



**INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT**

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

---

# **Indira College of Engineering & Management, (ICEM)**

**(An Autonomous Institute affiliated to SPPU)**



**Two Year**

**POST GRADUATE PROGRAMME  
MCA Curriculum for the Batch of 2024 - 2026**





# **MCA Curriculum for the Batch of 2024 - 2026**

**MASTER OF COMPUTER APPLICATIONS  
(CHOICE BASED CREDIT SYSTEM) (MCA-CBCS – 2024-25 ONWARDS)**

## **Preamble:**

### **Overview of Indira College of Engineering & Management Pune**

The Shree Chanakya Education Society (SCES) was established in February 1994 with the explicit vision to provide sustainable impetus to the corporate and entrepreneurial abilities in the youth. With the unfolding of the liberalization process in the Indian economy in 1991, Dr. Tarita Shankar, our founder Secretary and Chief Managing Trustee of SCES felt the need to prepare the youth of the country for opportunities and challenges that came along with the opening of our economy. It was her vision to spread quality higher education among the youth and to equip them with the knowledge and skill required to face the global competition. This vision found an expression with the establishment of Shree Chanakya Education Society in 1994 that envisaged to become a centre for quality education that creates productive participants for a globally competitive economy.

At the heart of its academic ethos lies a relentless pursuit of nurturing future-ready professionals who are equipped to thrive in the ever-evolving landscape of technology and business. With a rich legacy of academic prowess and industry relevance, Indira College of Engineering and Management has emerged as a preferred destination for those aspiring to embark on a transformative educational journey.

Indira College of Engineering and Management (ICEM), established in 2007, is a beacon of academic excellence, dedicated to fostering innovation, leadership, and holistic development. It has consistently upheld its commitment to providing dynamic and versatile programs in engineering, management, and computer applications.

One of the distinctive facets of this institution is its focus on the Master of Computer Applications (MCA) program launched in 2009. and it is affiliated to the Savitribai Phule Pune University (then University of Pune) approved by DTE and recognized by AICTE, New Delhi . Recognizing the pivotal role of computer applications in shaping the digital era, Indira College of Engineering and Management takes pride in offering a comprehensive MCA curriculum that blends theoretical insights with hands-on experiential learning.





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

---

The MCA program at ICEM is meticulously crafted to empower students with the requisite skills and knowledge to excel in diverse domains of computer science and information technology. Through a blend of rigorous coursework, practical projects, and industry-oriented training, students are primed to become adept problem solvers, innovators, and leaders in the realm of computing.

Moreover, the MCA program at ICEM goes beyond conventional boundaries, fostering interdisciplinary learning and collaboration. With a multidisciplinary approach, students can explore emerging fields such as data science, artificial intelligence, cloud computing, and cybersecurity, thereby staying abreast of the latest technological advancements.

Furthermore, the faculty members at ICEM bring a wealth of academic expertise and industry experience, ensuring that students receive mentorship and guidance of the highest caliber. Through interactive teaching methodologies and personalized attention, faculty members play a pivotal role in nurturing the intellectual curiosity and professional acumen of MCA students.

In addition to academic rigor, Indira College of Engineering and Management emphasizes holistic development. Beyond the classroom, students are encouraged to participate in co-curricular activities, industry internships, and research initiatives, thereby honing their soft skills, leadership abilities, and entrepreneurial spirit.

In essence, Indira College of Engineering and Management stands as a testament to academic excellence, innovation, and inclusive growth. Through its transformative MCA program and diverse educational offerings, it continues to inspire and empower the next generation of technocrats and thought leaders, shaping a brighter future for the global community. To keep pace with the current trends, the curriculum needs to be flexible and resilient.

In view of the above, and to enable the students to study in an application-oriented atmosphere, ICEM applied to University Grant Commission for grant of “Autonomous Status” and has been awarded the Autonomy status from Academic year 2024-25.





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

---

### PROGRAM EDUCATIONAL OBJECTIVES (PEOs):

The PEOs of MCA programme are:

- **PEO 1:** To solidify foundations in mathematical, computer science and application concepts necessary to effectively formulate, analyze and solve computer application problems.
- **PEO 2:** To impart advanced knowledge about several sub & dependent domains associated to the field of computer science and applications respect to industry trends
- **PEO 3:** To empower students with team building skills and leadership qualities that prepare them for employment, entrepreneurship and competent professionals to serve society and as per global needs.
- **PEO 4:** To acquaint students with the principles of system analysis, design, development and project management.
- **PEO 5:** To inculcate effectiveness to communicate effectively, work harmoniously in teams with ethical and professional attitude.

### PROGRAM SPECIFIC OBJECTIVES (PSOS):

At the completion of MCA programme, our students shall have:

- **PSO 1:** Ability to design and develop computing systems using fundamentals of Mathematics, Computer science and other related disciplines to meet customers' business objectives.
- **PSO 2:** Ability to test and analyze the quality of various subsystems and integrate them to evolve a larger computing system.
- **PSO 3:** Ability to work professionally with a positive attitude as an individual or in multidisciplinary teams and communicate effectively.





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

---

### Graduate Attributes (GAs)

The graduate attributes for students of MCA are:

- Computational Knowledge
- Problem analysis
- Design/development of solutions
- Conduct investigations of complex problems
- Modern tool usage
- Professional Ethics
- Life-Long Learning
- Project Management and Finance
- Communication Efficacy
- Societal and Environmental Concern
- Individual and Team Work
- Innovation and Entrepreneurship





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

### MCA Programme Outcomes (POs):

At the end of the MCA programme the learner will possess the following Program Outcome:

PO1	<b>Computational Knowledge:</b> Relate & apply fundamental knowledge of computing technology appropriate to the discipline
PO2	<b>Problem Analysis:</b> Ability to Analyze, identify and formulate tangible products/services/solutions/applications with computing requirements to solution.
PO3	<b>Design/development of solutions:</b> Ability to design, implement and evaluate a computer-based system, process, component or program to meet desired needs based solution with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
PO4	<b>Conduct &amp; investigate complex computing problems:</b> Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of information to provide valid conclusions.
PO5	<b>Modern Tool usage:</b> Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
PO6	<b>Professional Development Ethics:</b> Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practice.
PO7	<b>Lifelong learning:</b> Recognize the need, and have ability to engage in independent learning for continual development as a Computing professional.
PO8	<b>Project management and finance:</b> Demonstrate knowledge and understanding of computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO9	<b>Communicate Effectively:</b> Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.
PO10	<b>Social and environmental concern(S):</b> Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

PO 11	<b>Individual and Team Work:</b> Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.
PO 12	<b>Innovation and Entrepreneurship:</b> Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

## REGULATIONS PERTAINING TO MCA DEGREE PROGRAMME

- These regulations shall be cited as Academic Regulations Pertaining to MCA Degree of SCES's Indira College of Engineering & Management, Pune, an Autonomous Institute (2024-25onwards) under Choice Based Credit System of Faculty of Commerce and Management affiliated to the Savitribai Phule Pune University (Formerly University of Pune) duly approved by the Governing Body and Academic Council of ICEM
- These regulations will apply to Master of Computer Applications Programme being run by SCES's Indira College of Engineering & Management, Pune an Autonomous Institute having approval of the University/AICTE / UGC for the department of Master of Computer Applications program.
- **Choice Based Credit System (CBCS)**

Majority of Indian higher education institutions have been following marks or percentage- based evaluation system, which obstructs the flexibility for the students to study the subjects/courses of their choice and their mobility to different institutions. There is a need of flexibility in education system, which will allow students, depending upon their interests and aims, to choose interdisciplinary, intra-disciplinary and skill-based courses. This can be made possible through adoption of choice-based credit system (CBCS), an internationally acknowledged system. The choice-based credit system not only offers opportunities and avenues to learn core subjects but also exploring additional avenues of learning beyond the core subjects thus helping in holistic development of an individual.

The CBCS provides an opportunity for the students to choose from the prescribed courses, comprising core, elective/minor, or skill-based courses. The courses can be evaluated following the grading system, which is better than the conventional marks system.





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

---

- **Minimum Eligibility for admission and admission procedure:**

All admissions to Indira College of Engineering & Management, Pune will be made through an entrance test conducted by DTE, Maharashtra (<http://www.dtemaharashtra.gov.in/index.html>). Merit and the reservation rules based on the seat matrix as announced by the Government of Maharashtra from time to time, shall be applicable. The Institute reserves the right to admit students under Institutional level seats, but the aspirants need to clear the entrance examination of DTE, Maharashtra in accordance with the rules of AICTE.

### **Fee Structure:**

Courses Fees will be decided as per approval from Fees Regulatory Authority.

- **Duration of the Programme:**

The duration of the MCA programme shall extend over 4 semesters (two academic years) of 15 weeks or more each with a minimum of 90 actual working days of instruction in each semester and 2 to 3 weeks of examinations. The candidates shall complete the MCA Programme within 4 years from the date of admission.

### **INSTRUCTIONS FOR TEACHERS FOR INTERNAL EVALUATION:**

The aim of internal assessment is to gauge the extent of comprehension, knowledge, and awareness among learners. To achieve this objective, faculty members are encouraged to implement Continuous Internal Assessment (CIA) techniques, ensuring a fair and unbiased assessment of students.

Through CIA students' progress is continuously evaluated through various means such as quizzes, tests, assignments, and projects throughout a course or academic program. It emphasizes regular feedback, allowing students to understand their strengths and weaknesses and make necessary improvements.

By incorporating multiple assessment methods, CIA provides a holistic evaluation of students' knowledge and skills, aligning with modern educational approaches that prioritize formative assessment and lifelong learning.





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indiracem.ac.in](http://www.indiracem.ac.in)

### EXTERNAL EXAMINATION: -

There will be a written Examination of 50 marks duration of 2 hrs for a 3-credit course and 25 marks duration of 1.5 hours for a 2-credit course at the end of each semester.

- **Grade Calculation:**

The grading system is as follows:

### GRADE and GRADE POINTS for MCA (2024 – 25) Batch

#### Mark wise Grades & Grade Points

Marks out of 100	Grade	Grade Points	Performance
90-100	O	10	Outstanding
75 – 89	A	9	Excellent
65-74	B	8	Very Good
60 – 64	C	7	Good
50-59	P	6	Pass
0-49	F	0	Fail

Percentage Formula = SGPA/CGPA \* 9

Passing Criteria: 40% CIA, 40% TEA and 40% Overall

#### SGPA/CGPA wise Grades

SGPA/ CGPA	Grade
9.00 - 10.00	O
8.00 - 8.99	A
7.00 -7.99	B
6.50 - 6.99	C
6.00 - 6.49	P

SGPA- Semester Grade Point Average

CGPA- Cumulative Grade Point Average

TEA- Term End Assessment





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indiracem.ac.in](http://www.indiracem.ac.in)

---

### Award of Credits:

- Each course having 3&4 credits shall be evaluated out of 100 marks and student should secure at least 40 marks to earn full credits of that course.
- Each course having 2 credits shall be evaluated out of 50 marks and student should secure at least 20 marks to earn full credits of that course.
- Each course having 1 credit shall be evaluated out of 25 marks and student should secure at least 10 marks to earn full credits of that course.
- GPA shall be calculated based on the marks obtained in the respective subject provided that student should have obtained credits for that course.

