

The Kelkar Education Trust's V G Vaze College of Arts, Science and Commerce (Autonomous)

# Syllabus for S.Y.B.Sc. - I.T. (June 2024 Onwards)

Programme: B.Sc. Semester III & IV

**Subject : Information Technology** 

Semester III				
Code	Course Type	Course Title	Credit s	
VSIT200	Major Subject	Computer Network	2	
VSIT201	Major Subject Practical	Computer Network Practical	2	
VSIT202	Major Subject	Operating System	2	
VSIT203	Major Subject Practical	Operating System Practical	2	
VSIT204	Minor Subject	Python Programming	2	
VSIT205	Minor Subject Practical	Python Programming Practical	2	
VFLC230	Open Elective Subject	Law and Cyberspace -I	2	
VSIT206	Vocational Skill Enhancement Course Pr	Introduction to Java Programming Practical	2	
VFWS237	Ability Esterney of Course	Writing Skills in Marathi - I	2	
VFWS238	Ability Ennancement Course	Writing Skills in Hindi - I		
VSIT207	Field Project	Field Project	2	
VCA236	Cocurricular Activities	Community Engagement Activities	2	
VCE237	Cocurricular Activities	Cultural	2	
VNS238	Cocurricular Activities	National Service Scheme (NSS)	2	
VSA239	Cocurricular Activities	Sports Activities	2	
VYG240	Cocurricular Activities	Yoga	2	
VKB241	Cocurricular Activities	Performance Techniques I: Kathak & Bollywood	2	
VSS242	Cocurricular Activities	Sangeet Sadhana	2	
		Total Credits	22	

Semester IV			
Code	Course Type	Course Title	Credits
VSIT250	Major Subject	Data Structure	2
VSIT251	Major Subject Practical	Data Structure Practical	2
VSIT252	Major Subject	Software Engineering	2
VSIT253	Major Subject Practical	Software Engineering Practical	2
VSIT254	Minor Subject	Computer Oriented Statistical Techniques	2
VSIT255	Minor Subject Practical	Computer Oriented Statistical Techniques Practical	2
VFLC280	Open Elective Subject	Law and Cyberspace -II	2
VSIT256	Vocational Skill Enhancement Course Pr	Advanced Web Programming	2
VFLC280	Writing Skills in Marathi - II		2
VFWS290	Ability Emancement Course	Writing Skills in Hindi - II	2
VSIT257	Community Engagement Program	Community Engagement Program	2
VCA286	Cocurricular Activities	Community Engagement Activities	2
VCE287	Cocurricular Activities	Cultural Activities	2
VNS288	Cocurricular Activities	National Service Scheme (NSS)	2
VSA289	Cocurricular Activities	Sports Activities	2
VYG290	Cocurricular Activities	Yoga	2
VKB291	Cocurricular Activities	Performance Techniques II: Kathak & Bollywood	2
VSS292	Cocurricular Activities	Sangeet Sadhana	2
		Total Credits	22

# **SEMESTER-III**

B. Sc. (Information T	echnology)	Semester – I	II	
Course Name: Computer Networks		Course Code: VSIT200		
Periods per week (1 Period i	s 60 minutes)	2		
Credits	Credits		2	
		Hours	Marks	
Evaluation System	Theory Examination	2	60	
	Internal		40	

#### **Course Objective**

#### To make the learner

- 1. Aware of Understanding of the fundamental concepts of computer networking as well as will understand functionalities and various concepts of physical layer.
- 2. Aware of Bandwidth utilization, transmission media and switching concept as well as will understand functionalities and various concepts of Data link layer.
- 3. Familiarize with basic taxonomy and terminology of computer networking area.
- 4. Will understand functionalities and various concepts of Network Layer also become ware of advanced networking concepts and implement the same using CISCO packet tracer.
- 5. Will understand functionalities and various concepts of Transport Layer.

Unit	Details	Lectures
Ι	Introduction: Data communications, networks, network types, Internet	
	history, standards and administration.	
	Network Models: Protocol layering, TCP/IP protocol suite, The OSI model.	
	Introduction to Physical layer: Data and signals, periodic analog signals,	
	digital signals, transmission impairment, data rate limits, performance.	10
	Digital and Analog transmission: Digital-to-digital conversion,	10
	analog-to-digital conversion, transmission modes, digital-to-analog	
	conversion, analog-to-analog conversion.	
II	Bandwidth Utilization: Multiplexing and Spectrum Spreading:	
	Multiplexing, Spread Spectrum	
	Transmission media: Guided Media, Unguided Media	
	Switching: Introduction, circuit switched networks, packet switching,	10
	structure of a switch.	
	Introduction to the Data Link Layer: Link layer addressing, Data Link	
	Layer Design Issues, Error detection and correction, block	
	coding, cyclic codes, checksum, forward error correction, error correcting	
	codes, error detecting codes.	

III	<ul> <li>Introduction to the Network Layer: Network layer services, packet switching, network layer performance, IPv4 addressing, forwarding of IP packets, Internet Protocol, ICMPv4, Mobile IP</li> <li>Unicast Routing: Introduction, routing algorithms, unicast routing protocols.</li> <li>Introduction to the Transport Layer: Introduction, Transport layer protocols (Simple protocol, Stop-and-wait protocol, Go-Back-n protocol, Selective repeat protocol, Bidirectional protocols),Transport layer services,</li> </ul>	10
	Selective repeat protocol, Bidirectional protocols),Transport layer services, User datagram protocol, Transmission control protocol,	

Cours	Course Outcome			
Learn	Learner will be able to			
CO1	Understand and explain Data Communications System and its components.			
CO2	Identify the different types of network topologies and protocols.			
CO3	Identify the different types of network devices and their functions within a			
	network.			
<b>CO4</b>	Understand and building the skills of sub netting and routing mechanisms.			
CO5	Motivate the student to become network administrator.			

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Data Communication	Behrouz A.	Tata McGraw Hill	Fifth Edition	2013
	and Networking	Forouzan			
2.	TCP/IP	Behrouz A.	Tata McGraw Hill	Fourth Edition	2010
	Protocol Suite	Forouzan			
3.	Computer Networks	Andrew	Pearson	Fifth	2013
		Tanenbaum			

B. Sc. (Informa	tion Technology)	Semester – I	II
Course Name: Comp	outer Networks Practical	Course Code: VSIT201	
Periods per week (1	Period is 120 minutes)	2	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	100
	Internal		

#### List of Practical

# 1. IPv4 Addressing and Subnetting

a) Given an IP address and network mask, determine other information about the IP address such as:

- Network address
- Network broadcast address
- Total number of host bits
- Number of hosts

b) Given an IP address and network mask, determine other information about the IP address such as:

- The subnet address of this subnet
- The broadcast address of this subnet
- The range of host addresses for this subnet
- The maximum number of subnets for this subnet mask
- The number of hosts for each subnet
- The number of subnet bits
- The number of this subnet

2. Use of ping and tracert / traceroute, ipconfig / ifconfig, route and arp utilities.

**3.** Configure IP static routing.

**4.** Configure IP routing using RIP.

**5.** Configuring Simple OSPF.

**6.** Configuring DHCP server and client.

7. Create virtual PC based network using virtualization software and virtual NIC.

**8.** Configuring DNS Server and client.

**9.** Configuring OSPF with multiple areas.

**10.** Use of Wireshark to scan and check the packet information of following protocols

- HTTP
- ICMP
- TCP
- SMTP
- POP3

B. Sc (Information Technology)		Semester – III	
Course Name: Operating Systems		Course Code: VSIT202	
Periods per week 1 Period	is 60 minutes	2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal		40

Cours	e Objective
Learne	er should be able
1.	To learn the fundamentals of Operating Systems and the mechanisms of OS to handle
	processes and threads and their communication
2.	To learn the mechanisms involved in memory management and file management in
	contemporary OS.
3.	To learn handling of input a nd output as well as deadlock concept in operation system.
4.	To gain introductory knowledge on distributed operating system concepts that includes
	architecture.
5.	To gain introductory knowledge on Multiprocessor Systems.

