MAHAKAUSHAL

University

Faculty of & Computer Science Syllabus
& Prescribed Books
Subject-M.SC Data Science

Duration of Course: 2 Years

M.Sc. Data Science Semester Examination
2022-23

I, II,III & IV Semester

M.Sc. (Data Science) 1st semester

MDSC0101-T : Discrete Mathematical Structure

Unit No.	Topics
Unit 1	Set Theory: Introduction, Sets and Elements, Universal Set and Empty Set, Subsets, Venn Diagrams. Relations: Introduction, Product Sets, Relations, Pictorial Representation of Relations, Composition of Relations, Types of Relations, Partial Ordering Relations.
Unit 2	Functions: Introduction, One-to-One, Onto, and Invertible Functions, Cardinality. Logic and Propositional Calculus: Introduction, Propositions and Compound Propositions, Basic Logical Operations, Propositions and Truth Tables, Tautologies and Contradictions.
Unit 3	Counting: Introduction, Basic Counting Principles, Factorial Notation, Binomial Coefficients,Permutations and Combinations. Pigeon hole Principle.
Unit 4	Graph Theory: Introduction, Graphs and Multigraphs, Subgraphs, Paths, Connectivity, Weighted Graphs, Complete, Regular and Bipartite Graphs. Directed Graphs: Introduction, Rooted Trees, Graph Algorithms: Depth first and Breadth-First Searches.
Unit 5	TREES AND CUT - SETS : Paths and Circuits,ShortestPaths, Eulerian Pathsand Circuits, HamiltonianPathsand Circuits. Rooted Trees, PathLengthsin RootedTrees, BinarySearch Trees. Spanning Trees, Minimum Spanning Trees.

Reference books :

1.ElementsofDiscreteMathematics,C.L.Liu,SecondEdition,TMH
2. Discrete Mathematics and its applications, Kenneth H. Rosen, (Fifth Edition), Tata McGrawHill Publishing Company.
3. TheoryandProblemsofDiscreteMathematics,SemmourLipschutz,MarcLipson,Second Edition, Schaum‟s Outline, T.M.H.

M.Sc. (Data Science) 1st semester

MDSC0102-T:Operating System and System Software

Unit No.	Topics
Unit 1	Introduction to System Programs & Operating Systems, Evolution of Operating System (mainframe, desktop, multiprocessor, Distributed, Network Operating System, Clustered & Handheld System), Operating systemservices, operating systemstructure, System Call&System Boots,Operatingsystem design & Implementations, System protection, Buffering & Spooling. Types of Operating System:Bare machine, Batch Processing, Real Time, Multitasking & Multiprogramming, timesharing system.File: concepts, access methods, free space managements, allocation methods, directory systems, protection, organization,sharing & implementation issues.
Unit 2	Process: Concept, Process Control Blocks(PCB), Scheduling criteria Preemptive & non Preemptive process scheduling, Scheduling algorithms, algorithm evaluation, multiple processor scheduling, real time scheduling, operations on processes, threads, inter process communication, precedence graphs, critical section problem, semaphores, classical problems of synchronization. Deadlock: Characterization, Methods for deadlock handling, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock Process Management in Linux.
Unit 3	Memory Hierarchy, Concepts of memory management, MFT & MVT, logical and physical address space, swapping, contiguous and non-contiguous allocation, paging, segmentation, and paging combined with segmentation. Structure & implementation of Page table.Concepts of virtual memory, Cache Memory Organization, demand paging, page replacement algorithms, allocation of frames, thrashing, demand segmentation.
Unit 4	Mass Storage Structure: Disk Structure, Disk Scheduling- FCFS, SSTF, SCAN Scheduling, Disk Management, Swap-Space Management. Distributed operating system:-Types, Design issues, Filesystem, Remote file access, RPC, RMI, Distributed Shared Memory(DSM), Basic Concept of Parallel Processing &Concurent Programming.

Unit 5	System software and application software, layered organization of system software. Assemblers, Macros, Compilers, Cross compilers, Linking and loading, Relocation. Case study of Unix, Linux &Windows.

Reference books :

1. Operating Systems Concepts,A. Silberschatz, P.Galvin, G.Gagne,JohnWiley&Sons,Inc.
2. Systems Programming and Operating Systems (Part II - Operating Systems), Dhamdhere, 2ndEdition, TMH
3. Donovan,J.J.:Systemprogramming,McgrawHill,1972.
4. Dhamdhere.D.M.:Introductiontosystemsoftware,TataMcgrawHillPubl.comp.1986

MDSC0102-P :Operating System and System Software LAB

1. Introduction to Operating System Commands:
a. Familiarization with basic commands such as ls, cd, mkdir, rm, cp, mv in Unix/Linux or dir, cd, mkdir, del, copy, move in Windows.
b. Performing file and directory operations using command-line interfaces.

2. Process Management:
a. Writing shell scripts to automate process management tasks, such as starting, stopping, and monitoring processes.
b. Implementing process management utilities to list running processes, kill processes, and view process information.

3. Memory Management:
a. Writing programs to demonstrate memory allocation and deallocation using malloc and free in C/C++.
Analyzing memory usage and fragmentation in simulated memory environments.

4. File System Operations:
a. Implementing file system utilities to create, delete, read, and write files.
b. Demonstrating file manipulation operations such as copying, moving, and renaming files.

5. Shell Scripting:
a. Writing shell scripts to perform system administration tasks, automate backups, or manage user accounts.
b. Using conditional statements, loops, and functions in shell scripts.

6. Inter process Communication:
a. Implementing client-server communication using sockets in C/C++.
b. Building chat applications or file transfer utilities using TCP/IP or UDP protocols.

7. Device Management:
a. Writing device drivers or interfaces to interact with hardware devices such as printers, keyboards, or disk drives.
b. Implementing device management utilities to detect, configure, and control devices.

8. Concurrency and Synchronization:
a. Developing programs to demonstrate concurrency issues like race conditions and deadlocks.
b. Implementing synchronization mechanisms such as semaphores or mutexes to coordinate access to shared resources.

9. System Calls and API Usage:
a. Writing programs to demonstrate the usage of system calls for file I/O, process management, and interprocess communication.
b. Building applications using system APIs for networking, GUI development, or database access.

10. Virtua lization:
a. Setting up virtual machines using hypervisors like Virtual Box or VMware.
b. Installing and configuring guest operating systems, creating snapshots, and managing virtualized environments.

11. Security and Permissions:
a. Implementing access control mechanisms to restrict file access based on user permissions.
b. Configuring user accounts, groups, and permissions using tools like chmod, chown, and setfacl.

M.Sc. (Data Science) 1st semester

MDSC0103-T:Entrepreneurship Development

Unit No.	Topics
Unit 1	**Introduction:** Entrepreneurship meaning, nature ,importance, Specifictraits of Entrepreneurs, Role ofentrepreneurs in Indian Economy.
Unit 2	Analysis of Entrepreneur opportunities: Defining objectives, identification, process of sensing, accessing the impact of opportunities and threats.
Unit 3	**Search of Business Idea:** Preparing for businessplan, legal requirements forestablishin gofanew unit- procedure for registering business, starting of new venture, product designing / branding, researchand development, selection of forms of business organization.
Unit 4	RoleofSupportiveOrganizations:D.I.Candvariousgovernmentpoliciesforth edevelopmentof entrepreneurship, Government schemes and business assistance; subsidies, Role of Banks.
Unit 5	**Market Assessment :** Meaning of market assessment, components and dimensions of market assessment, Questionnaire preparations, survey of local market, Visit to industrial unit, business houses,servicesectoretc.SubmissionofSurveybasedreportononesuc cessful/oneunsuccessful entrepreneur.

Reference books :

1	Entrepreneurship Development	Dr. C.B. Gupta
2	Dynamic Entrepreneurial Development and Management	Vasant Desai
3	Innovation and Entrepreneurship	PeterF. Drucker
4	Entrepreneurship Development	G.A .Kaulgud
5	Entrepreneurship-Need of the Hour	Dr. Vidya Hattangadi
6	Entrepreneurship Development	DipeshD. Uike

M.Sc. (Data Science) 1st semester

MDSC0104-T :Object Oriented Programming Using C++

Unit No.	.Topics
Unit 1	**Object Oriented Systems Development :** Introduction to traditional programming with C. Objectives of OOP, Object Oriented Analysis, Object Oriented Programming in C++: Concepts of Objects, Classes, Data Abstraction, Encapsulation, Inheritance, Polymorphism, Dynamic Binding and Message passing.
Unit 2	Object modeling, Dynamic modeling, Events, Status, Scenarios, Event hate diagrams, Operations, State diagrams, Functional Models, Dataflow diagrams, Constraints specification, Relation of object, Functional and Dynamic models.
Unit 3	Tokens, Expressions and Control Structures, Classes and Objects, Overloading and information hiding, Function overloading, Operator overloading in C++,Memory Management: Constructors, Overloading of constructors, copy constructors, destructors.
Unit 4	**Inheritance :**Inheritance, Derived and base classes, Single, Multilevel, Hierarchical, Hybrid Inheritance, Protected member, overriding member function, class hierarchies, multiple inheritance, Containership
Unit 5	**Polymorphism :** virtual functions, late binding, pure virtual functions, abstract classes, friend functions, friend classes, static functions, this pointer, templates, function templates, Class templates.