Following are the session details per credit for each of L-P-T model

- 1) Every ONE-hour session per week of Lecture(L) amounts to 1 credit per semester,
- 2) Minimum of TWO hours per week of Practical(P) amounts to 1 credit per semester,
- 3) Minimum of ONE hours per week of Tutorial(T) amounts to 1 credit per semester

### Lecture-Practical/Project-Tutorial (L-P-T)

A course shall have either or all the three components, i.e. a course may have only lecture component, or only practical/project component or a combination of any two/three components

- **Lecture(L):** Classroom sessions delivered by faculty in an interactive mode. It should be conducted as per the scheme of lectures indicated in the respective course.
- **Practical/Project(P):** Practical / Project Work consisting of Hands-on experience /Field Studies / Case studies that equip students to acquire the much-required skill component. Besides separate Practical/Project course, three courses in each semester include few practical assignments and it will be evaluated under internal evaluation
- **Tutorial(T):** Session consisting of participatory discussion/ self-study/ desk work/ brief seminar presentations by students and such other novel methods that make a student to absorb and assimilate more effectively the contents delivered in the Lecture sessions





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

---

A Mini project is an assignment or part of a software product the student must complete at the end of every semester to strengthen the understanding of fundamentals through effective application of the courses learnt.

**The Project Work** to be conducted in the FINAL Semester and evaluated at the end of the semester. The detailed guidelines have been in the respective course structure.

The teaching / learning as well as evaluation are to be interpreted in a broader perspective as follows:

- i) Teaching – Learning Processes: Classroom sessions, Group Exercises, Seminars, Small Group Projects, Self-study, etc.
- ii) Evaluation: Tutorials, Class Tests, Presentations, Field work, Assignments, competency-based Activity, Research papers, Term papers, etc.

- **Evaluation Pattern:**

Each course carrying 100 marks shall be evaluated with Continuous Internal Assessment (CIA) and End Semester Exam (ESE) mechanism. Continuous Internal Assessment (CIA) shall be 50 marks while End Semester Exam (ESE) shall be 50 marks. To pass the course, a student must secure a minimum of 40 marks if he/she should secure a minimum of 40% of marks in the End Semester Exam (ESE) and Continuous Internal Assessment (CIA) each.

- CIA shall be assessed based on any of the following activities
  - 1. Internal tests
  - 2. Assignments,
  - 3. Case Study / Situation Analysis – (Group Activity or Individual Activity)
  - 4. Field Visit / Study tour and report of the same
  - 5. Small Group Project & Internal Viva-Voce
  - 6. Group Discussion
  - 7. Quiz
  - 8. Role Play / Story Telling
  - 9. Written Home Assignment
  - 10. Book Review





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

### 11. Seminar / Presentation

#### **Guidelines to conduct Mini-Project evaluation for Semester I, Semester II and Semester III Internal Evaluation**

The internal evaluation will be of 50 Marks. It will be distributed as follows

Description	Marks
Project Report	15
Viva	10
Working Demo	25
Total	50

Project Report should be evaluated as Textual chapters should be given 5 marks while diagrams, test cases/validations, screen designs should be evaluated for 10 marks and thus, totaling up to 15 marks. The evaluation will be conducted by an Internal Examiner. Working Demo is given maximum weightage to ensure each group submits an executable version of their project. Examiners should evaluate the efforts and contribution of every individual in the team (in case of group project). The examiner may review the code of the project while evaluating its working demo and modules.

**ATKT Rules:** The ATKT rules mentioned in CBCS handbook (available on university website) are applications to MCA Programme.

#### **Grade Improvement:**

- a. A Candidate who has secured any grade other than F (i.e. passed the MCA programme) and desires to avail the Grade Improvement option, may apply under Grade Improvement Scheme within four years from passing that Examination.
- b. He/she can avail not more than three attempts, according to the syllabus in existence, for grade improvement.
- c. He /she shall appear for Evaluation of at least 1/3+of the Core Mandatory Courses (except Project) for the purpose of Grade Improvement.
- d. Elective courses cannot be selected for grade improvement





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

---

### **Attendance:**

The student must meet the requirement of 75% attendance per semester per course for grant of the term. The Director shall have the right to withhold the student from appearing for examination of a specific course if the above requirement is not fulfilled. Since the emphasis is on continuous learning and concurrent evaluation, it is expected that the student's study all-round the semester. Weightage will be given in CIA for the attendance.

**Medium of Instruction:** The medium of Instruction & Assessment shall be English.

**Maximum Attempts per Course:** A student shall earn the credits for a given course in maximum FOUR attempts.

**Completion of Degree Programme:** A student who earns 98 credits shall be considered to have completed the requirements of the MCA degree program and CGPA will be calculated for such student.





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

## 2 yrs MCA Syllabus structure

(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

Semester-I										
Course Code	Course Type	Course	Teaching Scheme				TOTAL Credits	Evaluation Scheme		
			L	P	T	TOTAL		CIA	ESE	Total
MCA101	CM	JAVA Programming with Spring Framework and Hibernate	3	0	0	3	3	50	50	100
MCA102	CM	Data Structure & Algorithms	3	0	0	3	3	50	50	100
MCA103	CM	Object Oriented Software Engineering	3	0	0	3	3	50	50	100
MCA104	CM	Operating Systems and Shell Programming	3	0	0	3	3	50	50	100
MCA105	CM	Mathematics and Business Statistics	2	0	0	2	2	50	50	100
MCA101L	SEC	Advance JAVA Programming Lab	0	4	0	4	2	50	0	50
MCA102L	SEC	Data Structure & Algorithms Lab	0	4	0	4	2	50	0	50
MCA106	ME	(Major Elective1) Elective 1: Fundamentals of Cloud Computing & Networking Elective 2: Introduction to Web Technologies Elective 3: Introduction to data Science	3	1	0	4	4	50	50	100
MCA107	PROJ	Capstone Project	0	2	0	2	2	50	0	50
MCA108	CM	Soft Skills & Business Communication	1	0		1	1	25	0	25
MCA109	IKS	Yoga	1	0	0	1	1	25	0	25
		Non credit course(Bridge course)								
			<b>19</b>	<b>11</b>	<b>0</b>	<b>30</b>	<b>26</b>	<b>500</b>	<b>300</b>	<b>800</b>





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indiracem.ac.in](http://www.indiracem.ac.in)

Semester II										
Course Code	Course Type	Course	Teaching Scheme				TOTAL Credits	Evaluation Scheme		
			L	P	T	TOTAL		CIA	ESE	Total
MCA201	CM	Python Programming	3	0	0	3	3	50	50	100
MCA202	CM	Advance Database Management System	3	0	0	3	3	50	50	100
MCA203	CM	Software Testing & Tools	3	0	0	3	3	50	50	100
MCA204	RM	Research Methodology	3	0	1	4	4	50	50	100
MCA201L	CM	Python Programming Lab	0	4	0	4	2	50	0	50
MCA205	PROJ	Mini Project-2	0	2	0	2	2	50	0	50
MCA206	ME	(Major Elective2) <b>Choose Any One:</b> Elective 2.1: Cloud Computing Management and Security Elective 2.2: Advance Web Technologies Elective 2.3: Machine Learning	2	2	0	4	4	50	50	100
MCA207	ME	(Major Elective3) <b>Choose Any One:</b> Elective 3.1: IOT Elective 3.2: Goland/ <b>Blockchain</b> Elective 3.3: Power BI	2	2	0	4	4	50	50	100
MCA208	IKS	Vedic Mathematics	1	0	0	1	1	25	0	25
			17	10	1	28	26	425	300	725





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indraicem.ac.in](http://www.indraicem.ac.in)

Semester III										
Course Code	Course Type	Course	Teaching Scheme					Evaluation Scheme		
			L	P	T	TOTAL	TOTAL Credits	CIA	ESE	Total
MCA301	CM	Mobile Application Development	3	0	0	3	3	50	50	100
MCA302	CM	Software Project Management & DevOps	3	0	0	3	3	50	50	100
MCA303	ME	Choose Any 1: Elective 4.1 Cloud Migration and Management Elective 4.2 MERN Stack Development Elective 4.3 Artificial Intelligence and DL	2	2	0	4	4	50	50	100
MCA304	CM	Mobile Application Development Lab	0	4	0	4	2	50	0	50
MCA301L	ME	Choose Any 1: Elective 5.1 Cloud API & Services Elective 5.2 UI-UX Design Elective 5.3 Tableau	2	2	0	4	4	50	50	100
MCA305	ME	Choose Any 1: Elective 6.1 Cyber Security and Ethical Hacking Elective 6.2 E commerce & Digital Marketing Elective 6.3 Recent Technologies (Prompt Engineering, RPA & Generative AI)	2	2	0	4	4	50	50	100
MCA306	RP	Research Project	0	6	0	6	6	150	0	150
			<b>12</b>	<b>16</b>	<b>0</b>	<b>28</b>	<b>26</b>	<b>450</b>	<b>250</b>	<b>700</b>





## INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT

Parandwadi, Pune – 410506, Ph. 02114 661500, [www.indiracem.ac.in](http://www.indiracem.ac.in)

Semester IV										
Course Code	Course Type	Course	Teaching Scheme					Evaluation Scheme		
			L	P	T	TOTAL	TOTAL Credits	CIA	ESE	Total
MCA401	MOOC	Self-Learning Course/ MOOC-1	3	0	0	3	3	50	50	100
MCA402	MOOC	Self-Learning Course/ MOOC-2	3	0	0	3	3	50	50	100
MCA403	MOOC	Self-Learning Course/ MOOC-3	2	0	0	2	2	0	50	50
MCA404	OJT	Industrial Internship/On Job Training	0	24	0	24	12	250	200	450
			<b>8</b>	<b>24</b>	<b>0</b>	<b>32</b>	<b>20</b>	<b>350</b>	<b>350</b>	<b>700</b>
			<b>56</b>	<b>61</b>	<b>1</b>	<b>118</b>	<b>98</b>	<b>1725</b>	<b>1200</b>	<b>2925</b>

Abbreviations	Course Full Name
CM	Major Mandatory
ME	Major Elective
RM	Research Methodology
OJT	Industrial Internship/On Job Training
RP	Research Project
MOC	MOOC Course
CIA	Continuous Internal Evaluation



COURSE TITLE		JAVA PROGRAMMING WITH SPRING AND HIBERNATE FRAMEWORKS			CREDITS	3
COURSE CODE		MCA101	COURSE CATEGORY	CM	L-P-T	3-0-0
Version	1.0	Approval Details	07-2024			

#### ASSESSMENT SCHEME

First Periodical Assessment	Second Periodical Assessment	Seminar/ Assignments/ Project	Surprise Test / Quiz	Attendance	ESE
10%	10%	10%	10%	10%	50%

Course Description	This Advanced Java Programming course is designed to provide MCA students with a comprehensive understanding of advanced Java concepts, focusing on Object-Oriented Programming (OOP) features, multithreading, graphical user interface development with AWT and Swing, web technologies including Servlets and JSP, database connectivity using JDBC, and an introduction to the Spring and Hibernate frameworks.
--------------------	---

Course Objective	<ol style="list-style-type: none"> <li>To deepen the understanding of advanced OOP &amp; exception handling concepts in Java.</li> <li>To enhance skills in multithreading and concurrency in Java.</li> <li>To develop proficiency in building graphical user interfaces using AWT and Swing.</li> <li>To understand and implement web applications using Servlets and JSP.</li> <li>To establish a foundation in database connectivity with JDBC.</li> <li>To introduce the core concepts of the Spring and Hibernate frameworks.</li> </ol>
------------------	--

Course Outcome	<p>At the end of the course, students will be able to:</p> <p><b>CO1:</b> Demonstrate the application of advanced Object-Oriented Programming (OOP) features in Java. <b>(Apply)</b></p> <p><b>CO2:</b> Implement multithreading and manage concurrency in Java applications. <b>(Apply)</b></p> <p><b>CO3:</b> Design and develop graphical user interfaces using AWT and Swing. <b>(Create)</b></p> <p><b>CO4:</b> Perform database operations using Java Database Connectivity (JDBC). <b>(Apply)</b></p> <p><b>CO5:</b> Develop dynamic web applications using Servlets and JavaServer Pages (JSP). <b>(Create)</b></p> <p><b>CO6:</b> Understand and utilize basic concepts of the Spring and Hibernate frameworks. <b>(Understand)</b></p>
----------------	--

#### Prerequisite :

The prerequisite for an Advanced Java Programming subject is a foundational understanding of Java programming, including basic concepts of object-oriented programming (OOP), data structures, exception handling, and multithreading. Familiarity with core Java libraries and basic GUI development is also recommended.