Unit	Details	Lectures
	Introduction:	
	What is an operating system? History of operating system, computer hardware,	
т	different operating systems, operating system concepts, system calls, operating	10
	system structure.	
	Processes and Threads:	
	Processes, threads, inter process communication, scheduling, IPC problems.	
	Memory Management:	
	No memory abstraction, memory abstraction: address spaces, virtual memory,	
	page replacement algorithms, design issues for paging systems, implementation	
II	issues, segmentation.	10
	File Systems:	10
	Files, directories, file system implementation, file-system management	
	and optimization.	
	Input-Output:	
ш	Principles of I/O hardware, Principles of I/O software, I/O software layers, disks,	
	clocks, user interfaces: keyboard, mouse, monitor, thin clients, power	

management,	10
Deadlocks:	
Resources, introduction to deadlocks, the ostrich algorithm, deadlock detection	
and recovery, deadlock avoidance, deadlock prevention, issues.	
Multiple Processor Systems	
Multiprocessors, multi-computers, distributed systems.	

Cours	Course Outcome		
Learr	Learners should be able to		
CO1	Analyze the structure of OS and basic architectural components involved in OS		
	design.		
CO2	Understand the various modules in Operating system like process, memory, files,		
	device and resource management techniques or different types of OS.		
CO3	Understand the Mutual exclusion, Deadlock handling methods like Deadlock		
	detection, avoidance.		
CO4	Create virtual machine and install Linux / Windows OS.		
CO5	Know the basic commands in Linux and Windows OS as well as install and use		
	Windows / Linux Desktop and utilities.		

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Modern Operating Systems	Andrew S.	Pearson	4.1	2014
		Tanenbaum, Herbert		4 <sup>th</sup>	
		Bos			
2.	Operating Systems -	Willaim Stallings	Pearson	0.1	2009
	Internals and Design			ð <sup>tn</sup>	
	Principles				
3.	Operating System Concepts	Abraham	Wiley	Qth	
		Silberschatz,		om	
		Peter B. Galvineg			
		Gagne			
4.	Operating Systems	Godbole and Kahate	McGraw Hill	3rd	

B. Sc (Information	Technology)	Semester – I	II
Course Name: Operating Systems Practical		Course Code: VSIT203	
Periods per week (1 Peri	od is 120 minutes)	2	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	100
	Internal		

List	of Practical
1.	Installation of virtual machine software.
2.	Installation of Linux operating system (RedHat / Ubuntu) on virtual machine.
3.	Installation of Windows operating system on virtual machine.
4.	Linux commands: Working with Directories:
a.	pwd, cd, absolute and relative paths, ls, mkdir, rmdir,
b.	file, touch, rm, cp. mv, rename, head, tail, cat, tac, more, less, strings, chmod
5.	Linux commands: Working with files:
a.	ps, top, kill, pkill, bg, fg,
b.	grep, locate, find, locate.
с.	date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which.
d.	Compression: tar, gzip.
6.	Windows (DOS) Commands – 1
a.	Date, time, prompt, md, cd, rd, path.
b.	chkdsk, copy, xcopy, cls, defrag, del, move.
7.	Windows (DOS) Commands – 2
a.	diskcomp, diskcopy, diskpart, doskey, echo
b.	edit, fc, find, rename, set, type, ver
8.	Working with Windows Desktop and utilities
	Notepad, Wordpad, Paint, Taskbar, Adjusting display resolution, Using the browsers,
	Configuring simple networking, Creating users and shares
9.	Working with Linux Desktop and utilities
a.	The vi editor.
b.	Graphics
c.	Terminal
d.	Adjusting display resolution
e.	Using the browsers
f.	Configuring simple networking
g.	Creating users and shares
10.	Installing utility software on Linux and Windows

<b>B. Sc. (Information Technology)</b>		Semester – III	
Course Name: Python Programming		Course Code: VSIT204	
Periods per week (1 Period is 60	2		
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal		40

Course Objective
To make learner aware of
1. Acquire programming skills in core Python.
2. Understand and implement the concept of functions and strings in Python.
3. Understand and implement the concept of Lists, Tuples and Dictionaries.
4. Understand and implement the concepts of Classes and Objects, Multithreading
Programming and modules in python
5.Understand and implement the concepts of Graphical User Interface design and

Database(MySQL) programming in Python..

Unit	Details	Lecture		
		S		
	Introduction: The Python Programming Language, History, features,			
	Installing Python, Running Python program, Debugging: Syntax Errors,			
	Runtime Errors, Semantic Errors, Experimental Debugging, Formal and			
	Natural Languages.	10		
	Variables and Expressions: Values and Types, Variables, Variable Names			
	and Keywords, Type conversion, Operators and Operands, Expressions,			
	Interactive Mode and Script Mode, Order of Operations.			
	Conditional Statements, Looping, Control statements			
Ι	Functions: Function Calls, Type Conversion Functions, Math Functions,			
	Composition, Adding New Functions, Definitions and Uses, Flow of			
	Execution, Parameters and Arguments, Variables and Parameters Are Local,			
	Stack Diagrams, Fruitful Functions and Void Functions, Why Functions?			
	Importing with from, Return Values, Incremental Development,			
	Composition, Boolean Functions, More Recursion, Leap of Faith, Checking			
	Types			
	Strings: A String Is a Sequence, Traversal with a for Loop, String Slices,			
	Strings Are Immutable, Searching, Looping and Counting, String Methods,			

The in Operator, String Comparison, String Operations.	
Lists, Tuples and Dictionaries: Creating List, Tuple and	
Dictionary, Accessing element, Built-in List, Tuple and Dictionary functions	
and methods Tuples, Tuple Assignment, Tuples as return values, Variable-	
length argument tuples, Concatenation, Repetition, in Operator, Iteration,	
Updating Dictionary, Deleting Elements from Dictionary, Properties of 10	I
Dictionary keys, Operations in Dictionary.	
<b>Classes and Objects:</b> Overview of OOP (Object Oriented Programming).	
II Class Definition Creating Objects Instances as Arguments Instances as	
return values Built-in Class Attributes Inheritance Method Overriding Data	
Encapsulation Data Hiding	
Multithreaded Programming. Thread Module erecting a thread	
Multimeaded Programming: Intead Module, creating a thread,	
synchronizing inreads, multimeaded priority queue.	
Modules: Importing module, Creating and exploring modules, Math	
module, Random module, Time module.	
Creating the GUI Form and Adding Widgets:	
Widgets: Button, Canvas, Checkbutton, Entry, Frame, Label, Listbox,	
Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel,	
Spinbox, PanedWindow, LabelFrame, Messagebox. Handling Standard 10	
attributes and Properties of Widgets.	
Layout Management: Designing GUI applications with proper Layout	
Management features.	
Storing Data in Our MySQL Database via Our GUI : Connecting to a	
MySQL database from Python, Configuring the MySQL connection,	
Designing the Python GUI database, Using the INSERT command, Using the	
UPDATE command. Using the DELETE command, Storing and retrieving	
data from MySOL database.	

Course Outcome		
Learner will be able to		
CO1	Install and use Python for simple programming tasks.	
CO2	Extend the functionality of Python by using add-on packages.	
CO3	Develop database (MySQL) application in Python.	
CO4	Create GUI based application using python.	
C05	Develop a project / application using python.	

