object-oriented-python
15.1-15.2	Overriding methods, Object class , Isinstance function, Exploring Operator Overloading- Indexing And Slicing, Iterable Objects (implementation with Examples)	https://docs.python.org/3/tutorial/classes.html?highlight=object%20oriented	https://www.coursera.org/learn/object-oriented-python

	Introduction to GUI Programming (Tkinter), Creating user defined modules, importing existing modules, Creating root windows and calling various dialog boxes on button click response.	https://docs.python.org/3/library/tkinter.html?highlight=tkinter#module-tkinter https://www.coursera.org/projects/build-a-python-gui-with-tkinter https://www.coursera.org/projects/gui-programming-login-system-python-tkinter	https://www.youtube.com/watch?v=YXPYB4XeYLA
	Adding a Widget, Working with widgets(Displaying Text and Images With Label Widgets, Displaying Clickable Buttons With Button Widgets, Getting User Input With Entry Widgets, Getting Multiline User Input With Text Widgets, Assigning Widgets to Frames With Frame Widgets, Naming Conventions)	https://docs.python.org/3/library/tkinter.html?highlight=tkinter#module-tkinter https://www.coursera.org/projects/build-a-python-gui-with-tkinter https://www.coursera.org/projects/gui-programming-login-system-python-tkinter	https://www.youtube.com/watch?v=YXPYB4XeYLA
	Controlling Layout With Geometry Managers(.pack(), .place(), .grid()), Using event and event handlers(.bind())	https://docs.python.org/3/library/tkinter.html?highlight=tkinter#module-tkinter https://www.coursera.org/projects/build-a-python-gui-with-tkinter https://www.coursera.org/projects/gui-programming-login-system-python-tkinter	https://www.youtube.com/watch?v=YXPYB4XeYLA
	Dealing with text-boxes, list-boxes, menus and dealing with basic events.	https://docs.python.org/3/library/tkinter.html?highlight=tkinter#module-tkinter https://www.coursera.org/projects/build-a-python-gui-with-tkinter https://www.coursera.org/projects/gui-programming-login-system-python-tkinter	https://www.youtube.com/watch?v=YXPYB4XeYLA

	Building a simple App using Tkinter	https://docs.python.org/3/library/tkinter.html?highlight=tkinter#module-tkinter https://www.coursera.org/projects/build-a-python-gui-with-tkinter https://www.coursera.org/projects/gui-programming-login-system-python-tkinter	https://www.youtube.com/watch?v=YXPYB4XeYLA
	Connecting Python With Oracle 11G- Performing insertion, Updation and deletion operation using python GUI.	https://docs.python.org/3/library/pickle.html?highlight=database%20connectivity%20python	https://www.youtube.com/watch?v=C9op6I-4WM0

9. Action plan for different types of learners

Slow Learners	Average Learners	Fast Learners
Remedial Classes, Doubt Sessions, Guided Tutorials	Workshop, Doubt Session	Coding Competitions, Project

10. Evaluation Scheme & Components:

Evaluation Component	Type of Component	No. of Assessments	Weightage of Component	Mode of Assessment
Component 1	Continuous Lab Evaluations	02*	30%	Offline
Component 2	Sessional Tests (STs)	02**	20%	Offline
Component 3	End Term Examination	01***	50%	Offline
Total		100%		

* Continuous Lab Evaluations are mandatory evaluations undertaken by viva and File submission, taken twice in a semester, one will be considered as mid term evaluation and another one will be final evaluation.

** Out of 02 STs, the ERP system automatically picks the best 01 ST.

*** As per Academic Guidelines minimum 75% attendance is required to become eligible for appearing in the End Semester Examination.

#NPTEL Online Certification Courses (<https://onlinecourses.nptel.ac.in/>), Dean's Medal of Honor for Certified Students (OPTIONAL).

11. Syllabus of the Course:

Subject: Problem Solving using Python Programming			
S.No.	Topic (s)	No. of Sessions	Weightage %
1	Introduction to Python programming. Python environment setup, identifiers, reserved words ,lines and indentation, multiline statements, comments, quotation, input/output, output formatting(Integers and	12	20%

	string). Fundamentals: Variables(simple, assigning multiple values, output variables, global/local variables), Python decision making- If else and Nested, If else and elif. Match statement, Introduction to while and for loop, Concept of nested looping. For loop and range method, Break, Continue and Pass, Pattern Design,		
2	Defining and calling Functions, Type of arguments (required, keyword, default, variable length), Scope Rules(Global/local Statements), Passing Arguments(call by value/call by reference), Recursive Functions, Lambda Expressions. Lists: Introduction to Lists, Operations on Lists, Storage Structure, Resizing and Looping on Lists, Indexing and Slicing, Passing variables through Functions, passing Lists through Functions, List Comprehension, map, filter & reduce functions, Two Dimensional Lists: Introduction and Storage, Jagged Lists, String-Introduction, methods, comprehension and its relative properties, Tuples, Set and Dictionaries- Introduction, methods, comprehension and its relative properties.	24	40%
3	Modules: Creating Modules, Variables in Modules, Imports and Attributes, Namespaces , Reloading, Built in modules, Generating Random values, Files and Directories, File I/O, File positioning, File operators, Exception Handling: Introduction to Exception, Exception Handling, Except clause, Try, finally clause, User Defined Exceptions.	06	10%
4	Exploring Object-Oriented Programming In Python-Introduction, Creating classes, objects, attributes. Constructors., Idea of inheritance, Superclass And SubClass. Overriding methods, Object class , Isinstance function.	08	13%
5	Introduction to GUI Programming (Tkinter), Creating user defined modules, importing existing modules, Creating root windows and calling various dialog boxes on button click response. Adding a Widget, Working with widgets(Displaying Text and Images With Label Widgets, Displaying Clickable Buttons With Button Widgets, Getting User Input With Entry Widgets, Getting Multiline User Input With Text Widgets, Assigning Widgets to Frames With Frame Widgets, Naming Conventions) Controlling Layout With Geometry Managers(.pack(), .place(), .grid()), Using event and event handlers(.bind())Dealing with text-boxes, list-boxes, menus and dealing with basic events. Building a simple App using Tkinter Connecting Python With Oracle 11G- Performing insertion, Updation and deletion operation using python GUI.	10	17%

This Document is approved by:

Designation	Name	Signature
Course Coordinator	Dr. Renu Popli	
Head-Academic Delivery	Dr. Navjeet Kaur	
Dean	Dr. Monit Kapoor	
Date (DD/MM/YYYY)	28/07/2023	