SRM VALLIAMMAI ENGINEERING COLLEGE

(An Autonomous Institution)

SRM Nagar, Kattankulathur – 603 203

DEPARTMENT OF COMPUTER APPLICATIONS

QUESTION BANK

Academic Year 2025-2026

I SEMESTER

MC4163 - PYTHON PROGRAMMING

(Regulation – 2024)

(AY: 2025 – 26 ODD SEMESTER)

Prepared by

Mr.K.MANIRAJ MSc., M.Tech.,(Ph.D)
Assistant Professor

SRM VALLIAMMAI ENGINEERING COLLEGE

(An Autonomous Institution)

SRM Nagar, Kattankulathur

DEPARTMENT OF COMPUTER APPLICATIONS

QUESTION BANK (AY 25-26)

SUBJECT: MC4163 - PYTHON PROGRAMMING **YEAR / SEMESTER : I / I**

UNIT I

ALGORITHMIC

Introduction to Python Programming – Python Interpreter and Interactive Mode– Variables and Identifiers – Arithmetic Operators – Values and Types – Statements. Operators – Boolean Values Operator Precedence–Expression – Conditionals: If-Else Constructs –Loop Structures /Iterative Statements – While Loop – For Loop – Break Statement-Continue statement Function Call and Returning Values – Parameter Passing – Local and Global Scope – Recursive Functions

Part A

Q. No.	Questions	Marks	Competence	BT Level	CO’s
1	List out the features of Python.	2	Remember	BTL1	CO1
2	Write down the rules for naming a variable.	2	Understand	BTL2	CO1
3	Define recursion with example.	2	Remember	BTL1	CO1
4	Discuss different modes of operation in Python.	2	Understand	BTL2	CO1
5	Infer how does Python interpreter work?	2	Understand	BTL2	CO1
6	Write the standard data types in Python.	2	Remember	BTL1	CO1
7	State break and continue statements with a program example.	2	Remember	BTL1	CO1
8	Identify local and global variables.	2	Understand	BTL2	CO1
9	List the order of precedence of operators in Python.	2	Remember	BTL1	CO1
10	State how Comment is used in Python.	2	Understand	BTL1	CO1
11	Define function and state its use.	2	Remember	BTL1	CO1
12	Write a simple Python program using while loop	2	Understand	BTL2	CO1
13	Write the syntax for function definition.	2	Remember	BTL1	CO1
14	List the various control flow structures	2	Remember	BTL1	CO1
15	Describe a Boolean expression with an example.	2	Understand	BTL2	CO1

16	Write a function without argument and with return type.	2	Understand	BTL2	CO1
17	State the reasons to divide program in to functions.	2	Remember	BTL1	CO1
18	Discuss the membership and special operators in Python.	2	Understand	BTL2	CO1
19	What is lamda? Give an example.	2	Understand	BTL2	CO1
20	What is IDLE? Mention its features.	2	Understand	BTL2	CO1
21	Classify expressions by applying different operators.	2	Understand	BTL2	CO1
22	Describe the different types of parameters.	2	Understand	BTL2	CO1
23	Classify different types of statements in Python.	2	Understand	BTL2	CO1
24	List the types of operators available in Python.	2	Remember	BTL1	CO1
Part B					
1	Using a simple Python snippet, analyze different values, types and expression and explain them.	16	Analyze	BTL4	CO1
2	Explain the following				
	(i) Write a Python program to find the sum of N natural numbers.	8	Evaluate	BTL5	CO1
	(ii) What is the use of pass, break and continue statements? Illustrate with an example.	8	Evaluate	BTL5	CO1
3	List the different types of functions with suitable examples.	16	Analyze	BTL4	CO1
4	List the different types of operators in Python and explain them with suitable example.	16	Analyze	BTL4	CO1
5	What is the use of function? Explain the role of function definition and call for generating Fibonacci series.	16	Apply	BTL4	CO1
6	Explain the following				
	(i).Write a Python program to find the greatest among three numbers.	8	Apply	BTL4	CO1
	(ii). Develop a Python code to generate a series in between which are all divisible by an input number.	8	Create	BTL6	CO1
7	(i) Explain the various types of conditional control statements with example.	8	Analyze	BTL3	CO1
	(ii) Design a simple calculator with Python code by defining its different notations.	8	Create	BTL6	CO1
8	Rate the order of execution of different expressions by evaluating them through Python program.	16	Evaluate	BTL5	CO1