MODULE	Weightage(%)	CO Mapping
<b>MODULE 1: Overview of OOP Features &amp; Exception Handling</b>		
1.1 Inheritances and Polymorphism 1.2 Abstract Classes and Interfaces 1.3 Inner Classes and Anonymous Classes 1.4 Exceptions & Errors 1.5 Types of Exception 1.6 Use of try, catch, finally, throw, throws in Exception Handling 1.7 In-built and User Defined Exceptions 1.8 Checked and Un-Checked Exceptions 1.9 Java Collection Framework,	7	15%  <b>CO-1</b>

<b>MODULE 2: Multithreading and Concurrency</b>			
2.1 Understanding Multi-Threaded Programming 2.2 Thread Lifecycle 2.3 Thread Priorities 2.4 Synchronizing Threads 2.5 Inter Communication of Threads 2.6 Deadlock	4	10%	CO-2
<b>MODULE 3: AWT , Swing &amp; JDBC (Java Database Connectivity)</b>			
3.1 Introduction to Abstract Window Toolkit (AWT) 3.2 Event Handling in AWT 3.3 Introduction to Swing and Swing Components 3.4 Layout Managers 3.5 Building GUI Applications with AWT and Swing 3.6 Introduction to JDBC, JDBC Architecture and API, 3.7 Connecting to Databases 3.8 Executing SQL Queries Prepared Statements and Callable Statements 3.9 ResultSet and Metadata	7	15%	CO-3
<b>MODULE 4: Java Web Technologies (Servlets and JSP)</b>			
4.1 Introduction to Servlets 4.2 Request and Response Handling 4.3 Session Management 4.4 Introduction to Java Server Pages (JSP), JSP Tags, Scriptlets, and Expression Language 4.5 MVC Architecture in Java Web Applications	5	15%	CO-4
<b>MODULE 5: Spring</b>			
5.1 Overview of Spring Framework 5.2 Spring Core Concepts 5.3 Dependency Injection (DI) and Inversion of Control (IoC) 5.4 Spring Beans and Bean Lifecycle 5.5 Introduction to Spring MVC	10	20%	CO-5
<b>MODULE 6: Hibernate framework</b>			
6.1 Overview of Hibernate, Object-Relational Mapping (ORM) Concepts 6.2 Hibernate Architecture 6.3 Hibernate Configuration and Mapping, 6.4 CRUD Operations with Hibernate 6.5 Hibernate Query Language (HQL)	12	25%	CO-6
<b>TEXTBOOKS</b>			
1 "Modern Java in Action" by Raoul-Gabriel Urma, Mario Fusco, and Alan Mycroft 2 "Spring in Action" by Craig Walls "Java Persistence with Hibernate" by Christian Bauer, Gavin King, and Gary Gregory			
<b>REFERENCE BOOKS</b>			
1 "Java: The Complete Reference" by Herbert Schildt 2 "Java 8 Programming", BlackBook, DreamTech Press, Edition 2015			
<b>E BOOKS</b>			
1 Website: <a href="https://www.oracle.com/java/technologies/javase/jdk13-archive-downloads.html">https://www.oracle.com/java/technologies/javase/jdk13-archive-downloads.html</a> "Head First Java" by Kathy Sierra and Bert Bates Website: 3 <a href="https://www.oreilly.com/library/view/head-first-java/9780596009205/">https://www.oreilly.com/library/view/head-first-java/9780596009205/</a> <b>Online tutorials and resources:</b> Oracle Java Tutorials: <a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a>			

1	JavaTPoint: <a href="https://www.javatpoint.com/">https://www.javatpoint.com/</a>
2	Baeldung: <a href="https://www.baeldung.com/">https://www.baeldung.com/</a>
3	
<b>MOOC</b>	
1	edX: Course Title: "Advanced Java Programming" Provider: Hong Kong University of Science and Technology (HKUST) Link: Advanced Java Programming
2	Udacity: Course Title: "Java Programming Basics" Provider: Udacity Link: Java Programming Basics

COURSE TITLE		ADVANCE JAVA PROGRAMMING LAB			CREDITS	2
COURSE CODE		MCA101L	COURSE CATEGORY	SEC	L-P-T	0-4-0
Version	1.0	Approval Details		07-2024		

ASSESSMENT SCHEME					
Assignments	Attendance	Internal Exam		ESE	
40%	20%	40%		00%	
<p><b>Practical Assignment Questions</b></p> <p>1. Write a Java program that takes two integers as input from the user and performs division, handling division by zero and invalid input types.</p> <p>2. Create a Java program that reads from a user-specified file, implementing exception handling for file not found and I/O errors.</p> <p>3. Create a class hierarchy for animals. Design a base class Animal with properties like name and age. Then, create two subclasses: Dog and Cat. Each subclass should have a method sound() that returns the sound the animal makes.</p> <p>4. Design a class hierarchy for bank accounts. Create a base class BankAccount with properties like accountNumber and balance. Then, create two subclasses: SavingsAccount and CurrentAccount. Implement methods to deposit and withdraw money, and override a method to display account details specific to each account type.</p> <p>5. Develop a class hierarchy for geometric shapes. Create a base class Shape with a method area(). Then, implement two subclasses: Circle and Rectangle. Each subclass should have a constructor to initialize its dimensions and override the area() method to calculate the area of the shape.</p> <p>6. Implement a Java program demonstrating the use of abstract classes and interfaces in a banking application scenario. Define classes Account (abstract class), SavingsAccount, and CurrentAccount implementing different interfaces for operations like deposit, withdraw, and calculateInterest.</p> <p>7. Implement a Java program to demonstrate multithreading using the Runnable interface for printing numbers 1 to 10 using two threads.</p>					

**8.** Write a Java program that creates two threads. The first thread should print numbers from 1 to 10 with a delay of 500 milliseconds between each number. The second thread should print the letters from 'A' to 'J' with a delay of 700 milliseconds between each letter. Use the Thread class to create the threads.

**9.** Create a Java program that uses multiple threads to increment a shared counter. Implement a class Counter with a synchronized method increment() that increases the counter by 1. Create three threads that each increment the counter 1000 times. After all threads finish, print the final value of the counter to ensure it is correct.

**10.** Design a simple GUI application using Swing components that includes a JFrame with a JLabel, a JTextField, and a JButton. When the button is clicked, the text entered in the text field should be displayed in the label

- Create a JFrame.
- Add a JLabel to display instructions.
- Add a JTextField for user input.
- Add a JButton to trigger the action.
- Implement an ActionListener for the button to update the label with the text from the text field.

**11.** Experiment with different layout managers in Java to understand their behavior. Create a JFrame with multiple JButtons arranged using different layout managers such as BorderLayout, FlowLayout, GridLayout, and BoxLayout.

- a. Create a JFrame.
- b. Add multiple JButtons with different labels.
- c. Use different layout managers for each button to observe their arrangement.

**12.** Develop a menu-driven GUI application using Swing components. The application should include a menu bar with options for File (with sub-options New, Open, Save, Save As, Exit) and Edit (with sub-options Cut, Copy, Paste). Implement basic functionalities for each menu option.

- Create a JFrame.
- Add a JMenuBar.
- Add JMenu items for File and Edit.
- Add JMenuItems for the sub-options under each menu.
- Implement ActionListeners for each menu item to perform the respective actions (e.g., display a dialog for New/Open, save a file for Save, exit the application for Exit, etc.).

**13.** Develop a Java program that demonstrates basic event handling using buttons. Create a JFrame with two buttons labeled "Button 1" and "Button 2". When "Button 1" is clicked, display a message saying "Button 1 clicked!" and when "Button 2" is clicked, display a message saying "Button 2 clicked!"

- Create a JFrame.
- Add two JButtons with labels "Button 1" and "Button 2".
- Implement ActionListeners for each button to handle the click events.
- Display appropriate messages when each button is clicked.

**14.** Develop a Java program that demonstrates custom events and listeners. Create a scenario where an alarm system is triggered when a button is pressed. Implement custom event classes and listeners to handle the alarm eventCreate a JFrame.

- Add a JButton labeled "Trigger Alarm".

- Define a custom event class (e.g., AlarmEvent) and a corresponding listener interface (e.g., AlarmListener).
  - Implement the AlarmListener interface in a class responsible for handling the alarm event.
  - Trigger the custom event when the "Trigger Alarm" button is pressed.
  - Display a message or perform an action when the alarm event is triggered.
15. Develop a Java application to perform CRUD operations on a student database using JDBC.
- Create a database schema for a student table with fields like student\_id, name, age, and grade.
  - Establish a JDBC connection to the database.
  - Write SQL queries to create the student table, insert sample data, update records, and delete records.
  - Implement exception handling to manage SQL exceptions.
  - Execute the Java program to demonstrate CRUD operations.
16. Create a Java program to demonstrate transaction management and rollbacks using JDBC.
- Establish a connection to a database that supports transactions.
  - Write Java code to perform multiple SQL operations within a transaction, such as transferring funds between bank accounts.
  - Implement commit and rollback operations based on specific conditions (e.g., if a transaction fails).
  - Use SQL exceptions to handle errors and ensure data integrity.
  - Execute the program and observe the effect of commit and rollback operations on the database.
17. Create a database schema named "University" with tables for storing student records.
- d. Create a stored procedure named "getStudentById" that accepts a student ID as input and returns the corresponding student details.
  - e. Populate the student table with sample data.
  - f. Establish a JDBC connection to the "University" database.
  - g. Write a Java method to invoke the "getStudentById" stored procedure using CallableStatement.
  - h. Prompt the user to input a student ID.
  - i. Pass the input student ID to the CallableStatement as a parameter.
  - j. Execute the CallableStatement to retrieve the student details.
  - k. Display the retrieved student details (e.g., ID, name, age, etc.) to the user.
18. Develop a servlet that handles form submission from a web page. The servlet should extract form parameters (such as name, email, etc.), process them, and display the submitted data back to the user.
- Create a servlet class that extends HttpServlet.
  - Implement the necessary methods (e.g., doGet or doPost) to handle HTTP requests.
  - Read form parameters using the request object.
  - Process the form data (e.g., validate inputs, perform calculations).
  - Generate an HTML response to display the submitted data back to the user.
19. Develop a web application that includes user authentication using servlets and JavaServer Pages (JSP). Users should be able to log in with a username and password, and upon successful authentication, they should be redirected to a welcome page.
- Create a servlet to handle user authentication.

