GUJARAT UNIVERSITY

BCA V SYLLABUS

COURSE TITLE	**CC305 Python Programming Practicals**
COURSE CODE	CC-305
COURSE CREDIT	3
Sessions Per Week	3
Total Teaching Hours	40 Hours

AIM

To train the students from the basics of coding and executing Python scripts to the more advanced features of using libraries, handling errors and connecting to databases.

LEARNING OUTCOMES

On the completion of the course students will:

1. To learn how to design and implement efficient programming using python.
2. To learn working with the new datatypes in python.
3. To understand and use object based software concepts.
4. To work with the built in libraries and also prepare your own customised libraries.
5. Learning the importance of using different versions of python in a single system.
6. To connect python applications with database.

Note

The list in each unit is indicative only and **may or may not be asked in the examination**. The programs given below are only sample example for practice in lab.

DETAIL SYLLABUS

UNIT	**TOPIC/SUB TOPIC**		**TEACHING HOURS**
1	**Beginning with Python, Datatypes, Operators, I/O and Control statements**		**10**
	1.	Write a program to swap two numbers without taking a temporary variable.	
	2.	Write a program to display sum of two complex numbers.	
	3.	Write a program to create a byte type array, read, modify, and display the elements of the array	
	4.	Create a sequence of numbers using range datatype to display 1 to 30, with an increment of 2.	
	5.	Write a program to find out and display the common and the non common elements in the list using membership operators	
	6.	Create a program to display memory locations of two variables using id() function, and then use identity operators two compare whether two objects are same or not.	
	7.	Write a program that evaluates an expression given by the user at run time using eval() function. Example:	

		*Enter and expression: 10+8-9*2-(10*2)* *Result: -20*	
	8.	Write a python program to find the sum of even numbers using command line arguments.	
	9.	Write a menu driven python program which perform the following: Find area of circle Find area of triangle Find area of square and rectangle Find Simple Interest Exit.(Hint: Use infinite while loop for Menu)	
	10	Write a program to assert the user enters a number greater than zero.	
	11.	Write a program to search an element in the list using for loop and also demonstrate the use of "*else*" with for loop.	
	12.	Write a python program that asks the user to enter a length in centimeters. If the user enters a negative length, the program should tell the user that the entry is invalid. Otherwise, the program should convert the length to inches and print out the result. (2.54 = 1 inch).	
2	**Modules, Arrays, Functions, List, Tuples and Dictionaries**		**10**
	1.	Write a program to create one array from another array.	
	2.	Create a program to retrieve, display and update only a range of elements from an array using indexing and slicing in arrays.	
	3.	Write a program to understand various methods of array class mentioned: append, insert, remove, pop, index, tolist and count.	
	4.	Write a program to sort the array elements using bubble sort technique.	
	5.	Create a program to search the position of an element in an array using index() method of array class.	
	6.	Write a program to generate prime numbers with the help of a function to test prime or not.	
	7.	Write a python program that removes any repeated items from a list so that each item appears at most once. For instance, the list [1,1,2,3,4,3,0,0] would become [1,2,3,4,0].	
	8.	Write a program to pass a list to a function and display it.	
	9.	Write a program to demonstrate the use of Positional argument, keyword argument and default arguments.	
	10.	Write a program to show variable length argument and its use.	
	11.	Write a lambda/Anonymous function to find bigger number in two given numbers.	
	12.	Create a decorator function to increase the value of a function by 3.	
	13.	Create a program name "employee.py" and implement the functions DA, HRA, PF, and ITAX. Create another program that uses the function of employee module and calculates gross and net salaries of an employee.	