9	Write a simple program with and without recursion to calculate factorial of a number.	16	Apply	BTL3	CO1
10	Develop a Python code using Python for the following:				
	(i) Print multiplication table of a given number.	8	Apply	BTL3	CO1
	(ii) Print all prime numbers within a range.	8	Apply	BTL3	CO1
11	(i) Write a Python program to check a series of years in between which are all a leap year or not.	8	Create	BTL6	CO1
	(ii) Write a Python program to find the square root of a number.	8	Apply	BTL3	CO1
12	(i) Write a Python program to find GCD of two numbers.	8	Apply	BTL3	CO1
	(ii) Give the Python code to find the minimum among the list of numbers without using sequential data types.	8	Create	BTL6	CO1
13	(i) Identify the concept of local and global variables.	8	Apply	BTL3	CO1
	(ii) Write a Python program to delete a character from the given data which is duplicated.	8	Apply	BTL3	CO1
14	(i) Display the number of times a character repeated in the given string.	8	Apply	BTL3	CO1
	(ii) Calculate the multiplication of the given two matrices.	8	Apply	BTL3	CO1
15	Appraise with an example various looping statements in Python.	16	Analyze	BTL4	CO1
16	Explain the following with an example code				
	(i) To solve a quadratic equation.	8	Analyze	BTL4	CO1
	(ii) To search an element using binary search.	8	Analyze	BTL4	CO1
17	(i) Create two function with loop and else statement.	8	Create	BTL6	CO1
	(ii) Write a Python program to Create a function that returns multiple arguments.	8	Create	BTL6	CO1

UNIT II

DATA TYPES IN PYTHON

Lists, Tuples, Sets, Strings, Dictionary, Modules: Module Loading and Execution – Packages – Making Your Own Module – The Python Standard Libraries

Part A

Q No.	Questions	Marks	Competence	BT Level	CO's
1	Express a program to add two lists.	2	Understand	BTL2	CO2
2	Show the difference between string and list data types.	2	Understand	BTL2	CO2
3	List the basic methods that are used in Python lists.	2	Remember	BTL1	CO2
4	Express negative indexing in list with an example.	2	Understand	BTL2	CO2
5	How will you remove all duplicates from a list?	2	Remember	BTL1	CO2
6	Enumerate the occurrences of an element in a list.	2	Remember	BTL1	CO2
7	Express to create multiline strings in Python? Give example.	2	Understand	BTL2	CO2
8	Outline how Tuples are used as return values.	2	Understand	BTL2	CO2
9	List the usage of tuples as arguments to a function.	2	Remember	BTL1	CO2
10	Discuss the difference between lists and tuples.	2	Understand	BTL2	CO2
11	Write a program to create a list of even numbers in a given range.	2	Understand	BTL2	CO2
12	Express a Python program to check if a string is a palindrome.	2	Understand	BTL2	CO2
13	Outline the difference between mutable and immutable data types.	2	Understand	BTL2	CO2
14	Express a Python program to find the index of the first occurrence of a substring in a string	2	Understand	BTL2	CO2
15	Demonstrate how to create a dictionary in python.	2	Remember	BTL1	CO2
16	Demonstrate a dictionary with key-value pairs for a person's name and age.	2	Understand	BTL2	CO2
17	State and Check if a key exists in the person's dictionary	2	Understand	BTL2	CO2
18	List out the string methods in python.	2	Remember	BTL1	CO2
19	Define set with example	2	Remember	BTL1	CO2
20	Express different set functions	2	Understand	BTL2	CO2
21	What are modules?	2	Remember	BTL1	CO2
22	What is a package?	2	Remember	BTL1	CO2

