

Subject Code	Subject Name	Teaching Scheme (Hrs.)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ECL404	Skill Lab: Python Programming	-	04	--	--	02	--	02

Subject Code	Subject Name	Examination Scheme						
		Theory Marks				Term Work	Practical and Oral	Total
		Internal assessment			End Sem. Exam			
		Test 1	Test 2	Avg. of Test 1 and Test 2				
ECL404	Skill Lab: Python Programming	-	-	-	-	25	25	50

**NOTE:** Necessary theory part should be taught by the teacher at the beginning of the laboratory session.

#### Course pre-requisite:

1. ECL304 – Skill Lab: C++ and Java Programming.

#### Course Objectives:

1. Describe the core syntax and semantics of Python programming language.
2. Explore file handling in Python
3. Infer the Object-oriented Programming concepts in Python
4. Formulate GUI Programming and Databases operations in Python
5. Develop applications using variety of libraries and functions

#### Course Outcomes:

After successful completion of the course student will be able to:

1. Describe syntax and semantics in Python
2. Illustrate different file handling operations
3. Interpret object oriented programming in Python
4. Design GUI Applications in Python
5. Express proficiency in the handling Python libraries for data science
6. Develop machine learning applications using Python

Module No.	Unit No.	Topics	Hrs.
<b>1.0</b>		<b>Introduction to Python</b>	<b>6</b>
	1.1	Introduction to Python, Installation and resources, Identifiers and Keywords, Comments, Indentation and Multi-lining, Variables (Local and Global), data types, Arithmetic, Comparative, Logical and Identity Operators, Bitwise Operators, Expressions, Print statement and Formats, Input Statements in python	
	1.2	Strings, Lists, Tuples, Dictionaries, Sets, Accessing Elements, Properties, Operations and methods on these data structures.	
	1.3	Decision Flow Control Statement: if and else statement, Nested If statement, Loop Statement: While Loop, do and while loop, for loop statement, Continue, Break and pass Statement, Conditional Statements	
<b>2.0</b>		<b>Functions and File I/O Handling</b>	<b>8</b>
	2.1	Functions: Built-in-functions, library functions, Defining and calling the functions, Return statements, Passing the arguments, Lambda Functions, Recursive functions, Modules and importing packages in python code.	
	2.2	File Input/Output: Files I/O operations, Read / Write Operations, File Opening Modes, <i>with</i> keywords, Moving within a file, Manipulating files and directories, OS and SYS modules.	
<b>3.0</b>		<b>Object Oriented Programming</b>	<b>9</b>
	3.1	Classes and Objects, Public and Private Members, Class Declaration and Object Creation, Object Initialization, Class Variables and methods, Accessing Object and Class Attributes.	
	3.2	Intricacies of Classes and Objects, Inheritance, Constructor in Inheritance, Exception Handling, Link list, Stack, Queues.	
<b>4.0</b>		<b>Graphical User Interface and Image processing</b>	<b>9</b>
	4.1	Graphical User Interface using Tkinter Library module, creating simple GUI; Buttons, Labels, entry fields, widget attributes.	
	4.2	Database: Sqlite database connection, Create, Append, update, delete records from database using GUI.	
	4.3	Basic Image Processing using OpenCV library, simple image manipulation using image module.	
<b>5.0</b>		<b>Numpy, Pandas, Matplotlib, Seaborn, Scipy</b>	<b>10</b>
	5.1	Introduction to Numpy, Creating and Printing Narray, Class and Attributes of Narray, Basic operation, Copy and view, Mathematical Functions of Numpy.	
	5.2	Introduction to Pandas, Understanding Dataframe, View and Select Data, Missing Values, Data Operations, File read and write operation.	
	5.3	Introduction to Matplotlib library, Line properties, Plots and subplots, Types of Plots, Introduction to Seaborn.	
	5.4	Introduction to Scipy, Scipy Sub packages – Integration and Optimization, Eigen values and Eigen Vectors, Statistic, Weave and IO.	
<b>6.0</b>		<b>Python Applications</b>	<b>10</b>
	6.1	GUI based applications	
	6.2	Applications in Image Processing, Networking	
	6.3	Machine Learning, Linear Regression, Logistic Regression	
	6.4	Classification using K nearest neighbor,	
	6.5	Support Vector Machines	
<b>Total</b>			<b>52</b>

