ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

Model Syllabus for Computer Applications (Minor) in consonance with Curriculum framework w.e.f. AY 2025-26

COURSE STRUCTURE

Year	Semester	Course	Title of the Course	No. of Hrs /Week	No. of Credits
II	III	1	Computer Fundamentals and Office Automation	3	3
			Computer Fundamentals and Office Automation-Practical	2	1
	IV	2	Database Management System	3	3
			Database Management-Practical	2	1
III	V	3	Introduction to Web Programming	3	3
			Introduction to Web Programming-Practical	2	1
		4	Fundamentals of Python Programming	3	3
			Fundamentals of Python Programming-Practical	2	1
	VI	5	Cyber Security	3	3
			Cyber Security-Practical	2	1
		6	Data Visualization using Tableau & Power BI	3	3
			Data Visualization using Tableau & Power BI-Practical	2	1

SEMESTER-III

COURSE 1: COMPUTER FUNDAMENTALS AND OFFICE AUTOMATION

Theory **Credits: 3** **3 hrs/week**

Course Objectives:

1. **Understand foundational computing concepts** including number systems, evolution of computers, and architectural components.
2. **Explore basic computer organization and network fundamentals**, recognizing device functions, system types, and internet components.
3. **Demonstrate proficiency in word processing and presentation tools**, applying formatting techniques and design elements for professional outputs.
4. **Develop competency in spreadsheet operations**, employing formulas, charts, and data-handling techniques.
5. **Apply advanced data modeling and productivity features** to analyze and visualize data efficiently using modern tools.

Course Outcomes:

Learners will be able to:

1. **Convert between binary, decimal, octal, and hexadecimal systems**, and explain computer evolution and generations with examples.
2. Learners will demonstrate **basic blocks of a computer and fundamental networking knowledge.**
3. Create professional-level documents and **design visually appealing presentations** using word processing software and presentation software.
4. Manipulate data within spreadsheets, apply formulas, and **generate accurate summaries and visualizations.**
5. Apply data modelling techniques to **analyze, organize, and represent data effectively** in various scenarios.

Unit-I: Number Systems, Evolution , Block Diagram and Generations

Number Systems: Binary, Decimal, Octal, Hexadecimal; conversions between number systems.

Evolution of Computers: History from early mechanical devices to modern-day systems.

Block Diagram of a Computer: Input Unit, Central Processing Unit, Memory Unit, Output Unit.

Generations of Computers: First to Fifth Generation – Technologies, Characteristics, Examples.

Unit-II Basic Organization and Network Fundamentals

Computer Organization: Functional components: Input/Output devices, Storage types, Memory Hierarchy.

Types of Computers: Micro, Mini, Mainframe, and Supercomputers.

Networking Fundamentals: Definition, Need for Networks, **Key Components:** Nodes, Links, Protocols, Networking Devices. **Types of Computer Networks** : LAN, WAN, MAN.

Network Topologies: Bus. Ring, Star, Mesh..

Internet Basics: History, IP Address, URL, WWW, Web browsers, Search engines, E-mail, Internet Security.

Unit-III Word Processing and Presentations

Word Processing Basics: Definition, Using Microsoft Word / Google Docs. Templates for resumes, letters, reports. **Basic text editing and formatting -** Typing and editing text, Font styles, sizes, colors, and effects, Paragraph alignment, indentation, and spacing, Bullets, numbering, and text highlighting, Templates for resumes, letters and reports. **Working with Tables and Graphics -** Inserting and formatting tables, Adding images, shapes, icons, and SmartArt, Text wrapping and positioning graphics.

Document Layout and Design - Page setup, Headers, footers, and page numbering, Section breaks and columns, Applying themes and styles. **Advanced Features -** Spell check and grammar tools, Thesaurus, and Mail merge. **References and Citations** Footnotes, endnotes, and captions, Bibliography and citation tools, Table of contents and index creation.

Presentation Tools: Using PowerPoint/Google Slides – Creating, Saving and Opening presentations, Adding, deleting, and rearranging slides, Slide layouts and design themes, Using templates and master slides, Slide sorter and outline view, Applying transitions and Animations, Design and Layout.

Applications: Creating resumes, Reports, Brochures, and Presentations.

Unit-IV Spreadsheet Basics

Spreadsheet Concepts: Understanding rows, columns, cells in tools like MS Excel/Google Sheets, Workbook, Worksheet, **Cell referencing**- Relative, Absolute, Mixed.