Reference Books:

1. Object-Oriented Programming with C++:E. Balagurusamy, TMH,2005
2. Object-Oriented Programming in C++,Robert Lafore,Galgotia Publication.
3. Object-Oriented Programming,Tomothy Budd, Pearson education.
4. Object Oriented Modelling and Design, J. Rambaugh, M. Blaha, W. Premerlani, F. Eddy,W.Lorensen, P.H.I.

MDSC0104-P :Object Oriented Programming Using C++ LAB

1. Write a Program That Just Outputs `Hello World
2. Write a Program to Find Maximum and Minimum of Given 3 Numbers.
3. Write a Program That Output Value as Number and as Character.
4. Implementation of the Function That Calculates the Cross Sum of an Integer.(123 as 1+2+3).
5. Determine Number of Characters in a String.
6. Raising a Number N to a Power P is the Same as Multiplying N By Itself P Times. Write a Function Called Power ( ) That Takes a Double Value for N and an INT Value for P and Returns the Result as Double Value. use a Default Argument of 2 for P So That If This Argument is Omitted the Number Will Be Squared. Write a Main ( ) Function That Gets Values from the Userto Test This Function.
7. Write a C++ Program to Sort an Array of Integer in Ascending Order Using a Function Called Exchange( ) Which Accepts Two Integer Arguments By Reference.
8. Write a C++ Program to Implement Function Overloading in Order to Compute.
9. Write a C++ Program to Implement Power(MN) Where
   I) M is Double and N is Int II) M and N are Int.
10. Write a Program That Uses a Structure Called Point to Model a Point. Define Three Points and Have the User Input Values to Two of Them. Then Set the Third Point Equal to the Sum of the Other Two and Display the Value of the New Point. Interaction with the Program Might Look Like This:

    Enter Coordinates for P1: 3 4
    Enter Coordinates for P2: 5 7 Coordinates
    of P1 + P2 are: 8 11
11. Create the Equivalent of a Four Function Calculator. the Program Should Request the User to Enter a Number an Operator and Another Number. It Should Then Carry Out the Specified Arithmetical Operation: Adding Subtracting Multiplying Or Dividing the Two Numbers. (It Should use a Switch Statement to Select the Operation). Finally It Should Display the Result. When It Finishes the Calculation the Program Should Ask If the User Wants to Do Another Calculation. the Response Can Be Y Or N. Some Sample Interaction with the Program Might Look Like This.

    Enter First Number Operator Second Number: 10/ 3Answer
    = 3.333333
    Do Another (Y/ N)? Y
    Enter First Number Operator Second Number 12 + 100Answer = 112
    Do Another (Y/ N)? N
12. Create a 'Distance' Class with :
    - Feet and Inches as Data Members
    - Member Function to Input Distance
    - Member Function to Output Distance
    - Member Function to Add Two Distance Objects
    - Write a Main Function to Create Objects of Distance Class. Input Two Distances and Outputthe Sum.

13. Create a Class Called 'Time' That Has
    - Three Integer Data Members for Hours Minutes and Seconds
    - Constructor to Initialize the Object to Zero
    - Member Function to Add Two Time Objects
    - Member Function to Display Time in Hh:Mm:Ss Format
    - Write a Main Function to Create Two Time Objects Add Them and Display the Result inHh:Mm:Ss Format.

14. Create a Class Called 'Employee' That Has
    - Empcode and Empname as Data Members
    - Member Function Getdata( ) to Input Data
    - Member Function Display( ) to Output Data
    - Write a Main Function to Create Emp an Array of Employee Objects. Accept and

- Display the Details of At Least 6 Employees.

15. Create a Class Rational Which Represents a Numerical Value by Two Double Values- Numerator& Denominator. Include the Following Public Member Functions: Constructor with No Arguments (Default). Constructor with Two Arguments.
    - Void Reduce( ) That Reduces the Rational Number By Eliminating the Highest CommonFactor Between the Numerator and Denominator.
    - Overload + Operator to Add Two Rational Number.
    - Overload >> Operator to Enable Input Through Cin.
    - Overload << Operator to Enable Output Through Cout.
    - Write a Main ( ) to Test All the Functions in the Class.
16. Create a Class 'Complex' to Hold a Complex Number. Write a Friend Function to add TwoComplex Numbers. Write a Main Function to Add Two Complex Objects.
17. Create a 'Matrix' Class of Size M X N. Overload the „+" Operator to Add Twomatrix Objects.Write a Main Function to Implement It.
18. Create a 'String' Class Which Overloads „ = = ' Operator to Compare Two Stringobjects.
19. Create a Base Class Called 'Shape' Having
    - Two Data Members of Type Double.
    - Member Function *Get-Data*( ) to Initialize Base Class Data Members.
    - Pure Virtual Member Function *Display-Area*( ) to Compute and Display the Area of theGeometrical Object.

M.Sc. (Data Science) 1st semester

MDSC0105(A)-T :Computer Organization and Architecture

Unit No.	Topics
Unit 1	Binary Systems: Digital Computers and Digital Systems, Binary Numbers, Number Base Conversion, Octal and Hexadecimal Numbers, Complements, Binary Codes. Boolean Algebra and Logic Gates: Boolean Functions, Digital Logic Gates. Simplification of Boolean Functions: The Map Method, Two and Three Variable Maps, Four Variable Map, Product of Sums Simplification, NAND and NOR Implementation, Don‟t-Care Conditions.
Unit 2	Combinational Logic: Introduction, Design Procedure, Adders, Subtractors, Code Conversion, Analysis Procedure. Combinational Logic with MSI and LSI: Binary Parallel Adder, Decoders, Multiplexers. Sequential Logic: Introduction, Flip-Flops, Triggering of Flip-Flops.
Unit 3	Analysis of Clocked Sequential Circuits, State Reduction and Assignment, Flip-Flop Excitation Tables, Design Procedure, Design of Counters. Processor Logic Design: Introduction, Processor Organization, Arithmetic Logic Unit, Design of Arithmetic Circuit, Design of Logic Circuit, Design of Arithmetic Logic Unit, Status Register, Design of Shifter, Processor Unit.
Unit 4	Microcomputer System Design: Introduction, Microprocessor Organization, Basic Concept of Instruction, Instruction Types, Micro Instruction Format sand Addressing Modes, Subroutines Interrupt, Fetch and Execution cycle, Hardwired control unit, Micro-programmed Control unit- microprogram sequencer Control Memory, Sequencing and Execution of Micro Instruction.
Unit 5	Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory. Input Output Organization: Peripheral Devices, Input-Output Interface, Direct Memory Access (DMA), Input-Output Processors (IOP), Structure of Multiprocessor- Inter-processor Arbitration, Inter Processor Communication and Synchronization. Memory in Multiprocessor System, Concept of Pipelining, Vector Processing, Array Processing, RISC And CISC, Study of Multicore Processor – Intel, AMD.

Reference Books:

1. Digital Logic and Computer Design ,M.Morris Mano,P.H.I., Eastern Economy Edition.
2. Computer System Architecture(3rd ed..),M.Morris Mano,P.H.I., Eastern Economy Edition.
3. Computer Architecture and Organization, J.P.Hays,Mc GrawHill.
4. Digital Principle and Applications, Malvino and Leach
5. Digital Computer Fundamentals,Thomas C.Bartee

M.Sc. (Data Science) 1st semester

MDSC0105(B)-T :Techniques of Operation Research

Unit No.	Topics
Unit 1	Introduction: nature and meaning of O.R. Modelling in operations research, features of operationresearch ,scope of operations research. Linear Programming Problem: formulation of L.P.P. solutionof L.P.P. graphical method, simplex methods, duality.
Unit 2	Assignment problems: Mathematical formulation, Reduction theorem, methods of solving the assignments problems, Unbalanced assignment problem, Transportation problem: formulation, basic feasible solution: North-West-Corner method, least cost method, Vogel‟s approximation method, Optimum solution: Modi method.
Unit 3	Project management: introduction, network diagram representation, time estimates and critical path in network analysis, project evaluation and review techniques. Job sequencing: processing jobs through2 machines, processing n jobs through 3 machines, processing 2 jobs through m machines.
Unit 4	Queuing Theory: introduction, queuing system Transient and steady traffic inlets, Distribution of arrival distribution of departure, M/M/I: ∞/ FCFS model. Replacement problems: replacement policy for items whose maintenance cost increases with time and money value is constant.
Unit 5	Deterministic Inventory Models, what is inventory, types of inventory, inventory decisions, how to develop n variables model, costs involved in inventory problems, variables in inventory problem, classification of characteristics of inventory systems, EOQ model without shortage.

