COURSE BOOK MCA I YEAR

AUTONOMOUS

KIET GROUP OF INSTITUTIONS
www.kiet.edu
Delhi-NCR, Ghaziabad

KIET
GROUP OF INSTITUTIONS
Connecting Life with Learning

CELEBRATING 25 YEARS OF EXCELLENCE

CURRICULUM STRUCTURE & SYLLABUS

Effective from the Session: 2025-26

MCA, 1st Sem

S No.	Course Category (AICTE)	Course Category (UGC)	BOS	Course Code	Course Name	Type	Academic Learning (AL)			Continuous Internal Examination (CIE)			End Sem Examination (ESE)	Total Marks	Total Credits
							L	T	P	MSE	CA	TOTAL		CIE+ ESE	
1	PC	Major (Core)	CSE	CS206L	Operating System	L	3	0	0	60	15	75	75	150	3
2	BS	SEC	ASH	MA205L	Mathematical Foundation of Data Science	L	3	0	0	60	15	75	75	150	3
3	PC	Major (Core)	MCA	CA207L	Software Engineering	L	3	0	0	60	15	75	75	150	3
4	PW	Value Added Courses	CSIT	IT103L	Design Thinking	L	1	0	0	-	50	50	—	50	1
5	HS	Value Added Courses	ASH	HS107L	Quantitative Aptitude & Logical Reasoning-I	L	2	0	0	40	10	50	50	100	2
Blended															
6	ES	Minor Stream	EEE	EE111B	Essentials of IoT	B	1	0	2	40	10	50	50	100	2
7	PC	Major (Core)	MCA	CA101B	Object-Oriented Programming Concepts using Java	B	2	0	4	80	20	100	100	200	4
8	PC	Major (Core)	MCA	CA102B	Web Development-1	B	2	0	4	80	20	100	100	200	4
Lab/Practical															
9	PC	Major (Core)	CSE	CS206P	Operating System Lab	P	0	0	2	—	25	25	25	50	1
10	HS	Value Added Courses	ASH	HS101P	Communication Skills	P	0	0	4	40	10	50	50	100	2
Total Hours : 33 hrs.							**17**	**0**	**16**					**1250**	**25**

MCA, 2nd Sem

S No.	Course Category (AICTE)	CourseCategory(UGC)	BOS	Course Code	Course Name	Type	Academic Learning(AL)			Continuous Internal Examination(CIE)			End Sem Examination(ESE)	Total Marks	Total Credits
							L	T	P	MSE	CA	TOTAL		CIE+ ESE	
1	PC	Major (Core)	MCA	CA202L	Software Testing	L	3	0	0	60	15	75	75	150	3
2	PC	Major (Core)	MCA	CA201L	Artificial Intelligence	L	3	0	0	60	15	75	75	150	3
3	PC	Major (Core)	MCA	CA103L	Computer Networks Technologies	L	3	0	0	60	15	75	75	150	3
4	HS	Value Added Courses	ASH	HS201L	Quantitative Aptitude & Logical Reasoning-II	L	2	0	0	40	10	50	50	100	2
Blended															
5	PC	Major (Core)	MCA	CA301B	Data Structures using Java	B	2	0	4	80	20	100	100	200	4
6	PC	Major (Core)	MCA	CA203B	Database Management Systems	B	2	0	4	80	20	100	100	200	4
7	PC	SEC	MCA	CA105B	Advanced Java Programming	B	2	0	2	60	15	75	75	150	3
Lab/Practical															
8	PW	Value Added Courses	CSIT	IT104P	Innovation and Entrepreneurship	P	0	0	2	-	50	50	-	50	1
9	PC	Major (Core)	MCA	CA204P	Python Workshop	P	0	0	2	-	25	25	25	50	1
10	HS	Value Added Courses	ASH	HS202P	Campus to Corporate	P	0	0	2	40	10	50	—	50	1
Total Hours : 33 hrs.							**17**	**0**	**16**					**1250**	**25**

Theory Courses Detail Syllabus

Course Code: CS206L	Course Name: Operating System	L	T	P	C
		3	0	0	3

Pre-requisite: Basic knowledge on Computer System and system memory, Computer Organization and Logic Design (COLD)

Course Objectives:

This course aims to provide a comprehensive understanding of operating systems, their components, structures, and functionalities. It covers fundamental concepts such as process management, memory management, file systems, and concurrency. The course also emphasizes practical aspects, such as system calls, Linux commands, shell scripting, and scheduling algorithms. By the end of this course, students will have the knowledge and skills to understand, analyze, and apply operating system principles in real-world scenarios.

Course Outcome: After completion of the course, the student will be able to

1. Understand the need, evolution and design issues of various categories of operating systems.
2. Apply different CPU scheduling algorithms and deadlock handling methods.
3. Analyze the principles of concurrency control and process synchronization problem.
4. Analyze various memory management techniques for efficient memory allocation.
5. Apply the concept of various I/O management, Disk scheduling and file system.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1	1	1	-	-	-	-	-	2
CO2	3	3	3	3	3	1	-	-	-	-	-	2
CO3	3	3	2	3	3	2	-	-	-	-	-	2
CO4	3	3	2	3	3	2	-	-	-	1	-	2
CO5	3	3	2	2	2	2	-	-	-	1	-	2

Unit 1	Introduction of Operating System	09 hours

Introduction: Operating system Components and its services, Classification of Operating systems- Batch system, Time sharing, Real Time System, Multiprocessor Systems, Multiuser Systems, Multiprocess Systems, Multithreaded Systems, Operating System Structure- Layered structure, Reentrant Kernels, Monolithic and Microkernel Systems. System Calls, Elementary Linux commands and Shell Scripting.

Unit 2	Process Scheduling and Resource Management	09 hours

Introduction to Process: Process States, State Transition Diagram, Schedulers, Process Control Block (PCB), Threads and their management,

CPU Scheduling: Concepts, Performance Criteria, Scheduling Algorithms.

Deadlock: System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock.

Unit 3	Concurrent Processes	09 hours

Concurrent Processes: Principle of Concurrency, Critical Section Problem, Mutual Exclusion, Dekker's solution, Peterson's solution, Semaphores, Monitors, Test and Set operation; Classical Problem in Concurrency- Producer / Consumer Problem, Reader Writer Problem, Dining Philosopher Problem, Sleeping Barber Problem; Inter Process Communication models (IPC).

Unit 4	Memory Management	09 hours

Memory Management: Basic bare machine, Resident monitor, Multiprogramming with fixed partitions and variable partitions, Protection schemes, Paging, Segmentation, Paged segmentation, Virtual memory concepts, Demand paging, Performance of demand paging, Page replacement algorithms, Thrashing.

Unit 5	I/O Management and Disk Scheduling:	09 hours

File Systems and I/O Management of Windows and Linux.

Disk Scheduling: Disk storage structure and disk scheduling algorithms, RAID.

Case Study: Introduction to Android and Mac Operating System, The Evolution of Mobile Operating Systems: iOS vs. Android

Total Lecture Hours	45 hours

Text Book:

1. Silberschatz, Galvin and Gagne, "Operating Systems Concepts", Wiley
2. Harvey M Dietel, " An Introduction to Operating System", Pearson Education
3. William Stallings, "Operating Systems: Internals and Design Principles ", 6th Edition, Pearson Education

Reference Book:

1. Sibsankar Halder and Alex A Aravind, "Operating Systems", Pearson Education
2. D M Dhamdhere, "Operating Systems : A Concept based Approach", 2nd Edition, TMH.
3. Andrew S. Tanenbaum and Herbert Bros, Modern Operating Systems (4thEdition), Pearson

Mode of Evaluation

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3 (ATTN)		
30	30	6	6	3	75	150
60		15				

Course Code: MA205L	Course Name: Mathematical Foundation of Data Science	L	T	P	C
		3	0	0	3

Pre-requisite: Introductory knowledge of statistics, Basic logical reasoning and problem-solving skills

Course Objectives:

1. Equip students with a mathematical groundwork essential for AI, machine learning, and data science.
2. Able to solve real-world problems by using statistics, probability, Random number generation with data pre-processing.

Course Outcome:

Students will be able to:	
CO1	Apply the concept of matrix in eigen vectors and Inner Product.
CO2	Apply the knowledge of matrix factorization in data decomposition.
CO3	Employ the probability distribution in the field of data science.
CO4	Apply the concept of statistics in testing hypothesis.
CO5	Apply the postulates of partial derivatives in optimization.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	2	1	1	-	-	-	1
CO2	2	2	1	1	-	-	-	2
CO3	2	2	1	1	-	-	-	1
CO4	2	2	1	1	-	-	-	2
CO5	2	2	1	1	-	-	-	1

Detailed Syllabus

Unit 1	**Linear Algebra-I**	**9 hours**
Matrices and their properties, rank by echelon form, Rank, nullity, Eigen Values and Eigen Vectors, Inner Product and projections.		
Unit 2	**Linear Algebra-II**	**9 hours**
Similarity transformation, Diagonalization (2x2), EVD (Eigen Value Decomposition, SVD (Singular Value Decomposition), Matrix Factorization, LU- Doolittle Method.		
Unit 3	**Probability and Distributions**	**9 hours**
Probability and Conditional Probability, Bayes' Theorem, Discrete and Continuous Distributions, Binomial Distribution, Poisson Distribution., Normal Distribution , Expectation and variance.		
Unit 4	**Descriptive Statistics and Interference**	**9 hours**
Mean, Median, Mode, Standard Deviation, Correlation, Sampling Techniques Sampling Theory (Small and Large), Hypothesis, Null hypothesis, Alternative hypothesis, Testing a Hypothesis, Level of significance, Confidence limits, t-test.		
Unit 5	**Calculus and Optimization Basics**	**9 hours**
Partial Derivatives, Composite Function, Total Derivatives, Chain Rule and Gradient Computation, Gradient Descent Algorithm, Learning rate and convergence.		
	Total Lecture Hours	**45 hours**

Textbook:

1. Sheldon M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", Academic Press, 5th edition
2. S.C. Gupta & V.K. Kapoor, "Fundamentals of Mathematical Statistics", Sultan Chand & Sons
3. Han, J., Kamber, M., & Mining, D. (2012). Concepts and techniques. Morgan kaufmann, 340, 94104-3205.
4. Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). Mathematics for Machine Learning. Cambridge University Press.
5. Montgomery, D. C., & Runger, G. C. (2014). Applied Statistics and Probability for Engineers. Wiley.