Functions and Formulae: Mathematical, Statistical, Logical, Text, Date and Time, Financial.

Lookup and Reference : VLOOKUP, HLOOKUP, XLOOKUP, INDEX, MATCH

Visual representations: Creating a chart, common chart types, Column Chart, Bar Chart, Line Chart, Pie Chart, Scatter Chart, Histogram.

Data Handling: Sorting data, Filtering data, Grouping Data, **Conditional formatting:** Data Bars, Color Scales, Icon Sets, Custom Formulas.

Unit-V Data Modelling

Data Analysis Tools: Pivot Tables and Pivot Charts, Data Validation (Drop-downs, Input Messages, Error Alerts), **What-If Analysis**: Goal Seek, Scenario Manager, Data Tables

Charts and Dashboards: Creating Interactive Dashboards, Using slicers with Pivot Tables, Combo Charts and Sparklines.

Productivity Tips: Using Named Ranges, Freeze Panes, Split View.

Text Books:

1. **Thareja, R.** (Second Edition). *Fundamentals of Computers*. Oxford University Press.
2. **Rajaraman, V.** (n.d.). *Fundamentals of Computers*. PHI Learning.
3. **Norton, P.** (2017). *Introduction to Computers* (7th ed.). McGraw Hill Education.
4. **Nordell, R., Stewart, K., Easton, A., Graves, P. R., & Triad Interactive, Inc.** (2022). *Microsoft Office 365: In Practice* (1st ed.). New York: McGraw Hill Education.

References Books:

1. **Alexander, M., & Kusleika, R.** (2022). *Microsoft Excel 365 Bible* (2nd ed.). Wiley.
2. **Lowe, D.** (2021). *Networking All-in-One For Dummies* (8th ed.). Wiley.
3. **Microsoft Official Docs and Training**: https://learn.microsoft.com
4. **Google Workspace Learning Center**: https://support.google.com/a/users/

Activities:

Unit 1: Number Systems & Computer Evolution

Outcome: At the End of the Course, The Students will be able to **explain different number systems**, the historical evolution of computers, and identify key components in a block diagram.

Activity: Create a digital poster or infographic comparing number systems (binary, decimal, octal, hexadecimal) and illustrating the timeline of computer generations with key innovations.

Evaluation Method: Rubric-based assessment of the poster presentation on a 10-point scale focusing on:

- Accuracy of number system conversions
- Correct identification of block diagram components
- Visual organization and creativity

Unit 2: Computer Architecture & Networking Basics

Outcome: Learners will demonstrate **basic blocks of a computer and fundamental networking knowledge.**

Activity: Design a concept map showing the internal architecture of a computer and types of networks (LAN, WAN, MAN), including devices and topologies.

Evaluation Method: Checklist-based peer review and instructor validation:

- Completeness of the map
- Correctness of networking concepts
- Use of appropriate terminology
- Logical flow and structure of the map

Unit 3: Word Processing & Presentation Design

Outcome: Learners will create professional-level documents and **design visually appealing presentations** using word processing software and presentation software.

Activity: Prepare a formal report (e.g., project proposal) in a word processor and present it using a slide deck with transitions, embedded media, and design elements.

Evaluation Method: Performance-based evaluation using a 10-point scoring scale:

- Formatting and structure of the document
- Presentation aesthetics and clarity
- Communication skills during presentation

Unit 4: Spreadsheet Analysis & Visualization

Outcome: Learners will manipulate data within spreadsheets, apply formulas, and **generate accurate summaries and visualizations.**

Activity: Analyze a dataset (e.g., student scores or sales data) using spreadsheet software. Apply formulas (SUM, AVERAGE, IF, VLOOKUP) and create relevant charts.

Evaluation Method: Practical test with a rubric:

- Correct use of formulas
- Accuracy of data summaries

Unit 5: Data Analysis and Visualization:

Outcome: Learners will apply data modelling techniques to **analyze, organize, and represent data effectively** in various scenarios.

Activity: Prepare an interactive dashboard for a given data set using EXCEL.

