

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)**

**Under Faculty of Science**

Semester – V	Paper – I
Course Code: BSC-CS 501 T	Title of the Course: Operating System
Credits: 02 Credits	Total Lectures: 32Hrs.

**Course Outcomes (COs):**

On completion of the course, student will be able to–

- concept of operating system and its principle
- Processes and Thread Scheduling by operating system
- Synchronization in process and threads by operating system
- Memory management by operating system using with the help of various schemes

**Detailed Syllabus:**

**Unit I: Introduction to Operating Systems**

**(06)**

- 1.1 Types of program -  
System program and Application program,
- 1.2 Difference between system programming and application programming.
- 1.3 Operating Systems Overview- system Overview and Functions of operating systems
- 1.4 Operating system structure
- 1.5 Booting
- 1.6 System call in operating system
  - 1.6.1 What is system call?
  - 1.6.2 Types of System calls
    - Process Control –fork (), exit (), wait ()
    - File manipulation –open (), read (), write (), close ()
    - Device management-Ioctl(), read(),write()

**Unit II: Processes and Threads**

**(6)**

- 2.1 Process Concept – The processes, Process states, Process control block.
- 2.2 Process Scheduling – Scheduling queues, Schedulers, context switch
- 2.3 Operations on Process – Process creation with program using fork(), Processtermination

2.4 Thread Scheduling- Threads, benefits, Multithreading Models, Thread Libraries

**Unit III: Process Scheduling****(7)**

- 3.1 Basic Concept – CPU-I/O burst cycle
- 3.2 Scheduling Criteria
- 3.3 CPU scheduler,
- 3.4 Preemptive scheduling
- 3.5 Dispatcher
- 3.6 Scheduling Algorithms – FCFS, SJF, Priority scheduling, Round-robin scheduling

**Unit IV: Process Deadlock****(5)**

- 4.1 What is Deadlock?
- 4.2 Deadlock Characterization – Necessary conditions, Resource allocation graph
- 4.3 Deadlock Methods- Prevention and Deadlock Avoidance - Safe state, Resource allocation graph algorithm, Banker's Algorithm
- 4.4 Deadlock Detection
- 4.5 Recovery from Deadlock – Process termination, Resource preemption

**Unit V: Memory Management****(8)**

- 5.1 Background – Basic hardware, Address binding, Logical versus physical address space, Dynamic loading, Dynamic linking and shared libraries
- 5.2 Swapping
- 5.3 Contiguous Memory Allocation – Memory mapping and protection, Memory allocation, Fragmentation
- 5.4 Paging – Basic Method, Hardware support, Protection, Shared Pages
- 5.5 Segmentation
- 5.6 Virtual Memory Management – Background, Demand paging, Performance of demand paging, Page replacement – FIFO, Optimal, LRU, MFU

**Suggested Readings:**

1. Operating System Concepts, Avi Silberschatz, Peter Galvin, Greg Gagne, Student Edition, Wiley Asia
2. Operating Systems: Internals and Design Principles, William Stallings, Prentice Hall of India.
3. Advanced Concepts in Operating Systems, M Singhal and NG Shivaratri, Tata McGraw Hill Inc, 2001
4. The 'C' Odyssey, UNIX-the open boundless C, Meeta Gandhi, Tilak Shetty, Rajiv Shah, BPB publication

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – V	Paper – II
Course Code: BSC_CS 502 T	Title of the Course: Internet Programming - I
Credits: 02 Credits	Total Lectures: 30 Hrs.

### Prerequisites

- HTML basics for form designing

### Course Objectives

- To Design dynamic and interactive Web pages.
- To Learn Core-PHP, Server Side Scripting Language
- To Learn PHP-Database handling

### Course Outcomes

- On completion of the course, student will be able to–
- Understand how to develop dynamic and interactive Web Page

### Course Contents

#### Unit I: Overview of HTML, HTTP and PHP

(04)

- 1.1 Overview of HTML and Basic Tags, Creating Forms, Tables, HTML5 Semantics.
- 1.2 CSS basic concept, three ways to use CSS, Box Model, Navigation Bar.
- 1.3 Introduction to Web server and Web browser.
- 1.4 HTTP basics.
- 1.5 PHP Basics: Use of PHP, Lexical structure, Language basics.

#### Unit II: Function and String

(08)

- 2.1 Defining and calling a function Default parameter, Variable parameters, Missing parameters  
Variable function, Anonymous function
- 2.2 Types of strings in PHP
- 2.3 Printing functions
- 2.4 Encoding and escaping, comparing strings, Manipulating and searching strings  
Regular expressions

#### Unit III: Arrays

(06)

- 3.1 Indexed Vs Associative arrays
- 3.2 Identifying elements of an array, Storing data in arrays, Multidimensional arrays  
Extracting multiple values
- 3.3 Converting between arrays and variables, Traversing arrays,
- 3.4 Sorting, Action on entire array

**Unit IV: Files and database handling****(10)**

- 4.1 Working with files and directories
- 4.2 Opening and Closing, getting information about file, Read/write to file, splitting name and path from file, Rename and delete files
- 4.3 Reading and writing characters in file Reading entire file
- 4.4 Random access to file data Getting information on file Ownership and permissions Using PHP to access a database Relational databases and SQL
- 4.5 PEAR DB basics
- 4.6 Advanced database techniques

**Unit V: Files and database handling****(02)**

- 5.1 Email background
- 5.2 Internet mail protocol Structure of an email message
- 5.3 Sending email and validation of Email\_id with php

**Reference Books:**

- HTML & CSS: The Complete Reference, Fifth Edition Author: Thomas A. Powell First published: 01 Jan 2010.
- Programming PHP by Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication
- Beginning PHP 5, Wrox publication, PHP web services, Wrox publication
- Mastering PHP, BPB Publication, PHP cookbook, O'Reilly publication
- PHP for Beginners, SPD publication, Programming the World Wide Web, Robert W Sebesta (3rd Edition)
- HTML 5 Black Book : Covers Css3, JavaScript, XML, XHTML, Ajax, PHP And JQuery by Kogent Learning Solutions Inc., Published November 2011 by Dreamtech Press
- Spurlock Jake, Bootstrap: Responsive Web development. O'Reilly Media, Inc.

**Ref. Links**

- [www.php.net.in](http://www.php.net.in)
- [www.W3schools.com](http://www.W3schools.com)
- [www.wrox.com](http://www.wrox.com)
- <https://coreui.io/docs/layout/grid/#grid-options>
- <https://www.tutorialrepublic.com/twitter-bootstrap-tutorial/bootstrap-grid-system.php>

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

<b>Semester – V</b>	<b>Paper – III</b>
<b>Course Code: BSC_ CS – 503T</b>	<b>Title of the Course: Professional Practice in Python</b>
<b>Credits: 02 Credits</b>	<b>Total Lectures: 30 Hrs.</b>

**Course Objective:**

- Student should be able to develop object oriented Programming logic using python
- Student should able to handle database using python
- Student should able to do network programming in python

**Course Outcomes (COs):**

- Acquire knowledge and oops concept
- Students will learn the Advance concepts of python
- Student will learn Network programming and advance features in Python

**Unit I Object Oriented Concept**

**8 Hrs**

1.1 Python Classes and Object

1.1.1 Constructor

1.1.2 init () method and self-argument

1.1.3 Attributes and instance variable

1.1.4 instance methods ()

1.1.3 Class variables

1.1.4 Class methods ()

1.1.4.1 Class method Decorator

1.1.4.2 Class method as Constructor

1.1.5 Static method

1.2 Encapsulation

1.3 Abstraction

1.4 Naming Convention

1.5 Name Mangling

1.6 Inheritance

1.6.1 Single Inheritance

1.6.2 Multiple Inheritance

1.6.3 Multilevel Inheritance

1.6.4 Hierarchical Inheritance

1.6.5 Hybrid Inheritance

1.7 Polymorphism

- 1.7.1 Definition
- 1.7.2 inbuilt polymorphic functions
- 1.7.3 user-defined polymorphic functions
- 1.7.4 Polymorphism with class methods
- 1.7.5 Polymorphism with Inheritance
- 1.7.6 Polymorphism with a Function and objects
- 1.8 Abstract Class and interface
  - 1.8.1 Abstract Method and Abstract Class
  - 1.8.2 Interfaces in Python
  - 1.8.3 Abstract Classes vs. Interfaces

**Unit II Decorator, Iterator, Generator and comprehensions****4 Hrs**

- 2.1 Decorator
  - 2.1.1 function as object
  - 2.1.2 Pass Function as Argument
  - 2.1.3 Function returning a Function
  - 2.1.4 introduction to python
    - 2.1.3.1 Use of decorator
    - 2.1.3.2 Syntax of decorators
    - 2.1.3.3 decorate a function with argument
    - 2.1.3.3 decorator function returning a value
    - 2.1.3.4 use of built-in decoder wrap
    - 2.1.3.5 chaining decorators
    - 2.1.3.6 decorator with argument
    - 2.1.3.7 Memoization using decorators
- 2.2 Iterator
  - 2.2.1 definition of Iterator
  - 2.2.2 next () method
  - 2.2.3 iter() method
- 2.3 Generator
  - 2.3.1 yield keyword
  - 2.3.2 Generator verses List
- 2.4 comprehensions
  - 2.4.1 List comprehensions
  - 2.4.2 nested list comprehensions
  - 2.4.3 dictionary comprehensions
  - 2.4.3 set comprehensions
  - 2.4.4 Generator comprehensions

**Unit III Multithreading****4Hrs**

- 3.1 Introduction
  - 3.1.1 Use of thread
- 3.2 Thread modules
- 3.3 Methods form thread class
- 3.4 Function from thread class
- 3.5 Thread Local Data
- 3.6 Thread Objects in Python
- 3.7 Python Locks
- 3.8 Python RLocks and Semaphore Locks
- 3.9 Advantages and Disadvantages of Multithreading in Python

**Unit IV Database Connectivity****8 Hrs**

- 4.1 Introduction
  - 4.1.1 Generic database interface
  - 4.1.2 Relational database interface
  - 4.1.3 Benefits of python database programming
- 4.2 Advantages of a DBMS over files
- 4.3 Database Connection
- 4.4 Creating Cursor object
- 4.4 Creating table
- 4.5 Insert, read, update, delete operation
  - execute & close functions
  - reading single & multiple results of query execution
  - executing different types of statements
  - executing transactions
- 4.6 commit and rollback
- 4.7 Stored procedures
- 4.8 Disconnecting Database
- 4.9 Handling Errors
  - understanding exceptions in database connectivity

**Unit V Network programming****8Hrs**

- 5.1 Introduction
- 5.2 Socket
- 5.3 Socket module
- 5.4 Server socket method
- 5.5 Client socket method
- 5.6 General socket method
- 5.7 Creating socket object
- 5.8 Function to setup the sever
- 5.9 Function to create client
- 5.10 creating server-client programs
- 5.11 sending email,
- 5.12 reading from URL
- 5.10 Python internet modules

**Suggested Readings:**

- Learning Python, 5th Edition
- Step By Step Database Programming using Python GUI & MySQL: A practical approach to learn database programming through progressive project-based examples
- Foundations of Python Network Programming, Third Edition

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – V	Paper – IV
Course Code: BSC-CS 504 T	Title of the Course: Foundations of Data Science
Credits: 02 Credits	Total Lectures: 36 Hrs.