References:

1. https://www.udemy.com/course/mathematical-foundation-for-machine-learning-and-ai/
2. https://www.udemy.com/course/master-linear-algebra-and-probability-2-in-1-bundle/
3. Strang, G. (2005). Linear Algebra and its Applications. Cengage Learning.
4. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
5. Cover, T. M., & Thomas, J. A. (2006). Elements of Information Theory. Wiley.
6. Sastry, S. S. (2002). Introductory Methods of Numerical Analysis. PHI.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
30	30	6	6	3	75	150
60		15				

Course Code: CA207L	Course Name: Software Engineering	L	T	P	C
		3	0	0	3

Pre-requisite: None

Course Objectives:

1. Introduce basic software engineering concepts for software development process.
2. Analyze the system requirement specifications.
3. Design the system using modeling tools.
4. Explore the basic concept of software testing and debugging.
5. Understand the various project management activities.

Course Outcome:

Students will be able to:	
CO1	Understand the fundamental concepts of the software development process.
CO2	Demonstrate the concept of requirement engineering in the SRS document.
CO3	Demonstrate the concepts of software design.
CO4	Elaborate concept of software maintenance and software project management.
CO5	Illustrate the fundamental concepts of Agile models.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	-	-	-	-	-	-	1
CO2	2	3	2	-	1	-	-	3
CO3	3	3	3	3	2	-	-	3
CO4	3	2	1	1	1	-	-	3
CO5	2	1	1	-	1	3	-	2

Unit 1	Introduction	9 hours

Introduction to Software Engineering: Software Components, Software Characteristics, Conventional Vs Software Engineering approach.

Software Development Life Cycle (SDLC) Models: Waterfall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, Rapid Application Development.

Quality Assurance: Software Quality Assurance Models, SEI-CMM Model, ISO Models.

Unit 2	**Software Requirement Specifications**	**9 hours**

Software Requirements: Functional and Non-Functional Requirements, User Requirements, System Requirements, and Interface Requirements.

Requirement Engineering Process: Feasibility Study, Requirement Elicitation, Requirement Analysis, Requirement Management, Introduction to Software Requirements Specifications (SRS), Structure of SRS (IEEE Format).

Unit 3	**Software Design & Testing**	**9 hours**

Introduction to Design: Basic Concepts, Top-down & Bottom-Up Approach, Coupling, Cohesion, Structure Charts.

Introduction to Software Testing: Software Errors, Faults and Failures Concepts of Verification and Validation.

Levels of Testing: Unit Testing, Integration Testing, System Testing, Acceptance Testing. White Box Testing: Test coverage, code coverage, condition coverage, branch coverage, cyclomatic complexity; Black-Box Testing: Equivalence classes, boundary value tests, Testing for Non-Functional Requirements Load, Volume, Performance, Code Inspection, Test Plan, Test case, Test suite.

Unit 4	**Software Project Management**	**9 hours**

Maintenance: Introduction, Perfective, Preventive, Corrective, Adaptive.

Software Project Management Activities: Project planning and Tracking, Cost Estimation Management (Constructive Cost Model (COCOMO)), Scheduling Management (GANTT and CPM/PERT), **Configuration Management Activities**: Identification and Establishment, Version Control, Change Control, Configuration Auditing.

Unit 5	**Agile Process Model**	**9 hours**

Agile Model: Introduction to Agile, principles of agile manifesto, overview of Various Agile methodologies - Scrum, Extreme Programing, Lean, and Kanban

Key aspects of Scrum: roles - Product Owner, Scrum Master, Team, Manager in scrum and product backlog **Scrum process flow**: Product Backlog, Sprint Review, sprints backlog, scrum meetings, sprint review, sprint retrospective, Burn down chart.

Scaled agile frameworks: SAFe, Scrum@Scale, Disciplined Agile.

	Total Lecture Hours	**45 hours**

Textbook:

1. Roger S. Pressman, Software Engineering-A practitioner's Approach, 7th Edition, McGraw-Hill International, 2010.
2. K.K Aggarwal & Yogesh Singh, Software Engineering, 3rd Edition, New Age International Publishers, 2007.
3. Pankaj Jalote, "Software Engineering", 1st edition, Wiley, 2013.
4. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, 1st edition, Pearson Education, 2009.

Reference Books:

1. Sommerville, Software Engineering, 8th Edition, Pearson Education, 2009.
2. Aditya P Mathur, Fundamentals of Software Testing; 1st Edition, Pearson, 2011.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
30	30	6	6	3	75	150
60		15				

Course Code: IT103L	**Course Name: Design Thinking**	**L**	**T**	**P**	**C**
		1	**0**	**0**	**1**
Pre-requisite: NA					
Course Objectives:					

1. To expose the student with state of the art perspectives, ideas, concepts, and solutions related to the design and execution of projects using design thinking principles.
2. To prepare the mindset and discipline of systemic inspiration driven by a desire to identify new sources of ideas, and new models especially outside their regular working atmosphere.
3. To propose a concrete, feasible, viable and relevant innovation project/challenge.

Course Outcome: After completion of the course, the student will be able to

1. Understand the basic requirements of a good design.
2. Empathize and ideate the solutions to problems in his environment
3. Prototype and test the developed solutions.
4. Apply the principles of design thinking on developing innovative solutions to the real world problems.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	3	2	-	1	-	-	2	2	-	2
CO2	1	2	3	2	-	1	-	-	2	2	-	2
CO3	1	2	3	2	-	1	-	-	2	2	-	2
CO4	1	2	3	2	-	1	-	-	2	2	-	2

Unit 1	**Fundamentals Of Design Thinking**	**04 hours**

Concept of Design Thinking, Need of Design Thinking, Goal of Design thinking (Desirability, feasibility and viability), Design thinking Process model, Design thinking tools.

Activities: Identify an Opportunity, Scope of the Project, Explore the possibilities and prepare a design brief.

Unit 2	**Empathize And Define**	**04 hours**

Design thinking phases, how to empathize, Role of empathy in design thinking, the purpose of empathy maps, Things to be done prior to empathy mapping, Activities during and after the session, Understanding empathy tools: Customer Journey Map, Personas. Define- Methods of Define Phase: Storytelling.

Activities: Apply the methods of empathizing and Define Phases Finalize the problem statement.

Unit 3	**Ideation**	**04 hours**

Challenges in idea generation, Visualize, Empathize, and Ideate method, Importance of visualizing and empathizing before ideating, Applying the method, Create Thinking, Generating Design Ideas, Lateral Thinking, Analogies, Brainstorming, Mind mapping,

Ideation Tools: How Might We? (HMW), Storyboard, Brainstorming. What is design innovation? A mindset for innovation, and asking "What if?" asking "What wows?" and "What works?"

Activities: Apply the methods of Ideate Phase: Generate Innovative solution ideas.

Unit 4	**Prototyping And Testing**	**03 hours**

What is a prototype? - Prototyping as a mindset, prototype examples, prototyping for products; Why we prototype? Fidelity for prototypes, Process of prototyping- Minimum Viable prototype. Testing prototypes with users, Collect feedback; iterate and improve the ideas.

Activities:

1. Prototype: Apply the Methods of the Prototype Phase - Create prototypes for selected ideas.

Testing: Collect feedback; iterate and improve the ideas Present your solution using the Storytelling method.

Total Lecture Hours	**15 hours**

Textbook:

1. Design Thinking, A Beginner's Perspective, E Balaguruswamy, Bindu Vijayakumar, Mc Graw Hill,2024
2. The Design Thinking Playbook, Michael Lewrick (Author), Patrick Link (Author), Larry Leifer (Author) Publisher Wiley, Edition 2018.
3. Design Thinking For Dummies, Prof. Dr. Christian Müller- Roterberg, Wiley, 2021
4. The Design of Everyday Things, Don Norman(Author), Navol Books Trading, Edition 2022.

Reference Books:

1. Design Thinking, A Beginner's Perspective, E Balaguruswamy,Bindu Vijayakumar, Mc Graw Hill,2024
2. The Design Thinking Playbook, Michael Lewrick (Author), Patrick Link (Author), Larry Leifer (Author) Publisher Wiley, Edition 2018.
3. Design Thinking For Dummies, Prof. Dr. Christian Müller- Roterberg, Wiley, 2021
4. The Design of Everyday Things, Don Norman(Author),Navol Books Trading, Edition 2022
5. Designing Experiences, James Robert Rossman and Mathew D. Duerden, Columbia Business School Pub, Edition 2019.

6. Roger Martin, "The Design of Business: Why Design Thinking is the Next CompetitiveAdvantage", Harvard Business Press, Edition 2009.
7. Idris Mootee, Design Thinking for Strategic Innovation,2013, John Wiley & Sons Inc.

Mode of Evaluation

Evaluation Scheme					
MSE	**CA**			**ESE**	**Total Marks**
-	**CA1**	**CA2**	**CA3(ATT)**	-	**50**
	20	25	5		
	50				

Course Code: HS107L	**Course Name:** Quantitative Aptitude & Logical Reasoning-I	**L**	**T**	**P**	**C**
		2	0	0	2

Course Objectives:

1. Students will gain a strong foundation in basic mathematical concepts, and become proficient in performing calculations, solving equations, and manipulating numerical data.
2. Students will learn to think critically, make deductions, and draw logical conclusions based on given information. This will enable them to approach complex problems systematically and derive accurate solutions.
3. Students will learn how to interpret and analyze data presented in various formats such as tables, graphs, and charts. This skill is particularly useful for tasks involving statistical analysis and decision-making.
4. It will help students to improve their speed and accuracy in solving problems. Through practice and exposure to different types of questions, students will learn time-saving techniques and develop strategies to effectively manage time during exams or real-world scenarios.
5. It will help students to prepare for various competitive exams, such as aptitude tests for job placements, college admissions tests for higher education (GATE/CAT/GRE/GMAT), and government entrance exams. The course will provide the necessary knowledge and practice to perform well in these exams.