23	What is the special file that each package in Python must contain?	2	Understand	BTL2	CO2
24	Express a program to perform union, intersection and difference operation using set with symbols.	2	Understand	BTL2	CO2
Part B					
1	Describe the following				
	(i) Creating the list	4	Apply	BTL3	CO2
	(ii) Updating the list	4	Apply	BTL3	CO2
	(iii) Deleting the list elements	4	Apply	BTL3	CO2
	(iv) Access the elements	4	Apply	BTL3	CO2
2	(i) Write a Python program that interchanges the first and last characters of a given string.	8	Evaluate	BTL5	CO2
	(ii) How to create, index and split the string? Illustrate with example.	8	Evaluate	BTL5	CO2
3	(i) How to update a String? Explain with example code.	8	Apply	BTL3	CO2
	(ii) Develop a python code to delete all the duplicate elements in a list.	8	Create	BTL6	CO2
4	Develop a python code that takes a sentence as input from the user and computes the frequency of each letter. Use a Variable of dictionary type to maintain the count.	16	Create	BTL6	CO2
5	Explain the following with example				
	(i) Remove an element from a tuple.	8	Analyze	BTL4	CO2
	(ii). Find the common elements between two tuples.	8	Analyze	BTL4	CO2
6	(i). Write the code to illustrate the difference between discard () and remove () in sets.	8	Apply	BTL3	CO2
	(ii) Write a Python program to find common elements in three lists using sets.	8	Apply	BTL3	CO2
7	Write a function that takes a number as an input parameter and returns the corresponding text in words, for example, on input 452, the function should return ‘Four Five Two’. Use a dictionary for mapping digits to their string representation.	16	Create	BTL6	CO2
8	(i). Explain the basic Tuple operations with examples.	8	Apply	BTL3	CO2
	(ii). Illustrate a program to check whether an element ‘y’ and ‘a’ belongs to the tuple mytuple=(‘p’,’y’,’t’,’h’,’o’,n’) and after printing the result, delete the Tuple.	8	Apply	BTL3	CO2
9	Write a Python program to calculate the average value of the numbers in a given tuple of tuples. Original Tuple: ((1, 1, -5), (30, -15, 56), (81, -60, -39), (-10, 2, 3)) Average value of the numbers of the said tuple of tuples: [25.5, -18.0, 3.75].	16	Create	BTL6	CO2

10	Give a comparison between lists, tuples, dictionaries, and sets.	16	Evaluate	BTL5	CO2
11	Explain the operations of sets with a neat code by using symbols and functions.	16	Apply	BTL3	CO2
12	Assess the built-in functions of a Tuple.	16	Evaluate	BTL5	CO2
13	Explain the following:				
	(i). Predefined Modules	8	Analyze	BTL4	CO2
	(ii) User defined Modules.	8	Analyze	BTL4	CO2
14	Describe the methods and operations of Dictionaries.	16	Analyze	BTL4	CO2
15	Write a Python program to display a calendar of a month by importing a calendar package.	16	Create	BTL6	CO2
16	How to execute the module with the functions run_module() and run_path() with example code.	16	Apply	BTL3	CO2
17	Explain with example code to create and import user-defined module in detail.	16	Analyze	BTL4	CO2

UNIT III

FILE HANDLING AND EXCEPTION HANDLING

Files: Introduction – File Path – Opening and Closing Files – Reading and Writing Files –File Position –Exception: Errors and Exceptions, Exception Handling, Multiple Exception.

Part A

Q. No.	Questions	Marks	Competence	BT Level	CO's
1	List the different modes of file opening.	2	Remember	BTL1	CO3
2	What do you mean by file object in Python?	2	Remember	BTL1	CO3
3	Distinguish between files and modules.	2	Understand	BTL2	CO3
4	What is the default mode of opening a file?	2	Understand	BTL2	CO3
5	Describe renaming and deleting a file in python.	2	Remember	BTL1	CO3
6	Discover the format operator available in files.	2	Remember	BTL1	CO3
7	Develop a python code in python to read first 5 characters from a file.	2	Understand	BTL2	CO3
8	Write the symbols used in binary file mode for the different operations.	2	Understand	BTL2	CO3
9	Accept five names from the user and write in a file "name.txt".	2	Understand	BTL2	CO3
10	What do you mean by fileisatty() method?	2	Understand	BTL2	CO3