**Course Outcomes (COs):**

On completion of the course, student will be able to–

- Perform Exploratory Data Analysis.
- Obtain, clean/process, and transform data.
- Detect and diagnose common data issues, such as missing values, special values, outliers, inconsistencies, and localization.
- Demonstrate proficiency with statistical analysis of data.
- Present results using data visualization techniques.
- Prepare data for use with a variety of statistical methods and models and recognize how the quality of the data and the means of data collection may affect conclusions.

**Detailed Syllabus:**

**Unit I: Introduction to Data Science (06)**

- 1.1 The 3 V's: Volume, Velocity, Variety.
- 1.2 Why learn Data Science?
- 1.3 Applications of Data Science.
- 1.4 The Data Science Lifecycle.
- 1.5 Data Scientist's Toolbox.

**Unit II: Statistical Data Analysis (10)**

- 2.1 Role of statistics in data science
- 2.2 Descriptive statistics
  - 2.2.1 Measuring the Frequency.
  - 2.2.2 Measuring the Central Tendency: Mean, Median, and Mode
  - 2.2.3 Measuring the Dispersion: Range, Standard deviation, Variance, Interquartile Range
- 2.3 Inferential statistics.
  - 2.3.1 Hypothesis testing.
  - 2.3.2 Multiple hypothesis testing.
  - 2.3.3 Parameter Estimation methods.
- 2.4 Measuring Data Similarity and Dissimilarity.
  - 2.4.1 Data Matrix versus Dissimilarity Matrix.
  - 2.4.2 Proximity Measures for Nominal Attributes.
  - 2.4.3 Proximity Measures for Binary Attributes.
  - 2.4.4 Dissimilarity of Numeric Data: Euclidean, Manhattan, and Minkowski



distances.

2.4.5 Proximity Measures for Ordinal Attributes

2.5 Concept of Outlier, types of outliers, outlier detection methods.

**Unit III: Data Preprocessing (10)**

3.1 Data Preprocessing

3.1.1 Data Objects and Attribute Types: What Is an Attribute?

3.1.2 Nominal, Binary, Ordinal Attributes, Numeric Attributes.

3.1.3 Discrete versus Continuous Attributes.

3.2 Data Quality: Why Preprocess the Data?

3.3 Data munging/wrangling operations.

3.4 Cleaning Data.

3.4.1 Missing Values.

3.4.2 Noisy Data: Duplicate Entries, Multiple Entries for a Single Entity, Missing Entries,

NULLs, Huge Outliers, Out-of- Date Data, Artificial Entries.

3.4.3 Irregular Spacings: Formatting Issues - Irregular between Different Tables/Columns, Extra

Whitespace, Irregular Capitalization, Inconsistent Delimiters, Irregular NULL Format,

Invalid Characters, Incompatible Date times.

3.5 Data Transformation.

3.5.1 Rescaling, Normalizing.

3.5.2 Binarizing, Standardizing Label.

3.5.3 One Hot Encoding.

3.6 Data reduction.

3.7 Data discretization.

**Unit IV: Data Visualization (10)**

3.1 Introduction to Exploratory Data Analysis(EDA).

3.1.1 Purpose of Exploratory Data Analysis.

3.2 Data visualization and visual encoding.

3.3 Data visualization libraries.

3.4 Basic data visualization tools.

(Histograms, Bar charts/graphs, Scatter plots, Line charts, Area plots, Pie charts, Donut charts.)

3.5 Specialized data visualization tools.

Boxplots, Bubble plots, Heat map, Dendrogram, Venn diagram, Treemap, 3D scatter plots Advanced data visualization tools- Wordclouds.

3.6 Visualization of geospatial data.

3.7 Data Visualization types.

**Suggested Readings:**

1. Data Science Fundamentals and Practical Approaches, Gypsy Nandi, Rupam Sharma, BPB Publications, 2020.
2. The Data Science Handbook, Field Cady, John Wiley & Sons, Inc, 2017.
3. Data Mining Concepts and Techniques, Third Edition, Jiawei Han, Micheline Kamber, Jian Pei, Morgan Kaufmann, 2012.
4. A Hands-On Introduction to Data Science, Chirag Shah, University of Washington Cambridge University Press

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – V	Paper – V
Course Code: BSC-CS 505T	Title of the Course: Theoretical Computer Science
Credits: 02 Credits	Total Lectures: 36 Hrs.

### Prerequisites

- Mathematical Preliminaries Sets (Subset, Set Operations), Relations (Properties of Relations, Closure of Relations) and Functions
- Discrete Mathematics- Graphs, Trees, Logic and Proof Techniques

### Course Objectives

- To understand the Finite Automata, Pushdown Automata and Turing Machine.
- To understand the Regular Language, Context Free Language, Context Sensitive Language and Unrestricted Language.
- To understand the relation between Automaton and Language

### Course Outcomes

On completion of the course, student will be able to–

- Understand the use of automata during language design.
- Relate various automata and Languages.

### Course Contents

#### Unit I: Finite Automaton

(10)

1.1 Introduction: Symbol, Alphabet, String, Prefix & Suffix of Strings, Formal Language, Operations on Languages.

1.2 Deterministic finite Automaton – Definition, DFA as language recognizer, DFA as pattern recognizer.

1.3 Nondeterministic finite automaton – Definition and Examples.

NFA To DFA (Myhill Nerode Method)

NFA with  $\epsilon$ - transitions Definition and Examples. NFA with  $\epsilon$ -Transitions to DFA & Examples

1.4 Finite automaton with output – Mealy and Moore machine, Definition and Examples.

1.5 Minimization of DFA, Algorithm & Problem using Table Method.

#### Unit II: Regular Expressions and Languages

(06)

2.1 Regular Expressions (RE): Definition & Example Regular Expressions Identities.

2.2 Regular language-Definition and Examples.

Conversion of RE to FA-Examples.

2.3 Pumping lemma for regular languages and applications. Closure Properties of regular Languages

**Unit III: Context-Free Grammars and Languages (10)**

3.1 Grammar - Definition and Examples.

3.2 Derivation-Reduction - Definition and Examples. Chomsky Hierarchy.

3.3 CFG: Definition & Examples. LMD, RMD, Parse Tree Ambiguous

3.4 Grammar: Concept & Examples.

3.5 Simplification of CFG: Removing Useless Symbols, Unit Production,  $\epsilon$ -production and Nullable Symbol.

3.6 Normal Forms: Greibach Normal Form (GNF) and Chomsky Normal Form (CNF) Regular Grammar: Definition.

3.7 Left linear and Right Linear Grammar-Definition and Example. Equivalence of FA & Regular Grammar

3.8 Construction of regular grammar equivalent to a given DFA.

3.9 Construction of a FA from the given right linear grammar

**Unit IV: Push Down Automata (05)**

4.1 Definition of PDA and examples.

4.2 Construction of PDA using empty stack and final State method: Examples using stack method.

4.3 Definition DPDA & NPDA, their correlation and Examples of NPDA  
CFG (in GNF) to PDA: Method and examples

**Unit V: Turing Machine (05)**

5.1 The Turing Machine Model, Definition and Design of TM Problems on language recognizers.

Language accepted by TM.

5.2 Types of Turing Machines (Multitrack TM, Two-way TM, Multitape TM, Non-deterministic TM)

5.3 Introduction to LBA (Basic Model) & CSG. (Without Problems)

**Reference Books**

- Introduction to Automata Theory, Languages and Computation, John E. Hopcraft, Rajeev Motwani, Jeffrey D. Ullman, Third Edition, Pearson Education Publication, 2008
- Introduction to Automata theory, Languages and computation By John E. Hopcroft and Jeffrey Ullman – Narosa Publishing House, 1995
- Theory of Computer Science Automata, Languages and Computation, K.L.P. Mishra, N. Chandrasekaran, Publication- Prentice Hall of India, 2008
- Introduction to Computer Theory Daniel I. A. Cohen – 2<sup>nd</sup> edition – John Wiley & Sons, 1996
- Introduction to Languages and The Theory of Computation John C. Martin The McGraw-Hill, Fourth Edition, 2011

Ahmednagar Jilha Maratha Vidya Prasarak Samaj's

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – V	Paper – VI
Course Code: BSC-CS 506T	Title of the Course: Core JAVA
Credits: 02 Credits	Total Lectures: 36

**Prerequisites**

- Knowledge of C Programming language

**Course Outcomes (COs):**

- To learn Object Oriented Programming language
- To study various java programming concept like Interface, File and Exception Handling etc.
- To design User Interface using Swing and AWT

**Course Objectives**

- To learn Object Oriented Programming language
- To study various java programming concept like Interface, File and Exception Handling etc.
- To design User Interface using Swing and AWT

**Detailed Syllabus:**

**Unit I: An Introduction to Java (06)**

Object Oriented Programming Concepts A short history of Java .

Features OR Buzzwords of Java Java Environment Simple Java Program Java Tools – jdb, javap, javadoc Types of Comments Data Types Final Variable Declaring 1D, 2D Array Accepting Input (Command Line Arguments, BufferedReader, Scanner)

**Unit II: Objects and Classes (07)**

Defining your own classes Access Specifiers (public, protected, private, default) Array of Objects Constructors, Overloading Constructors and Use of 'this' keyword static block, static fields And methods Predefined Classes

Object Class, Methods (equals(), toString(),hashCode(), getClass())

String Class And StringBuffer Class,Formatting String data using format() method Creating , Accessing AndUsing Packages Wrapper Classes

**Unit III: Inheritance and Interface (08)**

Inheritance Basics (extends Keyword) and Types of Inheritance Superclass, Subclass and use of SuperKeyword Method Overriding and runtime polymorphism Use of final keyword related to method and class Use of abstract class and abstract methods Defining and Implementing Interfaces Runtime polymorphism using interface Concept of Marker and Functional Interfaces

**Unit IV: Exception and File Handling****(05)**

Dealing with errors , Exception class, Checked And Unchecked Exception Catching Exceptions, Multiple Catch Block, Nested try block Creating User Defined Exception Introduction to Files And Streams Input-OutputStream : FileInputStream/OutputStream, BufferedInput/OutputStream, DataInput/OutputStream Reader-Writer : FileReader/Writer, BufferedReader/Writer, InputStreamReader, OutputStreamWriter

**Unit V: User Interface with AWT and Swing****(10)**

What is AWT? What is Swing? Difference between AWT and Swing The MVC Architecture And Swing Layouts And Layout Managers Containers And Components – JFrame, JButton, JLabel, JText, JTextArea, JCheckBox And JRadioButton, JList, JComboBox, JMenu And related Classes Dialogs (Message, Confirmation, Input), JFileChooser, JColorChooser Event Handling: Event Sources, Listeners Adapters And Anonymous Inner Class.

**Suggested Readings:**

1. Complete reference Java by Herbert Schildt(5th edition)
2. Java 2 programming black books, Steven Horlzner
3. Programming with Java , A primer ,Forth edition , By E. Balagurusamy
4. Core Java Volume-I-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall,Sun Microsystems Pres

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)**

**Under Faculty of Science**

Semester – V	Paper – VII
Course Code: BSC-CS 507P	Title of the Course: Practical Based on BSC-CS 501 and BSC -CS 502
Credits: 02 Credits	Total Lectures: 30 Hrs.