Course Outcome:

Students will be able to:	
CO1	Analyze problems, identify relevant information, and apply appropriate mathematical methods to reach solutions.
CO2	Analyze tasks and activities by following a chain of thought process and find logical solutions to a problem.
CO3	Analyze trends, patterns, and relationships within the data.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	2	-	-	-	-	-	1
CO2	2	2	-	-	-	-	-	1
CO3	1	1	-	-	-	-	-	1

Unit 1	**Speed Math & Number System**	**6 hours**

Addition, Subtraction, Multiplication and Division shortcuts, Square and Square roots, Cubes and Cube roots, Classification of Numbers, Divisibility Rules, Prime and Composite No. Co-Prime pair, Factors, Finding HCF and LCM, Word problem based on HCF and LCM,

Unit 2	**Series, Coding and Decoding, Ranking and Order**	**6 hours**

Number and Letter Series, Word Series, Coding & Decoding, (Type-I)-: Basics of A-Z and Z-A, the position of alphabets including Reverse Order, the concept of 27, Short-cuts to learn positions of alphabets, (Type-II)-: Ascending and Descending calculation of position of alphabets, (Type-III)-: Fictitious Coding, (Type-IV)-: Substitution Method applied in Coding, (Type-V)-: finding position and identifying the Characters in Coded words, Ranking and Order based problems.

Unit 3	**Percentage, Profit, Loss & Discount (PPLD), Interest**	**6 hours**

Introduction & Definition of Percentage, Conversion of a fraction to percentage and vice versa, Growth and depreciation, Word problem based on percentage, Application of percentage in Change in Area, Perimeter, and Volume of different Geometrical shapes, Definition and Introduction of P & L, Application of P & L based on percentage, Introduction of discount, Discount Series, single

discount, Application based word problems on Discount, Introduction of Concept related to SI and CI, Generalized way to find the difference between SI and CI for 2 year and 3 year.		
Unit 4	**Ratio, Proportion and Variation, Mixture and Alligation**	**6 hours**
Definition of Ratio and Proportion, Type of Proportion (Direct Proportion, Inverse Proportion, Continued Proportion), Definition of Variance, Problem on coins, expenditure, and saving, etc, Introduction of Mixture and Alligation, Repletion process, Percentage used in Mixture and allegation.		
Unit 5	**Direction Sense, Data Arrangement, and Data Interpretation**	**6 hours**
Direction Sense based Problems, Linear Arrangement, Circular Arrangement, Introduction to Data Interpretation (DI), Questions based on Tabular charts, Bar Graphs, Pie charts, Line Graphs, and Mix graphs etc., Questions based on missing data.		
	Total Lecture Hours	**30 hours**

Textbooks:

1. "Quantitative Aptitude for Competitive Examinations" by R.S. Aggarwal, S. Chand Publication.
2. "A Modern Approach to Verbal & Non-Verbal Reasoning" by R.S. Aggarwal, S. Chand Publication.

Reference Books:

1. Arun Sharma, How to Prepare for Quantitative Aptitude for the CAT, 10th Edition, TMH Publication, 2022.
2. Arun Sharma, How to Prepare for Logical Reasoning for the CAT, 7th Edition, TMH Publication, 2024.
3. Arun Sharma, "How to Prepare for Data Interpretation for the CAT, 8th Edition, TMH Publication, 2024.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
20	20	4	4	2	50	**100**
40		**10**				

Course Code: EE111B	**Course Name: Essentials of IoT**	**L**	**T**	**P**	**C**
		1	0	2	2

Pre-requisite: Not Applicable

Course Objectives:

1. The course aims to provide exposure to the applications of IoT in smart cities and industrial applications.
2. It aims to train the students to the basic concepts of the Embedded C and Controllers.
3. This course is designed to give the students hands-on experience with the Software and Hardware concepts.

Course Outcome:

Students will be able to:	
CO1	Understand the basic concepts of sensors and transducers.
CO2	Understand basics of embedded system and different IoT boards.
CO3	Apply basic operations and programming techniques of IoT devices.
CO4	Apply smart technology knowledge through case studies.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	–	–	–	2	2	2	–
CO2	2	–	2	–	2	2	2	–
CO3	3	–	3	2	3	2	2	–
CO4	3	2	3	3	3	2	2	–

Unit-1	Sensing Devices & Transducers	07 hours

Introduction to breadboard, basic electrical connections using resistors, LEDs, and switches, Overview of power supply. Definition and basic concepts of sensors and actuators.

Hands-on/Case Study/ Mini-Project/ Problem solving:

1. Understanding Sensors and Transducers: Types, Operation Principles, and Selection Criteria.
2. Role of Sensors and Transducers in Internet of Things (IoT) Systems.
3. Hands-on Introduction to commonly used real world IoT Sensors.
4. Hands-on Introduction to different controllers used in IoT.

Unit 2	Embedded Systems Fundamentals	07 hours

Introduction to Embedded C: Interfacing Basics, Digital I/O, Analog I/O, Introduction to Arduino (ATmega328P), Arduino board components and architecture, Interfacing DHT11 with Arduino.

Hands-on/Case Study/ Mini-Project/ Problem solving:

1. Understanding the Architecture and Pin Configuration of Arduino Boards.
2. Analyze Digital signal data acquisition using Arduino.
3. Explore Digital signal generation using Arduino.
4. Analyze Analog signal data acquisition using Arduino.
5. Explore Analog signal generation using Arduino.

Unit 3	IoT Board	08 hours

Introduction to IoT in Modern Industry Applications, Basic Operations of IoT, Basics of ESP 8266 programming, Introduction to Blynk IoT, Interfacing with Different types of Sensors: Touch Sensor, Alcohol Sensor (MQ 3), LPG Sensor (MQ 6), Relay, Light Dependent Resistor (LDR), IR (Infrared) Sensors and PIR (Passive Infrared) Sensors.

Hands-on/Case Study/ Mini-Project/ Problem solving:

1. Understanding the Architecture and Pin Configuration of ESP8266.
2. Analyze Digital / Analog signal data acquisition using ESP8266.
3. Explore Digital / Analog signal generation using ESP8266.
4. Real-Time Data Logging Using ESP8266 and Arduino.

Unit 4	Smart Sensor Technologies	08 hours

Intelligent Sensors: General Structure of smart sensors & its components, Case study of Air Quality Monitoring System, Case study of Soil Health Monitoring System, Case study of Water Quality Monitoring System.

Hands-on/Case Study/ Mini-Project/ Problem solving:

1. Designing a Lighting Control System using LDR.
2. Designing a Multi-Sensor Alert System Using Touch, IR, PIR and Arduino.
3. Object Detection Using Ultrasonic Sensors with Arduino and ESP.
4. Building IoT Applications with Blynk: Monitoring Temperature and Humidity with DHT11 Sensor.
5. Building IoT Applications with Blynk: Smart Home Automation Using ESP8266 and Blynk.
6. Building a Soil Health Monitoring system using NPK sensor.

	Total Lecture Hours	30 hours

Textbook:

1. Raj Kamal, "Internet of Things: Architecture and Design Principles", McGraw Hill Education (India) Private Limited CHENNAI.
2. Waldemar Nawrocki , "Measurement Systems and Sensors" , Artech House Boston , London.
3. Rajkumar Buyya and Amir Vahid Dastjerdi , "Internet of Things: Key Applications and Protocols" Elsevier.

Reference Books:

1. Pethuru Raj and Anupama C. Raman. “The Internet of Things: Enabling technologies, platforms, and use cases”. Auerbach Publications.
2. "The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and Beagle Bone Black" by Donald Norris, McGraw-Hill Education TAB.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
20	20	4	4	2	50	100
40		10				

Course Code: CA101B	Course Name: Object Oriented Programming Concepts using Java	L	T	P	C
		2	0	4	4

Pre-requisite: NA

Course Objectives:

This course introduces students to object-oriented programming using Java, focusing on core concepts like fundamentals of basic programming, encapsulation, inheritance, polymorphism, and abstraction. Students will learn to design and develop clean, efficient, and maintainable Java applications using best practices and modern development tools.

Course Outcome:

Students will be able to:	
CO1	Understand programming concepts using Java.
CO2	Explain object-oriented programming concepts using Java.
CO3	Apply exception handling and file I/O operations in Java to develop robust and error-resilient applications.
CO4	Implement multi-threading programs in Java to solve concurrent processing problems efficiently.
CO5	Analyze Java programs to optimize data structures using the Collection Framework.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	1	2	1	2	-	-	-	1
CO2	2	2	2	2	-	-	-	2
CO3	2	2	2	2	1	-	-	3
CO4	1	2	1	2	1	-	-	1
CO5	2	2	3	3	2	-	-	3

Unit 1	Java Fundamentals	18 hours

Introduction to Java, history of Java, JVM and JRE, Java environment setup, Java source file structure, compilation process, defining classes, methods, variables and data types, conditional and looping statements, operators, access specifiers, static and final members, arrays, string handling

Hands-on: Create table of any number given by user

Unit 2	Object Oriented Concepts	18 hours

Object-oriented programming principles, class and object, inheritance, superclass and subclass, method overriding and overloading, encapsulation, polymorphism, abstraction, interfaces, abstract classes

Hands-on: Create calculator app using OOP concepts

Unit 3	Exception and File Handling	18 hours

Introduction to exceptions, exception types, error vs exception, try-catch-finally blocks, throw and throws keywords, in-built and user-defined exceptions, checked and unchecked exceptions, file handling with byte and character streams, reading and writing files, input from keyboard

Hands-on: Demonstrate use of try, catch and finally, create Java program to copy an image file

Unit 4	Multi-Threading and Package Management	18 hours

Thread basics, thread life cycle, creating threads, thread priorities, thread synchronization, inter-thread communication, defining packages, setting CLASSPATH, creating JAR files, import and static import, package naming conventions

Hands-on: Create a JAR file of your Java application

Unit 5	Collection Framework	18 hours

Overview of collection framework, collection hierarchy, iterator and collection interfaces, list interface with ArrayList, LinkedList, Vector, Stack, queue interface, set interface with HashSet, LinkedHashSet, TreeSet, map interface with HashMap, LinkedHashMap, TreeMap, Hashtable, sorting using collections

Hands-on: Demonstrate use of collections including Stack, Queue, Set and Map

	Total Lecture Hours	90 hours

Textbook:

1. Java: How to Program (Early Objects), Authors: Paul Deitel & Harvey Deitel, 11th Edition, Publisher: Pearson Education.