11	Write the need for exceptions using an example	2	Understand	BTL2	CO3
12	Compare text file and binary file.	2	Understand	BTL2	CO3
13	Write the Difference between built-in exceptions and user defined exception.	2	Remember	BTL1	CO3
14	Write with example code to fix value error exceptions in Python?	2	Understand	BTL2	CO3
15	Write a Python program that executes an operation on a list and to handle an IndexError exception	2	Understand	BTL2	CO3
16	What is an error? List the types of errors.	2	Remember	BTL1	CO3
17	What is meant by Assertions in Python?	2	Understand	BTL2	CO3
18	Compare exceptions vs syntax error.	2	Understand	BTL2	CO3
19	What is the use of finally?	2	Remember	BTL1	CO3
20	Write the purpose to raise KeyError in exception handling?	2	Understand	BTL2	CO3
21	List common types of Exceptions in python.	2	Remember	BTL1	CO3
22	How to Raise an Exception?	2	Understand	BTL2	CO3
23	Discover except clause with multiple exception.	2	Remember	BTL1	CO3
24	Define custom exception in python.	2	Understand	BTL2	CO3
Part B					
1	Write a Python program to demonstrate the file I/O operations.	16	Create	BTL6	CO3
2	(i) Discover a function in python to count the number of lines in a text file.	8	Apply	BTL3	CO3
	(ii) Write a function to read lines from the text file and display those word which are less than 4 characters.	8	Apply	BTL3	CO3
3	(i) Write a function to read contents from a file and to count and display the occurrence of your two input words.	8	Analyze	BTL4	CO3
	(ii) Discover a function to count the number of lines in a file which begins from upper case character.	8	Analyze	BTL4	CO3
4	Describe the following in detail				
	(i)Structure Renaming a file.	8	Understand	BTL2	CO3
	(ii)Explain about the files related methods.	8	Understand	BTL2	CO3
5	Explain with an example to copy the contents of one file to another.	16	Apply	BTL3	CO3

6	Write a function that reads a file sample.txt and displays number of words and vowels in the file.	16	Analyze	BTL4	CO3
7	Develop a python code that reads the contents of the file text.txt and counts the number of alphabets, blank spaces, lowercase letters and uppercase letters, the number of words starting with a vowel, and the number of occurrences of the word 'is' in the file.	16	Create	BTL6	CO3
8	(i). Discover a program to catch a divide by zero exception. Add a finally block too.	8	Apply	BTL3	CO3
	(ii). Write a function to print the hash of any given file in Python.	8	Apply	BTL3	CO3
9	Write a Python program that takes a list of numbers as input from the user and finds the maximum and minimum numbers in the list. Handle the exceptions that may occur during the program execution, such as invalid input or empty list error.	16	Evaluate	BTL5	CO3
10	Create a program to compute price per unit weight of an item using try – except – else block.	16	Create	BTL6	CO3
11	Examine the following function percentage:				

	def percentage(marks,total):
	try:
	percent=(marks/total)*100
	except ValueError: print('Value Error')
	except TypeError: print('TypeError')
	except ZeroDivisionError: print('ZeroDivisionError')
	except:
	print('any other error') else:
	print(percent) finally:
	print('Function percentage completed")
	Determine the output for the function calls:
		16	Analyze	BTL4	CO3

12	Illustrate a program to find the one's complement of binary number using file.	16	Evaluate	BTL5	CO3
13	(i) List the different types of exceptions in python.	8	Analyze	BTL4	CO3
	(ii) Write Advantages and disadvantages of exceptions in python.	8	Evaluate	BTL5	CO3
14	Analyze in detail how to handle Multiple Exceptions in Python.	16	Analyze	BTL4	CO3
15	(i) How to handle EOFError Exception in Python.	8	Apply	BTL3	CO3
	(ii) How to Fix "EOFError: EOF when reading a line" in Python.	8	Apply	BTL3	CO3
16	(i) How to pass argument to an Exception in Python?	8	Apply	BTL3	CO3
	(ii) How to Raise an Exception to Another Exception?	8	Apply	BTL3	CO3
17	Develop a python code and explain the following				
	(i) Syntax and logical errors.	8	Create	BTL6	CO3
	(ii) Exception Chaining.	8	Create	BTL6	CO3

UNIT IV

MODULES, PACKAGES AND FRAMEWORKS

Modules: Introduction – Module Loading and Execution – Packages – Making Your Own Module –The Python Libraries for data processing, data mining and visualization- NUMPY, Pandas, Matplotlib, Plotly-Frameworks-Django, Flask, Web2Py.