**Course Objectives:**

- To Design dynamic and interactive Web pages.
- To Learn Core-PHP, Server Side Scripting Language
- To Learn PHP- Database handling
- To apply statistical, data preprocessing and visualization techniques on data sets

**Course Outcomes (COs):**

- Student understand the system calls in operating system & operations on processes
- Student understand CPU scheduling algorithm & Student understand how to handle deadlock
- Student understand the page replacement algorithm
- Understand how to develop dynamic and interactive Web Page
- Prepare data for use with a variety of statistical methods and recognize how the quality of the data may affect conclusions, Perform exploratory data analysis

**Guidelines:**

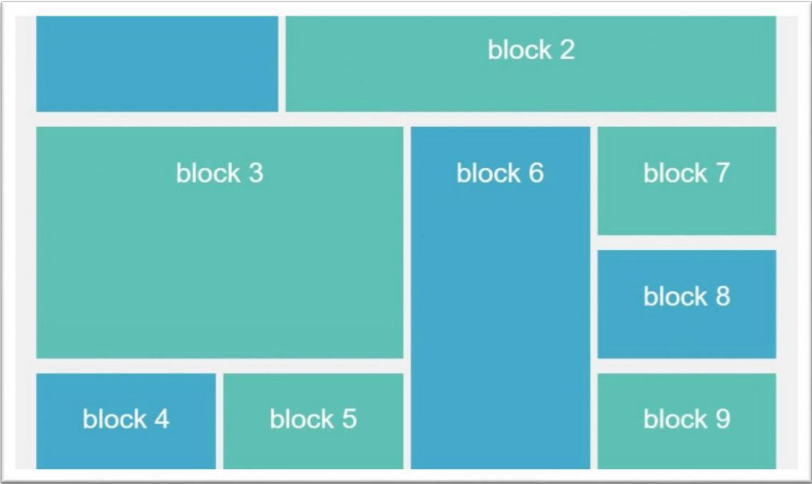
Operating Environment for web technologies: HTML5.0, PHP 5.0 and above ,  
Webserver

**Operating System Practical Assignment**

- 1 System calls in Operating Systems
  - Process control –fork(),exit(0,wait())
  - File manipulation –open(),read().write(),close()
  - Device management-ioctl(),read(),write()
- 2 Operations on process
  - Process creation
  - Process termination

- 3 Simulation of CPU Scheduling Algorithms
  - FCFS,
  - SJF,
  - Priority,
  - Round robin
- 4 Deadlock Handling
  - Bankers algorithm
- 5 Simulation of demand paging using memory page replacement algorithms
  - FIFO
  - Optimal
  - LRU
  - MFU

### Internet Programming-I (BSC-CS 502) Practical Assignments

Assignment 1:	Bootstrapping
	<p>1. Create following Bootstrap Web Layout Design. There are 9 blocks of the region in the arrangement. You can either place the images in them or the contents.</p> 
	2. Create following Bootstrap Web Page Design.



3. Create following layout using bootstrap



Assignment 2:

Using Functions and String

1. Write a script to accept two integers(Use html form having 2 textboxes).  
 Write a PHP script to,  
 a. Find mod of the two numbers.  
 b. Find the power of first number raised to the second.  
 c. Find the sum of first n numbers (considering first number as n)  
 d. Find the factorial of second number.  
 (Write separate function for each of the above operations.)

2. Write a PHP script for the following: Design a form to accept a string.  
 a. Write a function to find the length of given string without using built in functions.  
 b. Write a function to count the total number of vowels i.e. (a,e,i,o,u) ) from the string.  
 c. Convert the given string to lowercase and then to Title case.  
 d. Pad the given string with "\*" from left and right both the sides.  
 e. Remove the leading whitespaces from the given string.



	<p>3. Write a PHP script for the following: Design a form to accept two strings from the user.</p> <ol style="list-style-type: none"> <li>Find whether the small string appears at the start of the large string.</li> <li>Find the position of the small string in the big string.</li> <li>Compare both the string for first n characters, also the comparison should not be case sensitive.</li> </ol>
	<p>4. Write a PHP script having 3 textboxes. The first textbox should be for accepting name of the student, second for accepting name of college and third for accepting a proper greeting message. Write a function for accepting all the three parameters and generating a proper greeting message. If any of the parameters are not passed, generate the proper greeting message. (Use the concept of missing parameters)</p>
Assignment: 3	Using Arrays
	<ol style="list-style-type: none"> <li>Write a PHP script that inserts a new item in an array at any position. (hint : use array_splice())</li> <li>Write a menu driven program to perform the following stack and queue related operations:[Hint: use Array_push(), Array_pop(), Array_shift(), Array_unshift() ] <ol style="list-style-type: none"> <li>Insert an element in stack</li> <li>Delete an element from stack</li> <li>Display the contents of stack</li> <li>Insert an element in queue</li> <li>Delete an element from queue</li> <li>Display the contents of queue</li> </ol> </li> </ol>
	<p>3. Write a PHP script to sort the following associative array : array("Sophia"=&gt;"31","Jacob"=&gt;"41","William"=&gt;"39","Ramesh"=&gt;"40") in</p> <ol style="list-style-type: none"> <li>ascending order sort by value</li> <li>ascending order sort by Key</li> <li>descending order sorting by Value</li> <li>descending order sorting by Key</li> </ol>
Assignment: 4	Using Files and DATABASE(PostgreSQL)
	<ol style="list-style-type: none"> <li>Write a program to read one file and display the contents of the file with its size.</li> </ol>
	<ol style="list-style-type: none"> <li>Write a program to read a flat file "item.dat", which contains details of 5 different items such as Item code, Item Name, unit sold, and Rate. Display the Bill in tabular format.</li> </ol>

- |  |                                                                                                                                                                                                                                                                                                                                                             |
|--|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <p>3. Property (pno, description, area)</p> <ul style="list-style-type: none"><li>a. Owner (oname, address, phone)</li><li>b. An owner can have one or more properties, but a property belongs to exactly one owner.</li><li>c. Accept owner name from the user. Write a PHP script which will display all properties which are own by that owner</li></ul> |
|--|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

## New Arts, Commerce and Science College, Ahmednagar (Autonomous)

### Syllabus of T.Y.B.Sc. (Computer Science) Under Faculty of Science

Semester – V	Paper – VIII
<b>Course Code: BSC-CS 508P</b>	<b>Title of the Course:</b> Practical Course Based on BSC-CS 503 and BSC-CS 504
<b>Credits: 02Credits</b>	<b>Total Lectures: 30Hrs.</b>

#### Course Objectives:

- Details of decorators, lambdas, and comprehensions.
- Why and how to use iterators and generators.
- To introduce students to use of Python programming to solve data analytics problems
- To elaborate students to statistical analysis using Python programming

#### Course Outcomes (COs):

- The students will be able to improve Problem solving and programming capability
- The students will learn data analytics through python programming
- Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python

	Course Contents	Allotted Hours
	<p>Assignment 1 Understanding Object Oriented Programming, Python installation Differentiate procedure oriented and Object Oriented Programming Identify a simple real world scenario using the concept of classes and objects Demonstrate different types of inheritance in the scenario identified Practice Python Installation</p> <p>Assignment 2: Develop the programs to implement parent-child relationship Demonstrate different inheritance types Apply polymorphism and method overriding Create abstract classes</p> <p>Assignment 3: Implement the Python Classes and Objects to address the real world scenarios Define classes and objects using python for the real world scenario Defining constructors and using Self Understand public and private members Practice calling class methods from another class</p>	

Write built in functions to check, get, set and delete attributes

Assignment 4: Write the programs to address the exceptions via exception handling in the development of solutions and implement operator overloading

Write a simple exception handling program with try- except

Write a program for catching multiple exceptions

Demonstrate raising and re raising exceptions

Apply else and finally clauses

Demonstrate the usage of polymorphism in overloading of operator

Assignment 5: Write program using Decorator

Assignment 6: Write program using iterator and generator

Assignment 7: Write program using comprehensions

Assignment 8: Write program on multitreading

Assignment 9: Write a program to demonstrate database connectivity in python

Assignment 10 : demonstrate network programming using python

Write a program in Python to demonstrate following operations: a) Method overloading b) Method overriding

15. Write a program in Python to demonstrate following operations: a) Simple inheritance b) Multiple inheritance

### **Foundations of Data Science**

Assignment 1: The Data Science environment

Getting introduced to Python IDLE, command line, online tools like google colab and essential packages like NumPy, SciPy, pandas, scikit-learn, matplotlib, jupyter, beautiful-soup, etc.

Assignment 2: Loading the dataset

Select a dataset from a list of publicly available datasets at UCI Machine Learning Repository and load it using Pandas. (Import different data format files like .CSV, .htm, .json etc. Briefly describe what the dataset is about and size of the dataset (e.g. number of tables, number of instances and attributes, etc.)

Assignment 3: Basic statistical operations

Apply basic statistical operations on a dataset. For example - compute the mean, median, mode, range, quartiles, and variance for one or more attributes.

Assignment 4: Data preprocessing

Apply data preprocessing techniques that are likely required for the dataset. Partition them into appropriate number of bins by equal-frequency as well as equal-width partitioning.

Use smoothing by bin means to smooth the data based on the above partitioning, 3) Normalize the attribute based on min-max normalization and z-score normalization. Comment on which method you would prefer to use for partitioning,

	<p>Assignment 5: Data Visualization with matplotlib View the data using various 2-D, 3-D plots and charts, setting styles, saving the figures,customizing the legends, multiple subplots</p>	
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

Ahmednagar Jilha Maratha Vidya Prasarak Samaj's

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)**

**Under Faculty of Science**

Semester – V	Paper – IX
Course Code: BSC-CS 509P	Title of the Course: Practical Course Based on BSC - CS506 T
Credits: Credits	Total Lectures: 30 Hrs.

**Course Objectives:** Covers the complete scope of the syllabus. 1. Bringing uniformity in the way course is conducted across different colleges. 2. Continuous assessment of the students.

**Course Outcomes:** 1. Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs. 2. Read and make elementary modifications to Java programs that solve real-world problems. 3. Validate input in a Java program.

**Guidelines:** Operating Environment :  Operating system : Linux  Editor : Any linux based editor like vi, gedit and Use of IDE – Eclipse etc.  Compiler : javac

**List of Assignments:**

Assignment 1 :

Java Tools and IDE, Simple java programs [Slot – 1]

Introduction to the java environment Use of java tools like java, javac, jdb and javadoc

Defining simpleclasses and creating objects.

Assignment 2 :

Array of Objects and Packages [Slot – 2] Defining a class. Creating an array of objects.

Creating apackage.

Assignment 3 :

Inheritance and Interfaces [Slot – 2] To implement inheritance in java. To define abstract classes. Todefine and use interfaces and Functional Interface.

Assignment 4 :

Exception And File Handling [Slot – 2] Demonstrate Exception Handling Mechanism in Java.

Use of try, catch, throw, throws ,finally blocks Defining User defined Exception classes.

Creation of files anddemonstration of I-O operations

Assignment 5 :

GUI Designing, Event Handling [Slot – 5] To demonstrate GUI creation using Swing

Package andLayout managers. To understand Event handling mechanism in Java. Using

Event classes, Event Listeners and Adapters

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – V	Paper – X
Course Code: BSC-CS 510 T	Title of the Course: React JS
Credits: 02 Credits	Total Lectures: 30 Hrs.