Books and References:					
Sr.	Title	Author/s	Publisher	Edition	Year
No.					
1.	Think Python	Allen Downey	O'Reilly	1 <sup>st</sup>	2012
2.	An Introduction to	JasonMontojo,	SPD	1 <sup>st</sup>	2014
	Computer Science using	JenniferCampbell,			
	Python 3	Paul Gries			
3.	Python GUI Programming	Burkhard A. Meier	Packt		2015
	Cookbook				
4.	Introduction to Problem	E. Balagurusamy	ТМН	1 <sup>st</sup>	2016
	Solving with Python				
5.	Murach's Python	Joel Murach, Michael	SPD	1 <sup>st</sup>	2017
	programming	Urban			
6.	Object-oriented	Michael H.	Pearson	1 <sup>st</sup>	2008
	Programming in Python	Goldwasser, David	Prentice Hall		
		Letscher			
7.	Exploring Python	Budd	ТМН	1 <sup>st</sup>	2016

<b>B. Sc. (Information</b>	Technology)	Semester –	· III
Course Name: Python Pro	gramming Practical	Course Code: VSIT205	
Periods per week (1 Period	l is 120 minutes)	2	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	100
	Internal		

List	of Practical
1.	Write the program for the following:
a.	Create a program that asks the user to enter their name and their age. Print outa message
	addressed to them that tells them the year that they will turn 100 years old.
h	Enter the number from the user and depending on whether the number is even or odd, print
0.	out an appropriate message to the user.
c.	Write a program to generate the Fibonacci series.
d.	Write a function that reverses the user defined value.
0	Write a function to check the input value is Armstrong and also write the function for
e.	Palindrome.
f.	Write a recursive function to print the factorial for a given number.
2.	Write the program for the following:
а	Write a function that takes a character (i.e. a string of length 1) and returns True
а.	if it is a vowel, False otherwise.
b.	Define a function that computes the <i>length</i> of a given list or string.
	Define a <i>procedure</i> histogram() that takes a list of integers and prints a histogram to the
	screen. For example, histogram([4, 9, 7])should print the following:
c.	****
	*****
	****
3.	Write the program for the following:
	A pangram is a sentence that contains all the letters of the English alphabet at least once,
a.	for example: <i>The quick brown fox jumps over the lazy dog</i> . Your task here is to write a
	function to check a sentence to see if it is a pangram or not.
b.	Take a list, say for example this one: a=[1,1,2,3,5,8,13,21,34,55,89]
	and write a program that prints out all the elements of the list that are less than 5.
4.	Write the program for the following:
a.	Write a program that takes two lists and returns True if they have at least one

	common member.
1	Write a Python program to print a specified list after removing the 0th, 2nd, 4th and 5th
b.	elements.
с.	Write a Python program to clone or copy a list
5.	Write the program for the following:
a.	Write a Python script to sort (ascending and descending) a dictionary by value.
	Write a Python script to concatenate the following dictionaries to create a new one. Sample
	Dictionary :
h	dic1= $\{1:10,2:20\}$
0.	dic2= $\{3:30,4:40\}$
	dic3={5:50,6:60}
	Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
c.	Write a Python program to sum all the items in a dictionary.
6.	Write the program for the following:
a.	Write a Python program which shows how to create a thread using thread module.
b.	Write a Python program which shows how to create a thread using threading module.
с.	Write a Python program which shows the implementation of thread class methods.
d.	Write a Python program which creates a priority queue and retrieve its priority.
a.	Write a Python program which shows how to create a thread using thread module.
7.	Write the program for the following:
a.	Design a class that store the information of student and display the same
a. b.	Design a class that store the information of student and display the same         Implement the concept of inheritance using python
a. b.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and
a. b.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers).
a. b.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y.
a. b.	Design a class that store the information of student and display the sameImplement the concept of inheritance using pythonCreate a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers).Write a method called add which returns the sum of the attributes x and y.Write a class method called multiply, which takes a single number parameter a and returns
a. b. c.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two
a. b. c.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a
a. b. c.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter
a. b. c.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y.
a. b. c.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y. Write the program for the following:
a. b. c. <b>8.</b> a.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y. <b>Write the program for the following:</b> Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py
a. b. c. <b>8.</b> a.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y. <b>Write the program for the following:</b> Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions
a. b. c. <b>8.</b> a.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y. <b>Write the program for the following:</b> Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the "Control Flow and
a. b. c. <b>8.</b> a.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y. <b>Write the program for the following:</b> Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the "Control Flow and Functions" exercise into this file and save it.
a. b. c. <b>8.</b> a.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y. <b>Write the program for the following:</b> Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the "Control Flow and Functions" exercise into this file and save it. Now open a new file and save it in the same directory. You should now be able to import
a. b. c. <b>8.</b> a.	<ul> <li>Design a class that store the information of student and display the same</li> <li>Implement the concept of inheritance using python</li> <li>Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers).</li> <li>Write a method called add which returns the sum of the attributes x and y.</li> <li>Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y.</li> <li>Write the program for the following:</li> <li>Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the "Control Flow and Functions" exercise into this file and save it.</li> <li>Now open a new file and save it in the same directory. You should now be able to import your own module like this:import geometry</li> </ul>
a. b. c. <b>8.</b> a.	Design a class that store the information of student and display the same Implement the concept of inheritance using python Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). Write a method called add which returns the sum of the attributes x and y. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER. Write a static method called subtract, which takes two number parameters, b and c, and returns b -c. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and getter methods for manipulating the values of x and y. <b>Write the program for the following:</b> Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the "Control Flow and Functions" exercise into this file and save it. Now open a new file and save it in the same directory. You should now be able to import your own module like this:import geometry Try and add print dir(geometry) to the file and run it.

	square pyramid if squareBase is True and of a right circular cone if squareBase is False. x
	is the length of an edge on a square if squareBase is True and the radius of a circle when
	squareBase is False. y is the height of the object. First use squareBase to distinguish the
	cases. Use the circleArea and squareAreafrom the geometry module to calculate the base
	areas.
b.	Write a program to implement exception handling.
9.	Write the program for the following:
a.	Try to configure the widget with various options like: bg="red", family="times", size=18
h	Try to change the widget type and configuration options to experiment with other widget
0.	types like Message, Button, Entry, Checkbutton, Radiobutton, Scale etc.
10	Design the database applications for the following:
a.	Design a simple database application that stores the records and retrieves the same.
b.	Design a database application to search the specified record from the database.
с.	Design a database application that allows the user to add, delete and modify the records.

B. Sc (Information Technology)		Semester – III	
Course Name: Law and Cyberspace - I		Course Code: VFLC230	
Periods per week 1 Period is 60 minutes			2
Credits			2
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal		40

Course	Objective
--------	-----------

To make learner understand and use	
1. IT Act 2000 and its subsequent amendments.	
2. Contracts in the digital World.	
3. Concept of copyright, infringement of copyright, license of copyright	
4. Concept of Jurisdiction in cyberspace.	

5. Legal rights on cyberspace by recognizing the issues on it.

Unit	Details	Lectures
	Introduction:	
	What is Cyber law ? Need for cyber law, Power of Arrest Without Warrant	
	Under the IT Act, 2000: A Critique, Crimes of this Millennium, Section 80	10
	of the IT Act, 2000 - A Weapon or a Farce? Forgetting the Line Between	
	Cognizable and Non-Cognizable Offences, Necessity of Arrest without	
	Warrant from Any Place, Public or Otherwise, Check and Balances Against	
	Arbitrary Arrests,	
Ι	Cyber Crime and Criminal Justice: Penalties, Adjudication and Appeals	
	Under the IT Act,2000: Concept of "Cyber Crime" and the IT Act, 2000,	
	Hacking, Teenage Web Vandals, Cyber Fraud and Cyber Cheating, Virus on	
	the Internet, Defamation, Harassment and E- mail Abuse, Cyber	
	Pornography, Other IT Act Offences, Monetary Penalties, Adjudication and	
	Appeals Under IT Act , 2000, Network Service Providers, Jurisdiction and	
	Cyber Crime, Nature of Cyber Criminality, Strategies to Tackle Cyber Crime	
	and Trends, Criminal Justice in India and Implications on Cyber Crime.	
	Contract's and Jurisdiction's in Cyber World	
	Contracts in the Infotech World: What are Contracts in the Infotech World?	
II	, Click-Wrap and Shrink-Wrap Contract: Status under the Indian Contract	
	Act, 1872, Contract Formation Under the Indian Contract Act, 1872, Contract	10
	Formation on the Internet, Terms and Conditions of Contracts.	10