Part A

Q. No.	Questions	Marks	Competence	BT Level	CO's
1	What is NumPy? Illustrate	2	Understand	BTL2	CO4
2	What are modules and packages in Python?	2	Remember	BTL1	CO4
3	What is loading and execution?	2	Understand	BTL2	CO4
4	How many modules are in Python?	2	Remember	BTL1	CO4
5	What are the advantages of modules in Python?	2	Understand	BTL2	CO4
6	What is web2py framework?	2	Remember	BTL1	CO4
7	Summarize the benefits of web2py?	2	Remember	BTL1	CO4
8	How to load a package in Python?	2	Understand	BTL2	CO4
9	What is the difference between Python module and package?	2	Understand	BTL2	CO4
10	Define library in Python?	2	Remember	BTL1	CO4
11	What is a Python package?	2	Remember	BTL1	CO4
12	Write a python snippet to illustrate a python module	2	Understand	BTL2	CO4
13	List the advantages to modularizing code in python.	2	Remember	BTL1	CO4
14	Write Python libraries which are commonly used for data processing and data visualization?	2	Remember	BTL1	CO4
15	What is a pip in Python?	2	Remember	BTL1	CO4
16	What are the disadvantages of NumPy?	2	Remember	BTL1	CO4
17	What are data visualization techniques in data mining?	2	Understand	BTL2	CO4
18	List the Python libraries for data processing and visualization?	2	Understand	BTL2	CO4
19	Which data visualization library is best? Why?	2	Understand	BTL2	CO4
20	List Python Built-in modules.	2	Understand	BTL2	CO4
21	Which Python library is commonly used for image processing tasks?	2	Understand	BTL2	CO4

22	What is the difference between NumPy and array?	2	Understand	BTL2	CO4
23	What is the first Python data visualization library? Justify	2	Remember	BTL1	CO4
24	What are the key factors of data visualization?	2	Remember	BTL1	CO4
Part B					
1	Write Python snippet to create, use, access and rename the modules	16	Create	BTL6	CO4
2	Explain with example code for Built-in Modules.	16	Analyze	BTL4	CO4
3	Compare modules with packages in Python.	16	Analyze	BTL4	CO4
4	Discover how to write modules in Python?	16	Evaluate	BTL5	CO4
5	Explain with code to Access Modules from Another Directory.	16	Apply	BTL3	CO4
6	How to Create Package in Python. Explain in detail with example code.	16	Create	BTL6	CO4
7	List and narrate different type of Python packages for web frameworks.	16	Apply	BTL3	CO4
8	What is meant by Python Numpy? How to create a Numpy Array? Explain with code.	16	Evaluate	BTL5	CO4
9	How to Access the array Index in numpy array? Explain with example code.	16	Evaluate	BTL5	CO4
10	List and explain in detail on the Libraries for Data Visualization in Python programming.	16	Apply	BTL3	CO4
11	Write a short notes on the following Python libraries for data visualization.				
	(i) Matplotlib	8	Analyze	BTL4	CO4
	(ii) Plotly	8	Analyze	BTL4	CO4
12	How NumPy can be used to calculate the mean and standard deviation of a dataset?	16	Evaluate	BTL5	CO4
13	What are the advantages and disadvantages of Flask? Justify	16	Evaluate	BTL5	CO4
14	What is Flask? Write a short notes on features of Flask?	16	Apply	BTL3	CO4
15	Compare and contrast Flask with Django.	16	Evaluate	BTL5	CO4
16	List and explain the important features of Django.	16	Analyze	BTL4	CO4
17	What are the differences between Django, Flask and Web2py in terms of functionality, easiness?	16	Analyze	BTL4	CO4

UNIT V

OBJECT ORIENTED PROGRAMMING IN PYTHON

Creating a Class, Class methods, Class Inheritance, Encapsulation, Polymorphism, class method vs. static methods, Python object persistence.

Q. No.	Questions	Marks	Competence	BT Level	CO's
1	What is a class method, and how is it different from an instance method?	2	Understand	BTL2	CO5
2	What is meant by class?	2	Remember	BTL1	CO5
3	Define class inheritance and its usage	2	Remember	BTL1	CO5
4	What is polymorphism in Python, and give an example?	2	Remember	BTL1	CO5
5	State the features of Encapsulation with example	2	Understand	BTL2	CO5
6	Illustrate object-Oriented Programming features	2	Understand	BTL2	CO5
7	Compare abstract classes versus concrete classes in Python.	2	Understand	BTL2	CO5
8	Describe the use of super() in inheritance	2	Understand	BTL2	CO5
9	Define superclass and subclass.	2	Remember	BTL1	CO5
10	Illustrate the use of a constructor in a given Python class.	2	Understand	BTL2	CO5
11	What are the class members? How can you access them?	2	Remember	BTL1	CO5
12	Write a Python program to illustrate Inheritance.	2	Understand	BTL2	CO5
13	Write a short note on special class methods.	2	Remember	BTL1	CO5
14	What is class instantiation? How is it done?	2	Understand	BTL2	CO5
15	Write built in functions used in Python.	2	Remember	BTL1	CO5
16	Differentiate between class variables and instance variable.	2	Understand	BTL2	CO5
17	List the shelve module functions.	2	Remember	BTL1	CO5
18	What are pickle module functions?	2	Remember	BTL1	CO5
19	Develop a Python code to illustrate the use of __init__() method.	2	Understand	BTL2	CO5
20	Describe polymorphism in addition operator.	2	Understand	BTL2	CO5