	14.	Write a program to create a list using range functions and perform append, update and delete elements operations in it.	
	15.	Write a program to combine two List, perform repetition of lists and create cloning of lists.	
	16.	Create a sample list of 7 elements and implement the *List methods* mentioned: append, insert, copy, extend, count, remove, pop, sort, reverse and clear.	
	17.	Write a program to create nested list and display its elements.	
	18.	Write a program to accept elements in the form of a tuple and display its minimum, maximum, sum and average.	
	19.	Create a program to sort tuple with nested tuples.	
	20.	Write a program to create a dictionary from the user and display the elements.	
	21.	Create a dictionary that will accept cricket players name and scores in a match. Also we are retrieving runs by entering the player's name.	
	22.	Write a program to convert the elements of two lists into key-value pairs of a dictionary.	
	23.	Create a python function to accept python function as a dictionary and display its elements.	
3	**Classes, Inheritance and Polymorphism**		**10**
	1.	Write a program to create a Student class with name, age and marks as data members. Also create a method named display() to view the student details. Create an object to Student class and call the method using the object.	
	2.	Write a program to create Student class with a constructor having more than one parameter.	
	3.	Write a program to demonstrate the use of instance and class/static variables.	
	4.	Write a program to store data into instances using mutator methods and to retrieve data from the instances using accessor methods.	
	5.	Write a program to use class method to handle the common features of all the instance of Student class.	
	6.	Write a program to create a static method that counts the number of instances created for a class.	
	7.	Create a Bank class with two variables name and balance. Implement a constructor to initialize the variables. Also implement deposit and withdrawals using instance methods.	
	8.	Write a program to create a Emp class and make all the members of the Emp class available to another class (Myclass). *[By passing members of one class to another]*	
	9.	Create a Student class to with the methods set_id, get_id, set_name, get_name, set_marks and get_marks where the method name starting with set are used to assign the values and method name starting with get are returning the values. Save the program by *student.py*. Create another program to use the Student class which is already available in *student.py*.	

	10.	Write a program to access the base class constructor from a sub class by using *super()* method and also without using *super()* method.	
	11.	Write a program to override super class constructor and method in sub class.	
	12.	Write a program to implement single inheritance in which two sub classes are derived from a single base class.	
	13.	Write a program to implement multiple inheritance using two base classes.	
	14.	Write a program to understand the order of execution of methods in several base classes according to method resolution order (MRO).	
	15.	Write a program to check the object type to know whether the method exists in the object or not.	
	16	Write a program to overload the addition operator (+) to make it act on the class objects.	
	17.	Write a program to show method overloading to find sum of two or three numbers.	
	18.	Write a program to override the super class method in subclass.	
4	**Exception Handling, Standard Library, Creating Virtual Environment and Python Database connectivity**		**10**
	1.	Write a program to handle some built in exceptions like *ZeroDivisionError*.	
	2.	Write a program to handle multiple exceptions like *SyntaxError* and *TypeError*	
	3.	Write a program to import "*os*" module and to print the current working directory and returns a list of all module functions	
	4.	Write a program to provide a function for making file lists from directory wildcard searches.	
	5.	Write a program to import *datetime* module and format the date as required. Also use the same module to calculate the difference between your birthday and today in days.	
	6.	Write a program to create a database named "Sample_DB" in MySQL(). [First ensure connection is made or not and then check if the database *Sample_DB* already exists or not, if yes then print appropriate message]	
	7.	Write a program to retrieve and display all the rows in the employee table. [First create an *employee* table in the Sample_DB with the fields as eid, name, sal . Also enter some valid records]	
	8.	Write a program to insert several rows into *employee* table from the keyboard.	
	9.	Write a program to delete a row from an *employee* table by accepting the employee identity number (eid) from the user.	
	10.	Write a program to increase the salary (sal) of an employee in the *employee* table by accepting the employee identity number (eid) from the user.	

	11.	Write a program to create a table named *new_employee_tbl* with the fields eno , ename , gender and salary in Sample_DB database. The datatypes of the fields are eno-int, ename-char(30), gender-char(1) and salary-float.	

TEXT BOOK:

1) Core Python Programming
By, Dr. R. Nageswara Rao, 2017 edition
2) Python Tutorial (Release 3.6.4)
By, Guido van Rossum and the Python development team

REFERENCE BOOK:

1) A Byte of Python,
By Swaroop C H
2) Python Cookbook, Recipes of Mastering Python 3,
By David Beazely & Brian K. Jones

WEB RESOURCES:

https://www.python.org/about/apps/
https://www.w3schools.com/python/default.asp
https://www.tutorialspoint.com/python3/index.htm
https://www.programiz.com/python-programming/tutorial

REQUIRED SOFTWARES:

Python 3.4.1 or higher
IDE: IDLE
Database: MySQL