	Jurisdiction in the Cyber World: Questioning the Jurisdiction and Validity of	
	the Present Law of Jurisdiction, Civil Law of Jurisdiction in India, Cause of	
	Action, Jurisdiction and the Information Technology	
	IPR's in cyber space:	
	Battling Cyber Squatters and Copyright Protection in the Cyber World:	
	Concept of Domain Name and Reply to Cyber Squatters, legal issues related	
	to Meta-Tagging, Legislative and Other Innovative Moves Against	10
	Cybersquatting, The Battle Between Freedom and Control on the Internet,	10
	Works in Which Copyright Subsists and meaning of Copyright, Copyright	
111	Ownership and Assignment, License of Copyright, Copyright Terms and	
	Respect for Foreign Works, Copyright Infringement, Remedies and	
	Offences, Copyright Protection of Content on the Internet; Copyright Notice,	
	Disclaimer and Acknowledgement, Downloading for Viewing Content on	
	the Internet, Legal issues related to Hyper-Linking and Framing, Computer	
	Software Piracy.	

Course Outcome	
Learner will achieve competency in	
CO1	Knowledge of Cyberspace, Jurisdiction and basic concepts of Cyber Law
CO2	Jurisdiction in cyberspace, cybercrimes and their legal provisions in India.
CO3	Knowledge of legal provisions in Information Technology Act, 2000
CO4	Identify the issues on the Internet.
CO5	Contracts in the Digital world like Click-Wrap and Shrink-Wrap Contract.

Books and References:					
Sr.	Title	Author/s	Publisher	Edition	Year
No.					
1.	Cyber Law Simplified	Vivek Sood	TMH	First	2001
			Education	edition	
2.	Cyber Laws	Pavan Duggal	Universal's	3 <sup>rd</sup>	2023
				edition	
3.	The Information technology	Professional's	Professional's	Latest	2024
	Act, 2000 -Bare act 2024		Book	2024	
			Publisher		

<b>B. Sc. (Information Technology)</b>			ester – III
Course Name: Java Programming Practical			e Code: VSIT206
Periods per week (1 Period is 120 minutes)			2
Credits			2
		Hours	Marks
Evaluation System	Practical Examination	3	100

#### **Course Objective**

#### To make learner understand and use

- 1. Basics of JAVA
- 2. Control flow, looping statements and Classes in JAVA.
- 3. Concept of inheritance and packages.
- 4. Concept of Arrays, Multithreading.
- 5. Concept of event handling, abstract window toolkit and layouts.

#### List of Practical

	Java Basics-I:
1.	Data types: primitive data types, Object Reference Types, Strings, Auto boxing, Operators: Arithmetic operators, assignment operators, increment and decrement operator, relational operator, logical operator, bitwise operator, conditional operator
	Write a Loss and a statistic a new house in section desired its multiplication table
a.	write a Java program that takes a number as input and prints its multiplication table.
b.	Write a Java program to display the following pattern. * ** *** **** ****
c.	Write a Java program to display the following pattern. * * * * * * * * * * * * * * * * * * *
2.	Java Basics-II: Control Flow Statements, Iteration Statements (Loopings), Jump statements, Arrays.
a.	Write a Java program which shows that the given number is a Prime or not.
b.	Write a Java program to count the letters, spaces, numbers and other characters of an input string

с.	Write a Java program to find largest and smallest number from the array.
d.	Write a Java program to reverse a string or number
e.	Write a Java program which shows that the given number is a Palindrome or not.
f.	Write a Java program which shows the implementation of Armstrong number.
	Java Classes and Objects:
	Definition of class, syntax to write classes, Object, syntax to write object, Instantiating
3	Objects From A Class, Initializing The Class Object And Its Attributes, Class Methods,
5.	Accessing A Method, Method Returning A Value, Method's Arguments, Method
	Overloading, Constructors, this Instance, super Instance, this instance, static fields of a
	class, static methods of a class.
a.	Write a simple program in Java to demonstrate class and object.
b.	Write a simple program in Java to demonstrate class and object with its instance variables and instance methods.
с.	Write a Java program which shows the implementation of method overloading.
d.	Write a Java program which shows the implementation of static fields and static methods
	of a class.
4.	Constructor and Destructor:
	Purpose of constructor, types of constructor, use of super keyword in constructor,
	destructor.
a.	a) Write a Java program which shows the use of constructor and destructor.
	b) Write a Java program which shows the implementation of all types of constructors.
b.	Write a Java program which shows the implementation of constructor overloading.
c.	Write a simple Java program which shows the use of super keyword.
d.	Write a Java program which demonstrates the use of destructor.
	Inheritance:
	Inheritance basics, Types of Inheritance, Derived Class Objects, Inheritance and Access
5.	Control, Default Base Class Constructors, this and super keywords, Method overriding,
	inheritance Defining An Interface Implementing Interfaces Classes V/s Interfaces
	Interface can be extended.
a.	Write a java program to implement all types of inheritance.
b.	Write a java program to implement method overriding.
с.	Write a java program to implement method overriding using dynamic method dispatch.
d.	Write a java program which how to implement interface.
e.	Write a java program to implement multiple inheritance using interface.
6	Write a Java program which shows the implementation of interface extending other
f.	interface.

6.	Packages:
	Creating Packages, Default Package, Importing Packages, Using A Package.
a.	Create a package, Add the necessary classes and import the package in java class.
b.	Write a java program to add two matrices and print the resultant matrix.
с.	Write a java program for multiplying two matrices and print the product for the same.
7.	Multithreading:
	Multithreading: the thread control methods, thread life cycle, the main thread, creating a
	thread, extending the thread class.
a.	Write a java program to implement thread life cycle.
b.	Write a java program to implement multithreading.
8.	GUI programming(Part A):
	Abstract Window Toolkit: Introduction to Window Fundamentals, Component,
	Container, Panel, Window, Frame, Canvas. Components – Labels, Buttons, Check Boxes,
	Frames
0	Design a AWT program which shows the following components TextBox, Button, Label
а.	using Frame
h	Design a AWT program which shows the following components CheckBox Radio
0.	Buttons using Frame.
d.	Design a AWT program which shows the following components Choice Menus, Scrolling
	List, ScrollBars using Frame.
9.	GUI programming(Part B):
	Event Handling: Delegation Event Model, Events, Event classes, Event listener interfaces,
	Using delegation event model.
	Layouts: Flow Layout, Grid Layout, Border Layout, Card Layout.
a.	Design a AWT program to print the factorial for an input value.
b.	Design an AWT program to perform various string operations like reverse string, string
	concatenation etc.
d.	Design an AWT application that contains the interface to add student information and
	display the same.
	Design an AWT application that contains Choice Menu on selecting an option from menu
	change the value of a Label to a selected option.
e.	Design an AWT application to generate result marks sheet.
10.	Introduction to NetBeans and Eclipse:
	Swing: Introduction to Swing, Difference between AWT and Swing, Swing Components:
	JButton, JLabel, JTextFiled, JTextArea, JCheckBox, JRadioButton, JMenuBar, JMenu,
	Jivienuitem.

a.	Design a Swing application which shows the implementation of swing components using
	NetBeans/Eclipse.
b.	Design an AWT application which shows the implementation of AWT components using
	NetBeans/Eclipse.

Course	Course Outcome	
Learner will be able to		
CO1	Implement object oriented programming concepts in Java.	
CO2	Create and use package and interfaces in a Java.	
CO3	Develop Java application using graphical user interface.	
<b>CO4</b>	Develop windows application using AWT.	
CO5	Handling of events with controls.	