2. Programming with Java: A Primer, Author: E. Balagurusamy, 6th Edition, Publisher: McGraw-Hill Education

Reference Books:

1. Java: The Complete Reference" by Herbert Schildt, 12th Edition, Publisher: McGraw-Hill Education
2. Head First Java" by Kathy Sierra & Bert Bates, 3rd Edition, Publisher: O'Reilly Media

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
40	40	8	8	4	100	200
80		20				

Course Code: CA102B	Course Name: Web Development-1	L	T	P	C
		2	0	4	4

Pre-requisite: Basic concepts of web browser

Course Objectives:

Understand how to build scalable, high-performance, and interactive web applications by using JavaScript and React.js.

Course Outcome:

Students will be able to:	
CO1	Demonstrate static web pages using HTML.
CO2	Apply CSS to design responsive web pages.
CO3	Analyse client-side scripting using JavaScript.
CO4	Develop interactive web page using ES6.
CO5	Create dynamic website using React JS.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	3	-	-	1	1
CO2	3	2	2	3	-	-	1	1
CO3	3	2	2	3	-	-	1	1
CO4	3	2	2	3	-	-	1	1
CO5	3	2	1	3	-	-	1	2

Unit 1	HTML and GitHub	18 hours

Introduction: HTML Tags, Grouping Using Division tag and Span, Lists, Images, Hyperlink, Table, Form, XHTML Multimedia.
Web Designing: Introduction to Web Designing, Web Designing Strategies, Web Designing Issues, designing using Figma.
GitHub Introduction: Create a New Local Git Repository, Stage & Commit Files, push to a Remote Repository, Pull from a Remote Repository, Clone (Download) a Remote Repository, Version control.
Hands-on: Static website development using HTML.

Unit 2	CSS and Its Frameworks	18 hours

Introduction: Syntax, Types of CSS, Margins, Padding, Fonts, Text, Overflow, Float, Navigation Bar, Image Gallery, Rounded Corners, Multiple Backgrounds, Gradients, Shadow Effects, Transitions, Animations, Multiple Style Sheets, Selectors, Box Model.
Bootstrap: Introduction to Bootstrap, Bootstrap Grid layout, Bootstrap Components, Responsive pages.
Tailwind CSS: Introduction to Tailwind, Classes of Tailwind, Flex and Grid layout, Responsive pages.
Hands-on: Website designing using CSS.

Unit 3	JavaScript	18 hours

Introduction: Data types, operators, variables, substring, array, objects, conditional statements, loops, validation, Client-Side scripting versus Server-side scripting.

HTML DOM and JavaScript: Finding HTML Elements, Manipulating DOM Elements with JavaScript. Creating Functions in JavaScript, UI Events and event handling, Returning Data from Functions, working with Conditions, looping in JavaScript, Screen output and keyboard input, Block Scope Variables, working with Objects, Creating Object using Object Literals. **Hands-on:** Interactive web designing using JavaScript..		
Unit 4	**ECMA Script 6**	**18 hours**
Introduction: let, var, const., Default function parameters, Rest parameter, spread operator, object literal syntax extensions, Template literals, De-structuring, ES6 Modules, Arrow Functions, Async/Await, Promises, Callback method. **Hands-on:** Asynchronous Application Development		
Unit 5	**Introduction to React JS**	**18 hours**
React Fundamentals: Introduction to React, Running First React App using VS Code, Rendering, JSX & JSX Expression, Rendering Multiple Elements, Rendering List, Conditional Rendering, Functional Components, Props, Passing Props to Components, Bootstrap in React, Dynamic Class Rendering, Dynamic Expression Rendering, Event Handling, Component Composition, Passing Events to Components, Component Styling, Working with Forms. **Hands-on:** Dynamic website using React.		
	Total Lecture Hours	**90 hours**

Textbook:

1. Learning React: Modern Patterns for Developing React Apps" by Alex Banks and Eve Porcello, 2nd edition, published by O'Reilly Media on June 12, 2020.
2. HTML & CSS: The Complete Reference by Thomas Powell (Author), Edition 5, McGraw Hill Education, 2017.
3. JavaScript from Beginner to Professional by Laurence Lars Svekis (Author), Edition 1, Maaike van Putten (Author), Rob Percival (Author), Packt Publishing Limited, 2021.

Reference Books:

1. Full stack React: The Complete Guide to ReactJS and Friends" by Anthony Accomazzo, 1st Edition, published by Fullstack.io on September 12, 2017.
2. The Node Craftsman Book" by Manuel Kiessling, 1st Edition, published by Packt Publishing on April 28, 2017.
3. Beginning React (incl. Redux and React Hooks) by Greg Lim (Author), 2020.
4. Pro RESTful APIs: Design, Build and Integrate with REST, JSON, XML and JAX-RS by Sanjay Patni (Author), Edition 1, Apress, 2017.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
40	40	8	8	4	100	200
80		20				

Course Code: CA202L	Course Name: Software Testing	L	T	P	C
		3	0	0	3

Pre-requisite: Concept of any Programming Language

Course Objectives:

1. To provide a comprehensive understanding of the fundamental concepts, principles, and importance of software testing in the software development lifecycle (SDLC).
2. To equip students with knowledge of test design techniques, including black-box, white-box, and experience-based methods, for creating effective test cases.
3. To familiarize students with various types of testing (manual and automated) and their applications in functional and non-functional areas.
4. To introduce popular testing tools like Selenium, Bugzilla and demonstrate their practical applications in test automation.

Course Outcome:

Students will be able to:	
CO1	Understand software testing concepts, principles, and the testing lifecycle.
CO2	Apply black-box and white-box testing techniques to validate software functionality.
CO3	Understand levels of testing and regression testing techniques for ensuring software quality.
CO4	Apply test management strategies, including test planning and risk analysis, to optimize the testing process.
CO5	Apply automation testing tools like Selenium for functional testing.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	-	-	-	-	-	-	-
CO2	3	2	2	-	-	-	-	2
CO3	3	-	-	-	-	-	-	2
CO4	3	-	-	-	-	-	-	2
CO5	3	-	-	3	-	-	-	2

Unit 1	Introduction to Software Testing	09 hours

Introduction: Some popular worldwide software failure. What is software testing, Testing Objective, Role of a tester, Skills required by tester. Terminologies in testing: Error, Fault, Failure, Verification, Validation, Test Cases, Test Suits, Test Oracles, Testing Vs Debugging, Testing, Quality Assurance and Quality Control, Limitations of testing, Testing Life Cycle (Phases: Requirement analysis, test planning, test case design, test execution, defect reporting, and closure). Principles of Testing.

Unit 2	White Box and Black Box Testing	09 hours

Static vs Dynamic Testing, Black Box Testing Techniques: Boundary Value Analysis (Generalizing Boundary Value Analysis, Robust Boundary Value Testing, Worst-Case Boundary Value Testing, Robust Worst-Case Testing), Equivalence Class Testing, Decision Table based Testing, Cause-Effect Graphing Technique, State Transition Technique

White Box Testing – Need, Logic Coverage Criteria, Statement Coverage, Branch Coverage, Condition Coverage, Loop Coverage, Path Coverage, Graph Matrices, Cyclomatic Complexity, Data Flow Testing, Mutation Testing.

Unit 3	Testing Levels	09 hours

Levels of Testing: Unit Testing, Integration Testing, System Testing and Acceptance Testing

Regression Testing: Progressive Vs Regressive Testing, Objectives of Regression Testing, Regression testing Techniques

Experience Based Testing Techniques: Error Guessing, Exploratory Testing, Checklist Based Testing.

Unit 4	Test Management	09 hours

Test Management: Organization Structures for Testing Teams, Test Planning, Test case minimization, Test Case Prioritization, Risk Analysis.

Debugging: Debugging Process, Debugging Techniques, Debuggers.

Unit 5	Automation Testing	09 hours

Automation and Testing Tools: Need for automation, Testing Tool Classification, benefits and Risks of Test automation, Overview of some commercial testing tools, Introduction to Selenium, Functional testing using Selenium, Automation Testing using Bugzilla.

Total Lecture Hours	45 hours

Textbook:

1. Yogesh Singh, "Software Testing", Cambridge University Press, New York.
2. Naresh Chauhan, "Software Testing Principles and Practices".

Reference Books:

1. Wllian E Perry, "Effective Methods for Software Testing", Wiley
2. Roger S Pressman, Software Engineering – a Practitioners Approach", McGraw Hill Education.
3. Aditya P Mathur, "Foundation of Software Testing", Pearson Publications.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
30	30	6	6	3	75	150
60		15				

Course Code: CA201L	Course Name: Artificial Intelligence	L	T	P	C
		3	0	0	3

Pre-requisite: NA

Course Objectives: This course aims to introduce the foundational concepts, techniques, and applications of Artificial Intelligence (AI), including intelligent agents, search strategies, knowledge representation, reasoning, and basic machine learning. It enables students to model and solve real-world problems using AI principles and tools.

Course Outcome:

Students will be able to:	
CO1	Understand concepts of Artificial Intelligence, agents as a problem-solver
CO2	Explore uninformed searching strategies in AI based problems.
CO3	Demonstrate the concept informed searching strategies with different heuristics
CO4	Illustrate knowledge representation based on propositional logic and predicate logic.
CO5	Apply reasoning with uncertainty in learning taxonomy.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	-	-			-	-	-
CO2	-	3	3			-	-	-
CO3	-	-	3	3	3	-	-	-
CO4	-	-	3	2		2	-	-
CO5	-	-	-		3	-	3	3

Detailed Syllabus

Unit 1	Introduction to Artificial Intelligence	9 hours

Introduction to Artificial Intelligence, foundation and history of AI, Intelligent agents: Agents and Environments, concept of rationality, structure & types of agents, problem solving agents, State Space Search, Production Systems, Problem Characteristics & formulation, types of production system, Search strategies.