### Text Books:

1. Yashavant Kanetkar, "Let us Python: Python is Future, Embrace it fast", BPB Publications; 1 edition (8 July 2019).
2. Dusty Phillips, "Python 3 object-oriented Programming", Second Edition PACKT Publisher August 2015.
3. John Grayson, "Python and Tkinter Programming", Manning Publications (1 March 1999).
4. Core Python Programming, Dr. R. Nageswara Rao, Dreamtech Press
5. Beginning Python: Using Python 2.6 and Python 3.1. James Payne, Wrox publication
6. Introduction to computing and problem solving using python , E Balagurusamy, McGraw Hill Education.
7. Zed A. Shaw, "Learn Python the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code", Addison Wesley; 3 edition (1 October 2013).

### Reference Books:

1. Eric Matthes, "Python Crash Course A hands-on, Project Based Introduction to programming" No Starch Press; 1 edition (8 December 2015).
2. Paul Barry, "Head First Python" O'Reilly; 2 edition (16 December 2016)
3. Andreas C. Mueller, "Introduction to Machine Learning with Python", O'Reilly; 1 edition (7 October 2016)
4. David Beazley, Brian K. Jones, "Python Cookbook: Recipes for Mastering Python 3", O'Reilly Media; 3 edition (10 May 2013).
5. Bhaskar Chaudhary, "Tkinter GUI Application Development Blueprints: Master GUI programming in Tkinter as you design, implement, and deliver 10 real world application", Packt Publishing (November 30, 2015)

### Software Tools:

1. Python IDE: <https://www.python.org/downloads/>
2. Anaconda Environment: <https://www.anaconda.com/distribution/>

### Online Repository:

1. Github
2. Python 3 Documentation: <https://docs.python.org/3/>
3. "The Python Tutorial", <http://docs.python.org/release/3.0.1/tutorial/>
4. <http://spoken-tutorial.org>
5. Python 3 Tkinter library Documentation: <https://docs.python.org/3/library/tk.html>
6. Numpy Documentation: <https://numpy.org/doc/>
7. Pandas Documentation: <https://pandas.pydata.org/docs/>
8. Matplotlib Documentation: <https://matplotlib.org/3.2.1/contents.html>
9. Scipy Documentation : <https://www.scipy.org/docs.html>
10. Machine Learning Algorithm Documentation: <https://scikit-learn.org/stable/>
11. <https://nptel.ac.in/courses/106/106/106106182/>

**The following list of experiments and course project is for illustration purpose. Faculty members are required to introduce their own innovative list of experiments based on above curriculum.**

Sr. No.	Problem Statement	Module No.
1.	1. Write python programs to understand expressions, variables, quotes, basic math operations, list, tuples, dictionaries, arrays etc. 2. Write Python program to implement byte array, range, set and different STRING Functions (len, count, lower, sorted etc) 3. Write Python program to implement control structures.	Module 1

	<p>4. Assume a suitable value for distance between two cities (in km). Write a program to convert and print this distance in meters, feet, inches and centimetre.</p> <p>5. Write a program to carry out the following operations on the given set</p> <p style="padding-left: 40px;"><math>s = \{10, 2, -3, 4, 5, 88\}</math></p> <ol style="list-style-type: none"> <li>Number of items in sets s</li> <li>Maximum element in sets s</li> <li>Minimum element in sets s</li> <li>Sum of all elements in sets s</li> <li>Obtain a new sorted set from s, set s remaining unchanged</li> <li>Report whether 100 is an element of sets s</li> <li>Report whether -3 is not an element of sets s.</li> </ol>	
2.	<p>1. Write python program to understand different File handling operations</p> <p>2. Create 3 lists – a list of names, a list of ages and a list of salaries. Generate and print a list of tuples containing name, age and salary from the 3lists. From this list generate 3 tuples – one containing all names, another containing all ages and third containing all salaries.</p>	Module 2
3.	<p>1. Write Python program to implement classes, object, Static method and inner class</p> <p>2. If any integer is given as in input through the keyboard, write a program to find whether it is odd or even number.</p> <p>3. If ages of Ram, Shyam, and Ajay are given as an input through the keyboard, write a program to determine the youngest of the three.</p> <p>4. Write a program that prints square root and cube root of numbers from 1 to 10, up to 4 decimal places. Ensure that the output is displayed in separate lines, with number center-justified and square and cube roots right-justified.</p> <p>5. Write a program to find the factorial value of any number entered through the keyboard.</p> <p>6. Write a program that defines a function count_lower_upper( ) that accepts a string and calculates the number of uppercase and lowercase alphabets in it. It should return these values as a dictionary. Call this function for some sample strings.</p> <p>7. A 5-digit positive integer is entered through the keyboard, write a recursive function to calculate sum of digits of 5-digit number.</p>	Module 3
4.	<p>1. Write Python program to create, append, update, delete records from database using GUI.</p> <p>2. Write Python program to obtain histogram of any image</p> <p>3. Write Python Program to split color image in R,G,B and obtain individual histograms.</p> <p>4. Write Python program for histogram equalization</p> <p>5. Write Python Program for edge detection</p> <p>6. Write Python Program for image segmentation</p> <p>7. Write Python program to implement GUI Canvas application using Tkinter</p> <p>8. Write Python program to implement GUI Frame application using Tkinter</p>	Module 4
5.	<p>1. Write Python program to study define, edit arrays and perform arithmetic operations.</p> <p>2. Write python program to study selection, indexing, merging, joining, concatenation in data frames</p> <p>3. Evaluate the dataset containing the GDPs of different countries to:</p> <ol style="list-style-type: none"> <li>Find and print the name of the country with the highest GDP</li> <li>Find and print the name of the country with the lowest GDP</li> <li>Print text and input values iteratively</li> </ol>	Module 5