- Implement a login form using JSP.
- Use session management to keep track of authenticated users.
- Validate user credentials against a predefined set (e.g., in-memory storage or database).
- Upon successful authentication, redirect the user to a welcome page using JSP.

**20.** Create a dynamic web application for performing CRUD (Create, Read, Update, Delete) operations using servlets and JSP. The application should allow users to interact with a database to manipulate data records.

- Design a database schema for storing data records (e.g., user information, product details).
- Implement servlets to handle CRUD operations (e.g., adding new records, retrieving records, updating records, deleting records).
- Develop JSP pages to interact with users (e.g., display data, input forms for adding/updating records).
- Use JDBC (Java Database Connectivity) to connect to the database and perform database operations.
- Implement error handling and validation for user inputs.

**21.** Develop a simple Java application to demonstrate the usage of Spring IOC container and Dependency Injection (DI) features.

- Configure a Spring IOC container using XML-based configuration.
- Define two POJO classes: Employee and Address, with appropriate attributes and methods.
- Implement Dependency Injection using Setter Injection to inject Address object into the Employee class.
- Write a Java program to retrieve an Employee object from the Spring IOC container and display its details along with the associated Address.
- Test the application to ensure proper DI and object creation.

**22.** Implement a simple Java application using Spring Framework that demonstrates Dependency Injection (DI) using constructor injection.

**Instructions:**

1. Create an interface MessageService with a method sendMessage().
2. Create a class EmailService implementing MessageService that prints "Email message sent".
3. Create a class SMSService implementing MessageService that prints "SMS message sent".
4. Create a class MessageProcessor that depends on MessageService for sending messages.
5. Configure Spring to inject EmailService into MessageProcessor using constructor injection.
6. Test the application by creating an instance of MessageProcessor in main method and invoking sendMessage().

**23.** Create a Java application using Hibernate to perform CRUD operations on a Student entity.

**Instructions:**

1. Define a Student entity with fields id, name, email, and age.
2. Configure Hibernate to connect to a database (MySQL or H2).
3. Implement methods to perform CRUD operations:
  - o createStudent(Student student)
  - o readStudent(int studentId)
  - o updateStudent(Student student)
  - o deleteStudent(int studentId)
4. Test the CRUD operations by creating instances of Student and invoking these methods.

24. Develop a Spring MVC application to handle a simple "Hello World" request-response.

**Instructions:**

1. Create a controller HelloController with a method sayHello() mapped to URL /hello.
2. Configure Spring MVC to handle this request and respond with a view displaying "Hello, World!".
3. Implement a simple JSP view hello.jsp that displays the greeting message.
4. Test the application by accessing <http://localhost:8080/hello> in web browser.

25. Create a Java application using Hibernate to perform a CRUD operation using Hibernate Query Language (HQL).

**Instructions:**

1. Define a Product entity with fields id, name, price, and quantity.
2. Implement methods to:
  - o Insert new Product objects into the database.
  - o Retrieve all Product objects using HQL.
  - o Update a Product object.
  - o Delete a Product object by ID.
3. Test the CRUD operations by creating instances of Product and invoking these methods.

Note:- Additional lab experiments will be also given time to time according to topic as per the syllabus

## Department of MCA

### SYLLABUS of FY MCA 2024-25 (Autonomous)

#### SEMESTER-1

<b>COURSE TITLE</b>		<b>DATA STRUCTURES &amp; ALGORITHMS</b>			<b>CREDITS</b>	<b>3</b>		
<b>COURSE CODE</b>		<b>MCA102</b>	<b>COURSE CATEGORY</b>	<b>CM</b>	<b>L-P-T</b>	<b>3-0-0</b>		
<b>Version</b>	<b>1.0</b>	<b>Approval Details</b>		<b>07-2024</b>				
<b>ASSESSMENT SCHEME</b>								
<b>First Periodical Assessment</b>	<b>Second Periodical Assessment</b>	<b>Seminar/Assignments/Project</b>		<b>Surprise Test / Quiz</b>	<b>Attendance</b>	<b>ESE</b>		
<b>10%</b>	<b>10%</b>	<b>10%</b>		<b>10%</b>	<b>10%</b>	<b>50%</b>		
<b>Course Description</b>	<p>This course provides an in-depth understanding of data structures and algorithms using JavaScript. Students will learn to apply various algorithmic approaches, including greedy algorithms, dynamic programming, and divide and conquer, to solve real-world problems. By the end of the course, students will be proficient in implementing and analyzing data structures and algorithms, ensuring efficient optimized performance in JavaScript.</p>							
<b>Course Objective</b>	<ul style="list-style-type: none"> <li>To understand the core principles of data structures and their importance in software development.</li> <li>To explore and implement various data structures (arrays, linked lists, stacks, queues, trees, graphs) using JavaScript.</li> <li>To develop an ability to solve problems using algorithmic approaches such as greedy algorithms, dynamic programming, and divide and conquer.</li> <li>To enhance proficiency in memory management and understand the performance trade-offs of different data structures.</li> <li>To prepare students to write efficient code for competitive programming and real-time applications in JavaScript.</li> </ul>							
<b>Course Outcome</b>	<p>After end of this course student will be able:</p> <ul style="list-style-type: none"> <li><b>CO1:</b> Understand and implement fundamental data structures such as arrays, linked lists, stacks, and queues in JavaScript.</li> <li><b>CO2:</b> Apply sorting and searching algorithms (e.g., binary search, quick sort) and evaluate their performance using time and space complexity.</li> <li><b>CO3:</b> Design and analyze advanced data structures like trees, graphs, heaps, and hash tables for solving complex problems.</li> <li><b>CO4:</b> Develop efficient algorithms using different paradigms (greedy algorithms, dynamic programming, etc.) for problem-solving in JavaScript.</li> <li><b>CO5:</b> Optimize solutions for real-world applications by understanding memory management and algorithmic efficiency in JavaScript.</li> </ul>							
<b>Prerequisites:</b>	<ul style="list-style-type: none"> <li>Basic knowledge of programming languages, preferably C++.</li> <li>Understanding of fundamental data structures like arrays, linked lists, stacks, and queues.</li> <li>Familiarity with recursion and basic sorting/searching algorithms.</li> </ul>							

<b>MODULE: Topic</b>	<b>Sessions</b>	<b>Weig htage (%)</b>	<b>CO Mapping</b>
<b>MODULE 1: Arrays</b>			
<b>Topic:</b> 1.1 Introduction & Definition of an Array, 1.2 Memory Allocation & Indexing in JavaScript, 1.3 Operations on 1-D & 2-D Arrays (Array of Arrays), 1.4 Dynamic Arrays using JavaScript (Array Object methods like push(), pop(), etc.), 1.5 Memory management and Garbage Collection in JavaScript	<b>8</b>	<b>10%</b>	<b>CO-1</b>
<b>MODULE 2: Linked List</b>			
<b>Topic:</b> 2.1 Introduction & Definition of a Linked List, 2.2 Memory Allocation in JavaScript, 2.3 Types of Linked Lists (Singly, Doubly, Circular), 2.4 Operations on Singly Linked Lists (Insert, Delete, Search, Traverse), 2.5 Circular & Doubly Linked Lists (with JavaScript implementation)	<b>6</b>	<b>15%</b>	<b>CO-2</b>
<b>MODULE 3: Stacks and Queues</b>			
<b>Topic:</b> 3.1 Introduction and Definition of a Stack, 3.2 Implementation of Stacks using Arrays and Linked Lists (JavaScript code), 3.3 Applications of Stacks (Expression Conversion, String Reversal using JavaScript), 3.4 Introduction and Definition of a Queue, 3.5 Implementation of Queues using Arrays and Linked Lists (JavaScript code), 3.6 Advanced Queues (Priority Queue, Deque), 3.7 Stack and Queue implementations using JavaScript (Classes, Functions)	<b>6</b>	<b>15%</b>	<b>CO-3</b>
<b>MODULE 4: Trees and Graphs</b>			
<b>Topic:</b> 4.1 Tree Definition and Representation, 4.2 Binary Search Tree (BST) and its operations (Insertion, Deletion, Search), 4.3 Tree Traversal (Inorder, Preorder, Postorder), 4.4 AVL Tree and its Rotations (with code examples in JavaScript), 4.5 Directed and Undirected Graphs, 4.6 Graph Representations (Adjacency Matrix, List), 4.7 Graph Traversals (BFS, DFS with JavaScript), 4.8 Advanced Trees (Red-Black Trees, B-Trees), 4.9 Graph Algorithms (Dijkstra, Prim, Kruskal in JavaScript)	<b>10</b>	<b>25%</b>	<b>CO-3 and CO-4</b>
<b>MODULE 5: Searching and Sorting</b>			
<b>Topic:</b> 5.1 Linear Search (with examples in JavaScript), 5.2 Binary Search (with examples in JavaScript), 5.3 Interpolation Search (JavaScript implementation), 5.4 Merge Sort, 5.5 Quick Sort, 5.6 Bubble Sort (All in JavaScript), 5.7 Heaps (Min and Max), 5.8 Hash Tables (Implementation of Hash Maps in JavaScript using Objects and Map), 5.9 Hash Functions (Designing custom hash functions in JavaScript)	<b>10</b>	<b>25%</b>	<b>CO-4 and CO-5</b>
<b>MODULE 6: Advanced Algorithms and Optimization</b>			
<b>Topic:</b> 6.1 Greedy Algorithms, 6.2 Dynamic Programming (JavaScript-based solutions), 6.3 Divide and Conquer (JavaScript examples), 6.4 NP-Complete Problems, 6.5 Approximation Algorithms, 6.6 Optimization Techniques for Competitive Programming (focusing on JavaScript)	<b>5</b>	<b>10%</b>	<b>CO-5</b>