Books and	Books and References:				
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Core Java 8 forBeginners	Vaishali Shah, Sharnam	SPD	1 <sup>st</sup>	2015
		Shah			
2.	Java: The Complete	Herbert Schildt	McGrawHill	9 <sup>th</sup>	2014
	Reference				
3.	Murach's beginningJava	Joel Murach, Michael	SPD	1 <sup>st</sup>	2016
	with Net Beans	Urban			
4.	Core Java, Volume I:	Hortsman	Pearson	9 <sup>th</sup>	2013
	Fundamentals				
5.	Core Java, Volume II:	Gary Cornell and	Pearson	8 <sup>th</sup>	2008
	Advanced Features	Hortsman			
6.	Core Java: An Integrated	R. Nageswara Rao	DreamTech	1 <sup>st</sup>	2008
	Approach				

# **SEMESTER-IV**

B. Sc. (Information Technology)		Semester –	IV
Course Name: Data Structures		Course Code:	VSIT250
Periods per week (1 Period is 60 minutes)		2	
Credits		2	
		Hours	Marks
Evaluation System	Theory	2	60
	Examination		
	Internal		40

# Course Objective

#### To make learner understand and use

- 1. Basics concepts of algorithms and data structure and compute the complexity of different algorithm.
- 2. Concept of linked list and different types of linked lists as well as implementation.
- 3. Concept of stack and queue as well as its implementation in different ways.
- 4. Concept of Various sorting and searching techniques as well as trees and its implementation.
- 5. Concept of hashing techniques and graphs as well as its implementation.

#### Ghg

Unit	Details	Lectures
	Introduction: Data and Information, Data Structure, Classification of Data	
	Structures, Primitive Data Types, Abstract Data Types, Data structure vs.	
	File Organization, Operations on Data Structure, Algorithm, Importance of	
	Algorithm Analysis, Complexity of an Algorithm, Asymptotic Analysis and	
	Notations, Big O Notation, Big Omega Notation, Big Theta Notation, Rate	
	of Growth and Big O Notation.	
	Array: Introduction, One Dimensional Array, Memory Representation of	
	One Dimensional Array, Traversing, Insertion, Deletion, Searching, Sorting,	10
	Merging of Arrays, Multidimensional Arrays, Memory Representation of	10
Ι	Two Dimensional Arrays, General Multi- Dimensional Arrays, Advantages	
	and Limitations of Arrays.	
	Linked List: Linked List, One-way Linked List, Traversal of Linked List,	
	Searching, Memory Allocation and De-allocation, Insertion in Linked List,	
	Deletion from Linked List, Copying a List into Other List, Merging Two	
	Linked Lists, Splitting a List into Two Lists, Reversing One way linked List,	
	Circular Linked List, Applications of Circular Linked List, Two way Linked	
	List, Traversing a Two way Linked List, Searching in a Two way linked	
	List, Insertion of an element in Two way Linked List, Deleting a node from	
	Two way Linked List.	

	Stack: Introduction, Operations on the Stack Memory Representation of	
	Stack, Array Representation of Stack, Applications of Stack, Evaluation of	
	Arithmetic Expression, Matching Parenthesis, infix and postfix operations,	
	Recursion.	
	Queue: Introduction, Queue, Operations on the Queue, Memory	
	Representation of Queue, Array representation of queue, Linked List	10
11	Representation of Queue, Circular Queue, Some special kinds of queues,	10
	Deque, Priority Queue, Application of Priority Queue, Applications of	
	Queues.	
	Sorting and Searching Techniques :	
	Bubble, Selection, Insertion, Merge Sort. Searching: Sequential, Binary,	
	Indexed Sequential Searches, Binary Search.	
-	Tree:	
	Tree, Binary Tree, Properties of Binary Tree, Memory Representation of	
	Binary Tree, Operations Performed on Binary Tree, Reconstruction of	
	Binary Tree from its Traversals, Huffman Algorithm, Binary Search Tree,	
	Operations on Binary Search Tree, Heap, Memory Representation of Heap,	
	Operation on Heap, Heap Sort.	10
	Hashing Techniques	
III	Hash function, Address calculation techniques, Common hashing functions	
	Collision resolution, Linear probing, Quadratic, Double hashing, Bucket	
	hashing, Deletion and rehashing	
	Graph: Introduction, Graph, Graph Terminology, Memory Representation	
	of Graph, Adjacency Matrix Representation of Graph, Adjacency List or	
	Linked Representation of Graph, Operations Performed on Graph, Graph	
	Traversal, Applications of the Graph, Reachability, Shortest Path Problems,	
	Spanning Trees.	

Course	Course Outcome	
Learner will be able to		
CO1	Select appropriate data structure for a specific problem.	
CO2	Implement linear and Non-linear data structure.	
CO3	Implement appropriate searching / sorting technique for given problem.	
CO4	Determine the complexity of given algorithm.	
CO5	Implement different operations on various data structures.	

Books	Books and References:				
Sr.	Title	Author/s	Publisher	Edition	Year
No.					
1.	A Simplified Approach	Lalit Goyal,Vishal	SPD	1 st	2014
	to Data Structures	Goyal,Pawan Kumar		131	
2.	An Introduction to Data	Jean – Paul	Tata	<b>D</b> nd	2007
	Structure with	Tremblay and Paul	MacGraw	Zna	
	Applications	Sorenson	Hill		
3.	Data Structure and	Maria Rukadikar	SPD	1.	2017
	Algorithm			1 st	
4.	Schaum's Outlines Data	Seymour Lipschutz	Tata	0.1	2005
	structure		McGraw	Znd	
			Hill		
5.	Data structure – A	AM Tanenbaum,	Prentice	Ond	2006
	Pseudocode Approach	Y Langsamand MJ	Hall India	Zna	
	with C	Augustein			
6.	Data structure and	Weiss, Mark Allen	Addison	1 at	2006
	Algorithm Analysis inC		Wesley	1 st	

B. Sc. (Information Technology)		Semester – IV	
Course Name: Data S	Structures Practical	Course Code: VSIT251	-
Periods per week (1 F	Period is 120 minutes)	2	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	100
	Internal		

List of	Practical
1.	Implement the following:
a.	Write a program to store the elements in 1-D array and perform the operations like
	searching, sorting and reversing the elements. [Menu Driven]
b.	Read the two arrays from the user and merge them and display the elements in sorted
	order.[Menu Driven]
c.	Write a program to perform the Matrix addition, Multiplication and Transpose
	Operation. [Menu Driven]
2.	Implement the following for Linked List:
a.	Write a program to create a single linked list and display the node elements in
	reverse order.
b.	Write a program to search the elements in the linked list and display the same
c.	Write a program to create a double linked list and sort the elements in the linked list.
3.	Implement the following for Stack:
a.	Write a program to implement the concept of Stack with Push, Pop, Display and Exit
	operations.
b.	Write a program to convert an infix expression to postfix and prefix conversion.
4.	Implement the following for Queue:
a.	Write a program to implement the concept of Queue with Insert, Delete, Display and
	Exit operations.
b.	Write a program to implement the concept of Circular Queue
5	Implement the following govting techniques
5.	Implement the following sorting techniques:
a.	Write a program to implement bubble sort.
b.	Write a program to implement selection sort.
с.	Write a program to implement insertion sort.

6.	Implement the following data structure techniques:
a.	Write a program to implement merge sort.
b.	Write a program to search the element using sequential search.
с.	Write a program to search the element using binary search.
7.	Implement the following data structure techniques:
a.	Write a program to create the tree and display the elements.
b.	Write a program to construct the binary tree.
с.	Write a program for inorder, postorder and preorder traversal of tree
8.	Implement the following data structure techniques:
a.	Write a program to insert the element into the maximum heap.
b.	Write a program to insert the element into the minimum heap.
9.	Implement the following data structure techniques:
a.	Write a program to implement the collision technique.
b.	Write a program to implement the concept of linear probing.
10.	Implement the following data structure techniques:
a.	Write a program to generate the adjacency matrix.
b.	Write a program for the shortest path diagram.