Evaluation Method: Evaluation of the dashboard on a 10-point scoring scale:

- Presentation aesthetics and clarity
- Interactiveness
- Communication skills during presentation

SEMESTER-III

COURSE 1: COMPUTER FUNDAMENTALS AND OFFICE AUTOMATION

Practical **Credits: 1** **2 hrs/week**

List of Experiments:

1. Demonstration of Assembling and Dessembling of Computer Systems.
2. Identify and prepare notes on the type of Network topology of your institution.
3. Prepare your resume in Word by using the Resume template.
4. Using Word, write a letter to your higher official seeking 10-days leave.
5. Create a multi-page academic report and format it using headers and footers. The header will include the document title and author name, while the footer will contain page numbers and the date.
6. Prepare a formal invitation letter and use Mail Merge to personalize it for a list of recipients.
7. Prepare a report that includes: A table summarizing sales data, A graphic (image or chart) illustrating product performance with the proper formatting and alignment of both elements
8. Prepare a document and add Citations, Footnotes, and Bibliography in Word.
9. Create a PowerPoint Presentation on the Role of AI in Business Decision-Making.
10. Using a spreadsheet, prepare your class Time Table.
11. Using a Spreadsheet, calculate the Gross and Net salary of employees(Min 5) considering all the allowances.
12. Generate the class-wise and subject-wise results for a class of 20 students. Also generate the highest and lowest marks in each subject.
13. Using IF, AND, OR, and IFERROR to Automate Grade Evaluation.
    a. Create a table of student scores in different subjects.
    b. Use IF to assign grades (A/B/C/Fail).
    c. Use IFERROR to handle missing scores or invalid data.
14. Consider the problem of preparing a stationary order for the month of March. The item description, quantity and cost per item are available. The total cost per item is to be calculated and the final cost per item involves a sales tax of 2% over the total cost. The gross total and the net total are to be displayed.

Sl. No.	Description	Quantity	Cost Per Item
1	Notepad	202	2.85
2	Writing Pad	86	3.95
3	Ball point pen (Blue)	520	2.50
4	Cello-tape	75	2.95

5	A4 Refill pad	90	5.95
6	Pencils	603	0.50
7	Crayons	80	3.85
8	Stapler	30	9.95
9	Hole punch	25	14.95
10	Ring Binder	45	10.95

15. You are given the order details of a company in the below table.

Order Id	Product	Unit price	Quantity	Discount	Revenue	Tax (2% for each order)	Net Income
11250	A	8	10	0%	?	?	?
11251	B	20.8	1	0%	?	?	?
11252	C	7.7	16	25%	?	?	?
11253	D	15.6	50	0%	?	?	?
11254	E	39.4	15	25%	?	?	?
				Total	?		?

a. Calculate the revenue and tax on the revenue for each product.

b. Calculate the net income of each product.

c. Calculate the total revenue of all products.

d. Calculate the total net income of all products.

16. Create an Excel sheet with the following fields in the Sales table.

i)Month ii) Item iii) Quantity iv) Price v) Commission

Use Data Validation criteria for:

a) Quantity and Price should be whole numbers

b) Commission @3.5% of Price should be allowed only two decimals.

c) Price should accept 5000 and above values only.

17. Consider the problem of finding the total and average marks of five subject marks for five students. Calculate the Maximum mark, minimum mark, mean, median, Standard deviation and Variance for each subject.

Roll. No.	Name	Accounting	Income Tax	Business Law	Total	Average
100	Ramesh	85	75	60	?	?
101	Mahesh	100	78	85	?	?
102	Suresh	65	72	70	?	?
103	Ravi	90	80	85	?	?
104	Raju	80	76	90	?	?

18. The following are the details of Expenditure. Draw a Pie diagram with appropriate Formatting options, including Percentages and chart headings.

Product	Sales
Food	10000
Rent	5000
Clothing	1000
Fee	4000

19. The following are the marks obtained by the students of B. Com. In three subjects.

Roll. No.	Name	Accounts	IT	Economics
2001	Ramesh	65	85	75
2009	Mahesh	88	75	60
2004	Suresh	67	84	35
2002	Ravi	42	85	74
2007	Raju	88	89	90

1. Sort the above table on Roll. No.
2. Using Conditional formatting List out students who scored
    a. Less than 55 in Accounts
    b. More than 75 in IT
    c. Between 60 and 75 in Economics.