Unit 2	Uninformed Search Strategies	9 hours

Uninformed Search Strategies - Formulation of toy & real-world problems, Breadth First Search, Uniform Cost Search, Depth First Search, Depth Limited Search, Iterative Deepening Depth First Search, Bidirectional Search, Comparison of Uninformed search Strategies

Unit 3	Informed Search and Game Playing	9 hours

Informed Search Strategies – Generate & test, Hill Climbing, Best First Search, A* and AO* Algorithm, Constraint satisfaction problems, Game playing: Minimax Search, Alpha-Beta Cut-offs, waiting for quiescence

Unit 4	Knowledge Representation and Reasoning	9 hours

Knowledge based agents, Wumpus world. Propositional Logic: Representation, Inference, Reasoning Patterns, Resolution, Forward and Backward Chaining. First order Logic: Representation, Inference, Reasoning Patterns, Resolution, Forward and Backward Chaining.

Unit 5	Uncertainty and Learning	9 hours

Uncertain Knowledge, Reasoning and Machine Learning - Probabilistic Reasoning, Review of Probability Theory, Probabilistic Inference Rules, Bayes Theorem, examples of Bayes theorem, Introduction to Learning, Taxonomy of Learning Systems, Concept Learning, Find-S algorithm, Candidate Elimination Algorithm.

	Total Lecture Hours	45 hours

Textbook:

1. Stuart Russell and Peter Norvig – Artificial Intelligence a Modern Approach, PEARSON Education
2. N. P. Padhy – Artificial Intelligence and Intelligence Systems, OXFORD publication

Reference Books:

1. Eugene, Charniak, Drew Mcdermott: "Introduction to Artificial Intelligence.", Addison Wesley.
2. Nilsson: ―Principles of Artificial Intelligence‖, Morgan Kaufmann.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		

	30	30	6	6	3	75	150	
	60		15					

Course Code: CA103L	Theory Course Name: Computer Networks Technologies	L	T	P	C
		3	0	0	3

Pre-requisite: Basic knowledge of computer fundamentals.

Course Objectives:

The objective of this course is to provide insight about computer network concepts and to gain comprehensive knowledge of the layered communication architectures, their functionalities and key protocols.

Course Outcome:

Students will be able to:	
CO1	Understand networking concepts and functionality of the physical layer.
CO2	Illustrate the concept of the elementary data link layer protocol to build a robust network
CO3	Apply the concept of routing and IP addressing in the network layer.
CO4	Demonstrate the usage and working of the transport layer.
CO5	Determine the performance of different protocols used at the application layer.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	1	-	-	-	-	-	**1**
CO2	3	3	2	-	-	-	-	**3**
CO3	3	3	2	-	-	-	-	**3**
CO4	3	1	-	-	-	-	-	**2**
CO5	3	2	-	-	-	-	-	**2**

Unit 1	Introduction -Physical Layer	09 hours

Introduction to computer networks: Network applications and goals, modes of communications, LAN, MAN, WAN, Internet, network hardware, network software, Design issues of layers, reference models: OSI, TCP/IP layers, and characteristics.
Physical Layer: Network devices, Network topology, Transmission media, Signal transmission, Network performance and transmission impairments, Switching techniques.

Unit 2	Data Link Layer	09 hours

Data Link layer: Design issues, Framing, Error Detection and Correction, Flow control (Elementary Data Link Protocols, Sliding Window protocols).
Medium Access Control and Local Area Networks: Multiple access protocols: Random Access (ALOHA, CSMA, CSMA/CD, CSMA/CA), Controlled Access (Polling, Reservation, Token Passing), Channelization (FDMA, TDMA, CDMA) Overview of Wired and Wireless LAN, Bluetooth.

Unit 3	Network Layer	09 hours

Network Layer: Network layer design issues, Logical addressing: Basic internetworking, Subnetting, network layer protocols: IPv4, IPv6
Routing: Static and dynamic routing, Routing algorithms and protocols (Distance Vector Routing, Link State Routing, Hierarchical Routing)

Unit 4	Transport Layer	09 hours

Transport Layer: Process-to-process delivery, Transport layer protocols (UDP and TCP)
Congestion control in TCP: Congestion Control: Open Loop, Closed Loop; Quality of service: techniques to improve QoS: Leaky bucket algorithm, Token bucket algorithm

Unit 5	Application Layer	09 hours

Session Layer: Session Management, RPC
Presentation Layer: Cryptography: Symmetric-key cryptography, Asymmetric-key cryptography, Digital Signature, **Compression:** Lossless and Lossy Compression

Application Layer: Domain Name System, World Wide Web and Hyper Text Transfer Protocol, Electronic mail, File Transfer Protocol, Simple Network Management Protocol.

Total Lecture Hours	**45 hours**

Textbook:

1. Behrouz Forouzan, "Data Communication and Networking" Fourth Edition-2006, Tata McGraw Hill
2. Andrew Tanenbaum "Computer Networks", Fifth Edition-2011, Prentice Hall.
3. William Stallings, "Data and Computer Communication", Eighth Edition-2008, Pearson.

Reference Books:

1. Kurose and Ross, "Computer Networking- A Top-Down Approach", Eighth Edition-2021, Pearson.
2. Peterson and Davie, "Computer Networks: A Systems Approach", Fourth Edition-1996, Morgan Kaufmann
3. Behrouz Forouzan, "TCP/IP Protocol Suite", McGraw Hill.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
30	30	6	6	3	75	150
60		15				

Course Code: HS201L	Course Name: Quantitative Aptitude & Logical Reasoning-II	L	T	P	C
		2	0	0	2

Course Objectives:

1. Students will gain a strong foundation in basic mathematical concepts, and become proficient in performing calculations, solving equations, and manipulating numerical data.
2. Students will learn to think critically, make deductions, and draw logical conclusions based on given information. This will enable them to approach complex problems systematically and derive accurate solutions.
3. Students will learn how to interpret and analyze data presented in various formats such as tables, graphs, and charts. This skill is particularly useful for tasks involving statistical analysis and decision-making.
4. It will help students to improve their speed and accuracy in solving problems. Through practice and exposure to different types of questions, students will learn time-saving techniques and develop strategies to effectively manage time during exams or real-world scenarios.
5. It will help students to prepare for various competitive exams, such as aptitude tests for job placements, college admissions tests for higher education (GATE/CAT/GRE/GMAT), and government entrance exams. The course will provide the necessary knowledge and practice to perform well in these exams.

Course Outcome:

Students will be able to:	
CO1	Analyze problems, identify relevant information, and apply appropriate mathematical methods to reach solutions.
CO2	Analyze tasks and activities by following a chain of thought process and find logical solutions to a problem.
CO3	Analyze trends, patterns, and relationships within the data.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	2	-	-	-	-	-	1
CO2	2	2	-	-	-	-	-	1
CO3	1	1	-	-	-	-	-	1

Unit 1	Blood Relation, Dice, Cube, Clock, and Calendar	6 hours

Basic concepts, definition and terminology related to blood relationships, Conversation based blood relationships, Family Tree based problems, Coded relationships, Definition and Introduction of Concept and Relation of angle and time, Overtaking, overlapping, right-angle and straight Angle with respect to time, Error in clock (faster and slower), Correct time of clock, Mirror and Water Image of

clock, Introduction of Calendar, Concept of Normal and Leap Year, Finding Odd days, Finding the day of the week of given date with and without reference.		
Unit 2	**Time, Speed and Distance (TSD), Time and Work**	**6 hours**
Basic concepts, definition, and terminology related to TSD, Direct and Indirect relation of TSD, Unit Conversions, Average Speed, Relative Speed, Problem on Train, Problem on Boat & Stream, Race, and Games, Introduction to the Time and Work, Work and Wages, Efficiency based problems, Time and work-based problems, Pipe & Cistern based problems.		
Unit 3	**Verbal and Critical Reasoning**	**6 hours**
Syllogism, Argument – Identifying the Different Parts (Premise, assumption, conclusion), Course of Action, Cause and Effect, Assertion and Reason, Statement and Assumption.		
Unit 4	**Permutation, Combination, and Probability**	**6 hours**
Definition and Introduction of permutation and combination, Fundamental principle of counting by "AND" or "OR" rule, Forming of Numbers, Words, and Team, Problems related to linear and nonlinear arrangement, Match, handshake, Whole number, and Natural number solution, etc., Distribution: - Identical & Distinct, Basic concepts, definition and terminology related to probability, Problem based on Dice, Coins, Playing cards, Marbles, etc., Conditional Probability – Bayes theorem, Question based on Binomial theorem.		
Unit 5	**Analytical and Nonverbal Reasoning, Data Sufficiency**	**6 hours**
Analytical Reasoning Problems, Mirror Image, Water Image, Paper cutting and folding, Embedded Figures, Non-verbal series, Problems based on Data Sufficiency.		
	Total Lecture Hours	**30 hours**

Textbook:

1. "Quantitative Aptitude for Competitive Examinations" by R.S. Aggarwal, S. Chand Publication, 2017.
2. "A Modern Approach to Verbal & Non-Verbal Reasoning" by R.S. Aggarwal, S. Chand Publication, 2018.

