Bachelor of Science (Computer Science)
B.Sc. (CS) Semester VI

Course Code	**US06CCSC51**	Title of the Course	**Object Oriented Programming Using Java**
Total Credits of the Course	4	Hours per Week	4

Course Objectives:	1. To introduce basic programming concepts and necessary constructs of the Java programming language. 2. To understand the fundamental concepts of object-oriented programming using Java. 3. To describe exception handling and Input Output Management. 4. To understand Applet programming using Abstract Windows Toll Kit (AWT).

Course Content		
Unit	Description	Weightage* (%)
1.	**Introduction** – History of Java, features, the Java environment, the Java Virtual Machine (JVM) – Structure of a Java program, a simple Java program, implementing a Java program – Tokens, comments, constants, variables and data types – Scope of variables, type casting – Operators: arithmetic, relational, logical, assignment, increment/decrement, conditional, ternary operator & special operators – Decision making: if statement, if…else statement, nesting of if…else, the else if ladder, switch statement – Looping: while, do…while, for, for each loop jumps in loops, labeled loops	25
2.	**Arrays, Classes, Objects, Interfaces and Inheritance** – Arrays: one, two dimensional arrays – Defining a class, members of a class: variables and methods, creating objects, constructors, accessing class members – Static members v/s instance members – Introduction to inheritance, super keyword – Interfaces: Introduction, definition, extending, implementing & accessing – Final variables, methods and classes, abstract methods and classes – Introduction to method overloading and overriding	25

3.	**Exception Handling, I/O Management and Packages** – Managing errors & exceptions: introduction, types of errors, exceptions, syntax of exception handling construct, multiple catch statements, the finally clause, defining and throwing user-defined exceptions, the throw statement – Managing I/O files: introduction, concept of streams, Character stream classes – Introduction to the concept of package, Java API packages, using the System package – Using java. lang (String, Math)	25
4.	**Applet Programming** – Applet architecture and skeleton – java.awt package (Button, CheckBox, CheckBoxGroup, Choice, Color, Label, List, TextArea, TextField) – HTML applet tag, display techniques (DrawString, Lines, Rectangle, Ellipses, Circles, Arcs, Polygons, Color) – Introduction to event handling	25

Teaching-Learning Methodology	Blended learning approach incorporating both traditional classroom teaching as well as usage of ICT tools

Evaluation Pattern		
Sr. No.	Details of the Evaluation	Weightage
1.	Internal Written / Practical Examination (As per CBCS R.6.8.3)	15%
2.	Internal Continuous Assessment in the form of Practical, Viva-voce, Quizzes, Seminars, Assignments, Attendance (As per CBCS R.6.8.3)	15%
3.	University Examination	70%

Course Outcomes: Having completed this course, the learner will be able to develop	
1.	Understanding of the basic programming concepts and necessary constructs of the Java programming language.
2.	Understanding of the fundamental concepts of object-oriented programming using Java.
3.	Ability to describe exception handling and Input Output Management.
4.	Understanding of Applet programming using Abstract Windows Toll Kit (AWT)

Suggested References:	
Sr. No.	References
1.	Programming with Java- A Primer by E. Balaguruswami, 3rd Edition, TMH Publication.
2.	The Complete Reference – Java 2 7th Edition Herbert Schildt. TMH Publication.
3.	Saba Zame , Handbook of Object technology, CRC Press, Washington DC, 1999.
4.	Mary Campion and Kathy Walrath, Java tutorial, Second Edition, Addison Wesley Pun. 1998.
5.	Java 2 Programming Black Book, Steven Holzner.

On-line resources to be used if available as reference material
On-line Resources
1. https://www.tutorialspoint.com/
2. https://www.w3schools.com/
3. https://www.javatpoint.com/

*****

Bachelor of Science (Computer Science)
B.Sc. (CS) Semester VI

Course Code	**US06CCSC52**	Title of the Course	**PHP Using MySql**
Total Credits of the Course	4	Hours per Week	4

Course Objectives:	1. To introduce students to fundamental concepts related to PHP programming. 2. To impart basic knowledge of working with advanced features of PHP and interaction with forms. 3. To provide basic understanding of database access.

Course Content		
Unit	Description	Weightage* (%)
1.	**PHP Programming Concepts -1** – Introduction to open source – Advantage and capabilities of open source – Introduction to apache – Introduction to PHP: features, adding PHP to HTML – PHP Variables – Static & global variables – GET & POST method – PHP Operators – Conditional Structure & Looping Structures – User Defined Functions – Arrays – Strings and its functions	25
2.	**PHP Programming Concepts-2** – Site structure and basics of web site development using PHP – PHP and OOP templates – Error Handling – Authentication – Cookies and Session Management – Browser detection – Sending MIME Mail Message with Mail_mime, smtp	25

3.	**Working with images, pdf files, ajax, XML, CSS and Bootstrap** – Creating and Manipulating images – Using Text in Images – Creating database driven graph – Saving and building on existing image. – Generate PDF file. – PHP with XML – PHP with Ajax – PHP with CSS and Bootstrap	25
4.	**Accessing Database** – Working with MySQL using PhpMyAdmin – PHP-MySQL Connectivity – PHP-MySQL Functions	25

Teaching-Learning Methodology	Blended learning approach incorporating traditional classroom teaching and online /ICT-based teaching practices.