TEXT BOOKS	
1.	"Data Structures and Algorithms with JavaScript" by Michael McMillan <a href="https://www.amazon.com/Data-Structures-Algorithms-Michael-McMillan/dp/1449364934">https://www.amazon.com/Data-Structures-Algorithms-Michael-McMillan/dp/1449364934</a>
2	"JavaScript: The Good Parts" by Douglas Crockford <a href="https://www.amazon.com/JavaScript-Good-Parts-Douglas-Crockford/dp/0596517742">https://www.amazon.com/JavaScript-Good-Parts-Douglas-Crockford/dp/0596517742</a>
REFERENCE BOOKS	
1.	"Eloquent JavaScript" by Marijn Haverbeke <a href="https://eloquentjavascript.net">https://eloquentjavascript.net</a>
2.	"Introduction to Algorithms" by Cormen, Leiserson, Rivest, and Stein <a href="https://www.amazon.com/Introduction-Algorithms-3rd-MIT-Press/dp/0262033844">https://www.amazon.com/Introduction-Algorithms-3rd-MIT-Press/dp/0262033844</a>
E BOOKS	
1.	"JavaScript Data Structures and Algorithms" by Sammie Bae <a href="https://opendatastructures.org/ods-cpp.pdf">https://opendatastructures.org/ods-cpp.pdf</a>
2	"Open Data Structures (in pseudocode)" by Pat Morin: <a href="https://opendatastructures.org/ods-python.pdf">https://opendatastructures.org/ods-python.pdf</a>
MOOC	
1.	"Algorithms and Data Structures" on edX <a href="https://www.edx.org/course/algorithm-and-data-structures">https://www.edx.org/course/algorithm-and-data-structures</a>
2.	NPTEL Data Structures and Algorithms <a href="https://archive.nptel.ac.in/courses/106/102/106102064/">https://archive.nptel.ac.in/courses/106/102/106102064/</a>



2.1 Requirements elicitation and analysis 2.2 Requirement Engineering 2.3 Types of Requirements: Functional and Nonfunctional 2.4 Four Phases of Requirement Engineering 2.5 Software requirement Specification (SRS) Structure and contents of SRS 2.6 IEEE standard format SRS	9	20%	CO-2
<b>MODULE 3:</b> Use-case Driven Object-Oriented Analysis			
3.1 Introduction to oops concepts 3.2 Class and object - Abstraction and encapsulation -Method and messages - Interface, Inheritance and polymorphism Use case modeling 3.3 Object-oriented design principles 3.4 Structural Diagram - Class Diagram, Associations and links, Aggregation, Composition and containment- Inheritance, Sub Types and IS-A hierarchy 3.5 Behavioral Diagram - Use case Diagram, Sequence diagram 3.6 State modeling diagram - Activity Diagram	12	30%	CO-2 & CO-3
<b>MODULE 4:</b> SOFTWARE DESIGN Pattern			
4.1 Software Design process 4.2 Design concepts Coupling & Cohesion 4.3 Functional independence 4.4 Design patterns 4.5 Creational Patterns (Factory, Singleton) 4.6 Structural Patterns (Adapter, Decorator) 4.7 Behavioral Patterns (Observer, Strategy)	9	20%	CO-3, CO-4
<b>MODULE 5:</b> User Interface UI/UX Design			
5.1 Elements of good design 5.2 Eight golden rules for design 5.3 User interface design-Case Study UI/U X 5.4 Interactive UI design 5.5 Introduction to UI and UX- Principles of User Interface Design 5.6 Interaction Design 5.7 Usability and User Experience 5.8 Wireframing and Prototyping - Evaluation and Testing of UI/UX	6	10%	CO-4
<b>MODULE 6:</b> Agile Methodology			
6.1 Introduction to Agile Methodologies- Agile Manifesto and Principles 6.2 Scrum Framework - Roles: Product Owner, Scrum Master 6.3 Development Team 6.4 Agile Planning: Release and Sprint Planning 6.5 Agile Metrics: Burndown Charts, Velocity 6.6 Agile Development Practices: TDD 6.7 Continuous Integration	4	10%	CO-5
<b>TEXT BOOKS</b>			
1.	"Object-Oriented Software Engineering: An Agile Unified Methodology" by David Kung		
2.	"Software Engineering: A Practitioner's Approach" by Roger S. Pressman		
3.	"Object-Oriented Analysis and Design with Applications" by Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen, and Kelli A. Houston		
<b>REFERENCE BOOKS</b>			
1.	"UML Distilled: A Brief Guide to the Standard Object Modeling Language" by Martin Fowler		
2.	"Clean Code: A Handbook of Agile Software Craftsmanship" by Robert C. Martin		
3.	"Design Patterns: Elements of Reusable Object-Oriented Software" by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides		

<b>E BOOKS</b>	
1.	<a href="https://www.geeksforgeeks.org/software-design-patterns/">https://www.geeksforgeeks.org/software-design-patterns/</a>
2	Design Patterns: Elements of Reusable Object-Oriented Software by Erich Gamma et al.: [ <a href="https://example.com/design-patterns">https://example.com/design-patterns</a> ]
3.	"The UX Book: Process and Guidelines for Ensuring a Quality User Experience" by Rex Hartson and Pardha Pyla (E-book)
<b>MOOC</b>	
1.	Coursera - Object-Oriented Software Engineering: <a href="https://example.com/coursera-oose">https://example.com/coursera-oose</a>
2.	edX - Software Engineering Essentials: <a href="https://example.com/edx-soft-eng">https://example.com/edx-soft-eng</a>
3.	"Introduction to User Experience Design" by Georgia Institute of Technology on Coursera
4.	"Software Development Processes and Methodologies" by the University of Minnesota on Coursera

## Department of MCA

### SYLLABUS of FY MCA 2024-25 (Autonomous)

#### SEMESTER-1

<b>COURSE TITLE</b>		<b>SUBJECT TITLE/NAME</b> Operating System & Shell Programming			<b>CREDITS</b>	<b>3</b>
<b>COURSE CODE</b>		MCA104	<b>COURSE CATEGORY</b>	CM	L-P-T	3-0-0
Version	1.0	Approval Details		01-07-2024		
<b>ASSESSMENT SCHEME</b>						
<b>First Periodical Assessment</b>	<b>Second Periodical Assessment</b>	<b>Seminar/Assignments/Project</b>	<b>Surprise Test / Quiz / Activity Based</b>	<b>Attendance</b>	<b>ESE</b>	
<b>10%</b>	<b>10%</b>	<b>10%</b>	<b>10%</b>	<b>10%</b>	<b>10%</b>	<b>50%</b>
<b>Course Description</b>	This course will introduce you to modern operating systems. We will focus on UNIX-based operating systems, though we will also learn about alternative operating systems, including Windows. The course will begin with an overview of the structure of modern operating systems. Over the course of the subsequent units, we will discuss the history of modern computers, analyze in detail each of the major components of an operating system (from processes to threads), and explore more advanced topics in the field, including memory management and file input/output. The class will conclude with a discussion of various system-related security issues.					
<b>Course Objective</b>	1. To learn the fundamentals of Operating Systems and handle processes and threads and their communication 2. To learn the mechanisms involved in memory management in contemporary OS 3. To know the functionality of Multiprocessor OS and Mobile OS. 4. To gain knowledge of distributed operating system concepts. 5. To learn about the Basics of Linux. 6. To learn programmatically to implement Linux OS mechanisms.					
<b>Course Outcome</b>	CO1: <b>Understand</b> structure of OS, process management and synchronization. CO2: <b>Understand</b> multicore and multiprocessing OS. CO3: <b>Analyze</b> Real-time and embedded OS CO4: <b>Apply</b> Windows and Linux OS fundamentals and administration CO5: <b>Apply</b> to solve shell scripting problems					
<b>Prerequisites:</b> Basics of Operating system						
<b>Module, Topic</b>						<b>Sessions</b>
						<b>Weightage (%)</b>
						<b>Co-Mapping</b>
<b>MODULE 1: Overview of OS</b>						
1.1 Overview of operating systems, 1.2 Functionalities and Characteristics of OS , 1.3 Hardware concepts related to OS ,1.4 CPU states, I/O channels, 1.5 Memory Management, Memory Management Techniques, 1.6 Contiguous & Non-Contiguous					7	15%
						<b>CO-1</b>



allocation, Logical & Physical Memory - Conversion of Logical to Physical address, 1.7 Paging, 1.8 Demand Paging Page Replacement Concept, 1.9 Segmentation - Segment with paging 1.10 Virtual Memory Concept, Thrashing			
<b>MODULE 2: Process Management</b>			
2.1 Process Management and Synchronization, 2.2 PCB, 2.3 Job and processor scheduling. Scheduling Concept, 2.4 Process hierarchies, 2.5 Problems of concurrent processes, 2.6 Critical sections, Mutual exclusion Synchronization, 2.7 Deadlock, 2.8 Device and File Management, 2.9 Overview Techniques, File Systems	8	17%	CO-2
<b>MODULE 3: Processor Management</b>			
3. Multiprocessor and Multicore Operating Systems, 3.1 Introduction, Advantages and Disadvantages, Multicore System Vs. Multiprocessor System, 3.2 Types of Multiprocessors , Symmetric Multiprocessors, Asymmetric Multiprocessors 3.3 Basic Multicore Concepts: Memory Sharing Styles, Uniform Memory Access (UMA), Non-Uniform Memory Access (NUMA), No Remote Memory Access, (NORMA),3.4 Cache Coherence, Inter-Process and intercore Communication: Shared Memory, Message Passing, 3.5 Mobile Operating Systems, Concept Need and Features, Types of Mobile OS Overview of Android OS, Applications of Mobile OS 3.6 Distributed Operating Systems, Concept Need and Features, Examples of Distributed OS with brief introduction, Applications of Distributed OS	8	17%	CO2, CO-3
<b>MODULE 4: Real Time OS</b>			
4. Real Time OS,4.1 Introduction and use of RTOS, 4.2 Components of RTOS, Types of RTOS, Features of RTOS, Factors for selecting in RTOS, 4.3 Applications of RTOS, Disadvantages of RTOS, 4.4 Embedded OS, 4.5 Concept Need and Features of embedded OS, 4.6 Examples of embedded OS with brief introduction, 4.7 Applications of embedded OS	4	10%	CO-3
<b>MODULE 5: Types of OS</b>			
5.Windows OS and Windows Server, Architecture,5.1 Windows OS , Introduction, Windows OS Installation, Process Management, Control Panel Overview, Users, Security and Privacy Settings, Identify Accessibility Settings 5.2 Service Management, Syncing Devices and File Sharing 5.3 Windows Utilities (Accessories, Disk, Management, Resource Monitor, Backup, and Recovery), Basic Troubleshooting (Networking, Security, Device Driver). Introduction to Ubuntu, Introduction, Overview of Kernel,5.3 Installation of Ubuntu File system, Basic Commands of Linux, Managing Processes in Linux, 5.4 Installing and deleting software packages, User Management, File and Device Management, Backup and recovery,5.5 Introduction to Graphical Environment (GNOME), Ubuntu Utilities, (Virtual Box,	12	25%	CO-4