20. Prepare Pivot Table for the given data:

Department	Employee Name	Salary
HR	Ramesh	20,000
Finance	Mahesh	18,500
IT	Suresh	17,500
HR	Ravi	13,000
Finance	Raju	15,000
IT	Balu	10,000

21. Employee Database Search Using VLOOKUP, HLOOKUP, XLOOKUP, INDEX, and MATCH

a. Create a database of employees (Name, ID, Department, Salary).
b. Implement VLOOKUP to search by employee ID.
c. Use HLOOKUP to extract department heads by role.
d. Apply XLOOKUP for more flexible searches.
e. Use INDEX + MATCH as an alternative to VLOOKUP.

22. Sales Report Analysis Using Pivot Tables and Charts

a. Use a dataset of product sales (Product, Region, Date, Quantity, Revenue).
b. Create Pivot Tables to summarize data by region/product.
c. Insert Pivot Charts for visual analysis (e.g., bar, line).
d. Add slicers to make the dashboard interactive.

23. Designing a Data Entry Form with Drop-downs and Input Rules

e. Create a student registration form.
f. Add drop-down lists for course selection using Data Validation.

g. Add input messages to guide users.
h. Add error alerts for wrong entries.

24. Monthly Budget Planning using Goal Seek and Scenario Manager

i. Create a simple personal budget (income, expenses, savings).
j. Use Goal Seek to determine income needed to save a desired amount.
k. Use Scenario Manager to compare different budgeting scenarios (best/ worst/ realistic case).
l. Create a one-variable Data Table to analyze how different expenses affect savings.

25. Consider the monthly sales report

Month	**Region**	**Product**	**Units Sold**	**Unit Price (₹)**	**Total Sales (₹)**
Jan-25	North	Laptop	120	50,000	60,00,000
Jan-25	South	Tablet	80	30,000	24,00,000
Feb-25	North	Laptop	150	50,000	75,00,000
Feb-25	South	Tablet	90	30,000	27,00,000
Mar-25	North	Laptop	100	50,000	50,00,000
Mar-25	South	Tablet	110	30,000	33,00,000

1. Create PivotTables
2. Add Slicers
3. Create Combo Chart
4. Insert Sparklines
5. Assemble Dashboard

SEMESTER-IV

COURSE 2: DATABASE MANAGEMENT SYSTEMS

Theory **Credits: 3** **3 hrs/week**

Course Objectives:

1. Understand the fundamental concepts of data, databases, and the architecture of Database Management Systems (DBMS). Analyze and design database schemas using Entity-Relationship (E-R) and Extended E-R models.
2. Apply relational database principles, including normalization and integrity constraints, to ensure efficient schema design.
3. Develop SQL queries for data definition, manipulation, and control, including advanced operations like joins and nested queries.
4. Implement PL/SQL programming constructs and manage database transactions with a focus on ACID properties.

Course Outcomes:
The Learners will be able to:

1. Differentiate between file systems and DBMS, and explain the evolution, architecture, and components of modern database systems.
2. Design conceptual data models using E-R and Extended E-R diagrams, including specialization, generalization, and aggregation.
3. Normalize relational schemas up to BCNF and apply relational algebra operations to query and manipulate data.
4. Write efficient SQL queries using DDL, DML, DCL, and TCL commands, and utilize functions, joins, and views for data analysis.
5. Develop PL/SQL blocks with control structures, procedures, and triggers, and manage transactions ensuring atomicity, consistency, isolation, and durability.

Unit-I: Overview of Database Management System
Introduction: Data, Information, Database, Database Management System, Database System Applications, File Systems versus DBMS, Advantages of DBMS.

Evolution of Database Systems, Data Models, Data Abstraction, Database Architecture, Centralized and Client/Server Architectures for DBMSs, Database Users, Database Administrator, Components of DBMS, DBMS Vendors.

Unit-II: Database Design and the E-R Model
Overview of the Database Design Process, **The Entity-Relationship Model :** Entity Sets, Classification of Entity Sets, Attributes, Types of Attributes, Relationship, Relationship Set, Degree of a relationship set, Relationship Classification, Mapping cardinalities.

Extended Entity Relationship Model : Specialization, Generalization, Constraints, Attribute Inheritance and ISA relationship, Aggregation.

Case Study: Hospital Management System

Unit-III : Relational Database Design

Relational Model: Introduction to Relational Model, Concepts of Relation, Tuple, Attribute, Instance, Domain. Keys (Super key, Candidate Key, Primary Key, Foreign Key), Constraints (Domain constraints, Key constraints, Integrity constraints), selection, projection operations, Codd's rule set for relational databases.