**Course Outcomes (COs):**

- To build user interfaces specifically for single-page applications.
- To enable fast and quality assured application development that in turn saves time for both - clients and developers.
- To create reusable UI components and hence reduce the code.

**Detailed Syllabus:**

**Unit I: Introduction to ReactJS (04)**

- 1.1 What is ReactJS?
- 1.2 History of ReactJS.
- 1.3 Why to learn ReactJS?
- 1.4 Features of ReactJS.
- 1.5 Advantages of ReactJS.
- 1.6 Aspects of ReactJS.

**Unit II: ReactJS Installation (06)**

- 2.1 Setting up Environment for ReactJS
- 2.2 NodeJS Installation
- 2.3 Downloading and adding ReactJS Libraries.
- 2.4 Creating your first ReactJS project.
- 2.5 Use of create-react-app
- 2.6 ReactJS Project Structure

**Unit III: ReactJS ES6 (06)**

- 3.1 Classes
- 3.2 Arrow Functions
- 3.3 Variables
- 3.4 Array Methods
- 3.5 DeStructuring
- 3.6 Modules
- 3.7 Ternary Operators and Spread Operators.

**Unit IV: React JS Fundamentals****(08)**

- 4.1 React JSX
- 4.2 React Component
  - 4.2.1 Function Based Component
  - 4.2.2 Class Based Component
- 4.3 React Component API
- 4.4 Component Lifecycle
- 4.5 React State
- 4.6 React Props
- 4.7 React State vs React Props

**Unit V: Form Handling in ReactJS****(06)**

- 8.1 React Forms
- 8.2 React Events
- 8.3 React Lists
- 8.4 React Keys
- 8.5 React Refs
- 8.6 React Fragments

**Suggested Readings:**

1. “ FullStack React” The Complete Guide to ReactJS and Friends
2. The Road to Learn React: Your journey to master plain yet pragmatic React.js by RobinWieruch
3. React in Action by Mark Tielens Thomas
4. Learning React, 2nd Editi



*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – V	Paper – XI
Course Code: BSC-CS 511 P	Title of the Course: React JS (Lab.)
Credits: 02 Credits	Total Lectures: 30 Hrs.

**Course Outcomes (COs):**

- To build user interfaces specifically for single-page applications.
- To enable fast and quality assured application development that in turn saves time for both - clients and developers.
- To create reusable UI components and hence reduce the code.

**List of Assignments:**

- Setup the environment for ReactJS.
  - Install NodeJS.
  - Download ReactJS Libraries
  - Create ReactJS App using create-react-app
- Create a simple ReactJS app and display “Hello World” message on webpage.
- Create a list of programming languages you know and display the list on webpage.
- Create a class-based Component Car and display the car details.
- Create a function-based Component and display your own details like Name, DOB, Contact Number, Address etc.
- Add an attribute to pass a colour to the Car Component and use its render() function to display the Colour of Car.
- Create a class Component Car and display Car Component in root.
- Create a Constructor function in the Car Component and add a Colour property. Use the Colour property in render function to display it.
- Create a class Component called as Garage. Use the Car Component inside the Garage Component.
- Specify the state object in the constructor method. The state object can contain as many properties as you like. Add all properties of your Car Component like brand, model, colour, year, etc.
- Refer to the state object and display Car details using render() function.
- Add a button with an onClick() event that will change the colour property.
- Implement lifecycle methods of Component.
- Add a Brand attribute to the Car element. Use the props object to access brand attribute in Component.
- Send this Brand property from Garage Component to Car Component.

16. Create a Football Component. Put a shoot() function in it. Call this shoot() on a button click.
17. Create a list of Car brands in Garage Component. Render all the Car from Garage.
  
18. Add keys to above list items of Brands in Garage Component. Render all the Car from Garage.
19. Add an attribute that allows React to keep track of elements in list.
20. Create a form that allows user to enter their name. Use onChange() to manage the input.
21. Add a submit button and an event handler in the onSubmit attribute.

**NOTE: Students has to choice to opt for following DSEC courses  
with consent of Department**

**DSEC- 04: BSC-CS 504 T: Object Oriented Software Engineering**

**DSEC – 05: BSC – CS 506 T: Database Technology**

**DSEC – 09 Practical: BSC – CS 509 P Database Technology Practical**

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*  
**New Arts, Commerce and Science College, Ahmednagar**  
**(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)**

**Under Faculty of Science**

Semester – V	Paper – IV
Course Code: BSC-CS 504T	Title of the Course: Object Oriented Software Engineering
Credits: 02 Credits	Total Lectures: 30 Hrs.

**Pre-requisites:**

- a. Knowledge of Object Oriented Concepts.
- b. Knowledge of Classical Software Engineering

**Course Objectives:**

- a. Understanding importance of Object Orientation in Software engineering.
- b. Understand the components of Unified Modeling Language.
- c. Understand techniques and diagrams related to structural modeling.
- d. Understand techniques and diagrams related to behavioral modeling.
- e. Understand techniques of Object Oriented analysis, design and testing. □ □ □

**Course Outcomes (COs):**

- a. To Understand Object Oriented Modeling techniques and their applicability.
- b. To understand latest testing methods used in the software industries.

**Detailed Syllabus:**

**Unit I: Object Oriented Concepts and Principles**

**(03)**

- 1.1 What is Object Orientation? -Introduction, Object, Classes and Instance, Polymorphism, Inheritance
- 1.2 Object Oriented System Development- Introduction, Function/Data Methods(With Visibility), Object Oriented Analysis, Object Oriented Construction.
- 1.3 Identifying the Elements of an Object Model.
- 1.4 Identifying Classes and Objects.
- 1.5 Specifying the Attributes (With Visibility).
- 1.6 Defining Operations.
- 1.7 Finalizing the Object Definition.

**Unit II: Introduction to UML**

**(02)**

- 2.1 Concept of UML
- 2.2 Advantages of UML.

**Unit III: Basic Structural Modeling**

**(04)**

- 3.1 Classes
- 3.3 Relationship

- 3.4 Common Mechanism
- 3.5 Class Diagram (Minimum three examples should be covered)

**Unit IV: Advanced Structural Modeling (04)**

- 4.1 Advanced Classes
- 4.2 Advanced Relationship
- 4.3 Interface
- 4.4 Types and Roles
- 4.5 Packages
- 4.6 Object Diagram (Minimum three examples should be covered)

**Unit V: Basic Behavioral Modeling (06)**

- 5.1 Interactions
- 5.2 Use Cases and Use Case Diagram with stereo types (Minimum three examples should be covered)
- 5.3 Interaction Diagram (Minimum three examples should be covered)
- 5.4 Sequence Diagram (Minimum three examples should be covered)
- 5.5 Activity Diagram (Minimum three examples should be covered)
- 5.6 State Chart Diagram (Minimum three examples should be covered)

**Unit VI: Object Oriented Analysis (04)**

- 6.1 Interactive Development and the Rational Unified Process
- 6.2 Inception
- 6.3 Understanding Requirements
- 6.4 Use Case Model From Inception to Elaboration
- 6.5 Elaboration

**Unit VII: Object Oriented Design (03)**

- 7.1 The Booch Method, The Coad and Yourdon Method and Jacobson Method and Rumbaugh Method
- 7.2 The Generic Components of the OO Design Model
- 7.3 The System Design Process – Partitioning the Analysis Model, Concurrency and Sub-System Allocation, Task Management Component, The Data Management Component, The Resource Management Component, Inter Sub System Communication
- 7.4 Object Design Process

**Unit VIII: Architectural Modeling (04)**

- 8.1 Component
- 8.2 Component Diagram (Minimum three examples should be covered)
- 8.3 Deployment Diagram (Minimum three examples should be covered)
- 8.4 Collaboration Diagram (Minimum three examples should be covered)

**Reference Books:**

1. Grady Booch, James Rumbaugh, The Unified Modeling Language User/Reference Guide, Pearson Education INC
2. Ivar Jacobson, Object Oriented Software Engineering, Pearson Education INC
3. Craig Larman, Applying UML and Patterns Pearson Education INC  
Bennett, Simon, Object Oriented Analysis and Design McGraw Hill

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – V	Paper – VI
Course Code: BSC-CS 506T	Title of the Course: Database Technology
Credits: 02 Credits	Total Lectures: 34

**Prerequisites**

- Knowledge of the MySQL relational database management system.

**Course Outcomes (COs):**

- provide a quick introduction to MariaDB, and aid you in achieving a high level of comfort with MariaDB programming and administration.

**Course Objectives**

- It offers support for both small data processing tasks and enterprise needs.
- It aims to be a drop-in replacement for MySQL requiring only a simple uninstall of MySQL and an install of MariaDB.
- MariaDB offers the same features of MySQL and much more.

**Detailed Syllabus:**

**Unit I: Introduction to Database System (04)**

Introduction to RDBMS terminology, relational model, base tables, keys, primary keys, foreign keys, constraints, Codd Rules.

Normalization, Normal Forms- 1NF,2NF,3NF, BCNF,4NF and 5NF, E-R Diagram, Mapping ER- diagram to database tables.

**Unit II: MariaDB (05)**

Introduction to MariaDB, Data types, SQL commands, Create, Insert, Update, Delete, Drop, Alter, SQL functions (String functions, date functions), indexing, Keys, Primary Key, Foreign Keys.

**Unit III: Manipulating Data with MariaDB (10)**

SQL Statements, Select , like clause, groupby, orderby, joins –leftjoin, naturaljoin, rightjoin, union,correlated and nested queries, Backup and restore commands.

**Unit IV: NOSQL Database Technology (12)**

Introduction to NOSQL Databases, Difference between relational and NoSQL databases, NoSQL features, types , advantages,Cassandra-Cqlsh,Shell Commands, Cassandra Keyspace Operations, Cassandra Table Operations, Cassandra CRUD Operations, Cassandra CQL Types.

JSON File Format for storing documents, Inserting and Saving Documents, Batch Inert,

Insert Validation, Removing Documents Updating Documents, Document Replacement, Using Modifiers, Updating Multiple Documents, Returning updated Documents.

Introduction to Indexing, Introduction to Compound Indexes, Using Compound Indexes, Indexing Objects and Arrays, Aggregation Framework, pipeline Operations, \$match, \$project, \$group, \$unwind, \$sort, \$limit, \$skip. Using Pipeline, MongoDB and MapReduce, Aggregation Commands, Introduction to Replication, configuring a Replica Set, Member Configuration Options.

**Unit V: Selecting the Right Database****(03)**

Selection of right databases, RDBMS or NoSQL, selection of database based on performance, data size, type of data, frequency of accessing data, business needs, type of application.

**Suggested Readings:**

- 1) Database Technologies -BPB Publications.
- 2) <https://www.tutorialspoint.com/mariadb>
- 3) Getting Started with MariaDB, 2nd Edition by Daniel Bartholomew  
A beginner's guide to MariaDB. No prior MariaDB or MySQL experience necessary.  
An expanded update to the 1st edition.
- 4) MariaDB Beginners Guide by Rodrigo Ribeiro Gonçalves  
book for newcomers to MariaDB and SQL.

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)**

**Under Faculty of Science**

Semester – V	Paper – XII
Course Code: BSC-CS 512P	Title of the Course: Practical Course Based on BSC -CS508T
Credits: Credits	Total Lectures: 30 Hrs.

**Course Objectives:**

Covers the complete scope of the syllabus.