Evolution, Gimp, Bleach Bit, Unity Tweak Tool etc.), SAMBA Overview			
<b>Module 6: Shell Scripting</b>			
6. Linux Shell Scripting,6.1 Introduction,6.2 Variables, Flow Controls, Loops,6.3 Functions, 6.4 Lists,6.5 Manipulating Strings 6.6 Reading and Writing Files,6.7 Positional Parameters 6.8 Case statement,6.9 Real time scripts for different system administration activities	6	16%	CO-5
<b>TEXTBOOKS</b>			
1 Silberschatz, A., Galvin, P.B. and Gagne, G., Operating System Concepts (10 ed.), John Wiley, 2018. ISBN 978-1-119-32091-3 2 Stallings William, Operating Systems Internals and Design Principles (9 ed.), Prentice Hall, 2021. ISBN 978-0134670959 3 Abraham Silberschatz, Peter Baer Galvin, Greg Gagne (2006), Operating System Principles, 7th edition, Wiley India Private Limited, New Delhi.			
<b>REFERENCE BOOKS</b>			
1 Andrew S Tanenbaum and Herbert Bos, Modern Operating Systems (1 ed.), Pearson, 2021. ISBN 9789332575776. 2 Stallings (2006), Operating Systems, Internals and Design Principles, 5th edition, Pearson Education, India. 3 Andrew S. Tanenbaum (2007), Modern Operating Systems, 2nd edition, Prentice Hall of India, 4 "Operating Systems: Internals and Design Principles" by William Stallings. 5 P.C.P. Bhatt: Introduction to Operating Systems Concepts and Practice, 3rd Edition, PHI, 2010			
<b>E BOOKS</b>			
1 Introduction to operating system <a href="https://nptel.ac.in/courses/106/106/106106144/">https://nptel.ac.in/courses/106/106/106106144/</a> 2 <a href="https://www.freebookcentre.net/ComputerScience-Books-Download/Operating-System-Notes.html">https://www.freebookcentre.net/ComputerScience-Books-Download/Operating-System-Notes.html</a> 3 <a href="https://www.freebookcentre.net/ComputerScience-Books-Download/Introduction-to-Operating-Systems-Lectures.html">https://www.freebookcentre.net/ComputerScience-Books-Download/Introduction-to-Operating-Systems-Lectures.html</a>			
<b>MOOC/Coursera</b>			
1 <a href="https://www.coursera.org/learn/os-power-user">https://www.coursera.org/learn/os-power-user</a> 2 <a href="https://www.coursera.org/specializations/codio-introduction-operating-systems">https://www.coursera.org/specializations/codio-introduction-operating-systems</a> 3 <a href="https://www.coursera.org/learn/comparch">https://www.coursera.org/learn/comparch</a>			

COURSE TITLE		MATHEMATICS AND BUSINESS STATISTICS			CREDITS	2
COURSE CODE		MCA105	COURSE CATEGORY	CM	L P T	2-0-0
Version	1.0	Approval Details		01-07-2024		

ASSESSMENT SCHEME					
First Periodical Assessment	Second Periodical Assessment	Seminar/Assignments/Project	Surprise Test / Quiz	Attendance	ESE
10%	10%	10%	10 %	10%	50%
Course Description	This course provides an integrated introduction to key mathematical and statistical concepts essential for data analysis and problem-solving in the field of computer applications. Designed for MCA students, it combines foundational mathematical techniques with statistical methods commonly used in business and technology contexts. Students will develop skills to analyze data, apply mathematical models, and use statistical tools for decision-making and forecasting.				
Course Objective	<ol style="list-style-type: none"> <li>Develop a strong understanding of fundamental mathematical concepts and their applications.</li> <li>Gain proficiency in descriptive and inferential statistics relevant to business and technology.</li> <li>Apply mathematical and statistical methods to real-world problems in computer science and business.</li> </ol>				
Course Outcome	<ul style="list-style-type: none"> <li><b>CO1:</b> Apply fundamental mathematical techniques such as algebra and functions to solve complex problems related to computer applications and data analysis.</li> <li><b>CO2:</b> Demonstrate proficiency in descriptive statistics by effectively summarizing and visualizing data using appropriate statistical methods and tools.</li> <li><b>CO3:</b> Utilize probability theory and distributions to model and analyze random processes, assess risks, and make informed decisions based on probabilistic data.</li> <li><b>CO4:</b> Perform inferential statistical analyses, including hypothesis testing and confidence interval estimation, to draw valid conclusions and make data-driven decisions.</li> <li><b>CO5:</b> Implement regression analysis to explore and quantify relationships between variables, applying these techniques in practical business scenarios to support decision-making and forecasting.</li> </ul>				
Prerequisites:	Fundamental knowledge of algebra, basic statistics, introductory probability, and basic calculus.				
<b>MODULE 1: Fundamentals of Mathematics</b>					
1.1 Algebraic expressions 1.2 solving linear equations, matrix operations (addition, subtraction, multiplication, inversion), determinants, 1.3 systems of linear equations.				6	20% CO-1
1.4 Types of functions (linear, quadratic, exponential, logarithmic), 1.5 graphical representation and interpretation, 1.6 transformations of functions, optimization problems.					

<b>MODULE 2: Descriptive Statistics</b>			
2.1 Methods of data collection (surveys, experiments, observational studies) 2.2 types of data (qualitative, quantitative) 2.3 data organization (frequency tables, cross-tabulations). 2.4 Calculating and interpreting mean, median, mode, variance, standard deviation, range, Coefficient of Variance 2.5 Constructing and interpreting histograms, bar charts, pie charts, box plots, scatter plots.	6	20%	CO-2
<b>MODULE 3: Probability and Distributions</b>			
3.1 Basic probability rules (addition and multiplication rules), conditional probability, Bayes' theorem 3.2 probability of combined events (independent and dependent events). 3.3 Discrete distributions (Binomial distribution), continuous distributions (Normal distribution) 3.4 properties and applications of these distributions, Skewness and Kurtosis. 3.5 Definition of random variables, expected value, variance, covariance, probability mass and density functions.	6	20%	CO-3
<b>MODULE 4: Inferential Statistics</b>			
4.1 Formulating null and alternative hypotheses 4.2 Type I and Type II errors, significance levels, p-values 4.3 hypothesis tests for means (z-test, t-test), and proportions. 4.4 Constructing confidence intervals for population means and proportions 4.5 interpretation of intervals, margin of error.	6	20%	CO-4
<b>MODULE 5: Regression Analysis</b>			
5.1 Model formulation, least squares estimation, interpretation of regression coefficient 5.2 goodness-of-fit measures (R-squared, adjusted R-squared). 5.3 Using regression analysis for forecasting, trend analysis, and decision-making 5.4 practical case studies and examples.	6	20%	CO-5
<b>TEXT &amp; REFERENCE BOOKS</b>			
1. "Discrete Mathematics and Its Applications" by Kenneth H. Rosen			
2. "Business Statistics: A First Course" by David M. Levine, Kathryn A. Szabat, and David F. Stephan			
3. "Calculus: Early Transcendentals" by James Stewart			
<b>E BOOKS</b>			
1. "Discrete Mathematics and Its Applications" Kenneth H. Rosen			
2. "Business Statistics: A First Course" David M. Levine, Kathryn A. Szabat, David F. Stephan			
<b>MOOC</b>			
1. <b>Introduction to Statistics</b> Platform: Coursera			
2. <b>Mathematics for Data Science</b> Platform: edX			
3. <b>Data Science and Machine Learning Bootcamp with R</b> Udemy			

## Department of MCA

### SYLLABUS of FY MCA 2024-25 (Autonomous)

#### SEMESTER-1

<b>COURSE TITLE</b>		<b>Fundamentals of Cloud Computing</b>			<b>CREDITS</b>	<b>4</b>
<b>COURSE CODE</b>		MCA106	<b>COURSE CATEGORY</b>	ME	L-P-T	3-1-0
<b>Version</b>	<b>1.0</b>	<b>Approval Details</b>		<b>1-07-2024</b>	<b>LEARNING LEVEL</b>	
<b>ASSESSMENT SCHEME</b>						
First Periodical Assessment	Second Periodical Assessment	Seminar/Assignments/Project	Surprise Test / Quiz	Attendance	ESE	
10%	10%	10%	10%	10%	50%	
<b>Course Description</b>	This course provides an introduction to the principles and concepts of cloud computing, focusing on its role in modern IT infrastructure. It covers the architecture, deployment models, services, and security aspects of cloud computing, equipping students with the skills to design and deploy cloud-based solutions.					
<b>Course Objective</b>	<ol style="list-style-type: none"> <li>Understand the core concepts and architecture of cloud computing.</li> <li>Explore various cloud service models and deployment strategies.</li> <li>Learn about cloud storage, computing, and networking services.</li> <li>Address security, privacy, and compliance in cloud environments.</li> <li>Gain practical experience with leading cloud platforms.</li> </ol>					
<b>Course Outcome</b>	<ul style="list-style-type: none"> <li>CO1: <b>Understand</b> the foundational concepts and architecture of cloud computing.</li> <li>CO2: <b>Analyze</b> different cloud service and deployment models.</li> <li>CO3: <b>Utilize</b> cloud storage, computing, and networking services.</li> <li>CO4: <b>Identify</b> and implement security measures in cloud environments.</li> <li>CO5: <b>Develop</b> and deploy applications on cloud platforms.</li> </ul>					
<b>Prerequisites:</b> Basic knowledge of Networking						
<b>Module Name</b>					<b>Sessions</b>	<b>Weightage (%)</b>
<b>CO mapping</b>						
<b>MODULE 1: Introduction to Cloud Computing</b>						
1.1. Introduction to Cloud Computing ,Basic Networking Concepts, IPv4 & IPv6, IP Address Classes 1.2. Cloud Computing vs. Cluster Computing vs. Grid Computing 1.3. Characteristics, Pros and Cons of Cloud 1.4. Introduction to Dockers 1.5. Introduction to Container					10	10% <b>CO-1</b>
<b>MODULE 2: Cloud Models</b>						
2.1 Cloud Service Models - IAAS, PAAS, SAAS & difference 2.2 Cloud Deployment Models-Public, Private, Hybrid, Community 2.3. XAAS- Anything as a Service – Storage as a service, Network as a Service, Database as a Service etc 2.4. Cloud Storage Types : Block, File, Object Storage 2.5 Cloud Platforms					10	20% <b>CO-2</b>

<b>MODULE 3: Virtualization</b>			
3.1. Introduction to Virtualization concept & Hypervisors 3.2. Pros and Cons of Virtualization 3.3. Machine Image, Virtual Machine(VM). 3.4. Xen: Para virtualization, VMware: Full Virtualization 3.5. Microsoft Hyper-V 3.6. Open Source Virtualization Manager	10	20%	<b>CO-3</b>
<b>MODULE 4: Cloud Architecture &amp; Management</b>			
4.1. Introduction to Service Oriented Architecture , Web Services: SOAP and REST 4.2. Relating SOA and Cloud Computing. 4.3. Service Level Agreement (SLA), Billing, Pricing, and Support. 4.4. Cloud Computing Architecture. 4.5. Multi Cloud Environment 4.6. Edge Computing Concepts 4.7. Cloud Bursting	10	20%	<b>CO-4</b>
<b>MODULE 5: Cloud Storage, Computing, and Networking &amp; Emerging trends in cloud computing</b>			
5.1 Storage services, Computing services, Networking services, 5.2 Cloud service providers (AWS, Azure, Google Cloud), 5.3 Pricing and performance consideration 5.4 Omni Cloud. 5.5 Blockchain Technology. 5.6 Types of Blockchain technology. 5.7 Cloud AI	10	20%	<b>CO-3</b>
<b>MODULE 6 :Moving Applications to the Cloud</b>			
5.1. Cloud Migration Strategies and Process 5.2. Issues in Inter Cloud 5.3. Applications in the Clouds 5.4. Cloud Service Attributes 5.5. Data Migration in Cloud	10	10%	<b>CO-5</b>
<b>TEXT BOOKS</b>			
1. "Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl 2. "Architecting the Cloud" by Michael J. Kavis			
<b>REFERENCE BOOKS</b>			
1. "Mastering Cloud Computing" by Rajkumar Buyya 2. "Cloud Computing Bible" by Barrie Sosinsky			
<b>E BOOKS</b>			
1. "Cloud Computing Explained" by John Rhoton (computingclouds.files.wordpress.com/2012/05/cloud-computing-explained.pdf)			
2 "NIST Cloud Computing Standards Roadmap" (nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.500-291.pdf)			
<b>MOOC</b>			
1. "Cloud Computing Specialization" by the University of Illinois on Coursera (coursera.org/specializations/cloud-computing)			
2. "Introduction to Cloud Computing" by IBM on Coursera (coursera.org/learn/introduction-to-cloud)			