Reference Books:

1. Operations Researchby Taha.
2. Operations Researchby SD Sharma.
3. Introduction to Operations Research (SixthEdition) byF.S.HillierandG.J.Lieberman,Mc Graw Hill International Edition, Industrial Engineering Series, 1995.
4. Linear Programming by G. Hadley, Narosa Publishing House, 1995.

M.Sc. (Data Science) 2nd semester

MDSC0201-T: Advanced Machine Learning

Unit No.	Topics
Unit 1	**Foundations of Machine Learning:-** Introduction to Machine Learning, Probability and Statistics for Machine Learning, Linear Algebra for Machine Learning, Optimization Techniques.
Unit 2	**Supervised Learning:-** Regression Analysis, Classification Techniques, Support Vector Machines, Decision Trees and Random Forests, Ensemble Methods.
Unit 3	**Unsupervised Learning:-** Clustering Techniques, Dimensionality Reduction, Principal Component Analysis (PCA), K-means Clustering, Gaussian Mixture Models.
Unit 4	**Deep Learning:-** Neural Network Basics, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Autoencoders, Generative Adversarial Networks (GANs).
Unit 5	**Advanced Topics:-** Reinforcement Learning, Transfer Learning, Semi-supervised Learning, Bayesian Methods in Machine Learning, Ethical and Societal Implications of Machine Learning.

Reference Book:

"Pattern Recognition and Machine Learning" by Christopher M. Bishop
"Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy
"Introduction to Statistical Learning" by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani
"Elements of Statistical Learning" by Trevor Hastie, Robert Tibshirani, and Jerome Friedman
"Pattern Recognition and Machine Learning" by Christopher M. Bishop
"Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy
"Deep Learning" by Ian Good fellow, Yoshua Bengio, and Aaron Courville
"Neural Networks and Deep Learning: A Textbook" by Charu C. Aggarwal
"Reinforcement Learning: An Introduction" by Richard S. Sutton and Andrew G. Barto
"Bayesian Reasoning and Machine Learning" by David Barber

M.Sc. (Data Science) 2nd semester
MDSC0202-T Database Management System

Unit No.	Topics
Unit 1	**Introduction:** Advantages of DBMS approach, Various views of data, data independence, schema & sub-schema, primary concept of data models, database languages, transaction management, database administrator &user, data dictionary, database architectures. **ER model:** Basic concept, design issues, mapping constraint, keys, ER diagram, weak & strong entity-sets, specialization & generalization, aggregation, inheritance, design of ER schema, Reduction of ER Schema to tables. Domains, relation, kind of relation, Relational databases, Various types of keys: candidate, primary, alternate & foreign keys.
Unit 2	**Relational Algebra and SQL:** The structure, relational algebra with extended operations, modification of database, Idea of relational calculus. **Relational Database:** Basic structure of SQL, Set operation, Aggregate functions, Null values, Nested Sub queries, derived relations, views, Modification of database, join relation, Domain, relation & keys, DDL in SQL. Programming concepts of PL/SQL, Stored procedure, Database connectivity with ODBC/JDBC.
Unit 3	**Functional dependencies:** Basic definitions, Trivial & non trivial dependencies, closure set of dependencies & of attributes, Irreducible set of dependencies, FD diagram**.** **Normalization:** Introduction to normalization, non loss decomposition, First, second and third normal forms, dependency preservation, BCNF, multivalue dependencies and fourth normal form, join dependencies and fifth normal form. **Database Integrity:** general idea, integrity rules, Domain rules, Attributes rules, assertion, triggers, integrity & SQL
Unit 4	**Transaction Management:** basic concept, ACID properties, transaction state, Implementation of atomicity & durability, Concurrent execution, Basic idea of serializability. **Concurrency & Recovery:** Basic idea of concurrency control, basic idea of deadlock, Failure Classification, storage structure-types, stable storage implementation, data access, recovery & Atomicity: log based recovery, deferred database modification, immediate database modification, checkpoints.
Unit 5	**Storage Structure:** overview of physical storage media, magnetic disk: performance & optimization, RAID. **File Organization:** File organization, Organization of records in files, basic concept of Indexing, ordered indices: B+ tree & B tree index files. Query processing, Query optimization, Introduction to data mining and data warehousing.

Text Book :

"Database System concepts – Henry F. Korth, Tata McGraw Hill" 4th Edition

Reference books :

1. **"**Fundamentals of Database Systems", Elmasri R, Navathe S, Addison Wesley 4th Ed., ISBN 0321122267
2. An introduction to database system- Bipin C. Desai
3. An introduction to Database System - C.J Date
4. SQL, PL/SQL The programming language of Oracle- Ivan Bayross

Class Assignments:-

(Student should submit their assignment in hard copy (hand written) as well as soft copy to the respective faculty)

A. Prepare the following case study on ER diagram and normalized database design based on FD"s

(a) Database design for Retail Banking
(b) Database design for Technical Training Institute
(c) Database design for an Internet Book Shop
(d) Database design for Customer Order Warehouse
(e) Database design for University Registrar"s Office

B. Define the schema for the following databases with specific data type and constraints, the table name and its fields name are to be taken from database description which are given below :

1. Sales Information System

A database is being constructed for storing sales information system. A product can be described with a unique product number, product name, selling price, manufacturer name. The product can be sold to a particular client and each client have it own unique client number, client name, client addresses, city, pin code, state and total balance to be required to paid. Each client orders to buy product from the salesman. In the order, it has unique sales order number, sales order date, client number, salesman number (unique), billed whole payment by the party or not and its delivery date. The information associated with salesman is name, addresses, city, pin code, state, salary of the sales man, delivery date, total quantity ordered, product rate.

Write the SQL queries for the following –

a) Retrieve the list of names and the cities of all the clients.
b) List the various products available.
c) Find the names of all clients having „a" as the second letter in their names.
d) List all the clients who are located in „INDORE".
e) Find the products whose selling price is greater than 2000 and less than or equal to 5000.
f) Add a new column NEW_PRICE into the product _master table.
g) Rename the column product_ rate of Sales_ Order_ Details to new_ product_ rate.
h) List the products in sorted order of their description.
i) Display the order number and date on which the clients placed their order.
j) Delete all the records having delivery date before 25th March, 2010.
k) Change the delivery date of order number ON01008 to 16-05-10.

l) Change the bal_ due of client_ no CN01003 to 1200.
m) Find the product with description as „HDD1034" and „DVDRW".
n) List the names, city and state of the clients not in the state of „MP".
o) List of all orders that were cancelled in the month of March.

2. College Department Management

A student is described by a unique Roll Number, Name, Address, and Semester. Each student enrols himself/herself in an Academic programme offered by a Department. Academic programmes have programme name(unique), duration, a programme code(unique) and a list of subjects (both core and elective subject) while the departments have department code (unique), department name (unique), HoD who is a Teacher and list of courses offered by it. Each teacher is described by employee code (unique), name, department and designation. A student registers some courses in a semester. A course is described by a unique course number, title of the course, credit allotted for the course and offering department. Database stores the grades obtained by different students in different courses registered by him/her in different semesters. Database also stores information about the courses offered by a department ina semester, the corresponding teacher(s) for each course.

Write the SQL queries for the following –

a) Find all the students" name, city, course allotted from the SIT department. b) List the total number of Faculty in the SIT department.
c) List the available courses from the SIT department. d) List the all students in a particular semester.
e) List the students who earned CGPA greater than or equal to 8.5. f) How much subjects are registered by a student in each semester.
g) List the common students who are allotted the same courses of both the programme MCA and M. Tech.
h) List the total number of student enrolled in the subject DBMS. i) Retrieve the semester of the student under DBMS subject.
j) Retrieve the entire student name and arrange into ascending order.
k) Modify a student address DEWAS to INDORE where sdt_id="ITI08002".
l) Find the total credit point of student required to complete for a course like MCA. m) List the all courses which are related to computer science.
n) Retrieve all the students located at „INDORE".
o) Find the total number of department in our database.
p) List the all courses which are related to computer science.

MDSC0202-P Database Management System LAB

Database Management System LAB-

1. Introduction to SQL: Basic SQL queries including SELECT, INSERT, UPDATE, DELETE.
2. Table Creation and Data Manipulation: Creating tables, altering tables, inserting, updating, and deleting data.
3. Constraints and Keys: Implementing primary keys, foreign keys, unique constraints, and check constraints.
4. Joins: Performing INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL JOIN operations.
5. Aggregation and Grouping: Using GROUP BY and aggregate functions like SUM, AVG, COUNT, MAX, MIN.
6. Subqueries: Writing subqueries in SELECT, INSERT, UPDATE, DELETE statements.
7. Views: Creating and managing views to simplify complex queries and provide security.
8. Transactions and Concurrency Control: Understanding transaction management, COMMIT, ROLLBACK, and handling concurrency issues.
9. Indexes and Performance Tuning: Creating indexes, understanding query optimization, and analyzing query performance.
10. Stored Procedures and Triggers: Creating stored procedures and triggers to automate tasks and enforce data integrity rules.