1. Bringing uniformity in the way course is conducted across different colleges.
2. Continuous assessment of the students.

**Course Outcomes:**

1. Use an integrated development environment to write, compile, run, and test simple queries.
2. Elementary modifications to programs that solve real-world problems.
3. Validate input.

**Guidelines:**

If you have intimate knowledge of Linux/Unix systems, simply download source to build your install.

Our recommended way of installing is to utilize distribution packages.

MariaDB offers packages for the following Linux/Unix distributions-

- RedHat/CentOS/Fedora
- Debian/Ubuntu

The following distributions include a MariaDB package in their repositories-

- openSUSE
- Arch Linux
- Mageia
- Mint
- Slackware

**List of Assignments:**

Assignment 1 :

Assignment based on select database, datatype, create table, drop table, insert query, select query, where clause, update query, delete query, Alter command

Assignment 2 :

Assignment based on like clause, orderby, groupby, join. Null values

Assignment 3 :

Assignment based on Indexes and Statistics table, table cloning



Assignment 4 :

Assignment based on Useful functions – aggregate functions, age calculation, string calculation, date/time function, numeric functions. String functions

Assignment 5 :

Assignment based on Transaction – Structure of Transaction

Assignment 6:

Assignment on PHP syntax, Connection – PHP Connection Script

Assignment 7:

Assignment based on Cassandra keyspace, table, CRUD operations. Assignment based on CQL Types

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – VI	Paper – I
Course Code: BSC-CS 601 T	Title of the Course: Advanced Operating System
Credits: 02 Credits	Total Lectures: 32 Hrs.

**Course Outcomes (COs):**

On completion of the course, student will be able to–

- Understanding concept Unix Operating System.
- Understanding File and Directories
- Concept Process ,Process Control ,Process Context
- Concept Signal handling

**Detailed Syllabus:**

**Unit 1:Introduction to UNIX/LinuxKernel**

**(6)**

- 1.1 System Structure,
- 1.2 User Perspective,
- 1.3 Assumptions about Hardware
- 1.4 Architecture of UNIX Operating System
- 1.5 Concepts of Linux Programming-
  - 1.5.1 Files and the Filesystem, Processes
  - 1.5.2 Users and Groups
  - 1.5.3 Permissions, Signal
  - 1.5.4 Interprocess Communication

**Unit 2:File and Directory I/O**

**(13)**

- 2.1 Buffer headers
- 2.2 structure of the buffer pool,
- 2.3 scenarios for retrieval of a buffer
- 2.4 reading and writing disk blocks
- 2.5 inodes, structure of regular file, open, read, write, lseek, close, pipes, dup open, creat, file sharing, atomic special files. dup2, sync, fsync, and fdatsync, fcntl, /dev/fd, stat, fstat, lstat, file types, Set-User-ID and Set-Group-ID, 3.6 file access permissions, ownership of new files and directories,

access function, umask function, chmod and fchmod, sticky bit, chown, fchown, and lchown, file size, file truncation, file systems, link, unlink, remove, and rename functions, symbolic links, symlink and readlink functions, file times, utime, mkdir and rmdir, reading directories, chdir, fchdir, and getcwd, device special files

**Unit 3: Process Environment, Process Control**

(6)

3.1 What is Process

3.2 Process states and transitions, layout of system memory

3.3 The context of a process

3.3.1 saving the context of a process

- sleep
- process creation,
- signals,
- process termination,
- awaiting process termination,
- invoking other programs, the user id of a process
- changing the size of the process.

**Unit 4: Signal Handling**

(7)

4.1 Signal concepts

4.2 signal function

4.3 unreliable signals

4.4 interrupted system calls

4.5 reentrant functions

4.6 SIGCLD semantics

4.7 reliable-signal technology

4.7.1 kill and raise,

4.7.2 alarm and pause

4.7.3 signal sets

4.7.4 sigprocmask

4.7.5 sigpending

4.7.6 sigsetjmp

4.7.7 siglongjmp,

4.7.8 sigsuspend

4.7.9 abort

4.7.10 sleep

**Suggested Readings:**

1. The Design of the UNIX Operating System Maurice J. Bach. PHI
2. Advanced Programming in the UNIX Environment Richard Stevens Addison-Wesley
3. Linux System Programming Robert Love O'Reilly

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

<b>Semester – VI</b>	<b>Paper –II</b>
<b>Course Code: BSC-CS 602 T</b>	<b>Title of the Course: Internet Programming - II</b>
<b>Credits: 02 Credits</b>	<b>Total Lectures: 30 Hrs.</b>

**Prerequisites**

- HTML5,CSS
- Core PHP
- Bootstrap framework utility

**Course Objectives**

- To Learn different technologies used at client Side Scripting Language
- To Learn XML and XML parsers.
- To One PHP framework for effective design of web application.
- To Learn Java Script to program the behavior of web pages.
- To Learn AJAX to make our application more dynamic.
- Framework has

**Course Outcomes**

On completion of the course, student will be able to–

- Build dynamic website.
- Using MVC based framework easy to design and handling the errors in dynamic website.

**Course Contents**

**Unit I:Introduction to Web Techniques**

**(04)**

- 1.1 Variables
- 1.2 Server information
- 1.3 Processing forms
- 1.4 Setting response
- 1.5 headers Maintaining
- 1.6 PHP error handling

**Unit II:XML**

**(04)**

- 2.1 What is XML?
- 2.2 XML document
- 2.3 Structure PHP and
- 2.4 XML
- 2.5 XML parser
- 2.6 The document object
- 2.7 model The simple XML
- 2.8 Changing a value with simple XML

**Unit III: Java Script and JQuery****(09)**

## 3.1 Overview of JavaScript

3.1.1 Object Orientation and JavaScript Basic Syntax(JS data types, JS variables )

3.2 Screen Output and keyboard input(Verification and

3.3Validation)JS Control statements and JS Functions

3.4 JavaScript HTML DOM Events

3.5 JS Strings and JS String methods

3.6 JS popup boxes(alert, confirm, prompt).

3.6.1 JQuery library , Including JQuery library

in 3.6.2 page JQuery selector , DOM

manipulation using JQuery

**Unit IV:AJAX****(05)**

4.1 Introduction of AJAX

4.2AJAX web application

modelAJAX –PHP

framework 4.4Performing

AJAX validation

4.4.1Handling XML data using php

and 4.4.2AJAXConnecting database

using php and AJAX

**Unit V:PHP framework Code Igniter****(08)**

5.1 Code Igniter - Overview, Installing

Code 5.1.2 IgniteApplication

Architecture

5.1.3 MVC Framework , Basic concept of Code

Igniter, 5.1.4LibrariesWorking with databases

5.2 Load external JS and CSS page &amp; redirecting from controller , Adding JS

and CSS ,Page redirection.

5.3 Loading dynamic data on page &amp; session management, cookies management

**Reference Books:**

- **Programming PHP By Rasmus Lerdorf and Kevin Tatroe O'Reilly publication**
- **Beginning PHP 5, Wrox publication**
- **AJAX Black Book Kogent solution**
- **Mastering PHP BPB Publication**
- **Professional Code igniter By Thomas Myer ,Wrox Publication,**
- **Codeihniter 2 Cookbook By Rob Foster ,PACKT Publication ,**
- **Jquery Cookbook, O'reilly Publication.**

## Ref. Links:

- [www.php.net.in](http://www.php.net.in)
- [www.W3schools.com](http://www.W3schools.com)
- <https://www.tutorialspoint.com/codeigniter/index.htm>
- <https://api.jquery.com/>
- <http://codeigniter.com/docs>

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – VI	Paper – III
Course Code: BSC-CS 603T	Title of the Course: Data Analytics
Credits: 02 Credits	Total Lectures: 36

**Prerequisites**

- Basic of mathematics and statistics
- Basic programming Knowledge of python
- Knowledge of databases

**Course Outcomes (COs):**

On completion of the course, student will be able to–

- Use appropriate models of analysis, assess the quality of input, and derive insight from results.
- Analyze data, choose relevant models and algorithms for respective applications
- Understand different data mining techniques like classification, prediction, clustering and association rule mining
- Apply modeling and data analysis techniques to the solution of real world business problems

**Course Objectives**

- Deploy the Data Analytics Lifecycle to address data analytics projects.
- Develop in depth understanding of the key technologies in data analytics.
- Apply appropriate analytic techniques and tools to analyze data, create models, and identify insights that can lead to actionable results.

**Detailed Syllabus:**

**Unit I: Introduction to Data Analytics (06)**

Concept of data analytics Data analysis vs Data analytics Types of analytics Diagnostic Analytics, Predictive Analytics, Prescriptive Analytics, Exploratory Analysis, Mechanistic Analysis Mathematical models - Concept Model evaluation: metrics for evaluating classifiers -Class imbalance - AUC, ROC (Receiver-Operator Characteristic) curves, Evaluating value prediction models

**Unit II: Machine Learning Overview (06)**

Introduction to Machine Learning, deep learning, Artificial intelligence Applications for machine learning in data science The modeling process

Engineering features and selecting a model, Training the model, Validating the model, Predicting new observations

Types of machine learning Supervised learning, Unsupervised learning, Semi-supervised learning, ensemble techniques

Regression models

Linear Regressions

Polynomial Regression

Logistic Regressions

Concept of classification, clustering and reinforcement learning.

### **Unit III: Mining Frequent Patterns, Associations, and Correlations (12)**

What kind of patterns can be mined

Class/Concept Description: Characterization and Discrimination, Mining

Frequent Patterns, Associations, and Correlations, Classification and Regression

for Predictive Analysis, Cluster Analysis, Outlier Analysis

Mining frequent patterns - Market Basket Analysis.

Frequent Itemsets, Closed Itemsets, and Association

Rules Frequent Itemset Mining Methods

Apriori Algorithm

Generating Association Rules from Frequent

Itemsets Improving efficiency of apriori

algorithm

Frequent pattern growth (FP-growth) algorithm.

### **Unit IV: Social Media and Text Analytics (12)**

Overview of social media analytics

Social Media Analytics Process, Seven layers of social media analytics, accessing social

media data

Key social media analytics

methods Social network

analysis

Link prediction, Community detection, Influence maximization, Expert

finding, Prediction of trust and distrust among individuals

Introduction to Natural Language Processing

Text Analytics : Tokenization, Bag of words, Word weighting : TF-IDF, n-Grams, stop

words, Stemming and lemmatization, synonyms and parts of speech tagging Sentiment

Analysis Document or text summarization

Trend analytics

Challenges to social media analytics

### **Suggested Readings:**

- 1) Data Science Fundamentals and Practical Approaches, Gypsy Nandi, Rupam Sharma, BPB Publications, 2020.
  - 2) The Data Science Handbook, Field Cady, John Wiley & Sons, Inc, 2017
  - 3) Data Mining Concepts and Techniques, Jiawei Han, Micheline Kamber, Jian Pei, Morgan Kaufmann, Third Edition, 2012.
  - 4) A Hands-On Introduction to Data Science, Chirag Shah, University of Washington Cambridge University Press
  - 5) The Data Science Design Manual, Steven S. Skiena, Springer, 2017
- Introducing data science: big data, machine learning, and more, using Python tools, Cielen D., Meysman A.D., & Ali M., Manning Publications Co., 2016

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

<b>Semester – VI</b>	<b>Paper – IV</b>
<b>Course Code: BSC-CS 604 T</b>	<b>Title of the Course: Compiler Construction</b>
<b>Credits: 02 Credits</b>	<b>Total Lectures: 36 Hrs.</b>

**Prerequisites**

- Knowledge of Automata Theory and Languages.