Books and References:						
Sr. No.	Title	Author/s	Publishe	Editio	Yea	
			r	n	r	
1.	Data Structures and Algorithms Using Python	Rance Necaise	Wiley	First	201 6	
2.	Data Structures Using C and C++	Langsam, Augenstein, Tanenbaum	Pearson	First	201 5	

B.Sc.(Informati	on Technology)	Semester-I	V
Course Name: Softwa	are Engineering	Course Code: VSIT252	
Periods per week (1 l	Period is 60 minutes)	2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal		40

#### **Course Objective**

#### To make learner understand and use

- 1. Concept of Software engineering, types of software requirements, software development process models.
- 2. SocioTechnical and critical system along with requirement engineering process and different system models used for documentation.
- 3. Architectural and user interface design for system, project management and quality Management
- 4. Concept to software testing, cost estimation.
- 5. Concept of service oriented software engineering.

Unit	Details	Lectures
I	Introduction: Professional software development, Software engineering ethics Software Processes: Software process models, Process activities, Coping with change, Process improvement Agile software development: Agile methods, Agile development techniques, Agile project management, Scaling agile methods Project planning : Software pricing, Plan-driven development, Project scheduling Agile planning Estimation techniques COCOMO cost modeling	10
п	<ul> <li>Requirements Engineering: Functional and non-functional requirements, Requirements engineering processes, Requirements elicitation, Requirements specification, Requirements validation Requirements change Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.</li> <li>Socio-technical system: Essential characteristics of socio technical systems, Emergent System Properties, Systems Engineering, Components of system such as organization, people and computers, Dealing Legacy Systems.</li> <li>Critical system: Types of critical system, A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems.</li> </ul>	10

	System Modeling: Context models, Interaction models, Structural models,	
	Behavioral models, Model-driven engineering	
	Architectural Design : Architectural design decisions, Architectural views,	
	Architectural patterns, Application architectures	
	Design and Implementation : Object-oriented design using the UML,	
	Design Patterns, Implementation Issues, Open Source Development	
III	Software Testing : Development Testing, Test-driven development, Release	10
	testing, User testing	
	Software Evolution : Evolution processes, Legacy systems, Software	
	maintenance	
	Project management: Risk management, Managing people, Teamwork	

#### **Course Outcomes**

Learne	Learner will be able to		
CO1	Apply the software engineering life cycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment.		
CO2	Work in one or more significant application domains		
CO3	Describe the role of SDLC in project development		
CO4	Generate project schedule.		
CO5	Use the techniques and tools necessary for engineering practice.		

Bool	ks and References:				
Sr.	Title	Author/s	Publisher	Edition	Year
No.					
1.	Software Engineering, edition,	Ian Somerville	Pearson Education.	Ninth	
2.	Software Engineering	Pankaj Jalote	Narosa Publication		
3.	Software engineering, a practitioner's Approach	Roger Pressman	TataMcgraw-hill	Seventh	
4.	Software Engineering principles and practice	WS Jawadekar	Tata Mcgraw-hill		
5.	Software Engineering-A Concise Study	S.AKelkar	PHIIndia.		
6.	Software Engineering Concept and Applications	Subhajit Datta	Oxford Higher Education		
7.	Software Design	D.Budgen	Pearson Education	2nd	
8.	Software Engineering	K L James	PHI	EEE	2009

<b>B.Sc.</b> (Information 7	Technology)	Semester-IV	
Course Name: Software Engineering Practical		Course Code: VSIT	253
Periods per week 1 Period	is 120 minutes	2	
Credits	Credits 2		
		Hours	Marks
Evaluation System	Practical Examination	2	100

List of	Practical(To be executed using Star UML or any similar software)
1.	Study and implementation of class diagrams.
2.	Study and implementation of Use Case Diagrams.
3.	Study and implementation of Entity Relationship Diagrams.
4.	Study and implementation of Sequence Diagrams.
5.	Study and implementation of State Transition Diagrams.
6.	Study and implementation of Data Flow Diagrams.
7.	Study and implementation of Collaboration Diagrams.
8.	Study and implementation of Activity Diagrams.
9.	Study and implementation of Component Diagrams.
10.	Study and implementation of Deployment Diagrams.

Books and References:						
Sr.	Title	Author/s	Publisher	Edition	Year	
No.						
1.	Object-Oriented Modeling	Michael Blaha, James	Pearson		2011	
	and Design	Rumbaugh				

B. Sc. (Information Technology)		Semester – IV	
Course Name: Computer O	riented Statistical	Course Code: VSIT254	
Techniques			
Periods per week (1 Period	is 60 minutes)	2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal		40

Cours	e Objective
To ma	ke learner understand and use
1.	Concept of various measures of Central tendency (Mean, Median Mode), Dispersion
	(Range, Standard deviation, variance).
2.	Concept of moment, skewness and Kurtosis along with elementary probability and
	sampling theory.

- 3. Statistical Estimation and Decision theory .
- 4. The sampling theory and testing of hypothesis and making inferences and Chi-Square test.
- 5. Concept of Regression and Correlation Theory.

Unit	Details	Lectures
I	The Mean, Median, Mode, and Other Measures of Central Tendency: Index, or Subscript, Notation, Summation Notation, Averages, or Measures of Central Tendency ,The Arithmetic Mean , The Weighted Arithmetic Mean , Properties of the Arithmetic Mean , The Arithmetic Mean Computed from Grouped Data , The Median, The Mode, The Empirical Relation Between the Mean, Median, and Mode, The Geometric Mean G, The Harmonic Mean H ,The Relation Between the Arithmetic, Geometric, and Harmonic Means, The Root Mean Square, Quartiles, Deciles, and Percentiles, Software and Measures of Central Tendency. The Standard Deviation and Other Measures of Dispersion: Dispersion, or Variation, The Range, The Mean Deviation, The Semi- Interquartile Range, the 10–90 Percentile Range, The Standard Deviation, The Variance, Short Methods for Computing the Standard Deviation, Properties of the Standard Deviation, Empirical Relations Between Measures of Dispersion, Absolute and Relative Dispersion; Coefficient of Variation Moments, Skewness, and Kurtosis: Moments, Moments for Grouped Data, Relations Between Moments, Computation of Moments for Grouped Data,	10

	Skewness, and Kurtosis, Software Computation of Skewness and Kurtosis					
	Elementary Sampling Theory : Sampling Theory, Random Sa ples and					
	Random Numbers, Sampling With and Without Replacement, Sampling					
	Distributions, Sampling Distribution of Means, Sampling Distribution of					
	Proportions, Sampling Distributions of Differences and Sums, Standard					
	Errors, Software Demonstration of Elementary Sampling Theory.					
	Statistical Estimation Theory: Estimation of Parameters. Unbiased	10				
	Estimates, Efficient Estimates, Point Estimates and Interval Estimates; Their					
п	Reliability Confidence-Interval Estimates of Population Parameters					
	Probable Error.					
	Probable Error. Statistical Decision Theory: Statistical Decisions Statistical Hypotheses					
	Tests of Hypotheses and Significance, or Decision Rules, Type I and Type II					
	Errors, Level of Significance, Tests Involving Normal Distributions, Two-					
	Errors, Level of Significance, Tests Involving Normal Distributions, Two- Tailed and One-Tailed Tests, Special Tests, Operating-Characteristic Curves.					
	the Power of a Test, p- Values for Hypothesis Tests, Control Charts, Tests					
	Involving Sample Differences, Tests Involving Binomial Distributions					
	Small Sampling Theory: Small Samples, Student's t Distribution,					
	Confidence Intervals, Tests of Hypotheses and Significance, The Chi-Square					
	Distribution, Confidence Intervals for Sigma , Degrees of Freedom, The F					
	Distribution.	10				
	The Chi-Square Test: Observed and Theoretical Frequencies, Definition of					
	chi-square, Significance Tests, The Chi-Square Test for Goodness of Fit					
	Contingency Tables, Yates' Correction for Continuity, Simple Formulas for					
	Computing chi-square, Coefficient of Contingency, Correlation of Attributes,					
III	Additive Property of chi-square.					
	Correlation Theory: Correlation and Regression, Linear Correlation,					
	Measures of Correlation, The Least-Squares Regression Lines, Standard Error					
	of Estimate, Explained and Unexplained Variation, Coefficient of Correlation,					
	Remarks Concerning the Correlation Coefficient, Product-Moment Formula					
	for the Linear Correlation Coefficient, Short Computational Formulas,					
	Regression Lines and the Linear Correlation Coefficient, Correlation of Time					
	Series, Correlation of Attributes, Sampling Theory of Correlation, Sampling					
-						