Evaluation Pattern		
Sr. No.	Details of the Evaluation	Weightage
1.	Internal Written / Practical Examination (As per CBCS R.6.8.3)	15%
2.	Internal Continuous Assessment in the form of Practical, Viva-voce, Quizzes, Seminars, Assignments, Attendance (As per CBCS R.6.8.3)	15%
3.	University Examination	70%

Course Outcomes: Having completed this course, the learner will be able to develop	
1.	Understanding of the fundamental concepts related to PHP programming.
2.	Basic knowledge of working with advanced features of PHP and interaction with forms.
3.	Understanding of database access in PHP.

Suggested References:	
Sr. No.	References
1.	Essential PHP Tools Modules, extensions and Accelerators–David Sklar–APRESS (SPD)
2.	PHP advance for the World Wide Web – Larry Edward Ullman – peachpit press
3	Advance PHP for Web professionals – Christopher Cosentino – Pearson education
4	Expert PHP 5 Tools – Dirk Merkel – PACKT(SPD)
5	Learning PHP 5 – David Sklar –O'Reilly (SPD)
6	Beginning PHP 5.1 For Begginers – Iyan Byross, Sharanam Shah- The Team (SPD)

*****

Bachelor of Science (Computer Science)
B.Sc. (CS) Semester VI

Course Code	**US06CCSC53**	Title of the Course	**Python Programming**
Total Credits of the Course	4	Hours per Week	4

Course Objectives:	1. To learn the fundamentals of the Python programming language. 2. To study the concepts of object-oriented programming in Python. 3. To learn exception handling and file handling in Python. 4. To understand how to access files and databases from Python

Course Content		
Unit	Description	Weightage* (%)
1.	**Introduction & Basic Concept of Python** – Python Introduction – Python Features – Python Applications – Introduction to Python Libraries (PyTorch, Pandas, NumPy, Scikit-learn, TensorFlow, Keras, etc.) – Data type, Variable, keywords, literals, operators, comments – Flow control statements. If, Switch, while, for, do…while – Break, continue and pass statements	25
2.	**Inbuilt & UDF Functions** – Strings (String Operations & Functions) – Lists (List Operations & Functions) – Tuples (Tuple Operations & Functions) – Sets (Set Operations & Functions) – Dictionary (Dictionary Operations & Functions) – Functions (Built-in, User-define, Lambda)	25
3.	**File I/O Handling** – FILE & I/O Handling – Modules – Exceptions – Date – Regex – Read & Write CSV file – Use of different kind of modules	25

4.	**Working with Database** – Environment Setup – Database Connection – Creating new Database – Creating Tables – Working with table (Insert operations, Read operations, Update Operations, Join Operations) – Transaction	25

Teaching-Learning Methodology	Blended learning approach incorporating traditional classroom teaching and online /ICT-based teaching practices.

Evaluation Pattern		
Sr. No.	Details of the Evaluation	Weightage
1.	Internal Written / Practical Examination (As per CBCS R.6.8.3)	15%
2.	Internal Continuous Assessment in the form of Practical, Viva-voce, Quizzes, Seminars, Assignments, Attendance (As per CBCS R.6.8.3)	15%
3.	University Examination	70%

Course Outcomes: Having completed this course, the learner will be able to develop	
1.	Ability to develop computer programs using the Python programming language.
2.	Knowledge of manipulating different Python data types.
3.	Ability to develop object-oriented programs using Python.
4.	Basic knowledge of exception handling, file handling and database access in Python.

Suggested References:	
Sr. No.	References
1.	Paul A. DeBarry: Head First Python, 2010, O'Reilly Media, Inc.
2.	Martin C. Brown: The Complete Reference Python, McGraw Hill
3	David M. Beazley: Python Essential Reference, Pearson Addison-Wesley Professional
4	Exploring Python Book by Timothy Budd
5	Head First Python: A Brain-Friendly Guide by Aaul Barry
6	Introducing Python-Modern Computing in Simple Packages –Bill Lubanovic, O'Reilly Publication
7	Introduction to Computer Science Using Python-Charles Dierbach, Wiley Publication Learning with Python ", Green Tea Press, 2002
8	Beginning Programming with Python for DummiesPaperback–2015 by John Paul Mueller

On-line resources to be used if available as reference material
On-line Resources
Python Tutorial/Documentation www.python.or 2010 http://docs.python.org/3/tutorial/index.html http://www.javapoint.com/python-tutorial

*****

Bachelor of Science (Computer Science)
B.Sc. (CS) Semester VI

Course Code	**US06CCSC54**	Title of the Course	**Software Engineering**
Total Credits of the Course	4	Hours per Week	4