**Course Objectives**

- To understand design issues of a lexical analyzer and use of LEX tool.
- To understand design issues of a parser and use of YACC tool.
- To understand and design code generation and optimization techniques.

**Course Outcomes**

On completion of the course, student will be able to–

- Understand the process of scanning and parsing of source code.
- Learn the conversion code written in source language to machine language.
- Understand tools like LEX and YACC.

**Course Contents**

**Unit I: Introduction**

**(04)**

- 1.1 Definition of Compiler, Aspects of compilation.
- 1.2 The structure of Compiler.
- 1.3 Phases of Compiler – Lexical Analysis, Syntax Analysis, Semantic Analysis, Intermediate Code generation, code optimization, code generation, Error Handling.
- 1.4 Introduction to one pass & Multi pass compilers, cross compiler, Bootstrapping.

**Unit II: Lexical Analysis (Scanner)**

**(04)**

- 2.1 Review of Finite automata as a lexical analyzer,
- 2.2 Applications of Regular Expressions and Finite Automata (lexical analyzer, searching using RE), Input buffering, Recognition of tokens.
- 2.3 LEX: A Lexical analyzer generator (Simple Lex Program)

**Unit III: Syntax Analysis (Parser)**

**(14)**

- 3.1 Definition, Types of Parsers
- 3.2 Top-Down Parser –
  - 3.2.1 Top-Down Parsing with Backtracking: Method & Problems
  - 3.2.2 Drawbacks of Top-Down parsing with backtracking,
  - 3.2.3 Elimination of Left Recursion (direct & indirect) 3.2.4 Need for Left Factoring &



examples

### 3.3 Recursive Descent Parsing: Definition

#### 3.3.1 Implementation of Recursive Descent Parser Using Recursive Procedures

### 3.4 Predictive [LL (1)] Parser (Definition, Model)

#### 3.4.1 Implementation of Predictive Parser [LL (1)]

#### 3.4.2 FIRST & FOLLOW

### 3.5 Construction of LL (1) Parsing Table

#### 3.5.1 Parsing of a String using LL (1) Table.

### 3.6 Bottom-Up Parsers

### 3.7 Operator Precedence Parser -Basic Concepts

#### 3.7.1 Operator Precedence Relations form Associativity & Precedence Operator Precedence Grammar

#### 3.7.2 Algorithm for LEADING & TRAILING (with ex.)

#### 3.7.3 Algorithm for Operator Precedence Parsing (with ex.)Precedence Functions

### 3.8 Shift Reduce Parser

#### 3.8.1 Reduction, Handle, Handle Pruning

#### 3.8.2 Stack Implementation of Shift Reduce Parser (with examples)

### 3.9 LR Parser: Model, Types [SLR (1), Canonical LR, LALR]-Method & examples.

### 4.0 YACC (from Book 3) –program sections, simple YACC program for expression evaluation

## Unit IV:Syntax Directed Definition

(07)

### 4.1 Syntax Directed Definitions (SDD)

Inherited & Synthesized Attributes

### 4.2 Evaluating an SDD at the nodes of a Parse Tree, ExampleEvaluation

Orders for SDD's Dependency Graph

### 4.3 Ordering the Evaluation of Attributes

S-Attributed Definition, L-Attributed

DefinitionApplication of SDT

Construction of syntax trees,

The Structure of a Type

### 4. 4 Translation Schemes

#### 4.4.1 Definition, Postfix Translation Scheme

## Unit V: Code Generation and Optimization

(07)

### 5.1 Compilation of expression –

5.1.1 Concepts of operand descriptors and register descriptors with  
example. Intermediate code for expressions – postfix notations,  
Triples, Quadruples and Expression trees.

### 5.2 Code Optimization – Optimizing transformations – compile time evaluation, elimination of common sub expressions, dead code elimination, frequency reduction, strength reduction.

#### 5.2.1 DAG for Three address code

#### 5.2.2 The Value-number method for constructing DAG's.

#### 5.2.3 Definition of basic block, Basic blocks, and flow graphs Directed acyclic graph (DAG)

representation of basic block. Issues in design of code generator.

**Reference Books**

- Compilers: Principles, Techniques, and Tools, Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, 2004
- Principles of Compiler Design By: Alfred V. Aho, Jeffrey D. Ullman, Narosa Publication House, 2002 LEX & YACC, 2<sup>nd</sup> edition, O'reilly Publication, 2012

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – VI	Paper – V
Course Code: BSC-CS 605 T	Title of the Course: Android Programming
Credits: 02 Credits	Total Lectures: 34 Hrs.

**Course Outcomes (COs):**

After successful completion of this course, learner will be able to:

- a. Describe the process of developing mobile applications.
- b. Create mobile applications on the Android Platform.
- c. Design and implement mobile applications involving data storage in SQLite database
- d. Use location-based services while developing applications

**Detailed Syllabus:**

**Unit I: Introduction to Android**

**(05)**

- 1.1. Overview
- 1.2. History
- 1.3. Features of Android
- 1.4. Architecture of Android
  - Overview of Stack • Linux Kernel • Native Libraries • Android Runtime
  - Application Framework • Applications
- 1.5. SDK Overview
  - Platforms
  - Tools – (JDK, SDK, Eclipse/Android Studio, ADT, AVD, Android Emulator),
- Versions
- 1.6. Creating your first Android Application

**Unit II: Activities, Fragments and Intents**

**(07)**

- 2.1. Introduction to Activities
- 2.2. Activity Lifecycle
- 2.3. Introduction to Intents
- 2.4. Linking Activities using Intents
- 2.5. Calling built-in applications using Intents
- 2.6. Introduction to Fragments
- 2.7. Adding Fragments Dynamically
- 2.8. Lifecycle of Fragment
- 2.9. Toast

**Unit III: Android User Interface**

**(07)**

- 3.1. Understanding the components of a screen
  - Views and View Groups • Linear Layout • Absolute Layout • Table Layout

- Relative Layout • Frame Layout • Scroll Layout • Scroll View • Constraint Layout
- 3.2. Adapting to Display Orientation
- Anchoring Views
  - Resizing and Repositioning
- 3.3. Split Screen / Multi-Screen Activities

**Unit IV: Designing Your User Interface with Views (10)**

- 4.1. Using Basic Views
- TextView
  - Button, ImageButton, EditText, CheckBox
  - Switch, ToggleButton, RadioButton, and RadioGroup Views
  - ProgressBar View
  - AutoCompleteTextView View
- 4.2. Using Picker Views
- TimePicker View • DatePicker View
- 4.3. Using List Views to Display Long Lists
- ListView View • Using the Spinner View
- 4.4. Understanding Specialized Fragments
- Using a ListFragment • Using a DialogFragment
- 4.5. Displaying Pictures and Menus
- Using Image Views to Display Pictures**
- Gallery and ImageView views
  - Image Switcher
  - Grid View
- Using Menus with Views**
- Creating the helper methods • Options Menu • Context Menu
- 4.6. VideoView
- Play video from URL with using VideoView
  - VideoView Create
  - Optimized VideoView
  - Optimized VideoView in ListView

**Unit V: Databases – SQLite (05)**

- 5.1. Introduction to SQLite
- 5.2. SQLite Open Helper and SQLite Database
- 5.3. Creating, opening and closing database
- 5.4. Working with cursors, Insert, Update, Delete
- 5.5. Building and executing queries

**Suggested Readings:**

1. Beginning Android4 Application Development, By Wei-Meng Lee WILEY India Edition WROX Publication
2. Professional Android 4 Application Development, By Reto Meier WROX Publication
3. Head First Android Development: A Brain-Friendly Guide, By David Griffiths and Dawn Griffiths

Ahmednagar Jilha Maratha Vidya Prasarak Samaj's

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

<b>Semester – VI</b>	<b>Paper – VI</b>
<b>Course Code:</b> BSC-CS 606T	<b>Title of the Course:</b> Advanced JAVA
<b>Credits:</b> 02 Credits	<b>Total Lectures:</b> 36 Hrs.

**Prerequisites**

- Knowledge of Core Java (BSC-CS 505T)

**Course Objectives**

- To learn database programming using Java
- To study web development concept using Servlet and JSP
- To develop a game application using multithreading
- To learn socket programming concept

**Course Outcomes**

On completion of the course, student will be able to–

- To access open database through Java programs using Java Data Base Connectivity (JDBC) and develop the application.
- Understand and Create dynamic web pages, using Servlets and JSP.
- Work with basics of framework to develop secure web applications..

**Detailed Syllabus:**

**UNIT 1 Collections (06)**

Introduction to the Collection framework List - ArrayList, LinkedList Set - HashSet, TreeSet, Map -HashMap and TreeMap .

Interfaces such as Comparator, Iterator, ListIterator, Enumeration.

**UNIT 2 Multithreading (06)**

What are threads? Life cycle of thread Creating threads - Thread class , Runnable interface , Thread priorities, Running multiple threads Synchronization and interthread communication

**UNIT 3 Database Programming (06)**

The design of jdbc Types of drivers Executing sql statements, query execution Scrollable and updatable Resultset

**UNIT 4 Servlets and JSP (12)**

Introduction to Servlet and Hierarchy of Servlet, Life cycle of servlet , Handling get and post request

.(HTTP) Handling data from HTML to servlet , Retrieving data from database to servlet, Session tracking – User Authorization, URL rewriting, Hidden form fields, Cookies and HttpSession Introduction to JSP, Life cycle of JSP , Implicit Objects Scripting elements -

Declarations, Expressions,Scriptlets, Comments JSP Directives - Page Directive, include directive Mixing Scriptlets and HTML JSP Actions - jsp:forward , jsp:include, jsp:useBean, jsp:setProperty and jsp:getProperty

**UNIT 5 Spring Framework****(06)**

Introduction of Spring framework Spring Modules / Architecture ,Spring Applications ,Spring MVC Spring MVC Forms, Validation.

**Suggested Readings:**

1. Complete reference Java by Herbert Schildt(5th edition)
2. Java 2 programming black books, Steven Horlzner
3. Programming with Java , A primer ,Forth edition , By E. Balagurusamy
4. Core Java Volume-I-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall,Sun Microsystems Press
5. Core Java Volume-II-Advanced Features, Eighth Edition, Cay S. Horstmann, Gary Cornell, PrenticeHall, Sun Microsystems Press
6. Getting started with Spring Framework: covers Spring 5 by J Sharma and Ashish Sarin
7. Spring 4 for Developing Enterprise Applications: An End-to-End Approach by Henry H. Liu

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)**

Under Faculty of Science

Semester – VI	Paper – VII
Course Code: BSC-CS 607 P	Title of the Course: Practical course based on CS-601 & 602
Credits: 02 Credits	Total Lectures: 30 Hrs.

**Course Outcomes (COs):**

- Understanding Unix file and directories Concept.
- Understanding process context , signal handling.
  - To Learn XML and XML parsers.
- To One PHP framework for effective design of web application.
- To Learn Java Script to program the behavior of web pages, Learn AJAX to make our application more dynamic
- Using MVC based framework easy to design and handling the errors in dynamic website.