Reference Books:

1. "How to Prepare for Quantitative Aptitude for the CAT" by Arun Sharma, 10th Edition, TMH Publication, 2022.
2. "How to Prepare for Logical Reasoning for the CAT" by Arun Sharma, 7th Edition, TMH Publication, 2024.
3. "How to Prepare for Data Interpretation for the CAT" by Arun Sharma, 8th Edition, TMH Publication, 2024.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
20	20	4	4	2	50	100
40		10				

Course Code: CA301B	Course Name: Data Structures using Java	L	T	P	C
		2	0	4	4

Pre-requisite: NA

Course Objectives:

1. Provide the knowledge of basic data structures and their implementations.
2. Understand importance of data structures in context of writing efficient programs.
3. Develop skills to apply appropriate data structures in problem solving

Course Outcome:

Students will be able to:	
CO1	Understand basic concepts of OOPs.
CO2	Implement linear data structures like linked list, stacks and queues.
CO3	Implement nonlinear data structures like tree and graphs.
CO4	Analyze various sorting and searching algorithms.
CO5	Use Standard Template Library (STL) in Java

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	1	-	-	-	1
CO2	3	3	3	1	-	-	-	1
CO3	3	3	3	1	-	-	-	1
CO4	3	3	3	1	-	-	-	1
CO5	3	3	3	1	-	-	-	1

Unit 1	**Introduction to Data Structures and Java**	**18 hours**

Introduction: Data Structure, types of Data Structure, Algorithm, Abstract Data Types (ADTs), Analysis of an algorithm.
Java Fundamentals: Classes, Objects, Packages, Access Modifiers, Memory Management, Exception Handling, File I/O (Basics), Wrapper Classes, Auto-boxing/unboxing, Generics and Type Safety.
Arrays: Declaration, Initialization, Traversal, 1D and 2D arrays, ArrayList and Vector (from java.util), Sliding Window Technique (for competitive coding), Two Pointer Technique.
Strings: String and StringBuilder classes, String manipulation: substring, indexOf, charAt, equals, Anagram check, Palindrome check.
Hands-on Activities:
Implement array operations (insert, delete, search)
Create a simple class for student records
Pointer manipulation and dynamic memory allocation
String manipulation functions without using built-in libraries

Unit 2	**Recursion, Sorting and Searching**	**18 hours**

Recursion: Basics of recursion, Tower of Hanoi, Tail recursion, Recursion Tree & Time Complexity, Backtracking: N-Queens, Rat in a Maze.
Searching: Linear and Binary Search, Ternary Search, Applications in competitive programming: Search in Rotated Array, First/Last Occurrence
Sorting: Bubble, Selection, Insertion, Merge Sort, Quick Sort, Counting Sort, Radix Sort, Java built-in sort (Arrays.sort, Collections.sort with Comparators)
Hands-on Activities:
Implement recursive functions for Factorial, Fibonacci series, tower of Hanoi and Binary search.
Implement above mentioned Searching techniques.
Implement above mentioned sorting techniques and analyze sorting performance with large datasets.
Implement custom sorting using Java Comparators (Collections. sort (List<T>)).
Implement Rat in a Maze problem.

Unit 3	**Linear Data Structures**	**18 hours**

Linked Lists: Singly and Doubly Linked Lists, Creation, Insertion, Deletion, Traversal, Reversal, Detect and Remove Loop (Floyd's Cycle), Merge two sorted lists
Stacks: Stack implementation using arrays and LinkedList, Infix to Postfix, Postfix Evaluation, Nearest Greater/Smaller Element problems
Queues: Simple Queue, Circular Queue, Deque (Double-ended queue) – ArrayDeque, Priority Queue (Min/Max Heap)
Advance Concepts: Monotonic Stack/Queue (Stock Span, Histogram Area), Stack-based problems (Balanced Parentheses, Next Greater Element)
Hands-on Activities:
Implement linked list with all possible case for-insert, delete, traversal and search functions.
Build an Infix to postfix expression generator.
Build a postfix expression evaluator using stack
Develop a palindrome checker using stack

Unit 4	**Non-Linear Data Structures**	**18 hours**

Trees: Binary Tree and Binary Search Tree, Level, Height, Diameter, Tree Traversals (Inorder, Preorder, Postorder, Level Order), LCA (Lowest Common Ancestor), Balanced Trees (AVL basic), B+ Tree, Heap (Min/Max), Heap sort.
Graph Representations: Adjacency Matrix and List, **Graph Traversals:** BFS and DFS

Hands-on Activities:
Implement a binary search tree with insert, delete, and search
Perform and visualize tree traversals (inorder, preorder, postorder)
Create a min-heap and implement priority queue operations
Graph traversal using BFS and DFS with adjacency list

Unit 5	**Standard Template Library (STL) in Java**	**18 hours**

Introduction to Java Collections Framework: Overview of java.util package, Interfaces: Collection, List, Set, Map, Queue, Deque.
Collections class: sort(), reverse(), max(), min(), binarySearch()
List Implementations: ArrayList<E>, LinkedList<E>
Stack and Queue Implementations: Stack<E>, Queue<E>, PriorityQueue<E>, Use of Deque in sliding window problems.
Hands-on Activities:
Use vector and list to store and manipulate data,
Use set for duplicate elimination
Solve problems using STL algorithms (sort, find, etc.)
Integrate STL with user-defined classes.

Total Lecture Hours	**90 hours**

Textbook:
1. Data Structures and Algorithms Made Easy by Narasimha Karumanchi (Java version), CareerMonk Publications, 2016.
2. Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 3rd Edition, MIT Press, 2009.
3. Lipschutz Seymour, Data Structures, Revised First Edition, Schaum's Outline Series, McGraw Hill Education India, 2017.
4. Anany Levitin, Introduction to the Design and Analysis of Algorithms, 3rd Edition, Pearson Education, 2012.

Reference Books:
1. Michael T. Goodrich, Roberto Tamassia, and Michael H. Goldwasser, Data Structures and Algorithms in Java, 6th Edition, Wiley India, 2014.
2. Robert Lafore, Data Structures and Algorithms in Java, 2nd Edition, Sams Publishing, 2002.
3. Peter Drake, Data Structures and Algorithms in Java, 1st Edition, Pearson Education, 2006

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
40	40	8	8	4	100	**200**
80		**20**				

Course Code: CS203B	**Course Name: Database Management Systems**	**L**	**T**	**P**	**C**
		2	0	4	4

Pre-requisite: Concepts of any Programming Language.

Course Objectives:
1. Learn the difference between traditional file-based systems and modern database systems.
2. Understand the relational data model and its concepts, including tables, tuples, attributes, and keys.
3. Design E-R Diagrams and to convert it into relational model.
4. Understand normalization and the use of normal forms to reduce data redundancy and improve data integrity.
5. Apply SQL and PL/SQL on Oracle Database.

Course Outcome:

Students will be able to:	
CO1	Understand the concepts of ER modeling, relational data models, and database architecture.
CO2	Apply relational algebra and SQL commands for database creation and data retrieval.
CO3	Understand ER-to-relational mapping and enforce integrity constraints in databases.
CO4	Apply normalization techniques to optimize relational database design.
CO5	Apply SQL and PL/SQL concepts to solve database problems.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	-	-	-	-	2
CO2	3	2	2	3	-	-	-	2
CO3	3	2	2	3	-	-	-	2
CO4	3	3	3	-	-	-	-	2
CO5	3	3	3	3	-	-	-	2

Unit 1	**Introduction to DBMS and ER Modeling**	**18 hours**

Basics: Data, Information, Database, Database Management Systems. Database Systems Vs File Systems, Data Models, Instances & Schemas, Database Architecture: Levels of Abstraction and Data Independence.

Conceptual Data Modeling (E-R Modeling): Entity Types, Entity Sets, Attributes, Relationship Types, Roles and Mapping Constraints, Weak Entity.

Hands on:

- Case Studies: Design E-R diagrams for scenarios (e.g., Banking, Library, Airline).
- Practical Assignment 1: Analyze and create E-R diagrams for a given use case.
- Setting up databases in MySQL/Oracle.
- Practical Assignment 2: Use of DDL Commands.
- Practical Assignment 3: Use of DML Commands (SELECT, INSERT, UPDATE, DELETE).

Unit 2	**Relational Data Model and SQL Basics**	**18 hours**

Relational Model Concepts: Tables, Tuples, Attributes, Keys. Integrity Constraints: Entity Integrity, Referential Integrity, Domain Constraints. ER-to-Relational Mapping.

Hands on:

- Practical Assignment 4: SELECT with WHERE and DISTINCT clauses.
- Practical Assignment 5: Column Alias, LIKE, BETWEEN, IN clauses.
- Practical Assignment 6: Implementation of constraints in SQL.

Unit 3	**Relational Algebra and Advanced SQL**	**18 hours**

Relational Algebra: Operations (Select, Project, Union, Intersection, Cartesian Product, Join). Query Representation in Relational Algebra.

Introduction to SQL Joins, Subqueries, and Nested Queries, Built-in Functions

Hands on:

- Practical Assignment 7: String & Numeric Functions.
- Practical Assignment 8: Implementation of Joins.
- Practical Assignment 9: Implementation of Subqueries.

Unit 4	**Functional dependency & Database Normalization**	**18 hours**

Concept of Functional Dependencies: Inference Rules, Closure of Attributes, FD Equivalence & Minimal Cover.

Normalization Concepts: 1NF, 2NF, 3NF, BCNF. Lossless Decomposition and Dependency Preservation.

Hands on:

- Practical Assignment 10: Implementation of Aggregate Functions, Order by, GROUP BY, and HAVING.

Unit 5	**PL/SQL Programming and Advanced SQL**	**18 hours**

Introduction to PL/SQL: Procedures, Functions, Triggers. Cursors, Exception Handling Views, & Indexing.

Hands on:

- Practical Assignment 11: Implementation of Date & Conversion Functions.
- Practical Assignment 12: Implementation of PL/SQL.
- Practical Assignment 13: Implementation of Cursor, Function, Trigger, Views & Indexes.

	Total Lecture Hours	**90 hours**

Textbook:

1. Elmasri, Navathe, "Fundamentals of Database Systems", 6th Edition, Addition Wesley, 2010.
2. Korth, Silbertz, Sudarshan, "Database Concepts", 6th Edition, Tata McGraw Hill, 2010.