Course	Course Outcome		
Learner	will be able to		
CO1	Understand measures of central tendencies with the help of R programming.		
CO2	Learn Moments, skewness, kurtosis and importing data in R with the help of		
	Excel/CSV file		
CO3	Learn Regression, correlation Theory and small sampling theory.		
CO4	Understand statistical estimation with the help of hypothesis		
CO5	Understand concepts of hypothesis, power of test, critical region		

Books	Books and References:				
Sr.	TitleAuthor/sPublisherEdit		Editio	Yea	
No.				n	r
1.	A text book of	P. N. Wartikar and J.	Pune Vidyathi		
	Applied	N. Wartikar	Graha		
	Mathematics Vol I				
2.	Statistics	Murray R. Spiegel,	Mcgraw –Hill	Fourth	
		Larry J. Stephens.	Iternational		
3.	A Practical	R.B. Patil,	SPD	1	2017
	Approach using R	H.J. Dand and			
		R. Bhavsar			
4.	Fundamental of	S.C. GUPTA and	Sultan Chand	Eleven	2011
	Mathematical	V.K. KAPOOR	And Sons	th	
	Statistics			revised	
5.	Mathematical	J.N. Kapur and H.C.	S. Chand	Twenti	2005
	Statistics	Saxena		eth	
				revised	

B. Sc. (Information Technology)		Semester – IV	
Course Name: Compute	er Oriented Statistical	Course Code: VSIT255	
<b>Techniques Practical</b>			
Periods per week 1 Peri	iod is 120 minutes	2	
Credits			2
		Hours	Marks
Evaluation System	Practical Examination	2	100
	Internal	-	-

List of	Practical	
1.	Using R execute the basic commands, array, list and frames.	
2.	Create a Matrix using R and Perform the operations addition, inverse, transpose and	
	multiplication operations.	
3.	Using R Execute the statistical functions: mean, median, mode, quartiles, range, inter	
	quartile range histogram	
4.	Using R import the data from Excel / .CSV file and perform the above functions.	
5.	Using R import the data from Excel / .CSV file and calculate the standard deviation,	
	variance, co-variance.	
6.	Using R import the data from Excel / .CSV file and draw the skewness.	
7.	Import the data from Excel / .CSV and perform the hypothetical testing.	
8.	Import the data from Excel / .CSV and perform the Chi-squared Test.	
9.	Using R perform the binomial and normal distribution on the data.	
10.	Perform the Linear Regression using R.	
11.	Compute the Least squares means using R.	
12.	Compute the Linear Least Square Regression	

B. Sc (Information	n Technology)	Semester – IV	
Course Name: Law and Cyberspace- II		Course Code: VFLC280	
Periods per week (1 Per	riod is 60 minutes)	2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2 60	
	Internal		40

Course Objective		
To make learner understand		
1. E-Commerce Taxation real world problems, Concept of Permanent Establishment		
2. Digital Signatures, Digital Signature Certificate.		
3. Concept of copyright.		
4. Status of Electronic Records as evidence, The Indian Evidence Act of 1872 and The		
Information Technology Act, 2000.		
5 Directorian of Cuber Consumers in India		

5. Protection of Cyber Consumers in India.

Unit	Details	Lectures
	E-Commerce Taxation: Real Problems in the Virtual World: A Tug of	
	War on the Concept of Permanent Establishment", Finding the PE in Cross	
	Border E-Commerce, The United Nations Model Tax Treaty, The Law of	10
	Double Taxation Avoidance Agreements and Taxable Jurisdiction Over Non-	
Ι	Residents, Under the Income Tax Act, 1961, Tax Agents of Non-Residents	
	under the Income Tax Act, 1961 and the Relevance to E-Commerce, Source	
	versus Residence and Classification between Business Income and Royalty,	
	The Impact of the Internet on Customer Duties, Taxation Policies in India: At	
	a Glance.	
	The Indian Evidence Act of 1872 v. Information Technology Act, 2000:	
	Status of Electronic Records as Evidence, Proof and	
	Management of Electronic Records; Relevancy, Admissibility and	
п	Probative Value of E-Evidence, Proving Digital Signatures, Proof of	10
11	Electronic Agreements, Proving Electronic Messages, Other	10
	Amendments in the Indian Evidence Act by the IT Act, Amendments to the	
	Bankers Books, EvidenceAct,1891 and Reserve Bank of India	
	Act, 1934.	

	Digital Signature, Certifying Authorities and E-Governance:			
	Digital Signatures, Digital Signature Certificate, Certifying			
	Authorities and Liability in the Event of Digital Signature Compromise, E-			
ттт	Governance in India: A Warning to Babudom!	10		
111	Protection of Cyber Consumers in India: Are Cyber Consumers Covered			
	Under the Consumer Protection Act? Goods and Services, Consumer			
Complaint, Defect in Goods and Deficiency in Services, Restrictive and				
	Unfair Trade Practices, Instances of Unfair Trade			

**Course Outcome** 

Learner will achieve competency in

**CO1** Knowledge of E-taxation, The Law of Double Taxation Avoidance Agreements.

**CO2** Understanding the evidence in E-world

**CO3** Knowledge of Digital signature, certifying authority.

- **CO4** Identifying the Certifying Authorities and Liability in the Event of Digital Signature compromise.
- **CO5** Protection of cyber consumers.

#### **Books and References:**

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Cyber Law Simplified	Vivek Sood	TMH	First	2001
			Education	edition	
2.	Cyber Laws	Pavan Duggal	Universal's	3 <sup>rd</sup>	2023
				edition	
3.	The Information technology	Professional's	Professional's	Latest	2024
	Act, 2000 -Bare act 2024		Book	2024	
			publisher		

B. Sc. (Information	Technology)	Semester – IV	
Course Name: Advanced	Web Programming	Course Code: VSIT256	
Practical			
Periods per week (1 Perio	d is 120 minutes)	2	
Credits			2
		Hours	Marks
Evaluation SystemPractical Examination		3	100
	Internal	-	-

Course Objective
To make learner aware of
1. Basics of C# fundamentals.
2. Web form fundamentals and various form controls used in website development using C#.
3. M aster page and content page development in C#.
4. ADO.NET fundamentals, data binding and data controls.