**Assignments on Advanced Operating System**

- 1 Write a following program in 'C 1. To create 'n' children. When the children will terminate, display total cumulative time children spent in user and kernel mode.
- 2 Write a C program to display all the files from current directory which are created in particular month
- 3 To create a file with hole in it.
- 4 Takes multiple files as Command Line Arguments and print their inode number
- 5 . Print the type of file where file name accepted through Command Line
- 6 To demonstrate the use of atexit() function
- 7 Open a file goes to sleep for 15 seconds before terminating
- 8 Read the current directory and display the name of the files, no of files in current directory.
- 9 To print the size of File

- 10 Write a C program which receives file names as command line arguments and display those filenames in ascending order according to their sizes. i) (e.g \$ a.out a.txt b.txt c.txt
- 11 Write a C Program that demonstrates redirection of standard output to a file
12. Write a program that illustrates how to execute two commands concurrently with a pipe.
14. To generate parent process to write unnamed pipe and will read from it.
- 15 Write a C program to implement the following unix/linux command i. ls -l > output.txt

### Assignment on Internet Programming II

Assignment: 1	COOKIES and SESSIONS.
	1. Write a PHP script to keep track of number of times the web page has been access.  [Use session and cookies]
	2. Write a PHP script to change the preferences of your web page like font style, font size, font color, background color using cookie. Display selected setting on next web page and actual implementation (with new settings) on third page.
	3. Write a PHP script to accept username and password. If in the first three chances, username and password entered is correct then display second form with “Welcome message” otherwise display error message. [Use Session]
Assignment: 2	XML documents and DOM
	1) Write a script to create XML file named “Item.xml” <pre>&lt;Item&gt; &lt;ItemName&gt;          &lt;/ItemName&gt; &lt;ItemPrice&gt;         &lt;/Itemprice&gt; &lt;Quantity&gt;          &lt;/Quantity&gt; &lt;/Item&gt;</pre>
	Link “ Item. Xml” file to the CSS style sheet and get well formatted output as given below i) ItemName : Color : red; Font-family : copperplate Gothic Light; Font-size : 16pt; Font :bold; ii) ItemPrice and Quantity color:yellow;



	font-family:Arial; font-size:12 pt; font:bold;
	3. Write a PHP script to read “Movie.xml” file and print all MovieTitle and ActorName of file using DOMDocument Parser. “Movie.xml” file should contain following information with at least 5 records with values. MovieInfo MovieNo, MovieTitle, ActorName , ReleaseYear
Assignment: 3	JavaScript
	1. Write a javascript to display message ‘Exams are near, have you started preparing for?’ using alert, prompt and confirm boxes. Accept proper input from user and display messages accordingly.
	2. Add or append in paragraph text and also in the numbered(ordered) list in a given HTML document using jQuery selectors. [Hint : Use Append( ) method]
	3. Write a javascript function to validate username and password for a membership form.
	4. To insert text before and after an image using jQuery. [Hint : Use before( ) and after( )]
Assignment :4	Ajax
	1. Write AJAX program to read contact.dat file and print the contents of the file in a tabular format when the user clicks on print button. Contact.dat file should contain srno, name, residence number, mobile number, Address. [Enter at least 3 record in contact.dat file]
	2. Write AJAX program where the user is requested to write his or her name in a text box, and the server keeps sending back responses while the user is typing. If the user name is not entered then the message displayed will be, “Stranger, please tell me your name!”. If the name is Rohit, Virat, Dhoni, Ashwin or Harbhajan , the server responds with “Hello, master &lt;user name&gt;!”. If the name is anything else, the message will be “&lt;user name&gt;, I don’t know you!”.
	3. Create TEACHER table as follows TEACHER(tno, tname, qualification, salary). Write Ajax program to select a teachers name and print the selected teachers details.

Assignment :5	: PHP framework Code Igniter
	<p>1. Create a CSS file to apply the following styling for an HTML document. Background color: blue,</p> <pre>&lt;bookno&gt;2&lt;/bookno&gt; 1  H1 { Color : red, Font-family : Verdana, Font-size : 8 } P { Color : green, Font-family : Monospace, Font-size :10 }</pre>
	<p>2. Add a Javascript file in codeigniter. The javascript code should check whether a number is positive or negative.</p>
	<p>3. Create a table student having attributes (rollno, name, class). Assume appropriate data types for the attributes. Using Codeigniter , connect to the database and insert minimum 5 records in it.</p>
	<p>4. For the above table student, display all its records using Codeigniter.</p>

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)**

Under Faculty of Science

Semester – VI	Paper – VIII
<b>Course Code:</b> BSC-CS 608 P	<b>Title of the Course:</b> Practical Course Based on BSC CS 603 & BSC CS 606
<b>Credits:</b> 02 Credits	<b>Total Lectures:</b> 30Hrs.

**Course Objectives:**

1. Covers the complete scope of the syllabus.
2. Bringing uniformity in the way course is conducted across different colleges.
3. Continuous assessment of the students.
4. Advanced Java is designed to develop web based, network centric, Enterprise level applications

**Course Outcomes:**

1. To Learn database Programming using Java
2. Understand and Create dynamic web pages using Servlets and JSP.
3. Work with basics of framework to develop secure web applications

**Guidelines: Operating Environment :**

- Operating system: Linux □ Editor: Anylinux based editor like vi, gedit and Use of IDE – Eclipse etc.
- Compiler :javac
- Database :postgresql.

**List of Assignments:**

Assignment 1:

Database Programming [Slot-2] Study the Collection framework in java. To Implement Various Interfaces and classes through algorithms. To Demonstrate Cursor Objects (Enumeration, Iterator, ListIterator, Comparator)

Assignment 2:

Multithreading [Slot-2] To create and use threads in java. To demonstrate multithreading using ThreadSynchronization, Inter-thread. Communication, Thread Priorities.

Assignment 3:

Database Programming [Slot-2] To communicate with a database using java. To execute queries on tables. To obtain information about the database and tables.

Assignment 4:

Servlets [Slot-2] To understand server-side programming. Simple steps to create and execute servlets. How to pass parameters using doGet and doPost methods? Handling data from HTML to servlet. How to connect servlet to a database? Use of various session

tracking methods like Cookies.

Assignment 5:

Java Server Pages [Slot-2] JSP life-cycle. Use of JSP implicit objects. JSP Directives. Use of ScriptingElements. To understand actionstags in JSP. Understanding flow of JSP custom tags.

Assignment 6:

Spring Framework [Slot-2] To create and understand the steps to develop Spring application

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*  
**New Arts, Commerce and Science College, Ahmednagar**  
**(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)**  
**Under Faculty of Science**

Semester – VI	Paper – IX
Course Code: BSC-CS 609P	Title of the Course: Practical Course Based BSC_CS 605
Credits: 02 Credits	Total Lectures: 30Hrs.

**Course Outcomes (COs):**

After successful completion of this course, learner will be able to:

- a. Describe the process of developing mobile applications.
- b. Create mobile applications on the Android Platform.
- c. Design and implement mobile applications involving data storage in SQLite database
- d. Use location-based services while developing applications

**Lab Course Contents**

Assignment No 1. Activities, Fragments and Intents

Assignment No 2. Android User Interface

Assignment No 3. Designing Your User Interface with Views

Assignment No 4. Databases – SQLite

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

<b>Semester – VI</b>	<b>Paper – X</b>
<b>Course Code: BSC-CS 610T</b>	<b>Title of the Course: Software Testing</b>
<b>Credits: 02 Credits</b>	<b>Total Lectures: 32 Hrs.</b>

**Pre-requisites:**

- Basic knowledge of algorithms, problem solving, expected inputs/outputs.
- Knowledge of C and Java Programming Language, compilation, debugging

**Course Objectives:**

- To provide the knowledge of software testing techniques.
- To understand how testing methods can be used as an effective tools in quality assurance of software.
- To provide skills to design test case plan for testing software.
- To provide knowledge of latest testing methods.

**Course Outcomes (COs):**

- To understand various software testing methods and strategies.
- To understand a variety of software metrics and identify defects and managing those defects for improvement in quality for given software.
- To design test cases and test plans, review reports of testing for qualitative software,
- To understand latest testing methods used in software industries.

**Detailed Syllabus:**

**Unit I: Introduction to Software Testing (04)**

- 1.1 Basics of Software Testing- Faults, errors and failures
- 1.2 Testing objectives
- 1.3 Principles of testing
- 1.4 Testing and debugging
- 1.5 Testing metrics and measurements
- 1.6 Verification and Validation
- 1.7 Testing Life Cycle

**Unit II: Software Testing Strategies & Techniques (07)**

- 2.1 Testability- Characteristics lead to testable software
- 2.2 Test characteristics
- 2.3 Test Case Design for Desktop, Mobile, Web application using Excel
- 2.4 White box Testing- Basics of path testing, Control Structure Testing
- 2.5 Black Box Testing- Boundary Value Analysis, Equivalence partitioning,

## Differences between BBT &amp; WBT

- Unit III: Levels of Testing** (07)
- 3.1 A strategic Approach to Software Testing
  - 3.2 Test strategies for conventional Software Unit testing
  - 3.3 Integration testing- Top-Down, Bottom-up integration
  - 3.4 System Testing- Acceptance, performance regression, Load/Stress testing, Security testing, Internationalization testing
  - 3.5 Alpha & Beta Testing
  - 3.6 Usability and accessibility testing
  - 3.7 Configuration, compatibility testing
- Unit IV: Testing Web Applications** (05)
- 4.1 Dimension of quality
  - 4.2 Error within a WebApp Environment Testing
  - 4.3 Strategy for WebApp Test Planning
  - 4.4 The Testing Process- an overview
- Unit V: Agile Testing** (04)
- 5.1 Agile Testing
  - 5.2 Difference between Traditional and Agile testing
  - 5.3 Agile principles and values
  - 5.4 Agile Testing Quadrants, Automated Tests
- Unit VI: Testing Different domains** (05)
- 6.1 Web Application Testing
  - 6.2 Finance domain Application testing
  - 6.3 E-Commerce testing
  - 6.4 Telecomm and Healthcare and insurance domain testing
  - 6.5 8 VI Automated soft

**Reference Books:**

1. Software Engineering- A Practitioners Approach, Roger S. Pressman, 7<sup>th</sup> Edition, Tata McGraw Hill, 20
2. Effective Methods of Software Testing, William E. Perry, 3<sup>rd</sup> Edition, Wiley Publication Inc.
3. Managing the Testing Process- Practical Tools and Techniques for Managing Hardware and Software Testing, Rex Black, Microsoft Press, 1999
4. Agile Testing: A Practical Guide for Testers and Agile Teams, Lisa Crispin and Janet Gregory, 1<sup>st</sup> Edition, Addison-Wesley Professional, 2008
5. Software Testing Principles and Practices By Srinivasan Desikan, Gopalswamy Ramesh, Pearson
6. Kshirsagar Naik and Priyadarshi Tripathy, "Software testing and quality assurance: Theory and Practice", A John Wiley & Sons Inc., ISBN 978-0-471-78911-6, 2008

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

<b>Semester – VI</b>	<b>Paper – XI</b>
<b>Course Code: BSC_CS 611 Pr</b>	<b>Title of the Course: Project</b>
<b>Credits: 02 Credits</b>	<b>Total Lectures: 30 Hrs.</b>

Teaching Scheme 03 Lect/ week/  
Examination Scheme

Batch Size : 20  
IA : 15 marks UE: 35 marks

**Project Guidelines:**

- Students should work in a team of minimum 3 and maximum 4 students.
- Students can choose a project topic and implement the same using any language/technology covered in the curriculum so far. The operating environment must be linux.
- The student group will work independently throughout the project work including: problem identification, information searching, literature study, design and analysis, implementation, testing, and the final reporting.
- Project guide must conduct project presentations (minimum 2) to monitor the progress of the project groups.
- At the end of the project, the group should prepare a report which should conform to international academic standards. The report should follow the style in academic journals and books, with clear elements such as: abstract, background, aim, design and implementation, testing, conclusion and full references, Tables and figures should be numbered and referenced to in the report.
- The final project presentation with demonstration (UE) will be evaluated by the project guide (appointed by the college) and one external examiner (appointed by the University).