	<p>d. Print the entire list of the countries with their GDPs</p> <p>e. Print the highest GDP value, lowest GDP value, mean GDP value, standardized GDP value, and the sum of all the GDPs</p> <p>4. Analyze the Federal Aviation Authority (FAA) dataset using Pandas to do the following:</p> <ol style="list-style-type: none"> <li>a. View: aircraft make name, state name, aircraft model name, text information, flight phase, event description type,</li> <li>b. fatal flag</li> <li>c. b. Clean the dataset and replace the fatal flag NaN with "No".</li> <li>d. c. Find the aircraft types and their occurrences in the dataset</li> <li>e. d. Remove all the observations where aircraft names are not available</li> <li>f. Display the observations where fatal flag is "Yes"</li> </ol> <p>5. Analyze the "auto mpg data" and draw a pair plot using seaborn library for mpg, weight, and origin.</p> <p>(a) Origin: This dataset was taken from the StatLib library maintained at Carnegie Mellon University.</p> <ul style="list-style-type: none"> <li>• Number of Instances: 398</li> <li>• Number of Attributes: 9 including the class attribute</li> <li>• Attribute Information:</li> <li>• mpg: continuous</li> <li>• cylinders: multi-valued discrete</li> <li>• displacement: continuous</li> <li>• horsepower: continuous</li> <li>• weight: continuous</li> <li>• acceleration: continuous</li> <li>• model year: multi-valued discrete</li> <li>• origin: multi-valued discrete</li> <li>• car name: string (unique for each instance)</li> </ul> <p>5. Write python program to use SciPy to solve a linear algebra problem.</p> <p>6. There is a test with 30 questions worth 150 marks. The test has two types of questions: 1. True or false – carries 4 marks each</p> <p>2. Multiple-choice – carries 9 marks each. Find the number of true or false and multiple-choice questions.</p>	
6.	<p>1. Write python program to study linear regression</p> <p>2. Write python program to study multiple linear regression</p> <p>3. Write python program to study logistic regression</p> <p>4. Write python program to study Support Vector Machine</p> <p>5. Write python program to study decision tree algorithm</p> <p>6. Write python program to study two-way communication between client and server.</p> <p>7. Write Python Program to study image morphological operations.</p>	Module 6

Suggested list of course projects:

- Speed typing Test using Python
- Music player in Python
- Calculator app using tkinter
- Train announcement system using python
- Dice rolling simulator
- Expense tracker
- Contact book using python
- Develop classification model using freely available datasets
- Develop python application for sentiment analysis

**Note:**

1. Use of free cloud service such as Google Colab to run python scripts is encouraged.
2. Necessary theory part should be taught by the teacher at the beginning of the laboratory session.

**Term Work (25-Marks):**

At least **12 experiments and 01 course project** should be performed. Term work assessment must be based on the overall performance of the student with every experiment and project graded from time-to-time. The grades will be converted to marks as per “**Credit and Grading System**” manual and should be added and averaged. Based on above scheme grading and term work assessment should be done.

**The practical and oral examination will be based on entire syllabus.**