Normalization: Purpose of Normalization/ Schema refinement, Functional dependencies, Normal Forms : 1NF, 2NF, 3NF and BCNF, Denormalization.

Case Study: College Student Enrollment System

Unit-IV : Introduction to SQL

Structured Query Language (SQL) : Overview of the SQL, Data types, Operators, Basic structure of SQL Query,

Commands: Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL), Transaction Control Language (TCL).

Functions: Numeric, String, Date Functions. Set operators, Aggregate functions, Nested queries, Joins, GROUP By Clause, ORDER By Clause, views.

Case Study : Retail Store Database Management (Design and query a relational database for a retail store that manages products, customers, orders, and employees. Apply SQL concepts to extract insights, maintain data integrity, and support decision-making.)

Unit-V Advanced SQL and Transaction Management

PL/SQL: Structure of PL/SQL block, Control Structures, Procedures, Functions, Exception handling, Cursors, Triggers.

Transactions : Transaction concept, Simple transaction model, State diagram of a transaction, ACID properties: Atomicity, Consistency, Isolation, Durability.

Textbooks:

1. Database System Concepts, Avi Silberschatz, Henry F. Korth,S. Sudarshan, Seventh Edition, McGraw-Hill
2. Database Management Systems by Raghu Ramakrishnan, McGrawhill

Reference Books:

1. Fundamentals of Database Systems, Elmasri Navathe Pearson Education
2. An Introduction to Database systems, C.J. Date, A.Kannan, S.Swami Nadhan, Pearson

Activities:

Outcome: Describe the fundamentals of data, database systems, and the differences between file-based and database approaches. Compare and classify various DBMS architectures, data models, and their components, including the three-schema architecture.

Activity: Create a comparative presentation or infographic illustrating:

- File-based vs. DBMS approaches
- Types of DBMS architectures (1-tier, 2-tier, 3-tier)
- Data models and the three-schema architecture

Evaluation Method: Rubric-based assessment of the presentation covering clarity, accuracy, and depth of comparison. Include a short quiz to test conceptual understanding.

Outcome: Design conceptual data models using Entity-Relationship and Enhanced ER diagrams, applying generalization, specialization, and constraints.

Activity: Model a university or hospital database using ER and Enhanced ER diagrams that shows:

- Entity sets, relationships
- Generalization/specialization
- Participation and cardinality constraints

Evaluation Method: Diagram submission with peer review and instructor feedback. Use a checklist to assess completeness, correctness, and notation usage.

Outcome: Apply relational model concepts, including CODD rules, relational algebra, relational calculus, and normalization techniques.

Activity: Normalize a given unstructured dataset up to 3NF. Then, write relational algebra expressions for sample queries.

Evaluation Method: Written assignment graded on:

- Correctness of normalization steps
- Accuracy of relational algebra expressions
- Short-answer questions on CODD rules and relational calculus

Outcome: Construct and execute SQL queries for data definition, manipulation, aggregation, joining, and subqueries, including views and set operations.

Activity: Implement a mini-project (e.g., Library or Inventory DB) using SQL. Include:

- Table creation (DDL)
- Data manipulation (DML)
- Aggregation, joins, subqueries, views, and set operations

Evaluation Method: Lab-based practical test with query execution and output validation. Include a viva to explain logic and optimization.

Outcome: Develop PL/SQL programs incorporating control structures, procedures and functions to manage database behaviour effectively.

Activity: Build a PL/SQL-based payroll or student grading system using:

- Procedures and functions
- Control structures (IF, LOOP)
- Triggers for automated updates

Evaluation Method: Code review and demonstration. Evaluate based on:

- Syntax correctness
- Logical flow

SEMESTER-IV

COURSE 2: DATABASE MANAGEMENT SYSTEMS

Practical **Credits: 1** **2 hrs/week**

Experiment 1 : Database: Inventory Management

Table 1: Products

Structure:

Column Name	Data Type	Constraints
product_id	INT	PRIMARY KEY
product_name	VARCHAR(50)	NOT NULL
price	DECIMAL(10,2)	CHECK(price > 0)
stock_qty	INT	CHECK(stock_qty >= 0)

Sample Data:

product_id	product_name	price	stock_qty
1	Pen	10.00	100
2	Notebook	50.00	200
3	Stapler	120.00	50
4	Marker	25.00	80
5	File Folder	60.00	150