**Recommended Documentation contents:**

**Abstract**

**Introduction**

- motivation
- problem statement
- purpose/objective and goals
- literature survey
- project scope and limitations

**System analysis**

- Existing systems
- scope and limitations of existing systems
- project perspective, features
- stakeholders



- Requirement analysis - Functional requirements, performance requirements, security requirements etc.

### System Design

- Design constraints
- System Model: Using OOSE
- Data Model
- User interfaces

### Implementation details

- Software/hardware specifications

### Outputs and Reports Testing

- Test Plan, Black Box Testing or Data Validation Test Cases, White Box Testing or Functional Validation Test cases and results

### Conclusion and Recommendations

### Future Scope

### Bibliography and References

## Project Related Assignments

### Guidelines:

- The project assignments are a compulsory part of the project course and should be carried out by each project group.
- Project assignments are to be given by the guide for continuous internal evaluation.
- The project assignments are to be allotted to each group separately by the project guide on the basis of the implementation technology. A suggested list of assignments is given below.
  1. Project Time management: plan (schedule table), Gantt chart, Roles and responsibilities, data collection, Implementation
  2. Simple assignments to evaluate choice of technology
  3. Assignments on UI elements in chosen technology
  4. Assignments on User interfaces in the project
  5. Assignments on event handling in chosen technology
  6. Assignments on Data handling in chosen technology
  7. Online and offline connectivity
  8. Report generation
  9. Deployment considerations
  10. Test cases

Each student within the group must work actively and contribute to the assignments, project work and report writing

### Evaluation guidelines:

IA (15 marks)			UE (35 marks)		
First presentation	Second presentation	Assignments	Project Logic/ Presentation	Assignments and Project Documentation	Viva
05	05	05	20	10	05

**NOTE: Students has to choice to opt for following DSEC courses with  
consent of Department**

**DSEC- 11: BSC-CS 602 T: Block Chain Technology**

**DSEC – 12: BSC – CS 603 T: Cloud Computing**

**DSEC – 18 Practical: BSC – CS 609 P Cloud Computing Practical**

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

<b>Semester – VI</b>	<b>Paper – II</b>
<b>Course Code: BSC_CS 602T</b>	<b>Title of the Course: Blockchain Technology</b>
<b>Credits: 02 Credits</b>	<b>Total Lectures: 36 Hrs.</b>

**Prerequisite:**

**Understanding of Object Oriented Programming Concepts Knowledge of Python**

**Course Objectives**

1. Understand what and why of blockchain technology.
2. Explore major components of blockchain.
3. Learn about Bitcoin, Cryptocurrency and Ethereum.

**To learn block chain programming using Python, Flask Web Framework, and HTTP client Postman.**

**Course Outcomes (COs):**

On completion of the course, student will be able to–

1. Learn the fundamentals of Blockchain Technology.
2. Learn Blockchain programming.
3. Basic knowledge of Smart Contracts and how they function.

**Course Contents**

<b>Unit-I</b>	<b>Introduction to Blockchain</b>	<b>(07)</b>
	1.1 Foundational Computing Concepts (Client-Server systems vs Peer to Peer Systems)	
	1.2 Evolution of Blockchain	
	1.3 Blockchain Vs Database	
	1.4 Essentials of Blockchain (Blockchain generations, types of blockchain, benefits and challenges of blockchain usage)	
	1.5 Types of Networks	
	1.6 Layered Architecture of Blockchain Ecosystem	
	1.7 Components of blockchain	
	1.8 Cryptography (private and public keys, Hashing & Digital Signature)	
	1.9 Consensus Mechanisms	
	1.10 Cryptocurrency, Digital Currency Bitcoin and Ethereum, Smart Contracts	
	Blockchain use cases	

**Unit-II                      How Blockchain Works?                      (05)**

- 2.1 Understanding SHA256 Hash
- 2.2 Immutable Ledger
- 2.3 Distributed P2P Network
- 2.4 How Mining Works? (The NONCE and Cryptographic Puzzle)
- 2.5 Byzantine Fault Tolerance
- 2.6 Consensus Protocols: Proof of Work, Proof of State, Défense Against Attackers, Competing Chains
- 2.7 Blockchain Demo

**Unit-III                      Smart Contracts                      (06)**

- 3.1 Ethereum Network
- 3.2 What is a Smart Contract?
- 3.3 Ethereum Virtual Machine, Ether, Gas DApps
- 3.4 Decentralized Autonomous Organizations (DAO)
- 3.5 Hard and Soft Forks
- 3.6 Initial Coin Offerings
- 3.7 Demo of Smart Contracts

**Demonstration    Programming Assignments:                      (18)**

**Out of 36 lectures, 18 are assigned for demonstration. Teacher should give demonstration of various programs mentioned below in the classroom or in the laboratory as per their convenience.**

**Assignment 1** –Demonstration of Blockchain

<https://andersbrownworth.com/blockchain>

**Assignment 2** – Installation of Ganache, Flask and Postman

**Assignment 3** –Write a Simple Python program to create a Block class that contains index, timestamp, and previous hash. Connect the blocks to create a Blockchain.

**Assignment 4** –Demo of Remix-Ethereum IDE <https://remix.ethereum.org> and Test Networks

**Assignment 5**–1. Write a Simple Smart Contract for Bank with withdraw and deposit functionality.

**Assignment 6** – 2. Write a Smart Contract for storing and retrieving information of Degree Certificates.

**Suggested Readings:****Textbook:**

1. **Beginning Blockchain : A Beginner's Guide to Building Blockchain Solutions By Bikramaditya Singhal, Gautam Dhameja, Priyansu Sekhar Panda, Apress Media**

**Reference Books:**

2. Mastering Blockchain by Imran Bashir, Third Edition, Packt Publication
3. Waterhole, The Science of the Blockchain
4. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System
5. Mastering Ethereum: Building Smart Contracts and DAPPS, by Andreas Antonopoulos, Dr.Gavid Wood, Oreilly Publication

**Reference Web Links**

1. <https://www.investopedia.com/terms/b/blockchain.asp>

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

<b>Semester – VI</b>	<b>Paper –III</b>
<b>Course Code: BSC-CS 603 T</b>	<b>Title of the Course: Cloud Computing</b>
<b>Credits: 02 Credits</b>	<b>Total Lectures: 45 Hrs.</b>

**Course Outcomes (COs):**

On completion of the course, student will be able to–

- To understand the principles and paradigm of Cloud Computing.
- To understand the Service Model with reference to Cloud Computing.
- To appreciate the role of Virtualization Technologies.
- Ability to design and deploy Cloud Infrastructure.
- Understand cloud security issues and solutions

**Detailed Syllabus:**

**Unit I: Introduction to Cloud Computing (10)**

- 1.8 Basic Concepts of Cloud Computing, Concept of Cloud Computing and its necessity.
- 1.9 Different Types of Model: Development model and service model.
- 1.10 Different Types of Clouds: Public, private, hybrid and community.
- 1.11 IaaS, PaaS, SaaS

**Unit II: Introduction to Computer Network Basics (10)**

- 2.6 What is networking?
- 2.7 Concept of distributed Systems.
- 2.8 Cloud Service Providers in use and their significance.
- 2.9 Cloud Infrastructure, Cloud Pros and Cons.

**Unit III: Cloud Storage Management (10)**

- 3.8 Concept of Virtualization and Load Balancing.
- 3.9 Key Challenges in managing information.
- 3.10 Identifying the problems of scale and management in big data.

**Unit IV: Study and need of Cloud Security (15)**

- 4.1 Infrastructure Security, Network level security, Host level security, Application level security.
- 4.2 Data privacy and security issues.
- 4.3 Access Control and authentication in cloud computing.

**Suggested Readings:**

1. Cloud Computing: Technologies and Strategies of the Ubiquitous Data Center, Brian J.S. Chee and Curtis Franklin, Jr., , CRC Press, ISBN :9781439806128.
2. Mastering Cloud Computing: Foundations and Applications Programming, Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, McGraw Hill, ISBN: 978 1259029950, 1259029956.
3. Distributed and Cloud Computing, From Parallel Processing to the Internet of Things, Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Morgan Kaufmann
4. Publishers, 2012.
5. Cloud Computing: Implementation, Management, and Security, John W.Rittinghouse and James F.Ransome, CRC Press, 2010.
6. Cloud Computing Principles and Paradigms- Rajkumar Buya, James Broberg, Andrzej Goscinski. Willey publication.
7. Mastering Cloud Computing by Rajkumar Buyya, C. Vecchiola & S. Thamarai SelviMcGRAW Hill Publication.
8. Cloud Computing: Web Based Applications that Change the Way You Work and Collaborate Online, Miller Michael, Pearson Education India

*Ahmednagar Jilha Maratha Vidya Prasarak Samaj's*

**New Arts, Commerce and Science College, Ahmednagar  
(Autonomous)**

**Syllabus of T.Y.B.Sc. (Computer Science)  
Under Faculty of Science**

Semester – V	Paper – IX
Course Code: BSC_CS 609 P	Title of the Course: Practical Course based on BSC-CS 608T.
Credits: 02 Credits	Total Lectures: 30Hrs.

**Course Outcomes (COs):**

On completion of the course, student will be able to–

- Describe the main concepts, key technologies, strengths, and limitations of cloud computing and the possible applications for state-of-the-art cloud computing
- Explain the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud, etc.
- Collaboratively research and write a paper on the state of the art (and open problems) in cloud computing.
- Identify problems, and explain, analyze, and evaluate various cloud computing solutions.
- Choose the appropriate technologies, algorithms, and approaches for the related issues.
- Display new ideas and innovations in cloud computing.

**Detailed Syllabus:**

1. Working and Implementation of Infrastructure as a service.
2. Working and Implementation of Software as a service.
3. Working and Implementation of Platform as a services.
4. Practical Implementation of Storage as a Service.
5. Working of Google drive to make spreadsheet and notes.
6. Working and Implementation of identity management.
7. Write a program for web feed.
8. Execute the step to Demonstrate and implementation of cloud on single sign on.
9. Practical Implementation of cloud security.
10. Installing and Developing Application Using Google App Engine.
11. Implement VMWareESXi Server
12. Using OpenNebula to manage heterogeneous distributed data center Infrastructure.
13. Implementation of Cloud Failure Cluster.
14. Managing and working of cloud xen server.
15. Working with Aneka and demonstrate how to Managing cloud computing Resources.
16. Installation and configuration of cloud Hadoop and demonstrate simple query.
17. Create a sample mobile application using Amazon Web Service (AWS) account as a cloud service. Also provide database connectivity with implemented mobile application.