## **Department of MCA**

## **SYLLABUS of FY MCA 2024-25 (Autonomous)**

SEMESTER-1



2.1 Introduction to CSS3, 2.2 Architecture of CSS, 2.3 CSS Modules, 2.4 CSS Framework, Selectors and Pseudo Classes, 2.5 Fonts and Text Effects, Colors, Background Images, and Masks	5	5%	CO-2
<b>MODULE 3: JavaScript</b>			
3.1 Concept of script, Types of Scripts, 3.2 Introduction to JavaScript, Variables, identifiers constants in JavaScript and examples of each. 3.3 Operators in JavaScript's, various types of JavaScript operator, Examples on JavaScript operators, 3.4 Control and looping structure, examples on control and looping structures (if, if...else, for, while, do while, switch) 3.5 Concept of array, how to use it in JavaScript, types of an array, examples , Methods of an array, examples on it. 3.6 Event handling in JavaScript with examples, Math and date object and examples on it. 3.7 String object and examples on it, and some predefined functions, 3.8 DOM concept in JavaScript, DOM objects, 3.9 Validations in JavaScript, examples on it	15	25%	CO2, CO-3
<b>MODULE 4: Web Servers</b>			
4.1 Web server architecture, approaches , 4.2 Working with web browser, types of web servers, 4.3 Features of web servers, Benefits of web servers, 4.4 Uses of web servers, when to use web servers	5	5%	CO-5
<b>MODULE 5: Ajax</b>			
5.1 Introduction to AJAX: Exploring different web technologies, 5.2 Creating a simple AJAX application, 5.3 Interacting with the Web Server Using the XMLHttpRequest Object, Create an XMLHttpRequest Object, Interact with the Web Server. 5.4 Differentiating AJAX and Non-AJAX application.5.5 Working with PHP and AJAX: Introduction, 5.6 Process Client Requests, Accessing Files Using PHP, 5.7 Implementing Security and Accessibility in AJAX applications: 5.8 Introduction, Secure AJAX Applications, and Accessible Rich Internet Applications	15	30%	CO-4
<b>Module 6: PHP</b>			
6.1 Installation of Apache Tomcat (Xampp/Lampp/MySQL) 6.2 Installing and Configuring PHP, Introduction, PHP and the Web Server Architecture, PHP Capabilities, 6.3 PHP and HTTP Environment 6.4 Variables , Variables, Constants, Data Types, Operators, Working with Arrays , Decision Making, Flow Control and Loops , Introduction to Laravel , 6.5 Creating a Dynamic HTML Form with PHP, Database Connectivity with MySQL, Performing basic database operations (CRUD) , 6.6 Using GET, POST, REQUEST, SESSION, and COOKIE Variables	15	30%	CO-5
<b>TEXT BOOKS</b>			
1	Complete reference HTML, TMH 2. HTML5 & CSS3, Castro Elizabeth 7th Edition		
2	Beginning Node.js by Basarat Ali Syed 100 45		
3	Angular: Up and Running- Learning Angular, Step by Step by Shyam Seshadri		
4	Beginning PHP, Apache, MySQL web development		
<b>REFERENCE BOOKS</b>			
1	Introducing HTML5 - Bruce Lawson, Remy Sharp		
2	Node.js in Action, 2ed by Alex Young, Bradley Meck		
3	Mastering Node.js by Pasquali Sandro		
4	Angular Essentials by Kumar Dhananjay		
5	Complete Ref. PHP		
<b>E BOOKS</b>			



1.	<a href="https://nodejs.org/en/docs/guides/">https://nodejs.org/en/docs/guides/</a>
2	<a href="https://www.coursera.org/learn/web-development">https://www.coursera.org/learn/web-development</a>
<b>MOOC</b>	
1.	<a href="https://www.coursera.org/learn/angular">https://www.coursera.org/learn/angular</a>
2	<a href="https://www.coursera.org/specializations/web-applications">https://www.coursera.org/specializations/web-applications</a>
3	<a href="https://www.coursera.org/learn/database-applications-php">https://www.coursera.org/learn/database-applications-php</a>
4	<a href="https://www.coursera.org/learn/server-side-nodejs">https://www.coursera.org/learn/server-side-nodejs</a>

### List of Practicals (Tentative)

1. Using basic HTML elements headings, paragraphs, line break, colour, fonts, links, Images.
2. Creating Lists using HTML Tags.
3. Creating Tables using HTML Tags.
4. Creating Frames in HTML.
5. Creating Forms using HTML.
6. Designing of HTML form using CSS.
7. Using Functions in JavaScript.
8. Carryout Validation and event handling using JavaScript.
9. Create a basic calculator with JavaScript that performs addition, subtraction, multiplication, and division.
10. Designing website using basic elements of HTML, CSS and JavaScript.
11. Develop a webpage that changes its content or style based on user interactions (e.g., mouse clicks, keypresses, hover events) using JavaScript.
12. Build a basic to-do list where users can add, edit, and delete items using JavaScript and save the list to 'localStorage'.
13. Write a script that allows users to upload a file (like an image or PDF), handle file validation, and store it on the server.
14. An example of sending form data to a PHP processing page using AJAX without reloading the page. Create an AJAX request on a test page to a PHP page that returns a message, and display the response without refreshing the test page.
15. Create a simple HTML form (with fields like name, email, etc.) and process the input using PHP.
16. Build a basic user registration form with validation, and store the data in a MySQL database using PHP.
17. Create a login form and implement a session-based authentication system using PHP and MySQL.
18. Build a PHP application that allows users to Create, Read, Update, and Delete (CRUD) records from a MySQL database.
19. Implement client-side form validation using JavaScript to check fields such as name, email, and password strength.
20. Write a script to dynamically manipulate the Document Object Model (DOM), like changing text, styles, or content on button clicks.

<b>COURSE TITLE</b>		<b>INTRODUCTION TO DATA SCIENCE</b>			<b>CREDITS</b>	<b>4</b>
<b>COURSE CODE</b>		MCA106	<b>COURSE CATEGORY</b>	ME3	<b>L-P-T</b>	<b>3-1-0</b>
<b>Version</b>	<b>1.0</b>	<b>Approval Details</b>		01/07/2024		
<b>ASSESSMENT SCHEME</b>						
<b>First Periodical Assessment</b>	<b>Second Periodical Assessment</b>	<b>Seminar/Assignments/Project</b>	<b>Surprise Test / Quiz</b>	<b>Attendance</b>	<b>ESE</b>	
10%	10%	10%	10%	10%	50%	
<b>Course Description</b>	This course aims to provide MCA students with a fundamental understanding of Data Science, including data manipulation, statistical analysis, machine learning, and data visualization. The curriculum is designed to introduce students to the tools and techniques used in data science and their practical applications..					
<b>Course Objective</b>	<ul style="list-style-type: none"> <li>To introduce the basic concepts and techniques of data science.</li> <li>To develop skills in data manipulation and analysis.</li> <li>To provide an understanding of statistical methods for data analysis.</li> <li>To introduce machine learning algorithms and their applications.</li> <li>To develop skills in data visualization and communication of results.</li> </ul>					
<b>Course Outcome</b>	<p>By the end of the course, students will be able to:</p> <ul style="list-style-type: none"> <li>CO1: <b>Understand</b> the fundamental concepts of data science, its applications, and lifecycle.</li> <li>CO2: <b>Learn</b> to handle, clean, and preprocess data for analysis.</li> <li>CO3: <b>Apply</b> exploratory data analysis and statistical techniques to derive insights.</li> <li>CO4: <b>Understand</b> the basics of machine learning algorithms and implement simple models.</li> <li>CO5: <b>Analyze</b> with data visualization techniques to communicate results.</li> <li>CO6: <b>Evaluate</b> real-world data science case studies and apply learned techniques.</li> </ul>					
<b>MODULE</b>				<b>No. of lectures(%)</b>	<b>Weightage (%)</b>	<b>CO Mapped</b>
<b>MODULE 1: Introduction To Data Science and Python</b>						
1.1 Definition of Data Science, 1.2 Data Science vs. Data Analytics, 1.3 Applications of Data Science and Case Studies, 1.4 Data Science Lifecycle, 1.5 Data-driven decision making, 1.6 Introduction to Python for Data Science, 1.7 Libraries: Data Manipulation Libraries (Pandas, NumPy), Machine Learning Frameworks (scikit-learn, TensorFlow, Pytorch), 1.8 Working with Jupyter Notebooks,				10	15%	CO1, CO2
<b>MODULE 2 : Data Handling &amp; Preprocessing</b>						
2.1 Data Collection Methods, 2.2 Types of Data: Structured, Unstructured, Semi-structured, 2.3 Data Cleaning: Handling missing values, outliers, Data Wrangling 2.4 Data Transformation: Normalization, Standardization, 2.5 Feature Engineering: Encoding categorical variables, Feature scaling				10	15%	CO2, CO3

**MODULE 3: Exploratory Data Analysis (EDA)**

3.1 Understanding Data Distributions: Mean, Median, Mode, Variance, Standard deviation, 3.2 Visualizing Distributions: Histograms, Box plots, Scatter plots, 3.3 Identifying Patterns and Relationships in Data, 3.4 Correlation and Covariance, 3.5 Introduction to Statistical Inference, Descriptive and Inferential Statistics

12

20%

CO2,CO3

**MODULE 4: Introduction to Machine Learning**

4.1 Introduction to Machine Learning and its Types (Supervised, Unsupervised, Reinforcement Learning), 4.2 Supervised Learning: Regression (Linear, Logistic), Classification, 4.3 Unsupervised Learning: Clustering (K-means, Hierarchical), 4.4 Overfitting and Underfitting, 4.5 Model Evaluation Metrics (Accuracy, Precision, Recall, F1-score)

12

20%

CO4

**MODULE 5: Data Visualization**

5.1 Importance of Data Visualization (Seaborn, Matplotlib), 5.2 Types of Plots: Line plot, Bar plot, Pie chart, Scatter plot, 5.3 Visualization Tools: Matplotlib, Seaborn, 2, 5.4 Best Practices in Visualization, 5.5 Storytelling with Data

8

15%

CO5

**MODULE 6: Tools & Techniques in Data Science**

6.4 Implementing Data Science Pipelines, 6.5 Introduction to Big Data & Tools (Hadoop, Spark), 6.6 Ethical Issues in Data Science, Real world Data Science Projects and Applications in Industry

8

15%

CO6

**TEXT BOOKS**

- 1 Python for Data Analysis by Wes McKinney
- 2 Data Science from Scratch by Joel Grus
- 3 Introduction to Machine Learning with Python by Andreas C. Müller and Sarah Guido

**REFERENCE BOOKS**

- 1 Jure Leskovek, Anand Rajaraman and Jeffrey Ullman. Mining of Massive Datasets. v2.1, Cambridge University Press.