3. Ivan Bayross, "SQL PL/ SQL", 4th Edition, BPB Publications, 2010.

Reference Books:

1. Bipin C. Desai, "An Introduction to Database Systems", Galgotia Publications, 1981.
2. Majumdar & Bhattacharya, "Database Management System", TMH, 2017.
3. Date C J, "An Introduction to Database Systems", 8th Edition, Pearson Education, 2003.
4. Ramakrishnan Raghu, Gehrke Johannes, "Database System Concepts", 2nd Edition, Tata McGraw Hill, 1999.
5. Leon & Leon, "Database Management Systems", 1st Edition, Tata McGraw Hill, 2008.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
40	40	8	8	4	100	**200**
80		**20**				

Course Code: CA105B	Course Name: Advanced Java Programming	L	T	P	C
		2	0	2	3

Pre-requisite: Java OOP's

Course Objectives:

Upon successful completion of this course, students will gain a comprehensive understanding of modern Java programming, including the latest features introduced in recent Java versions. They will be proficient in using the Stream API to write efficient, functional-style code for processing collections. The course also covers JDBC for database connectivity, enabling students to perform CRUD operations and manage relational data in Java applications. Additionally, learners will acquire hands-on experience with Spring Boot, allowing them to build robust, scalable, and production-ready RESTful web applications.

Course Outcome:

Students will be able to:	
CO1	Apply new java features to build java programs.
CO2	Execute the Stream API to process and transform collections using functional programming constructs in Java.
CO3	Implement JDBC for database operations in Java
CO4	Analyze Spring-based Java applications using Java configuration, XML configuration, annotation-based configuration, beans and their scopes, and properties.
CO5	Test web and RESTful Web Services with Spring Boot using Spring Framework concepts.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	1	2	1	2	-	-	-	1
CO2	2	2	2	2	-	-	-	2
CO3	2	2	2	2	1	-	-	3
CO4	1	2	1	2	1	-	-	1
CO5	2	2	3	3	2	-	-	3

Unit 1	Functional Programming in Java	12 hours

Java New Features: Functional Interfaces, Lambda Expression, Method References, Stream API, Default Methods, Static Method, Private Method, For Each Method, Java Module System, Local Variable Type Inference, Switch Expressions, Yield Keyword, Text Blocks, Records, Sealed Classes.

Hands-on: Steam API using Lambda

Unit 2	Stream API in Java	12 hours

Stream API: Creating Streams, use of filter(), map(), reduce(), sorted(), sorted(Comparator), forEach(), toArray(), collect(), distinct(), limit(), skip(), count(), max() and min() method.

Hands-on: Find out list of pass and fail students in given list those attendance is more than 60%.

Unit 3	**JDBC**	**12 hours**

JDBC: Introduction, JDBC Architecture, Types of Drivers, Statement, ResultSet, Read Only ResultSet, Updatable ResultSet, Forward Only ResultSet, Scrollable ResultSet, PreparedStatement, Connection Modes, SavePoint, Batch Updations, CallableStatement, BLOB & CLOB**.**

Hands-on: CRUD App using JDBC

Unit 4	**Spring Framework**	**12 hours**

Spring: Spring Core Basics-Spring Dependency Injection concepts, Introduction to Design patterns, Factory Design Pattern, Strategy Design pattern, Spring Inversion of Control, AOP, Bean Scopes- Singleton, Prototype, Request, Session, Application, Web Socket, Auto wiring, Annotations, Life Cycle Call backs, Bean Configuration styles

Hands-on: Spring IOC Programming using XML and Annotation

Unit 5	**Spring Boot**	**12 hours**

Spring Boot: Spring Boot- Spring Boot Configuration, Spring Boot Annotations, Spring Boot Actuator, Spring Boot Build Systems, Spring Boot Code Structure, Spring Boot Runners, Logger, BUILDING RESTFUL WEB SERVICES, Rest Controller, Request Mapping, Request Body, Path Variable, Request Parameter, GET, POST, PUT, DELETE APIs, Build Web Applications

Hands-on: CRUD API using Spring Boot

	Total Lecture Hours	**60 hours**

Textbook:

1. Deitel & Deitel, "Java How to program", Prentice Hall, 4th Edition, 2000.
2. Modern Java in Action – Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft, 3rd Edition
3. Java: The Complete Reference, Eleventh Edition, 11th Edition. by Herbert Schildt. Released, 2018.
4. Craig Walls, "Spring Boot in Action", 5th Edition, Manning Publication, 2015.

Reference Books:

1. Steven Holzner, "Java Black Book", Edition, Dreamtech, 2001.
2. Balagurusamy E, "Programming in Java", 5th Edition, McGraw Hill, 2017.
3. Java: A Beginner's Guide by Herbert Schildt, 9th Edition, Oracle Press, 2022.
4. Greg L. Turnquist "Learning Spring Boot 2.0 – 2nd Edition", Packt Publication, 2017.
5. AJ HenleyJr(Author), Dave Wolf, "Introduction to Java SpringBoot, Learning by Coding", Independently Published,2019.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
30	30	6	6	3	75	150
60		15				

Practical Courses Detail Syllabus

Course Code: CS206P	Course Name: Operating System Lab	L	T	P	C
		0	0	2	1

Pre-requisite: Basic concept of C language and data structure

Course Objectives: This course provides hands-on experience with operating system concepts, focusing on UNIX/Linux environments. After this course student will be able to apply the operating system algorithms to solve the real-life problems.

Course Outcome: After completion of the course, the student will be able to

1. Apply knowledge of basic UNIX System calls and Shell programming.
2. Implement various CPU scheduling algorithms and deadlock handling techniques.
3. Implement memory management, process synchronization techniques, page replacement techniques, disk scheduling

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	3	-	-	1	1	1	1	2
CO2	3	3	1	2	2	-	-	1	2	2	1	2
CO3	3	3	-	2	2	-	-	1	2	2	1	2

List Of Practical's (Indicative & Not Limited To)

1. Study of hardware and software requirements of different operating systems (UNIX, LINUX, WINDOWS, Android, IOS
2. Execute various UNIX system calls
3. Write a basic program on VI Editor in Linux.
4. Shell Scripting.
5. Implement FCFS CPU Scheduling Policy
6. Implement SJF CPU Scheduling Policy
7. Implement Priority CPU Scheduling Policy
8. Implement Round Robin CPU Scheduling Policy
9. Implementation of Banker's algorithm
10. Implement the solution for Bounded Buffer (producer-consumer) problem using inter process communication techniques- Semaphores
11. Implement the solutions for Readers-Writers problem using inter process communication technique – Semaphore
12. Implementation First Fit contiguous allocation technique
13. Implementation Best Fit contiguous allocation technique
14. Implementation Worst Fit contiguous allocation technique
15. Implement file storage allocation technique:
    -Contiguous (using array)
16. Implement file storage allocation technique:
    - Linked –list (using linked-list)
    -Indirect allocation (indexing)
17. Comparison of Disk Scheduling Algorithms.

Total Hours: 30 hours

Mode of Evaluation

CA1	CA2	ESE	Total
12	13		
25		25	50

Course Code: HS101P	Course Name: Communication Skills	L	T	P	C
		0	0	4	2

Pre-requisite: NA

Course Objectives:

1. Develop a foundational understanding of communication, including voice dynamics for effective speech delivery.
2. Enhance reading comprehension, note-making, and idea organization skills for academic and professional use.
3. Strengthen writing proficiency through vocabulary building and application of effective writing techniques.
4. Build essential listening and professional communication skills for employability.

Course Outcome: After completion of the course, the student will be able to

1. Understand the essentials of communicating in a professional setting.
2. Employ correct English usage and formal style of Listening - speaking.
3. Apply the usage of verbal and non-verbal cues in presentation and day-to-day communication.
4. Illustrate Communication skills that meet the nature and objectives of the workplace.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	-	-	-	-	-	-	2	3	-	1
CO2	-	-	-	-	-	-	-	-	2	3	-	1
CO3	-	-	-	-	-	-	-	-	2	3	-	1
CO4	-	-	-	-	-	-	-	-	2	3	-	1

Unit-1	**Echoes of Understanding** *Listening, Self-Introduction, Reading Skills and Vocabulary Enhancement*	**15 hours**

Listening Comprehension using videos from software. Ice Breaking Session: Assessment of communication skills through brief self-introductions stage time (1-2 min per student); Reading Comprehension Exercise – Analytical Understanding of Informational Text using books/scientific text.Self-Analysis using SWOT/SWOC (Identification of Individual Strength, Weakness, Opportunity and Threats/Challenges);**Read Aloud Activity and Assessment** using newspapers like ***The Times of India***, ***The Hindu***, **and** ***The Indian Express u****sing* Reading methods -Skimming, Scanning, Churning and Assimilation.
Vocabulary Enrichment using Gamified methods; Doubt clearing session, Vocabulary quiz using Cohort.

Unit 2	**The Art of Expression** *Writing, Grammar concept building and Speaking Skills*	**15 hours**

Writing practices: Paragraph development in 100-120 words, Explaining complex concepts simply; **Grammar concept relevant to scientific communication:** Identifying and correcting grammar errors using worksheet or quiz related to: Parts of Speech, Tenses, voice, Preposition, Articles & conjunctions; Transforming sentences simple to complex, Extempore: Interactive and Communicative Practical with emphasis on Oral Presentation, based on International Phonetic Alphabets (I.P.A.)Written Professional Communication: Email, Formal Letters, and Applications used in professional settings
Speaking: Team presentation enhancing collaborative presentation skills; Introducing person, places and events.; Impromptu Speaking; "Just a Minute" speaking game on general awareness/social/scientific topics.

Unit 3	**Beyond the Lines** *Formal Speaking and Critical Thinking*	**15 hours**

Team presentations for idea pitching using visual aids and supporting materials; Practice of kinesics, paralinguistics,
Peer feedback, Strategies for answering questions effectively
Voice modulation through Role Play Presentation with Stress and Intonation activities based on real-time scenarios;
Development of persuasive communication skills through structured debates.
Script/Essay writing on PESTLE(Political, Economic, Scientific, Technical, Literary and Engineering) range of topics.