21	What is the output of the following program?	2	Understand	BTL2	CO5

	# define a class
	class Employee:
	  # define a property
	  employee_id = 0
	# create two objects of the Employee class
	employee1 = Employee()
	employee2 = Employee()
	# access property using employee1
	employee1.employeeID = 1001
	print("Employee ID: {employee1.employeeID}")
	# access properties using employee2
	employee2.employeeID = 1002
	print(employee ID: {employee2.employeeID}")				

22	What is the output of the below code?	2	Understand	BTL2	CO5

	class Person:
	  def __init__(self, first_name, last_name):
	    self.first_name = first_name
	    self.last_name = last_name
	first_name = "XYZ"
	person = Person(first_name, "ABC")
	first_name = "LMN"
	person.last_name = "PQR"
	print(person.first_name,person.last_name)				

23	What is the use of class or static method?	2	Remember	BTL1	CO5
24	Define Python object persistence.	2	Remember	BTL1	CO5
Part -B					
1	(i). Analyze the relationship between inheritance and polymorphism in Python OOPS.	8	Analyze	BTL4	CO5
	(ii). Explain how encapsulation and abstraction are related in Python OOPS.	8	Analyze	BTL4	CO5
2	(i). What other paradigms of programming exist besides OOPs?	8	Apply	BTL3	CO5
	(ii). What are the differences between procedural and Object-Oriented Programming?	8	Apply	BTL3	CO5
3	Develop a menu-driven program to read, display, add, and subtract two time objects with example.	16	Create	BTL6	CO5
4	(i).Interpret access specifiers, and When should we use these?	8	Evaluate	BTL5	CO5
	(ii). What is method Overloading? Conclude when it will be used?	8	Evaluate	BTL5	CO5
5	(i).Examine what is method Overriding?	4	Analyze	BTL4	CO5
	(ii).Analyze the differences between method overriding and method overloading in Python.	12	Analyze	BTL4	CO5
6	Analyze the different types of Inheritance with illustrative program.	16	Analyze	BTL4	CO5

7	Develop a Python code to read two points and calculate the distance between them using class and methods.	16	Create	BTL6	CO5
8	Make a class triangle, enter its three sides and calculate its area using class and methods.	16	Evaluate	BTL5	CO5
9	Write a class that has a list of integers as data members and read (), display (), find_largest(), find_smallest(), sum(), and find_mean() as its member functions.	16	Apply	BTL3	CO5
10	Develop a Python code that has a class point with attributes as the x and y co-ordinates. Make two objects of this class and find the midpoint of both the points.	16	Evaluate	BTL5	CO5
11	Develop a Python code to find mean of two numbers belonging to two different objects of the same class.	16	Create	BTL6	CO5
12	Develop a Python code that swaps two members of a class.	16	Create	BTL6	CO5
13	(i). Develop a Python code to call a class method from another method of the same class.	8	Evaluate	BTL5	CO5
	(ii). Develop a Python code to add variables to a class at run time.	8	Evaluate	BTL5	CO5
14	(i). Develop a Python code to illustrate the modification of an instance variable.	8	Create	BTL6	CO5
	(ii). Develop a Python code to modify a mutable type attributes.	8	Create	BTL7	CO5
15	How do you make use of persistent objects in Python?	16	Apply	BTL3	CO5
16	(i).What is class method and static methods in Python? Give examples.	10	Apply	BTL3	CO5
	(ii).Compare the differences between them with examples.	6	Analyze	BTL4	CO5
17	Write a static method that checks whether all words in a list starts with a vowel.	16	Evaluate	BTL5	CO5

Ω Ω Ω