GUJARAT TECHNOLOGICAL UNIVERSITY
MASTERS IN COMPUTER APPLICATION
Year – 2 (Semester – III) (W.E.F. JULY 2018)

Subject Name: Programming in Python
Subject Code: 4639304

1. **Learning Objectives:**
   - To develop proficiency in creating based applications using the Python Programming Language.
   - To be able to understand the various data structures available in Python programming language and apply them in solving computational problems.
   - To be able to draw various kinds of plots using PyLab and Pandas
   - To be able to understand the creation DB API in Python
   - To be able to understand the applications of advanced concepts like networking, multithreading and data science in python

2. **Prerequisites**: Knowledge of some programming Concepts

3. **Contents:**

Unit No.	Course Content	Weightage Percentage
1	**Introduction to Python:** The basic elements of Python, Objects, expressions and numerical Types, Variables and assignments, IDLE, Branching programs, Strings and Input, Iteration **Structured Types, Mutability and Higher-order Functions:** Tuples, Lists and Mutability, Functions as Objects, Strings, Tuples and Lists, Dictionaries	10%
2	**Functions, Exception, Modules and Files** **Functions:** Difference between a Function and a Method, Defining a Function, Calling a Function, Returning Results from a Function, Returning Multiple Values from a Function, Functions are First Class Objects, Pass by Object Reference, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Variable Length Arguments, Local and Global Variables, The Global Keyword, Passing a Group of Elements to a Function, Recursive Functions, Anonymous Functions or Lambdas (Using Lambdas with filter() Function, Using Lambdas with map() Function, Using Lambdas with reduce() Function), Function Decorators, Generators, Structured Programming, Creating our Own Modules in Python, The Special Variable __name__ **Exceptions:** Errors in a Python Program (Compile-Time Errors, Runtime Errors, Logical Errors),Exceptions, Exception Handling, Types of Exceptions, The Except Block, The assert Statement, User-Defined Exceptions, Logging the Exceptions	20%

	Files: Files, Types of Files in Python, Opening a File, Closing a File, Working with Text Files Containing Strings, Knowing Whether a File Exists or Not, Working with Binary Files, The with Statement, Pickle in Python, The seek() and tell() Methods, Random Accessing of Binary Files, Random Accessing of Binary Files using mmap, Zipping and Unzipping Files, Working with Directories, Running Other Programs from Python Program	
3	**Classes and Object-oriented Programming:** **Classes:** Creating a Class, The Self Variable, Constructor, Types of Variables, Namespaces, Types of Methods (Instance Methods, Class Methods, Static Methods), Passing Members of One Class to Another Class, Inner Classes **Inheritance and Polymorphism:** Constructors in Inheritance, Overriding Super Class Constructors and Methods, The super() Method, Types of Inheritance, Single Inheritance, Multiple Inheritance, Method Resolution Order (MRO), Polymorphism, Duck Typing Philosophy of Python, Operator Overloading, Method Overloading, Method Overriding **Abstract Classes and Interfaces**: Abstract Method and Abstract Class, Interfaces in Python, Abstract Classes vs. Interfaces,	15%
4	**Advanced Topics I: Plotting and Data Science** Plotting using PyLab, Plotting mortgages and extended examples **Data Science Using Python:** Data Frame (Creating Data Frame from an Excel Spreadsheet, Creating Data Frame from .csv Files, Creating Data Frame from a Python Dictionary, Creating Data from Python List of Tuples, Operations on Data Frames), **Data Visualization** : Bar Graph, Histogram, Creating a Pie Chart, Creating Line Graph	15%
5	**Advanced Topics II: Regular Expressions** **REs and Python:** Regular Expressions, Sequence Characters in Regular Expressions, Quantifiers in Regular Expressions, Special Characters in Regular Expressions, Using Regular Expressions on Files, Retrieving Information from a HTML File **Threading :** Concurrent Programming and GIL, Uses of Threads, Creating Threads in Python, Thread Class Methods, Single Tasking using a Thread, Multitasking using Multiple Threads, Thread Synchronization Deadlock of Threads, Avoiding Deadlocks in a Program, Communication between Threads, Thread Communication using notify() and wait() Methods, Thread Communication using a Queue, Daemon Threads **Networking:** Protocol, Sockets, Knowing IP Address, URL, Reading the Source Code of a Web Page, Downloading a Web Page from Internet, Downloading an Image from Internet, A TCP/IP Server, A TCP/IP Client, A UDP Server, A UDP Client, File Server, File Client, Two-Way Communication between Server and Client, Sending a Simple Mail,	25%

6	**Python's Database Connectivity** Verifying the MySQLdb Interface Installation, Working with MySQL Database, Using MySQL from Python, Retrieving All Rows from a Table, Inserting Rows into a Table, Deleting Rows from a Table, Updating Rows in a Table, Creating Database Tables through Python	15%

4. Text Book:

1) John V Guttag. "Introduction to Computation and Programming Using Python", Prentice Hall of India
2) R Nageswara Rao, Core Python Programming, 2nd Edition, Dreamtech Press

5. Reference Books:

1) Wesley J Chun, Core Python Applications Programming, 3rd Edition.Pearson
2) Luke Sneeringer, Professional Python, WROX
3) Robert Sedgewick, Kevin Wayne, Robert Dondero, Introduction to Programming in Python, Pearson
4) Doug Hellmann, The python 3 standard Library by example, Pearson Education
5) Alex Martelli, Python Cookbook, O'REILLY
6) Laura Cassell, Python Projects, WROX
7) Daniel Y Chen, Pandas for Everyone: Python Data Analysis , 1st Edition, Pearson Eduction

Web References:

1) Charles Severance, Python for informatics: www.pythonlearn.com
2) Swaroop C H. "A Byte of Python", http://www.swaroopch.com/notes/python
3) "Python Programming", http://en.wikibooks.org/wiki/Python_Programming
4) "The Python Tutorial", http://docs.python.org/release/3.0.1/tutorial/
5) "Learn Python the Hard way", http://learnpythonthehardway.org/
6) Dive Into Python 3: http://www.diveintopython.net/

6. Chapter wise Coverage from Text Book:

Unit #	Book#	Chapter
1	1	2,5
2	2	9,16,17
3	2	13,14,15
4	1	11
	2	25
5	2	Chapter: 18, Chapter: 21(Page 542 to 563), Chapter: 23
6	2	Chapter 24 (Page 663 to 681)

7. Accomplishments of the student after completing the course:

- Ability to create roboust applications using the Python programming language
- Ability to test and debug applications written using the Python programming language
- Ability to create applications for solving computational problems using the Python Programming Language.