Table 2: Suppliers

Structure:

Column Name	Data Type	Constraints
supplier_id	INT	PRIMARY KEY
supplier_name	VARCHAR(50)	NOT NULL
contact_no	VARCHAR(20)	UNIQUE

product_id	INT	FOREIGN KEY REFERENCES Products(product_id)

Sample Data:

supplier_id	supplier_name	contact_no	product_id
101	StationeryMart	9876543210	1
102	PaperWorld	9876500000	2
103	OfficeSupplies	9876512345	3
104	MarkerHub	9876522222	4
105	FileDepot	9876533333	5

Section A: DDL (Data Definition Language)

1. Create a database called InventoryDB.
2. Create a table Products and table Suppliers with the specified columns and constraints:

Section B: DML (Data Manipulation Language)

4. Insert at least 5 rows into the Products table.
5. Insert at least 5 rows into the Suppliers table.
6. Update the stock quantity of product ‘Pen’ to 120.
7. Delete a supplier with a specific supplier_id.
8. Write a query to rename ‘Notebook’ to ‘NoteBook A4’

Section C: DQL (SELECT Queries)

9. Display all records from the Products table.
10. Display only product_name and price of all products.
11. List all products that have a stock quantity less than 100.
12. Show all products between 20 and 100 price range.
13. Find all suppliers whose contact number starts with ‘98765’.
14. Find the average price of products.
15. Display the total number of products in the inventory.
16. Show the maximum and minimum stock quantities.

17. Count how many suppliers supply each product.
18. Show all products where price > 50 AND stock_qty > 100.
19. Show all products where price < 20 OR stock_qty < 80.
20. Display suppliers whose supplier_name contains the word ‘Mart’
21. List all suppliers along with the product they supply (use INNER JOIN).
22. Display suppliers whose name starts with ‘S’.
23. Find products whose name has exactly 5 characters
24. Find suppliers who supply products costing more than 100.

Experiment 2 : ONLINE BOOKSTORE DB

An online book store wants to implement a BOOKSTORE DB for managing their online transactions by using the following tables.

Authors Table

Column Name	Data Type	Constraints
author_id	INTEGER	PRIMARY KEY
first_name	VARCHAR	NOT NULL
last_name	VARCHAR	NOT NULL
nationality	VARCHAR	NULL allowed

Books Table

Column Name	Data Type	Constraints
book_id	INTEGER	PRIMARY KEY
Title	VARCHAR	NOT NULL
author_id	INTEGER	FOREIGN KEY REFERENCES Authors
publication_year	INTEGER	
Price	DECIMAL	

Customers Table

Column Name	Data Type	Constraints
customer_id	INTEGER	PRIMARY KEY
first_name	VARCHAR	NOT NULL
last_name	VARCHAR	NOT NULL
Email	VARCHAR	UNIQUE, NOT NULL
Address	VARCHAR	NOT NULL

Orders Table

Column Name	Data Type	Constraints
order_id	INTEGER	PRIMARY KEY
customer_id	INTEGER	FOREIGN KEY REFERENCES Customers
book_id	INTEGER	FOREIGN KEY REFERENCES Books
order_date	DATE	NOT NULL
quantity	INTEGER	NOT NULL

SAMPLE DATA SET for BOOKSTORE DB

Authors Table

author_id	first_name	last_name	nationality
1	Jane	Austen	British
2	George	Orwell	British
3	Gabriel	Garcia Marquez	Colombian
4	Toni	Morrison	American
5	Mark	Twain	American

6	Harper	Lee	American
7	Fyodor	Dostoevsky	Russian

Books Table

book_id	Title	author_id	publication_year	price
101	Pride and Prejudice	1	1813	12.99
102	1984	2	1949	9.50
103	One Hundred Years of Solitude	3	1967	15.00
104	Beloved	4	1987	11.25
105	Animal Farm	2	1945	8.75
106	Adventures of Huckleberry Finn	5	1884	10.50
107	To Kill a Mockingbird	6	1960	14.00

Customers Table

customer_id	first_name	last_name	Email	address
201	Alice	Smith	alice.s@example.com	12 Oak St, London
202	Bob	Johnson	bob.j@example.com	45 Pine Ave, Oxford
203	Charlie	Brown	charlie.b@example.com	78 Maple Rd, Bristol
204	Diana	Prince	diana.p@example.com	34 Queen St, York
205	Edward	Norton	edward.n@example.com	22 River Ln, Leeds
206	Fiona	Hall	fiona.h@example.com	56 Lake Dr, Bath
207	Greg	Miller	greg.m@example.com	89 Park Ave, Glasgow