**E BOOKS**

- 1 "Data Science from Scratch: First Principles with Python" by Joel Grus
- 2 "Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking" by Foster Provost and Tom Fawcett

**MOOC**

- 1 Coursera: "Introduction to Data Science" by University of Washington
- 2 edX: "Data Science and Machine Learning Essentials" by Microsoft  
Coursera: "Data Science Specialization" by Johns Hopkins University  
edX: "Introduction to Data Science"

## Department of MCA

### SYLLABUS of FY MCA 2024-25 (Autonomous)

#### SEMESTER-1

COURSE TITLE		CAPSTONE PROJECT			CREDITS	3
COURSE CODE		MCA107	COURSE CATEGORY	PR OJ	L-P-T	0-2-0
Version	1.0	Approval Details		07-2024		
<b>ASSESSMENT SCHEME</b>						
Review 1		Review 2	Review 3	Final Review	Attendance	ESE
10%		10%	20%	40%	20%	--

### Capstone Project Guidelines

1. Students can work individually or in pairs (maximum of 2) for the Capstone Project. Projects must relate to both laboratory subjects being taught in the current semester. The student may take up the project individually or in group. If project is done in group, each student must be given a responsibility for distinct modules.
2. Few hours per week is dedicated to project work. During this time, students must report to their assigned guides for support and discuss project progress. Attendance is mandatory for all review sessions and weekly project hours; project diaries must be presented at each review. (check class time table for more details)
3. Students must submit a clear and self-descriptive project title by the end of the first week of the semester. Following guide allotment, a project synopsis must be submitted by the second week, including sections on existing systems, need for the project, scope of work, objectives, and user requirements.
4. There will be a total of 4 reviews throughout the semester. Specific deliverables are required at each stage, including design documents, implementation details, and testing strategies. Students are expected to create a PowerPoint presentation for each review, summarizing their progress.
5. Documentation must adhere to specified standards: use Times New Roman, size 12 pt, with 1.5line spacing and justified alignment. Margins should be set to 1" (left and right), 2" (top), and 1.5" (bottom). All documentation must be spiral-bound and submitted by the final deadline.
6. The final evaluation will consist of a project presentation (PowerPoint) and a viva voce conducted by faculty members. Students must submit a comprehensive project report, including all relevant diagrams and design documents, on the final submission date.
7. All project work must be original and must adhere to the ICEM's academic integrity policies. Plagiarism will not be tolerated and may result in academic penalties.
8. Students are encouraged to utilize resources such as textbooks, online articles, and tutorials to support their project development. A list of recommended resources should be submitted along with the final project report.

9. Feedback from guides must be incorporated into subsequent project phases. Students should schedule regular meetings (in Capstone Project slot) with their respective guides to ensure alignment and address any challenges faced during project development.
10. Any changes to the project scope or objectives must be discussed with and approved by the guide before implementation. Documentation of such changes should be maintained in the project diary.

All project-related communications, including submission links and updates, will be shared via email and designated online platforms. Students should regularly check these platforms for important announcements.**MOOC**

- |    |  |
|----|--|
| 1. | <b>Coursera - "Project Management Principles and Practices" by Google</b><br><a href="https://www.coursera.org/specializations/project-management">https://www.coursera.org/specializations/project-management</a> |
| 2. | <b>edX - "Capstone Project: Data Science for Everyone" by IBM</b><br><a href="https://www.edx.org/professional-certificate/ibm-data-science">https://www.edx.org/professional-certificate/ibm-data-science</a>     |

## Capstone Project Schedule

Sr.No	Reviews	Topic	Documents Required
1		Project Title Submission	<b>Project Title</b>
2	Review 1	Project Synopsis	<p><b>INTRODUCTION</b></p> <ul style="list-style-type: none"> <li>Existing System and Need for System</li> <li>Scope of Work</li> <li>Operating Environment - Hardware and Software Detail Description of Technology Used</li> </ul> <p><b>PROPOSED SYSTEM</b></p> <ul style="list-style-type: none"> <li>Proposed System</li> <li>Objectives of System</li> <li>User Requirements</li> </ul>
3.	Review 2	Design	<p><b>ANALYSIS &amp; DESIGN</b></p> <ul style="list-style-type: none"> <li>Module Hierarchy Diagram</li> <li>Use Case Diagrams</li> <li>Class Diagram</li> <li>E-R Diagram</li> <li>Activity Diagram</li> <li>Sequence Diagram</li> <li>Web Site Map Diagram (in case of Web Site)</li> </ul>
5	Review 3	Database Connectivity	<ul style="list-style-type: none"> <li>User Interface Design (Screens etc.)</li> <li>Database Table Structure</li> <li>Database Connectivity</li> <li>Form Validation, Reports</li> </ul>
6	Review 4	Final Submission	Final Project Execution, checking Complete Documentation of Project with all Diagrams and PPT, Viva and submission of Spiral Binding documentation.



## **Department of MCA**

## **SYLLABUS of FY MCA 2024-25 (Autonomous)**

SEMESTER-1

Course Title		Soft Skills & Business Communication			Credits	1					
Course Code		MCA108	Course Category	CM	L-P-T	1-0-0					
Version	1.0	Approval Details	07-2024								
Assessment Scheme											
Seminar		Surprise Test / Quiz		Internal Exam	Attendance	ESE					
20%		20%		20%	40%	--					
Course Description	This course focuses on developing essential soft skills and effective business communication techniques required for professional growth. It emphasizes verbal and non-verbal communication, presentation skills, team collaboration, and leadership, preparing students for the corporate environment.										
Course Objective	<ul style="list-style-type: none"> <li>To enhance students' communication skills in professional settings.</li> <li>To develop interpersonal and team-building skills.</li> <li>To prepare students for effective public speaking and presentations.</li> <li>To improve time management and problem-solving abilities.</li> <li>To build confidence and leadership qualities in a business context.</li> </ul>										
Course Outcome	<p>After end of this course student will be able:</p> <ul style="list-style-type: none"> <li><b>CO1:</b> Demonstrate effective verbal and written communication in a business setting.</li> <li><b>CO2:</b> Exhibit interpersonal skills and teamwork in professional interactions.</li> <li><b>CO3:</b> Apply presentation and public speaking skills in business scenarios.</li> <li><b>CO4:</b> Develop leadership and problem-solving skills.</li> <li><b>CO5:</b> Manage time and stress efficiently in a workplace environment.</li> </ul>										
Prerequisites:	<ul style="list-style-type: none"> <li>Basic knowledge of the English language.</li> <li>Familiarity with email and digital communication tools.</li> </ul>										
Module: Topic				Sessions	Weightage (%)	CO Mapping					
<b>Module 1: Introduction to Soft Skills</b>											
<b>Topic:</b> 1.1 Definition and importance of soft skills, 1.2 Overview of communication skills, 1.3 Importance of emotional intelligence, 1.4 Understanding self-awareness and self-regulation, 1.5 Soft skills in the workplace.				2	10%	CO-1					

<b>MODULE 2: Verbal and Non-Verbal Communication</b>			
<b>Topic:</b> 2.1 Verbal communication techniques, 2.2 Importance of tone, pitch, and clarity, 2.3 Non-verbal cues: body language, facial expressions, and gestures, 2.4 Barriers to effective communication, 2.5 Active listening skills and their significance, 2.6 Techniques for overcoming communication barriers.	<b>3</b>	<b>20%</b>	<b>CO-2</b>
<b>MODULE 3: Business Communication Tools</b>			
<b>Topic:</b> 3.1 Writing professional emails: structure and etiquette, 3.2 Creating effective business reports and proposals, 3.3 Conducting meetings: agenda setting and minutes writing, 3.4 Presentation tools and techniques (e.g., PowerPoint, Prezi), 3.5 Utilizing digital communication platforms (e.g., Slack, Zoom), 3.6 Visual communication: charts, graphs, and infographics.	<b>2</b>	<b>15%</b>	<b>CO-3</b>
<b>MODULE 4: Teamwork and Collaboration</b>			
<b>Topic:</b> 4.1 Dynamics of team collaboration and roles, 4.2 Building trust and rapport within a team, 4.3 Conflict resolution strategies: negotiation and mediation, 4.4 Leadership styles and their impact on teamwork, 4.5 Strategies for fostering inclusive team environments, 4.6 Tools for effective team collaboration (e.g., Trello, Asana).	<b>3</b>	<b>20%</b>	<b>CO-3 and CO-4</b>
<b>MODULE 5: Time and Stress Management</b>			
<b>Topic:</b> 5.1 Understanding the importance of time management, 5.2 Prioritization techniques: Eisenhower Matrix and ABCD method, 5.3 Goal setting and action planning, 5.4 Managing workplace stress: techniques and tools, 5.5 Work-life balance strategies, 5.6 The role of mindfulness in stress management.	<b>2</b>	<b>15%</b>	<b>CO-4 and CO-5</b>
<b>MODULE 6: Public Speaking and Presentation Skills</b>			
<b>Topic:</b> 6.1 Structuring an effective presentation: introduction, body, conclusion, 6.2 Techniques for engaging your audience, 6.3 Overcoming stage fear: tips and tricks, 6.4 Using visual aids and technology effectively, 6.5 Rehearsal techniques for effective delivery, 6.6 Handling Q&A sessions and audience feedback.	<b>3</b>	<b>20%</b>	<b>CO-5</b>
<b>TEXT BOOKS</b>			
1. <b>Soft Skills: Enhancing Employability by M.S. Rao, I.K. International Publishing House.</b>			
2. <b>Business Communication by Meenakshi Raman &amp; Sangeeta Sharma, Oxford University Press.</b>			
<b>REFERENCE BOOKS</b>			
1. <b>Developing Soft Skills by Dr. K. Alex, S. Chand Publishing.</b>			
2. <b>Effective Business Communication by Asha Kaul, Prentice Hall India.</b>			
<b>E BOOKS</b>			
1. <b>Soft Skills Development by Kul Bhushan</b> <a href="https://www.pdfdrive.com/soft-skills-development-e53627497.html">https://www.pdfdrive.com/soft-skills-development-e53627497.html</a>			
2. <b>Effective Communication Skills by MTD Training</b> <a href="https://www.pdfdrive.com/effective-communication-skills-e15874210.html">https://www.pdfdrive.com/effective-communication-skills-e15874210.html</a>			



1.	<b>Effective Communication in the Workplace</b> <a href="https://www.edx.org/course/effective-communication-in-the-globalized-workplace">https://www.edx.org/course/effective-communication-in-the-globalized-workplace</a>
2.	<b>NPTEL Data Structures and Algorithms</b> <a href="https://archive.nptel.ac.in/courses/106/102/106102064/">https://archive.nptel.ac.in/courses/106/102/106102064/</a>