M.Sc. (Data Science) 2nd semester
MDSC0-203T Communication Skill

Unit No.	Topics
Unit 1	**Introduction:**Definition, nature,objects,elementsandimportanceofcommunication,principlesandpractices, models of communication, types of communication,.
Unit 2	**Communication Skills and Soft Skills:** Interviewing and group discussion, resume preparation , etiquetteandmanners,self-management,bodyandsignlanguage,presentationskills,feedback&questioning technique: objectiveness in argument (Both one on one and in groups).
Unit 3	**Concept to Effective Communication :**Dimensions and directions of communication ,means of communication,7C"sfor effective communication.
Unit 4	**Listening Skills:** Importance of listening skills, good & bad listening, communication channels, types of communication medium- audio, video, digital, barriers of communication.
Unit 5	**Public Speaking and Reporting :**Effective Public Speaking and itsprinciples ,interpretation and techniques of report writing, letter writing, negotiation skills.

Text Book :

Suggested Reading:

BusinessCommunicationRoyanandV.lesikar,JohnD.Pettit,JR.RichardD.Irwin,I

NC Businesscommunication- K.K. Sinha

BusinessEtiquettes–DavidRobinson

BusinesscommunicationDr.NageshwarRaoandDr.R.P.D

as Effectivebusiness communication- Morphy Richards

M.Sc. (Data Science) 2nd semester
MDSC0204-T : Data Mining and data warehousing

Unit No.	Topics
Unit 1	**Introduction to Data Mining and Data Warehousing:** Overview of Data Mining and Data Warehousing, Data Preprocessing Techniques, Data Cleaning, Integration, and Transformation, Data Warehouse Architecture.
Unit 2	**Data Warehousing Design and Implementation:-** Dimensional Modeling, Fact and Dimension Tables, Star and Snowflake Schema, ETL (Extract, Transform, Load) Processes.
Unit 3	**Data Mining Techniques:-** Overview of Data Mining Algorithms, Classification and Prediction, Clustering Analysis, Association Rule Mining, Anomaly Detection.
Unit 4	**Advanced Data Mining Concepts:-** Text Mining and Sentiment Analysis, Web Mining, Time Series Analysis, Ensemble Learning Techniques, Big Data Analytics.
Unit 5	**Applications and Case Studies:-** Real-world Applications of Data Mining and Data Warehousing, Case Studies and Practical Examples, Ethical and Legal Issues in Data Mining, Future Trends in Data Mining and Data Warehousing.

Reference Book:

"Data Mining: Concepts and Techniques" by Jiawei Han, Micheline Kamber, and Jian Pei
"The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling" by Ralph Kimball and Margy Ross
"The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling" by Ralph Kimball and Margy Ross
"Building the Data Warehouse" by W. H. Inmon "Data Mining: Practical Machine Learning Tools and Techniques" by Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal
"Introduction to Data Mining" by Pang-Ning Tan, Michael Steinbach, and Vipin Kumar
"Data Mining: Concepts and Techniques" by Jiawei Han, Micheline Kamber, and Jian Pei
"Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking" by Foster Provost and Tom Fawcett.
"Data Mining Applications with R" by Yanchang Zhao
"Data Mining: Concepts and Techniques" by Jiawei Han, Micheline Kamber, and Jian Pei.

MDSC0204-P : Data Mining and data warehousing LAB

1. Data Preprocessing Techniques: Implementing data cleaning, integration, and transformation on a dataset.
2. Dimensional Modeling: Designing and creating fact and dimension tables using star schema or snowflake schema.
3. ETL Process: Building an ETL pipeline to extract data from multiple sources, transform it, and load it into a data warehouse.
4. SQL Queries for Data Analysis: Writing SQL queries to perform basic data analysis tasks such as filtering, grouping, and aggregating.
5. Association Rule Mining: Implementing Apriori algorithm to discover frequent itemsets and association rules from transactional data.
6. Classification and Prediction: Building and evaluating classification models using algorithms like decision trees, logistic regression, or Naive Bayes.
7. Clustering Analysis: Applying clustering algorithms such as K-means or hierarchical clustering to group similar data points.
8. Text Mining and Sentiment Analysis: Performing text mining tasks like tokenization, stemming, and sentiment analysis on textual data.
9. Web Mining: Extracting useful information from web data using techniques like web scraping and web usage mining.

M.Sc. (Data Science) 2nd semester
MDSC0205(A)-T : Theory of Computation

Unit No.	Topics
Unit 1	Automata: Basic machine, FSM , Transition graph, Transition matrix, DeterministicandnondeterministicFSM‟S,EquivalenceofDFAandNDFA, Mealy&Mooremachines,minimizationof finite automata, Two-way finite automata. Regular Sets and Regular Grammars: Alphabet, words, Operations, Regular sets, Finite automata and regular expression, Myhill-Nerode theorem Pumping lemma and regular sets, Application of pumping lemma, closure properties of regular sets.
Unit 2	Regular Expressions, Two-way Finite Automata, Crossing Sequence of Two way Finite Automata Finite Automata with Output, Applications of Finite Automata, Closure Properties of Regular Sets.
Unit 3	. Context Free Grammars: Motivation and Introduction, Context-free Grammars, Derivation trees and Ambiguity, Normal Forms (Chomsky Normal Form and Greibach Normal forms), Unit Production Chomsky Normal Forms, The existence of inherently ambiguous context-free languages, Closure properties of Context Free Languages, Construction of Reduced Grammars, Elimination of null production.
Unit 4	Pushdown Automata: Definition of PDA, Deterministic Pushdown Automata, PDA corresponding to given CFG, CFG corresponding to a given PDA. Context Free Languages: The pumping lemma for CFL‟s, Closure properties of CFL‟s, Decision problems involving CFL‟s.
Unit 5	Turing Machines: Introduction, TM model, representation and languages acceptability of TM Design of TM,Universal TM & Other modification, Church‟s hypothesis, composite & iterated TM. Turing machineas enumerators. Propertiesof recursive & recursively enumerable languages, Universal Turing Machine.

Reference Books:

1. IntroductiontoAutomataTheory,Languages&Computation,JEHopcraft&JDUllman, Narosa Publications.
2. Theory of Computer Science, KLP Mishra &N ChandraSekhar, PHI
3. Mathematical Foundations of Computer Science, Beckman
4. JohnCMartin,"Introdutiontolanguagesandtheoryofcomputation",McGrawHill
5. Anami & Aribasappa, "Formal Languages and Automata Theory", Wiley India

M.Sc. (Data Science) 2nd semester
MDSC0205(B)-T :Internet Programming

Unit No.	Topics
Unit 1	Introduction to Internet Programming- Client-Server model, Browsers- Graphical and Hypertext Access to the Internet, HTTP–Hyper Text Transfer Protocol (how it actually works), The Phases of Web Site Development
Unit 2	Creating Internet World Wide Web pages- HTML - Hypertext Markup Language , Basic HTML Concepts, HTML: Structured Language ,headers, body, html tags, tables , Text, graphics, sounds, video clips, multi- media ,Client side image mapping
Unit 3	HTML forms programming: Building a form, Text fields and value, size, max length html buttons, radio, checkboxes, Selection lists. CSS: Introduction To Style sheet, types of style sheets- Inline, External, Embedded CSS, text formatting properties, CSS Border, margin properties, Positioning Use of classes in CSS, color properties, use of &
Unit 4	Intro to script, types, intro of JavaScript, JavaScript identifiers, operators, control &Looping structure, Intro of Array, Array with methods, Math, String, Date Objects with methods User defined & Predefined functions, DOM objects, Window Navigator, History, Location, Event handling, Validations On Forms
Unit 5	Intro & features of XML, XML writing elements, attributes etc. XML with CSS, DSO, XML Namespaces XML, DTD, XML Schemas, Writing Simple sheets using XSLT, SAX & DOM Parsers, SOAP Introduction.