5. ASP.NET AJAX basics.

### List of Practical

1.	Working with basic C# :	
	The C# Language: C# Language Basics, Variables and Data Types, Variable Operations,	
	Object-Based Manipulation, Conditional Logic, Loops, Methods, Class and Object.	
a.	Create an application that obtains four int values from the user and displays the product.	
b.	Create an application to demonstrate string operations.	
d.	Create an application to demonstrate following operations	
	i. Generate Fibonacci series. ii. Test for prime numbers.	
	iii. Test for vowels. iv. Use of foreach loop with arrays	
	v. Reverse a number and find sum of digits of a number.	
e.	Create an application to demonstrate method overloading.	
2.	Working with Object Oriented C#:	
	Constructor, destructor, Inheritance and all types of inheritance, use of Interface.	
a.	Create simple application to perform following operations	
	i. Finding factorial Value ii. Money Conversion iii. Temperature Conversion	
b.	Create simple application to demonstrate use of following concepts	
	i. Inheritance (all types) ii. Constructor overloading iii. Interfaces	
3.	Working with Web Forms and Controls(Part A):	
	Web Form Fundamentals: Writing Code, Using the Code-Behind Class, Adding Event	
	Handlers, Introducing Server Controls with its properties, AutoPostBack mechanism.	

a.	Create a simple web page with sever controls TextBox, Button, Label, CheckBox,		
	RadioButton to demonstrate setting and use of their properties)		
b.	Create a simple web page with sever controls ListBox, DropDownList, CheckBoxList,		
	RadioButtonList, BulletedList to demonstrate setting and use of their properties)		
c.	Create a web page with sever controls TextBox, Label, and implement AutoPostBack		
	property of TextBox.		
d.	Create a web page which shows the implementation of SelectedIndex, SelectedValue,		
	SelectedItem properties for ListBox control and Dropdownlist control.		
e.	Create a simple web page with sever controls ListBox, DropDownList to demonstrate the		
	event handlers for both the controls.		
4.	Working with Web Forms and Controls(Part B):		
	Rich Controls: Calendar Control, AdRotator Control, User Control.		
a.	Create Web Form to demonstrate use of Adrotator Control.		
b.	Create Web Form to demonstrate use User Controls.		
c.	Demonstrate the use of Calendar control to perform following operations.		
	a) Display messages in a calendar control b) Display vacation in a calendar		
	control		
	c) Selected day in a calendar control using style		
5.	Working with Navigation and Validation Control		
	Understanding Validation Control and Navigation Control with all its types with its		
	properties.		
a.	Create Web Form to demonstrate use of Website Navigation controls and Site Map.		
b.	Create a Registration form to demonstrate use of various Validation controls.		
6.	Working with Database:		
	ADO.NET Fundamentals: Understanding Databases, Configuring Your Database,		
	Understanding SQL Basics, Understanding the Data Provider Model, Using Direct Data		
	Access, Using Disconnected Data Access.		
a.	Create a web application which shows the use of connected architecture.		
	Create a web application which shows the use of connected architecture.		
b.	Create a web application for to display the phone no of an author using database.		
c.	Create a web application for inserting and deleting record from a database. (Using		
	Execute- Non Query).		
7.	Working with data binding :		
	Introducing Data Binding, Using Single-Value Data Binding, Using Repeated-Value Data		
	Binding, Working with Data Source Controls		
a.	Create a web application to demonstrate Single Value Data Binding.		
b.	Create a web application to demonstrate Repeated-Value Data Binding using dropdown		
	list control.		

с.	Create a web application to demonstrate Repeated-Value Data Binding with a Dictonary
	Collection.
d.	Create a web application to demonstrate various uses and properties of SqlDataSource.
с.	Create a web application to display Using Disconnected Data Access and Databinding
	using GridView.
8.	Working with Data controls:
	The GridView, Formatting the GridView, selecting a GridView Row, Editing with the
	GridView, Sorting and Paging the GridView, Using GridView Templates, The
	DetailsView and FormView
a.	Create a web application to demonstrate use of GridView control template and GridView
	hyperlink.
b.	Create a web application to demonstrate use of GridView button column and GridView
	events.
c.	Create a web application to demonstrate GridView paging and Creating own table format
	using GridView.
9.	Working with Database more examples.
a.	Create a web application bind data in a multiline textbox by querying in another textbox.
b.	Create a web application to display records by using database.
с.	Demonstrate the use of Datalist link control.
10.	Working with AJAX and Master Page:
	Understanding Ajax, Using Partial Refreshes, Using Progress Notification, Implementing
	Timed Refreshes, Working with the ASP.NET AJAX ControlToolkit.
	Master Page: What is Master Page, Content Page , How to create Master Page and Content
	Page.
a.	Create a web application to demonstrate use of Master Page with applying Styles and
	Themes for page beautification.
b.	Create a web application to demonstrate use of various Ajax controls.
c.	Create a web application to demonstrate use of various ASP.NET AJAX ControlToolkit.
11.	Programs to create and use DLL

Cours	Course Outcome		
Learn	Learner will be able to		
CO1	Design a responsive web site and demonstrate Rich Internet Application.		
CO2	Implement interactive and dynamic web page(s) using HTML,CSS,AJAX, C# with .NET		
	concepts.		
CO3	Develop database oriented web application.		
CO4	Build dynamic web site using server side .NET technology and Database connectivity.		
CO5	Motivate to work as website developer.		

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Beginning ASP.NET	Matthew MacDonald	Apress		2012
	4.5 in C#				
2.	C# 2015	Anne Bohem and Joel	Murach	Third	2016
		Murach			
3.	Murach"s ASP.NET 4.6Web	Mary Delamater and Anne	SPD	Sixth	2016
	Programming in C#2015	Bohem			
4.	ASP.NET4.0	J. Kanjilal	Tata McGraw-		2011
	programming		Hill		
5.	Programming ASP.NET	D.Esposito	Microsoft		2011
			Press		
			(Dreamtech)		
6.	Beginning Visual C#2010	K. Watson, C. Nagel,	Wrox (Wiley)		2010
		J.H Padderson, J.D.			
		Reid, M.Skinner			

# **Evaluation scheme**

(Major, Minor, VSEC)

#### 1. Internal examination: (40marks)

#### Test: 1 Class test of 15 marks. (Can be taken online)

Q	Attempt <i>any three of</i> the following:	15
a.		
b.		
c.		
d.		
e.		
f.		

i.

i.

15 marks project/presentation/assignment.

ii. 10 marks: Active participation in the class, overall conduct, attendance.

#### 2. External Examination: (60marks)

	All questions are compulsory	
Q1	(Based on Unit 1) Attempt <u>any four</u> of the following:	20
a.		
b.		
C.		
d.		
e.		
f.		
Q2	(Based on Unit 2) Attempt <u>any two</u> of the following:	20
Q3	(Based on Unit 3) Attempt <u>any two</u> of the following:	20

#### 3. Practical Exam: 100marks

A Certified copy journal is essential to appear for the practical examination.

1.	Practical Question 1	40
2.	Practical Question 2	40
3.	Journal	10
4.	Viva Voce	10
	OR	

1	Practical Question	80
---	--------------------	----

2.	Journal	10
3.	Viva Voce	10

# **Evaluation scheme**

(AEC)

#### 1. Internal Evaluation (50 marks).

i.

i.

#### Test: 1 Class test of 15 marks. (Can be taken online)

Q	Attempt <i>any three of</i> the following:	15
a.		
b.		
c.		
d.		
e.		
f.		

- i. 25 marks project/presentation/assignment.
- ii. 10 marks: Active participation in the class, overall conduct, attendance.

# **Evaluation scheme**

(CC)

#### <u>1.</u> Internal Evaluation (50 marks).

#### Test: 1 Class test of 15 marks. (Can be taken online)

Q	Attempt <i>any three of</i> the following:	15
a.		
b.		
c.		
d.		
e.		
f.		

ii.

35 marks Activities

d

KET's V. G. Vaze College(Autonomous) Board of Studies – Information
Technology comprised of following members
Mrs. Pournima P Bhangale, Chairperson
Mrs. Vandana Y Kadam, Member
Mrs. Rakhee D Rane, Member
Mrs. Nanda A Rupnar, Member
Ms. Mohini Bhole, Member
Ms. Pranali Pawar, Member
Dr. Hiren Dand, VC nominee
Professor (Dr.) Ajay S Patil, Subject Expert, North Maharashtra University
Mr. Milind Narayan Kolambe, Subject Expert, Pune University
Mr. Tejpal Khachane, Industry Expert
Mr. Abhishek Ghorpade, Postgraduate meritorious alumnus

ownimal

Mrs. Pournima P Bhangale Chairperson

- A our

Dr. Hiren Dand VC Nominee