Orders Table

order_id	customer_id	book_id	order_date	Quantity
301	201	101	2025-07-20	1
302	202	102	2025-07-21	2
303	201	105	2025-07-22	1
304	203	103	2025-07-23	1
305	204	106	2025-07-24	1
306	205	107	2025-07-25	3
307	206	104	2025-07-26	2

Section A: DDL (Schema Design & Constraints)

1. Write SQL statements to create all 4 tables (Authors, Books, Customers, Orders) with:

- Primary Keys
- Foreign Keys
- Appropriate data types
- NOT NULL constraints where necessary.

2. Alter the Books table to add a constraint that price must be greater than 0.
3. Add a new column phone_number to the Customers table (VARCHAR(15)) and ensure it is unique.
4. Drop the phone_number column from the Customers table.

Section B: DML (Data Manipulation)

5. Insert at least 7 records for each table (use sample dataset above).

6. Update the price of the book titled *Animal Farm* by increasing it by 10%.
7. Delete all orders made before 2025-07-21.
8. Change the nationality of Gabriel Garcia Marquez to “Latino-American”.

Section C: SELECT Queries (Data Querying)

9. List all books published between 1900 and 2000.
10. Find all customers whose email contains “example.com”.
11. Retrieve books whose price is between 10 and 15 and published before 1950.

12. Show authors who are either ‘British’ or ‘American’.
13. Find books that have a price less than 10 or are published after 1980.
14. Display all orders placed after 2025-07-22.
15. List all books written by author with author_id = 2.
16. Find customers whose last name starts with B.
17. Show all books with a price NOT between 9 and 13.
18. Display books whose publication_year is in (1813, 1945, 1987).
19. Find authors whose nationality is NOT ‘British’.
20. List customers whose address contains the word Park.
21. Show all books sorted by price in descending order.
22. List authors in alphabetical order by last_name.
23. Display orders sorted by order_date (latest first).

Use of Date Functions

24. Show all orders placed in July 2025.
25. Show all orders with an estimated delivery date (5 days after order date).
26. Show customers who placed an order on a weekend.
27. Calculate how many days have passed since the last order was placed.

Aggregate Functions (COUNT, SUM, AVG, MIN, MAX)

28. Count the total number of books in the database.
29. Find the average price of all books.
30. Show the highest-priced book.
31. Count how many orders each customer has placed.
32. Calculate the total sales (price × quantity) for each customer.

GROUP BY and HAVING

33. Count how many books are written by each author.
34. Group orders by customer_id and display total quantity ordered.
35. Show customers who have ordered more than 2 books in total (use HAVING).
36. Find the total number of books sold per author (GROUP BY author).

Experiment 3: EMPLOYEE DB

An enterprise wants to automate its employee management process by implementing an Employee Database.The goal is to replace manual record-keeping with a centralized system that stores employee, department, and project details. Use the following table structures and data set to implement Employee DB.

EmployeeDB – Table Structures

1. Departments Table

Column	Type	Constraints
dept_id	INT	PRIMARY KEY
dept_name	VARCHAR	UNIQUE, NOT NULL
location	VARCHAR	NOT NULL

2. Employees Table

Column	Type	Constraints
emp_id	INT	PRIMARY KEY
first_name	VARCHAR	NOT NULL
last_name	VARCHAR	NOT NULL
email	VARCHAR	UNIQUE, NOT NULL
phone	VARCHAR	CHECK (phone LIKE '--____')
hire_date	DATE	NOT NULL
job_title	VARCHAR	NOT NULL
salary	DECIMAL	CHECK (salary > 0)
dept_id	INT	FOREIGN KEY REFERENCES Departments(dept_id)
manager_id	INT	FOREIGN KEY REFERENCES Employees(emp_id) (self-referential)

3. Projects Table

Column	Type	Constraints
project_id	INT	PRIMARY KEY
project_name	VARCHAR	NOT NULL
start_date	DATE	NOT NULL
end_date	DATE	NULL

dept_id	INT	FOREIGN KEY REFERENCES Departments(dept_id)