Reference Books:

1. JoeFawcett, Danny Ayers, LiamR.E.Quin,"BeginningXML"Wrox Press,5thEd.,2012
2. Deitel & Deitel,"XML howtoprogram" ,Pearson, 2000
3. Hofstetterfred,"InternetTechnologyatwork",Osbornepub.,ISBN:9780072229998, 2004
4. IvanBayross,"HTML, DHTML,JavaScript,Perl&CGI",BPBpub.3rd Ed.,2004
5. IvanBayross,"WebenabledcommercialapplicationdevelopmentusingHTML,DHTML, JavaScript, PERL-CGI", BPB pub., 2nd Ed., 2000

M.Sc. (Data Science)3rd semester
MDSC0301-T RDBMS and NOSQL

Unit No.	Topics
Unit 1	**Introduction to Databases:-** Overview of Databases and Database Management Systems, Evolution of Database Systems, Types of Databases: RDBMS, No SQL, New SQL, ACID and CAP Theorem.
Unit 2	**Relational Database Management Systems (RDBMS):-** Relational Model and Relational Algebra, SQL Basics: Data Definition Language (DDL) and Data Manipulation Language (DML), Normalization and Database Design, Transaction Management and Concurrency Control.
Unit 3	**No SQL Databases: Key-Value Stores and Document Stores:-** Introduction to NoSQL Databases, Key-Value Stores: Redis, DynamoDB, Document Stores: MongoDB, Couchbase, Data Modeling in NoSQL Databases.
Unit 4	**NoSQL Databases: Column-Family Stores and Graph Databases:-** Column-Family Stores: Apache Cassandra, HBase, Graph Databases: Neo4j, Amazon Neptune, Use Cases and Applications of NoSQL Databases.
Unit 5	**Advanced Topics and Case Studies:-** Polyglot Persistence, Data Warehousing with NoSQL Databases, Distributed Database Systems, Case Studies and Real-world Applications.

Reference Book:

"Database Systems: The Complete Book" by Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom
"Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses" by Michael Minelli, Michele Chambers ,and Ambiga Dhiraj, "Database Management Systems" by Raghu Ramakrishnan and Johannes Gehrke "SQLforMereMortals"byJohnL.ViescasandMichaelJ.Hernandez, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence" by Pramod J. Sadalage and Martin Fowler
"ScalingMongoDB"byKristinaChodorow, "Cassandra: The Definitive Guide" by Jeff Carpenter and Eben Hewitt "GraphDatabases"byIanRobinson,JimWebber,andEmilEifrem"Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems" by Martin Kleppmann "Big DataAnalytics:TurningBigDataintoBigMoney"byFrankJ.Ohlhorst

M.Sc. (Data Science)3rd semester
MDSC0301-P RDBMS and NOSQL

1. RDBMS: SQL Basics
•Creating and manipulating tables using SQL (e.g., CREATE TABLE, INSERT, UPDATE, DELETE).
2. RDBMS: Querying and Joins
•Writing SQL queries to retrieve data from multiple tables using INNER JOIN, LEFT JOIN, etc.
3. RDBMS: Normalization
•Normalizing a denormalized database schema to third normal form (3NF).
4. RDBMS: Transactions and Concurrency Control
•Implementing transactions using SQL (BEGIN TRANSACTION, COMMIT, ROLLBACK) to maintain data consistency.
5. NoSQL: Key-Value Stores
•Working with a key-value store database (e.g., Redis) to store and retrieve data.
6. NoSQL: Document Stores
•Using a document store database (e.g., MongoDB) to insert, update, and query documents.
7. NoSQL: Column-Family Stores
•Exploring a column-family store database (e.g., Apache Cassandra) and performing CRUD operations.
8. NoSQL: Graph Databases
•Working with a graph database (e.g., Neo4j) to create nodes, relationships, and execute graph queries.
9. Data Integration: RDBMS and NoSQL
•Integrating data between an RDBMS and a NoSQL database using ETL processes.
10. Data Warehousing with RDBMS and NoSQL
•Designing and implementing a data warehousing solution using both RDBMS and NoSQL databases.

M.Sc. (Data Science)3rd semester
MDS0302-T: Cloud native and development

Unit No.	Topics
Unit 1	**Introduction to Cloud-Native Development:-** Understanding Cloud-Native Architecture, Principles of Microservices, Containers and Orchestration, DevOps Culture and Practices, Cloud-Native Tools and Technologies Overview.
Unit 2	**Containerization and Orchestration:-** Introduction to Docker and Containerization, Container Orchestration with Kubernetes, Container Networking and Storage, Managing Containerized Applications.
Unit 3	**Microservices Architecture:-** Principles of Microservices, Service Decomposition and Domain-Driven Design (DDD), API Gateways and Service Meshes, Service Discovery and Load Balancing.
Unit 4	**Cloud-Native Development Tools and Technologies:-** Serverless Computing, Continuous Integration and Continuous Delivery (CI/CD), Infrastructure as Code (IaC), Observability and Monitoring in Cloud-Native Applications.
Unit 5	**Cloud-Native Security and Governance:-** Security Best Practices in Cloud-Native Applications, Identity and Access Management (IAM), Compliance and Governance in the Cloud, Security Automation and Threat Detection.

Reference Books:

"Cloud Native Patterns: Designing Change-tolerant Software" by Cornelia Davis
"Cloud Native Application Architecture" by Boris Scholl, Trent Swanson, and Ralph Squillace
"Docker Deep Dive" by Nigel Poulton
"Kubernetes: Up and Running" by Kelsey Hightower, Brendan Burns, and Joe Beda
"Building Microservices" by Sam Newman
"Domain-Driven Design: Tackling Complexity in the Heart of Software" by Eric Evans
"Serverless Architectures on AWS" by Peter Sbarski
"Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation" by Jez Humble and David Farley
"Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance" by Tim Mather, Subra Kumaraswamy, and Shahed Latif
"Security Engineering: A Guide to Building Dependable Distributed Systems" by Ross J. Anderson

M.Sc. (Data Science)3rd semester
MDSC303-T: Big data Technologies

Unit No.	Topics
Unit 1	Introduction Introduction to Big Data Platform – Challenges of Conventional Systems - Intelligentdata analysis – Nature of Data - Analytic Processes and Tools - Analysis vs Reporting – Modern Data AnalyticTools.BigDataAnalyticsProcess,BigDataAnalyticsforBusiness.Identifyingproblemand solving problem in Big Data environment. Analyzing Unstructured vs. Structured Data, Databases.
Unit 2	Hadoop and MapReduce Introduction to Hadoop, Hadoop architecture, A Brief History of Hadoop, Apache Hadoop and the Hadoop Ecosystem, Hadoop Releases; Hadoop Distributed File system: Design of HDFS, HDFS Concepts.
Unit 3	IntroductiontoMapReduce:MapReduceBasicConcepts,UnderstandingtheMapReducearchitecture,Writing MapReduce Programs. understanding Map phase, shuffling, sorting, and reducing phase.
Unit 4	Spark Introduction to Spark, Resilient Distributed Dataset (RDD), RDD Operations: actions and transformation functions. Spark Data frames, operations on Data frames: Join, group by, aggregate, handling missing data.
Unit 5	Sparks and MLLib Sparks and its basic operations. MLLib: Data types, Basic statistics,Classification(Logistic regression, Decision tree classifier)and linear regression model generation, Model Evaluation, Collaborative filtering, and Clustering.

Reference Books:

1. ArvindSathi,"BigDataAnalytics:DisruptiveTechnologiesforChangingtheGame",1stEdition, IBM Corporation, 2012.

2. BillFranks,"TamingtheBigDataTidalWave:FindingOpportunitiesinHugeDataStreamswith Advanced Analytics", 1st Edition, Wiley and SAS Business Series, 2012.

3. TomWhite,"Hadoop:TheDefinitiveGuide",3rdEdition,O'ReillyMedia, 2012

4. DonaldMiner,AdamShook,EricSammer,"HadoopOperation",O"Reilly2012.

5. DonaldMiner,AdamShook"MapReduceDesignPatterns:BuildingEffectiveAlgorithms and Analytics for Hadoop and Other Systems", O"Reilly 2012.

6. ChuckLam,"HadoopinAction",ManningPublications,2010.

7. https://spark.apache.org/docs/2.0.0/programming-guide.html

MDSC303-P: Big data Technologies LAB

1. Introduction to Hadoop Ecosystem:
- Setting up a Hadoop cluster using Apache Hadoop or a distribution like Cloudera or Hortonworks.
- Writing and executing MapReduce programs to process large datasets.
2. Apache Spark Fundamentals:
- Introduction to Apache Spark and its core concepts like Resilient Distributed Datasets (RDDs) and transformations.
- Writing Spark applications in Scala or Python to perform data processing tasks.
3. Data Ingestion and Streaming:
- Setting up Apache Kafka for real-time data ingestion.
- Building streaming data pipelines with Apache Kafka and Apache Spark Streaming or Apache Flink.
4. Data Processing with Apache Hive:
- Introduction to Apache Hive for querying and analyzing data stored in Hadoop Distributed File System (HDFS).
- Writing HiveQL queries to perform data analysis tasks like aggregation, filtering, and joins.
5. Data Visualization with Apache Zeppelin:
- Setting up Apache Zeppelin for interactive data visualization and analysis.
- Creating notebooks in Apache Zeppelin to visualize data from various sources including HDFS and Apache Spark.
6. Machine Learning with Apache Mahout:
- Introduction to Apache Mahout for scalable machine learning algorithms.
- Implementing machine learning algorithms such as clustering, classification, and recommendation using Apache Mahout.
7. Data Warehousing with Apache Druid:
- Setting up Apache Druid for real-time analytics and data warehousing.
- Creating data ingestion tasks and querying data stored in Apache Druid.
8. Graph Processing with Apache Giraph:
- Introduction to Apache Giraph for processing large-scale graph data.
- Implementing graph algorithms such as PageRank and shortest path in Apache Giraph.
9. Distributed Deep Learning with TensorFlow and Apache Spark:
- Setting up TensorFlow on a distributed cluster for training deep learning models.
- Integrating TensorFlow with Apache Spark for distributed deep learning tasks.
10. Data Security and Governance:

- Implementing data security measures such as encryption, access control, and authentication in a big data environment.
- Setting up data governance policies and tools for data quality management and compliance.