Unit 4	**Express & Engage** – *Practical Communication Approach*	**15 hours**

Official/Public Speaking based on suitable Rhythmic Patterns, Turn-a-coat.
Group Discussion: Practical based on Accurate and Current Grammatical Patterns

	Total Hours	**60 hours**

Mode of Evaluation

MSE1	MSE2	CA1	CA2	ESE	Total
40	-	-	10		
50				50	100

Course Code: IT104P	Course Name: Innovation and Entrepreneurship	L	T	P	C
		0	0	2	1

Pre-requisite: NA

Course Objectives:

The course will provide hands-on learning experiences, problem-solving skills, product development knowledge, and interpersonal skills necessary for future entrepreneurs. By the end of the course, students will be equipped to start working on their start-up ideas or develop entrepreneurial competencies that will be beneficial for careers in both industry and business.

Course Outcome: After completion of the course, the student will be able to

1. Understand different types of innovation, innovative thinking and their role in solution of real-world challenges.

2. Understand creative problem-solving skills and use the Business Model Canvas to shape viable business ideas.
3. Analyze market research, identify target customers, and validate business ideas using data-driven insights.
4. Understand pitch business ideas, integrate expert feedback, and pursue funding or mentorship opportunities.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High)

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	-	-	-	2	2	-	2	-	2	2
CO2	-	-	-	-	-	2	2	-	2	-	2	2
CO3	-	-	-	-	-	2	2	-	2	-	2	2
CO4	-	-	-	-	-	2	2	-	2	-	2	2

Unit 1	**Innovation & Creativity**	**07 hours**

- Resource Person: Technical Expert/ Innovator/Entrepreneur
- Content Overview: Introduction to Innovation, the importance of Innovation in life, Type of Innovation, Stages of Innovation, success stories, and opportunities available to students.

Unit 2	**Idea/ Innovation Generation, Commercialization & Business Model Canvas Workshop**	**07 hours**

- Resource Person: Innovation Coaches/Startup Mentors
- Content Overview: Techniques for brainstorming, creativity exercises, introduction to the Business Model Canvas, and developing business concepts.

Unit 3	**Market Research and Validation Workshop**	**08 hours**

- Resource Person: Market Research Analysts/Marketing Professors
- Content Overview: Conducting market research, understanding target customers, market segmentation, and validating business ideas.

Unit 4	**Prototype Development & Pitching Workshop**	**08 hours**

- Resource Person: Product Developers/Venture Capitalists
- Content Overview: Creating a minimum viable product (MVP), hands-on prototyping, crafting, and delivering a compelling pitch.

Total Hours	**30 hours**

For reference: To ensure maximum engagement and learning, the course will be delivered through:

- Ignite (Master Class)
- Startup Interactions: lectures from successful entrepreneurs, startup founders, and investors
- IPR workshop
- Team Formation (Interdisciplinary minimum 3 dept)
- Mentorship & Guidance: Faculty mentors
- Hackathons: Prototyping, branding, and pitching

Mode of Evaluation

MSE	CA1	CA2	ESE	Total
-	25	25		
50			-	50

Course Code: CA204P	Course Name: Python Workshop	L	T	P	C
		0	0	2	1

Pre-requisite: NA

Course Objectives:

To equip learners with practical knowledge of Python programming and essential data science tools, enabling them to build machine learning models, perform data analysis, and explore modern AI techniques including deep learning and large language models using Google Colab.

Course Outcome:

Students will be able to:	
CO1	Write basic Python programs using loops, conditionals, and functions.
CO2	Analyze data using Pandas and NumPy.
CO3	Build classification and clustering models using scikit-learn.
CO4	Create simple CNN models for image classification.
CO5	Use large language models for text-based tasks in Google Colab.

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	1	1	-	-	2	-	-	1
CO2	1	1	-	-	2	-	-	-
CO3	2	2	-	-	2	-	-	1
CO4	1	2	-	-	1	-	-	-
CO5	1	-	-	-	-	-	-	1

List of Practicals

Practical 1: Python Basics & Google Colab
- Set up Google Colab and create your first notebook.
- Write Python scripts using variables, basic data types, and type casting.
- Practice arithmetic, logical, and comparison operations.
- Write input/output code to interact with users.
- **New Addition: Introduce string operations, including concatenation, indexing, slicing, and common string methods.**

Practical 2: Conditionals and Loops
- Write scripts using if, elif, and else conditions.
- Create loops with for and while constructs.
- New Addition: Implement nested loops and use control statements like break and continue.

Practical 3: Functions & Data Structures
- Define and call Python functions with arguments and return values.
- Write lambda functions for simple calculations.
- Create and manipulate lists, tuples, sets, and dictionaries.
- **New Addition: Introduce list comprehensions and dictionary comprehensions for concise data processing.**

Practical 4: NumPy for Numerical Computing
- Create 1D and 2D arrays using NumPy.
- Perform indexing, slicing, and array reshaping.
- Use broadcasting and arithmetic operations on arrays.
- Calculate mean, median, standard deviation using NumPy functions.

Practical 5: Data Cleaning with Pandas
- Load CSV data into a Pandas DataFrame.
- Inspect and clean missing or duplicate values.
- Filter and transform data types.
- Perform basic EDA like describe(), value_counts(), and groupby().

Practical 6: Data Visualization & Preprocessing
- Plot histograms, bar charts, and heatmaps using Matplotlib and Seaborn.
- Apply feature scaling and encoding to dataset columns.
- Split a dataset into training and testing sets.
- Demonstrate overfitting and underfitting on a small model.

Practical 7: Classification Models – SVM & Decision Tree
- Train a Support Vector Machine on a classification dataset.
- Train and visualize a Decision Tree classifier.
- Use sklearn to evaluate models (accuracy, confusion matrix).

Practical 8: Clustering Techniques
- Perform K-Means clustering on a customer dataset.
- Visualize clusters using scatter plots.

Practical 9: Convolutional Neural Networks (CNNs)
- Build and compile a CNN using Keras.
- Train the CNN on MNIST datasets.
- Evaluate model accuracy and loss over epochs.
- Visualize predictions using matplotlib.

Practical 10: Large Language Models (LLMs)
- Tokenize and summarize text using Hugging Face Transformers.
- Use an OpenAI or Hugging Face LLM to generate text in Colab.
- Implement a chatbot or summarizer using a pre-trained model.

Total Lecture Hours	**30 hours**

Textbook:
1. C. R. Severance, Python for Everybody: Exploring Data Using Python 3. CreateSpace Independent Publishing, 2016.
2. A. Sweigart, Automate the Boring Stuff with Python: Practical Programming for Total Beginners, 2nd ed. No Starch Press, 2019.

Reference Books:
1. J. VanderPlas, Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media, 2016.
2. A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 2nd ed. O'Reilly Media, 2019.

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
NA	NA	12	13	NA	25	50
NA		25				

Course Code: HS202P	Course Name: Campus to Corporate	L	T	P	C
		0	0	2	1

Pre-requisite: Graduate students are expected to have intermediate proficiency in English Communication Skills.

Course Objectives:
- To enhance the students' employability skills through practical implementation of their learning.

Course Outcome: After completion of the course, the student will be able to

Students will be able to:	
CO1	Articulate thoughts to display effective communication skills in several professional contexts.
CO2	Organize their thoughts effectively into writing specifically for professional contexts.
CO3	Adapt to the recruitment scenario in the industry for an enhanced ability to meet the industry requirements

CO-PO Mapping (Scale 1: Low, 2: Medium, 3: High):

CO-PO Mapping	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	-	1	-	-	2	-	-	1
CO2	-	1	-	-	2	-	-	1
CO3	-	1	-	-	2	-	-	2

Activity 1	**Elevator Pitch**	**3 hours**
Activity: Individual presentations		
Activity 2	**Public Speaking (Impromptu)** ***(CA-1)***	**3 hours**
Activity: Overcoming stage fright by speaking on a topic for at least a minute		
Activity 3	**Group Discussion** ***(MSE 1)***	**3 hours**
Activity: Students will participate in GDs based on abstract topics/current topics		
Activity 4	**Group presentations** ***(MSE-1)***	**3 hours**
Activity: Each team will do a deep dive into their dream roles and dream organisations, and present their findings in front of the class		
Activity 5	**Personal Branding and Networking**	**2 hours**
Activity: Optimizing LinkedIn profiles and creating e-portfolios		

Activity 6	**Video Resume – Dos and Don'ts**	**2 hours**
Activity: Students need to make their video resumes		
Activity 7	**Role plays (conversation skills)**	**2 hours**
Activity: Students will prepare a skit and act it out in class		
Activity 8	Essentials of Interview Skills – dos and don'ts, art of handling interview questions, Handling HR FAQs	**4 hours**
Activity: Discussion with students		
Activity 9	**Interview Simulation Scenarios & Practice: Peer interviews/pair interviews *(MSE 2)***	**6 hours**
Activity: Conduct peer interviews among students in class		
Activity 10	**Company-based Essay Writing: Tips to write effectively**	**2 hours**
Activity: Students will need to write a paragraph/essay of 200 – 250 words on a given topic		
	Total Lecture Hours	**30 hours**

Reference Book & Links:

1. Soft Skills for Everyone, Cengage Learning, by Jeff Butterfield
2. https://www.mindtools.com/a0uur35/why-soft-skills-matter
3. https://www.dol.gov/agencies/odep/publications/fact-sheets/soft-skills-the-competitive-edge
4. Personality Development and Soft Skills and Soft Skills, By Barun Mitra
5. Business Communication for Managers; Payal Mehra, Pearson Delhi, 2012
6. https://www.indeed.com/career-advice/career-development/professional-skills
7. https://www.peoplescout.com/insights/soft-skills-in-the-workplace/
8. https://www.youtube.com/watch?v=dQ6RNltrXro
9. https://prepinsta.com/capgemini/capgemini-essay-writing-topics-sections/
10. https://virtualspeech.com/blog/impromptu-speech-topics-and-ideas
11. https://www.youtube.com/watch?v=lWFbwcjWtmg

Mode of Evaluation:

MSE		CA			ESE	Total
MSE1	MSE2	CA1	CA2	CA3(ATT)		
20	20	5	5	NA	NA	50
40		10				