Course Objectives:	1. To study the fundamental concepts related to software engineering, different phases of software development and various process models. 2. To learn the basic concepts related to requirement specification and software project planning. 3. To acquire basic knowledge about the concepts related to system design, coding and testing

Course Content		
Unit	Description	Weightage* (%)
1.	**Introduction** – Introduction: Software and Software Engineering – General Characteristics of Software Process – Phases in Software development – Effort and Error Distribution – Process Models: Waterfall, Prototype, Iterative enhancement, spiral	25
2.	**Requirement Specification and Software Project Planning** – Introduction: Software Requirement Specification (SRS) and Needs – Problem Analysis - Structuring Information – Software Requirement Specifications (SRS), Characteristics and Components of SRS – Specification language (Structured English, Regular Expression and Decision Table) – Introduction: Software Projects, Planning, Categories of Software projects – Project Monitoring Plan: Time sheets, Reviews, Cost- schedule milestone and Earned value method – Software Quality Assurance Plans (SQAP) – Overview of Risk Management	25
3.	**System Design, Coding and Testing** – Introduction: System Design – Design Objectives and Design Principles – Design Concepts - Top down and Bottom up approach, Problem	25

	Partition, Abstraction, Modularity, Module Level concept, Coupling, Cohesion – Overview of structured design – Introduction: Coding, Top Down and Bottom Up approach for coding – Structured programming, Information Hiding – Programming style, Internal documentation – Introduction: Testing, Error, Fault, Failure & Reliability – Testing process, Top down and bottom up approach for testing – Levels of Testing – Functional Testing v/s. Structural testing	
4.	**The Concepts of a System, System Analysis and Development Life Cycle (SDLC)** – The concept of a system – The elements and characteristics of a system – Types of systems – Meaning of systems analysis – Role of a systems analyst – SDLC - Introduction – Stages of SDLC	25

Teaching-Learning Methodology	Blended learning approach incorporating both traditional classroom teaching as well as usage of ICT tools.

Evaluation Pattern		
Sr. No.	Details of the Evaluation	Weightage
1.	Internal Written / Practical Examination (As per CBCS R.6.8.3)	15%
2.	Internal Continuous Assessment in the form of Practical, Viva-voce, Quizzes, Seminars, Assignments, Attendance (As per CBCS R.6.8.3)	15%
3.	University Examination	70%

Course Outcomes: Having completed this course, the learner will be able to develop	
1.	understanding of the fundamental concepts related to software engineering, different phases of software development and various process models.
2.	understanding of the basic concepts related to requirement specification and software project planning.
3.	basic knowledge about the concepts related to system design, coding and testing.

Suggested References:	
Sr. No.	References
1.	An Integrated Approach to Software Engineering by PankajJalote ,Narosa Publishing House, Second Edition,1997.
2.	Software Engineering a practitioner's approach by Roger S. Pressman, Tata McGraw-Hill, Fifth Edition, 2001.
3.	Software Engineering Fundamentals by Richard Fairley, Tata McGraw- Hill.
4.	Software Engineering, By Ian Somnmerville, Addition-Wesley, Fifth Edition, 2000.
5.	S. Parthasarthy & B. W. Khalkar: System Analysis & Design, 1st Edition, Master Ed. Cons., Nashik.
6.	James A. Sen: Analysis & Design of Information System 2nd Edition, McGraw-Hill Int.

On-line resources to be used if available as reference material
On-line Resources
1. https://www.tutorialspoint.com/
2. https://www.w3schools.com/
3. https://www.javatpoint.com/

******

Bachelor of Science (Computer Science)
B.Sc. (CS) Semester-VI

Course Code	**US06CCSC55**	Title of the Course	**Praticals**
Total Credits of the Course	8	Hours per Week	16

Course Objectives:	1. To study the concepts of object-oriented programming concepts using Java. 2. To learn PHP web developments scripting language. 3. To learn Python Programming Language. 4. To learn the implementation of Software Development process model through the case study.

Course Content		
	Description	Weightage (%)
	Part-1: Practical based on US06CCSC51 Part-2: Practical based on US06CCSC52 Part-3: Practical based on US06CCSC53 Part-4: Practical based on US06CCSC54	100%

Teaching-Learning Methodology	Hands on training through ICT tools.

Evaluation Pattern		
Sr. No.	Details of the Evaluation	Weightage
1.	Internal Written / Practical Examination (As per CBCS R.6.8.3)	30%
2.	Internal Continuous Assessment in the form of Practical, Viva-voce, Quizzes, Seminars, Assignments, Attendance (As per CBCS R.6.8.3)	-
3.	University Examination	70%

Course Outcomes: After completing this course, the learner will be able to gain	
1.	knowledge of object-oriented programming concepts using Java.
2.	knowledge of PHP web development scripting language.
3.	knowledge of Python Programming Language.
4.	knowledge of analyzing system through the case study.

*** * * ***