M.Sc. (Data Science)3rd semester
MDSC0304-T Advanced java Programming

Unit No.	Topics
Unit 1	**JAVABASICSREVIEW Java streaming**-Networking-Event handling - Multithreading – Byte code Interpretation - Customizing application - Data Structures - Collection classes.
Unit 2	**DISTRIBUTED COMPUTING Custom sockets** - Remote Method Invocation - Activation - Object serialization -Distributed garbage collection - RMI - IIOP - Interface definition language – CORBA-JINI overview.
Unit 3	**JAVA BEANS AND SWING Bean concepts** - Events in bean box -Bean customization - Persistence - Application - deployment using swing - Advanced swing techniques - JAR file handling.
Unit 4	**JAVAENTERPRISEAPPLICATIONS**-JNI-Ser vlets-Java Server Pages - JDBC - Session beans - Entity beans - Programming and deploying enterprise Java Beans - Java transactions. RELATEDJAVATECHNIQUES.
Unit 5	**Graphics** Java Media Frame work - 3D graphics - Internationalization - Case study - Deploying n-tierapplication, E- commerce applications.

REFERENCES:

Deitel&Deitel,"JavaHowtoprogram",PrenticeHall,4thEdition,2000.

GaryCornellandCayS.Horstmann,"CoreJavaVol1andVol2",Sun Microsystems Press, 1999.

StephenAsbury,ScottR.Weiner,Wiley,"DevelopingJavaEnterpriseApplications", 1998.

MDSC0304-P Advanced java Programming LAB

1. Multithreading and Concurrency:

•Implement multithreaded programs using Java's Thread class and Runnable interface.

•Explore synchronization mechanisms such as synchronized blocks and methods to ensure thread safety.

•Practical exercises on solving concurrency issues like race conditions and deadlocks.

2. Networking and Socket Programming:

•Develop client-server applications using Java's Socket API for network communication.

•Implement protocols like TCP and UDP for reliable and connectionless communication.

•Practical exercises on creating chat applications, file transfer utilities, or network-based games.

3Java Database Connectivity (JDBC):

•Integrate Java applications with relational databases using JDBC.

•Perform database operations like querying, updating, and executing stored procedures.

•Practical exercises on connecting to a database, executing SQL queries, and handling result sets.

4. Java EE Technologies:

•Explore Java Enterprise Edition (Java EE) technologies like Servlets, JavaServer Pages (JSP), and JavaServer Faces (JSF).

•Develop web applications using Servlets and JSP for server-side processing and dynamic content generation.

•Practical exercises on building web forms, handling user input, and displaying dynamic content.

5. Enterprise Application Development with Spring Framework:

•Introduction to the Spring Framework for building enterprise Java applications.

•Implementing dependency injection, aspect-oriented programming (AOP), and inversion of control (IoC) with Spring.

•Practical exercises on developing Spring-based web applications, RESTful services, and data access layers.

6. Web Services Development with JAX-RS and JAX-WS:

•Implement RESTful and SOAP web services using Java API for RESTful Web Services (JAX-RS) and Java API for XML Web Services (JAX-WS).

•Practical exercises on creating and consuming web services for inter-application communication.

7. JavaFX GUI Development:

•Build modern and rich graphical user interfaces (GUIs) using JavaFX.

•Create interactive UI components, layouts, and animations with JavaFX.

•Practical exercises on developing desktop applications with advanced UI features using JavaFX.

8. Integration Testing and Test-Driven Development (TDD):

•Implement unit tests and integration tests for Java applications using frameworks like JUnit and

Mockito.

•Practice test-driven development (TDD) by writing tests before implementing code.

•Practical exercises on writing test cases, mocking dependencies, and ensuring code quality through automated testing.

9. Security and Authentication in Java Applications:

•Explore security features and best practices for Java applications, including authentication, authorization, and encryption.

•Implement authentication mechanisms like username/password, OAuth, or JWT (JSON Web Tokens).

•Practical exercises on securing web applications, protecting sensitive data, and preventing common security vulnerabilities.

10. Asynchronous Programming with CompletableFuture and Reactive Streams:

•Learn asynchronous programming techniques using CompletableFuture and Reactive Streams in Java.

•Implement non-blocking I/O operations and reactive programming patterns for handling asynchronous events.

•Practical exercises on developing responsive and scalable applications using asynchronous programming paradigms.

Class Assignments:

1. Write a program that produces the following output:Hello World!
   "It's been nice knowing you ".
   Goodbye world!
2. State the order of evaluation of the operations in each of the following Java statements and implement them to show the value of x after each statement.
   x = 7 + 3 * 6 / 2 – 1;
   x = 2 % 2 + 2 * 2 – 2 / 2;
   x = (3 * 9 * (3 + (9 * 3 / (3))));
3. Write an application that declares 5 integers, determines and prints the largest and smallest inthe group.
4. Write an application that takes 3 parameters as sides of triangle and calculate area of triangle.
5. Write an application that declares two integers, determines whether the first is a multiple ofthe second and print the result. [Hint: Use the remainder operator.]
6. Write a program to find all prime numbers between 100 to 1000.
7. Write a program to check whether the given number is palindrome or not.
8. Write an application that evaluates the factorial of the integers from 1 to 5.
9. Write a program that accepts an integer from user and check whether the number is Fibonaccinumber or not.
10. Read a positive integer value, and compute the following sequence: If the number is even, halve it; if it's odd, multiply by 3 and add 1. Repeat this process until the value is 1, printing out each value. Finally print out how many of these operations you performed.
    Typical output might be:
    Initial value is 9

Next value is 28
Next value is 14
Next value is 7
Next value is 22
Next value is 11
Next value is 34
Next value is 17
Next value is 52
Next value is 26
Next value is 13
Next value is 40
Next value is 20
Next value is 10
Next value is 5
Next value is 16
Next value is 8
Next value is 4
Next value is 2
Final value 1, number of steps 19

If the input value is less than 1, print a message containing the word Error and perform an exit(0);

11. Write a program which will read an integer value for a base, then read a positive integer written to that base and print its value. Read the second integer a character at a time; skip over any leading nonvalid (i.e. not a digit between zero and ``base-1") characters, then read valid characters until an invalid one is encountered.

Input Output
========== ======
10 1234 1234
8 77 63 (the value of 77 in base 8, octal)
2 1111 15 (the value of 1111 in base 2, binary)

The base will be less than or equal to 10.

12. Write an application that uses *String* method *compareTo* to compare two strings defined by the user.
13. Write an application that uses String method *equals* and *equalsIgnoreCase* to tests any two string objects for equality.
14. Write an application that uses *String* method *indexOf* to determine the total number of occurrences of any given alphabet in a defined text.
15. Write an application that uses *String* method *concat* to concatenate two defined strings.
16. Write an application that finds the length of a given string.
17. Write an application that uses *String* method *charAt* to reverse the string.
18. Write an application that finds the substring from any given string using *substring* method and *startsWith* & *endsWith* methods.
19. Write an application that changes any given string with uppercase letters, displays it , changes it back to lowercase letters and displays it.
20. Create a class called *Employee* that includes three pieces of information as instance variables – a first name (type String), a last name (type String) and a monthly salary (double)

21. Write an application that uses *String* method *compareTo* to compare two strings defined by the user.
22. Write an application that uses String method *equals* and *equalsIgnoreCase* to tests any two string objects for equality.
23. Write an application that uses *String* method *indexOf* to determine the total number of

occurrences of any given alphabet in a defined text.

24. Write an application that uses *String* method *concat* to concatenate two defined strings.
25. Write an application that finds the length of a given string.
26. Write an application that uses *String* method *charAt* to reverse the string.
27. Write an application that finds the substring from any given string using *substring* method and *startsWith* & *endsWith* methods.
28. Write an application that changes any given string with uppercase letters, displays it , changes it back to lowercase letters and displays it.
29. Create a class called *Employee* that includes three pieces of information as instance variables – a first name (type String), a last name (type String) and a monthly salary (double)

    write application that take a series of integers in pair(e.g. (a,b)) and find the following

    a. Is first one is greater
    b. Is both are even
    c. They are relatively prime or not
30. Write an application that uses *String* method *compareTo* to compare two strings defined by the user.
31. Write an application that uses String method *equals* and *equalsIgnoreCase* to tests any two string objects for equality.
32. Write an application that uses *String* method *indexOf* to determine the total number of occurrences of any given alphabet in a defined text.
33. Write an application that uses *String* method *concat* to concatenate two defined strings.
34. Write an application that finds the length of a given string.
35. Write an application that uses *String* method *charAt* to reverse the string.
36. Write an application that finds the substring from any given string using *substring* method and *startsWith* & *endsWith* methods.
37. Write an application that changes any given string with uppercase letters, displays it , changes it back to lowercase letters and displays it.
38. Create a class called *Employee* that includes three pieces of information as instance variables – a first name (type String), a last name (type String) and a monthly salary (double)
39. Create a constructor in above class to initialize the three instance variables. Provide a get method for each instance variable.
40. Write a test application named *EmployeeTest* that demonstrates class Employee‟s capabilities. Create two employee objects and display each object‟s yearly salary.
41. Give each employee a 10% raise and display each Employee‟s yearly salary again.
42. Create a class *Account* with an instance variable *balance* (double). It should contain a constructor that initializes the *balance*, ensure that the initial balance is greater than 0.0.

M.Sc. (Data Science)3rd semester
MDSC0305(A) –T Machine Learning

Unit No.	Topics
Unit 1	Learning Problems - Perspectives and Issues - Concept Learning - Version Spaces and Candidate Eliminations-Inductivebias-DecisionTreelearning-RepresentationAlgorithm-HeuristicSpace Search.
Unit 2	NeuralNetworkRepresentation-Problems-Perceptrons– MultilayerNetworksandBackropagation Algorithms-Advanced Topics-Genetic Algorithms –Hypothesis Space Search Genetic Programming Models of Evalution and Learning.
Unit 3	Bayes Theorem-Concept Learning-Maximum Likelihood–Minimum Description Length Principle-Bayes Optimal Classifier - Gibbs Algorithm - Naïve Bayes Classifier – Bayesian Belief Network –EM Algorithm-Probability Learning-Sample Complexity-Finite and Infinite Hypothesis Spaces- Mistake Bound Model.
Unit 4	K-Nearest Neigh bour Learning-Locally weighted Regression- Radial Bases Functions-Case Based Learning.
Unit 5	Learning Set sofRules-SequentialCoveringAlgorithm-LearningRuleSet-FirstOrderRules-Sets of First Order Rules - Induction on Inverted Deduction - Inverting Resolution -Analytical Learning - Perfect Domain Theories - Explanation Base Learning – FOCL Algorithm-Reinforcement Learning - Task - Q-Learning - Temporal Difference Learning

TEXTBOOKS:

1. Machine Learning-Tom M. Mitchell,-

MGHREFERENCE BOOKS

1. Machine Learning: An Algorithmic Perspective, Stephen Marsl and, Taylor& Francis

M.Sc. (Data Science)3rd semester
MDSC0305(B)-T Cloud Computing

Unit No.	Topics
Unit 1	**Cloud Service Models**: Overview of IaaS, PaaS, and SaaS. , Security and management challenges across these models. **Cloud Resource Management**: Provisioning and managing cloud resources efficiently. Techniques for monitoring, scaling, and optimizing cloud resources.
Unit 2	**Service Level Agreements (SLAs) in Cloud**: Key concepts of SLAs (performance, availability, reliability). , How to define and negotiate SLAs with cloud providers. **Cost Management in Cloud**: Techniques for managing and optimizing cloud costs. , Tools for monitoring and forecasting cloud expenditures.
Unit 3	**Cloud Automation and Orchestration**: Tools for automating cloud infrastructure management (e.g., AWS Cloud Formation, Terra form)., Orchestrating multi-cloud environments and managing complex workflows. **Cloud Governance Models**: How governance is applied in a cloud environment., Policies and frameworks to ensure cloud operations are secure and compliant.
Unit 4	**Risk Management in Cloud**: Identifying risks in cloud computing environments., Techniques for assessing, mitigating, and managing cloud-related risks. **Auditing Cloud Services**: How to audit cloud providers and services to ensure compliance with regulations., Tools and best practices for cloud auditing.
Unit 5	**Legal and Ethical Issues in Cloud Computing**: Legal implications of using cloud services., Intellectual property rights, data sovereignty, and jurisdiction issues in the cloud.

Reference Books for Cloud Computing

1. Thomas Erl, Zaigham Mahmood, and Ricardo Puttini
2. Satyanand (S. S.) and Ranjan Parekh
3. Barrie Sosinsky
4. Tim Mather, Subra Kumaran, and Shahed Latif

M.Sc. (Data Science)4th semester

MDSC0401-T Advance statistical Model and Analysis

Unit No	TOPIC
Unit 1:	Probability and probability distribution probability : classical ,relative frequency and axiomatic definition of probability, Bayes Theorem and Independence. Probability distribution: binomial,, geometric, negative binomial uniform exponential, normal and log normal distribution.
Unit 2	Introduction tonon-linear modelling; sampling methods:basic sampling algorithms, rejections sampling, adaptive rejection sampling.
Unit 3	Linear statistical models; multiple linear regression, inference technique for the general linear model, genera lised linear models, inference procedure, special case of generalized linear models leading to Logistic regression and log linear model.
Unit 4	Random variables: discrete, continuous and mixed random variable, probability mass, probability density and cumulative distribution functions, mathematical expectation ,moments median, quartiles, Markov inequality, correlation and , independence of random variable.
Unit 5	Hypothesis testing basic: ideas of testing hypothesis, null and alternative hypothesis, the critical and acceptance regions, two types of errors, Test for one sample and two sample problems for normal population ,test for proportions.

Refrences Book:

1. Krzanowski, W.J.,An Introduction to Statistical Modeling, Wiley(2010).
2. Hastie,T .,Tibshirani,R. ,and Friedman,J., The Elements of Statistical Learning: Data Mining,Inference, and Prediction, Springer (2002).

3. Bishop, C.M., Pattern Recognition and Machine Learning, Springer(2006).

MDSC0401-P Advance statistical Model and Analysis lab

1. Exploratory Data Analysis (EDA) of Time Series Data:
- Load and preprocess time series data.
- Visualize time series data using line plots, histograms, and box plots.
- Analyze trends, seasonality, and stationarity using decomposition techniques and statistical tests.

2. Time Series Forecasting with Classical Methods:
- Implement classical forecasting methods such as Moving Average (MA), Exponential Smoothing (ETS), and Autoregressive Integrated Moving Average (ARIMA).
- Tune model parameters using techniques like grid search or auto ARIMA.
- Evaluate forecast accuracy using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

3. Advanced Forecasting Models:
- Introduction to advanced forecasting models like Seasonal ARIMA (SARIMA), Seasonal Exponential Smoothing (ETS), and Prophet.
- Implement these models and compare their performance with classical methods.

4. Machine Learning Approaches for Time Series Forecasting:
- Introduction to machine learning algorithms such as Support Vector Machines (SVM), Random Forests, and Gradient Boosting for time series forecasting.
- Preprocess time series data for machine learning models (e.g., feature engineering, rolling window approach).
- Implement machine learning models and optimize hyperparameters using techniques like cross-validation.

5. Evaluation and Validation of Forecasting Models:
- Split time series data into training and testing sets.
- Evaluate forecasting models on testing data using appropriate metrics.
- Perform cross-validation to ensure model robustness.

6. Model Interpretation and Visualization:
- Visualize forecasted values along with actual values.
- Interpret model results and understand the implications of forecasting outputs.
- Use diagnostic plots to assess model performance and identify areas for improvement

M.Sc. (Data Science)4th semester

MDSC0402-T Advanced python programming for spatial analytics

Unit No	TOPIC
Unit 1:	**Introduction:** check icon History, Features, Settingup path, Variable and Data Types, Operator. **Conditional Statements:** if, if-else, if-elif, nested if-else and Looping: for, while, nested loops with break, continue and pass keyword. **String Manipulation:** Accessing Strings, Basic Operations, String slices, Function and Methods. **Functions:** Defining and Calling of a function, Types of functions, Function Arguments, Anonymous functions, Global and local variables.
Unit 2	**Collection:** list, tuple , Dictionaries. Introduction, Accessing values, Working, Properties, Functions and Methods. **Modules:** Importing module, Math module, Random module, os module, date-time module, calendar module, Packages, user defined module, introduction of pip.**IO:** Printing on screen and Reading data from keyboard, Opening and closing file, Reading and writing files, Functions.
Unit 3	**Exception Handling:** Except, Try, else, finally clause, User Defined Exceptions, raise user- defined exception, nested try-except. **OOPs concept:** Class and object, Attributes, Inheritance, Overloading, Overriding, Data hiding, final and abstract class. Database: Introduction, Connections with MYSQL, Executing queries, Transactions) web-designing: HTML, CSS, JAVASCRIPT.
Unit 4	**CGI:** Introduction, Architecture, CGI environment variable, GET and POST methods. application using CGI: signup, login and session tracking with server side programming.

Unit 5	**DJANGO:** working of MVT, Environment setting and installation, creating a Project, Apps Life Cycle, Admin Interface, Views, URL Mapping. **Template System:** DTL and JINJA. Models, Page Redirection, Form Processing, project with signup and login.

References:

1. ProgrammingandProblemSolvingwithPython(AshokNamdevKamthaneandAmitAshok Kamthane) McGraw Hill publication
2. LetUs Python(Kanetkar Yashavant)BPBPublication
3. Python Complete Refernce (Brown MartinC.) McGrawHillpublication
4. PythonProgrammingAModularApproach(NaveenandKumarandTanejaSheetal)PEARSON
5. Beginning Django (RubioDaniel) Apress

MDSC0402-P Advanced python programming for spatial analytics LAB

1. ntroduction to Data Visualization: Briefly discuss the importance of data visualization in analytics and decision-making processes. Introduce Matplotlib and Seaborn libraries.
2. Installation and Setup: Guide students through installing Matplotlib and Seaborn libraries if they are not already installed. Demonstrate how to import these libraries into Python scripts.
3. Line Plots and Scatter Plots with Matplotlib: Show how to create line plots and scatter plots using Matplotlib. Explain customization options such as labels, titles, colors, and markers.
4. Histograms and Density Plots with Matplotlib: Demonstrate how to create histograms and density plots to visualize the distribution of numerical data. Discuss binning strategies and kernel density estimation.
5. Bar Plots and Pie Charts with Matplotlib: Guide students through creating bar plots and pie charts for visualizing categorical data. Discuss different types of bar plots (e.g., grouped, stacked) and pie chart customization options.
6. Introduction to Seaborn: Introduce Seaborn library and its advantages over Matplotlib for statistical data visualization. Discuss its built-in themes and color palettes.
7. Statistical Plots with Seaborn: Demonstrate various statistical plots available in Seaborn, such as box plots, violin plots, and swarm plots. Discuss when to use each type of plot based on the characteristics of the data.
8. Pair Plots and Heatmaps with Seaborn: Show how to create pair plots and heatmaps using Seaborn to explore relationships between multiple variables. Discuss correlation matrices and clustering.
9. Customizing Visualizations: Discuss advanced customization options for visualizations, such as adding annotations, adjusting axis scales, and incorporating multiple subplots.
10. Interactive Visualization with Plotly (Optional): Optionally, introduce Plotly library for creating interactive visualizations. Demonstrate how to create interactive line plots, scatter plots, and heatmaps.
11. Applying Visualizations to Real Data: Provide students with a dataset and task them with visualizing various aspects of the data using Matplotlib and Seaborn techniques learned in the lab.
12. Q&A and Further Exploration: Allow time for questions and encourage students to explore additional visualization techniques and libraries beyond the scope of the lab.

M.Sc. (Data Science)4th semester

MDSC0403-T Programming in SAS for Analytics

Unit No	TOPIC
Unit 1:	**Introduction to SAS Programming:-** Introduction to SAS software and environment, SAS data sets and libraries, SAS programming fundamentals: DATA step and PROC step, Reading and writing data in SAS.
Unit 2	**Data Manipulation and Management:-** Data manipulation techniques using SAS functions and formats, Combining and merging datasets, Handling missing values and outliers, Data cleaning and validation in SAS.
Unit 3	**Descriptive Analytics with SAS:-** Descriptive statistics and data exploration using SAS, Frequency analysis and summary statistics, Creating graphical visualizations with SAS procedures.
Unit 4	**Predictive Analytics with SAS:-** Introduction to predictive modeling concepts, Building predictive models using SAS procedures like PROC REG, PROC LOGISTIC, and PROC GLM, Model assessment and validation techniques.
Unit 5	**Advanced Analytics and Reporting:-** Advanced analytics techniques such as text analytics and time series analysis, Generating advanced reports using PROC TABULATE and PROC REPORT, Exporting SAS results to different formats.

Reference Book:

"The Little SAS Book: A Primer" by Lora D. Delwiche and Susan J. Slaughter
"Learning SAS by Example: A Programmer's Guide" by Ron Cody
"Carpenter's Guide to Innovative SAS Techniques" by Art Carpenter
"PROC SQL by Example: Using SQL within SAS" by Howard Schreier
"The SAS Programmer's PROC REPORT Handbook: Basic to Advanced Reporting Techniques" by Jane Eslinger
"SAS Certification Prep Guide: Base Programming for SAS 9" by SAS Institute
"Predictive Modeling with SAS Enterprise Miner: Practical Solutions for Business Applications" by Kattamuri S. Sarma
"Applied Analytics Using SAS Enterprise Miner" by Patrick Hall, Wayne Winston, and Brett Wujek
"Text Mining and Analysis: Practical Methods, Examples, and Case Studies Using SAS" by Goutam Chakraborty, Murali Pagolu, and Satish Garla
"SAS Programming for Enterprise Guide Users" by Neil Constable

MDSC0403-P Programming in SAS for Analytics

1. Introduction to SAS Environment
   - Setting up SAS environment, navigating SAS Studio or SAS Enterprise Guide.
2. Working with SAS Data Sets
   - Creating SAS data sets, importing and exporting data, defining variables.
3. Data Manipulation with SAS
   - Using SAS functions to manipulate data, handling missing values, sorting and filtering data.
4. Combining and Merging Datasets
   - Merging datasets using different techniques (e.g., MERGE, SQL joins).
5. Descriptive Statistics and Data Exploration
   - Calculating descriptive statistics, generating frequency distributions, exploring data visually.
6. Predictive Modeling Basics
   - Building simple predictive models using PROC REG or PROC LOGISTIC.
7. Model Assessment and Validation
   - Assessing model fit, validating models using techniques like cross-validation.
8. Advanced Analytics: Text Mining
   - Analyzing text data using SAS Text Miner, extracting insights from unstructured text.
9. Time Series Analysis
   - Analyzing time series data, forecasting future values using PROC TIMESERIES or PROC ARIMA.
10. Reporting and Visualization
    - Creating custom reports with PROC REPORT, generating graphical visualizations with PROC SGPLOT.

M.Sc. (Data Science)4th semester

MDSC0404-T Bayesian Statistics

Unit No	TOPIC
Unit 1:	**Unit 1: Introduction to Bayesian Statistics:-** Overview of Bayesian statistics: Principles, advantages over frequentist statistics, Bayes' theorem and its applications in text analytics, Prior, likelihood, and posterior distributions, Bayesian inference: Parameter estimation and hypothesis testing, Introduction to probabilistic programming with tools like PyMC3.
Unit 2	**Unit 2: Bayesian Text Classification:-** Introduction to text classification: Applications and challenges, Naive Bayes classifier: Theory and implementation for text classification, Multinomial and Bernoulli Naive Bayes models for text data, Handling uncertainty in text classification with Bayesian methods, Case studies and practical exercises on Bayesian text classification using real-world datasets.
Unit 3	**Unit 3: Bayesian Topic Modeling:-** Introduction to topic modeling: Latent Dirichlet Allocation (LDA) and its probabilistic formulation, Bayesian approach to topic modeling: Hierarchical LDA, Bayesian Non-negative Matrix Factorization (NMF), Inference in Bayesian topic models: Variational inference, Gibbs sampling, Advanced topics in Bayesian topic modeling: Dynamic topic models, supervised topic models, Hands-on projects on implementing Bayesian topic models and interpreting results.
Unit 4	**Unit 4: Bayesian Sentiment Analysis:-** Overview of sentiment analysis in text: Challenges and techniques, Bayesian methods for sentiment analysis: Bayesian Neural Networks, Bayesian Optimization, Modeling uncertainty in sentiment analysis with Bayesian approaches, Transfer learning and Bayesian methods for sentiment analysis on limited labeled data, Case studies and projects on Bayesian sentiment analysis using social media data and product reviews.
Unit 5	**Unit 5: Advanced Bayesian Text Analytics:-** Bayesian text summarization: Extractive and abstractive summarization using Bayesian methods, Named Entity Recognition (NER) with Bayesian models: Conditional Random Fields (CRF), Bayesian sequence labeling, Bayesian inference for text generation: Text generation with Recurrent Neural Networks (RNNs) and Bayesian optimization, Bayesian deep learning for text analytics: Bayesian Convolutional Neural Networks (CNNs), Bayesian Recurrent Neural Networks (RNNs), Practical projects on advanced Bayesian text analytics techniques and applications.

Reference Book:

"Bayesian Data Analysis" by Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin
"Probabilistic Programming & Bayesian Methods for Hackers" by Cameron Davidson-Pilon
"Bayesian Methods for Hackers: Probabilistic Programming and Bayesian Inference" by Cameron Davidson-Pilon
"Bayesian Reasoning and Machine Learning" by David Barber
"Bayesian Methods for Statistical Analysis" by B. E. Lunn, A. Thomas, N. Best, and D. Spiegelhalter.