4. Employee_Project Table (Many-to-Many)

Column	Type	Constraints
emp_id	INT	FOREIGN KEY REFERENCES Employees(emp_id), PRIMARY KEY(emp_id, project_id)
project_id	INT	FOREIGN KEY REFERENCES Projects(project_id)
hours_allocated	INT	CHECK (hours_allocated > 0)

Sample Data Set

Departments Table

dept_id	dept_name	Location
1	HR	New York
2	IT	San Francisco
3	Finance	Chicago
4	Marketing	Boston
5	Operations	Seattle
6	Legal	Washington D.C.
7	Sales	Dallas
8	R&D	Austin
9	Procurement	Denver
10	Customer Care	Miami

2. Employees Table

emp_id	first_name	last_name	Email	phone	hire_date	job_title	salary	dept_id	manager_id
101	Alice	Johnson	alice.j@corp.com	123-456-7890	2020-03-15	HR Manager	75000	1	NULL
102	Bob	Smith	bob.s@corp.com	234-567-8901	2019-05-20	IT Analyst	65000	2	104
103	Charlie	Brown	charlie.b@corp.com	345-678-9012	2021-01-10	Finance Executive	58000	3	106
104	Diana	Prince	diana.p@corp.com	456-789-0123	2018-07-12	IT Manager	90000	2	NULL
105	Ethan	Hunt	ethan.h@corp.com	567-890-1234	2022-02-25	Marketing Lead	62000	4	NULL
106	Fiona	Hall	fiona.h@corp.com	678-901-2345	2017-11-01	Finance Manager	85000	3	NULL
107	Greg	Miles	greg.m@corp.com	789-012-3456	2023-04-15	IT Support	45000	2	104
108	Hannah	White	hannah.w@corp.com	890-123-4567	2021-09-05	HR Executive	50000	1	101
109	Ian	Scott	ian.s@corp.com	901-234-5678	2020-11-20	Operations Analyst	56000	5	NULL
110	Julia	Adams	julia.a@corp.com	012-345-6789	2019-12-18	Legal Advisor	70000	6	NULL

3. Projects Table

project_id	project_name	start_date	end_date	dept_id
201	Payroll System	2023-01-01	NULL	3
202	Website Upgrade	2023-02-10	NULL	2
203	Recruitment Drive	2023-03-05	NULL	1
204	Ad Campaign	2023-05-20	NULL	4
205	New CRM Tool	2023-04-15	NULL	7
206	Compliance Portal	2023-06-10	NULL	6
207	Inventory System	2023-07-01	NULL	5
208	AI Research	2023-08-05	NULL	8
209	Customer Feedback	2023-09-10	NULL	10
210	Procurement System	2023-10-01	NULL	9

4. Employee_Project Table

emp_id	project_id	hours_allocated
102	202	120
104	202	80
103	201	100
106	201	150
101	203	50
105	204	70
107	202	60
109	207	90
110	206	110

108	203	40

Section A: DDL (Schema Creation & Modification)

1. Write SQL statements to create the above tables with the specified constraints
2. Alter the Employees table to add a column bonus DECIMAL(8,2) with default value0.
3. Drop the column bonus from Employees.

Section B: DML (Insert, Update, Delete)

4. Insert at least 10 rows into Departments, Employees, Projects, and Employee_Project.(use the above data set)
5. Try inserting an employee with a negative salary (should fail due to CHECK constraint).
6. Update the salary of the employee with emp_id = 103 by 15%.
7. Delete an employee record who has resigned (choose any emp_id).
8. Increase all employees' salaries in the IT department by 5%.
9. Change the department of an employee to "Research".(should fail due to FK constraint)

Section C: DQL (Select Queries)

10. List all employees and their details.
11. Show all employees in the "HR" department.
12. Find employees with salaries between 50,000 and 80,000.
13. Retrieve employees hired after 2020.
14. Show employees who are in either the IT or Finance department.
15. Find employees whose email ends with "@corp.com".
16. List all employees with salary > 60,000 AND located in "New York".
17. Display employees in descending order of salary.
18. Count the number of employees in each department.
19. Show the average salary of employees department-wise.

20. Display departments where the average salary is greater than 70,000.
21. Find the number of employees in each project.
22. Display departments with more than 3 employees.
23. Show the sum of all salaries department-wise.
24. List all distinct department IDs from the Employees table.
25. Show employee names with the year they were hired.
26. Show employees grouped by the year of hire.
27. List employees hired in the last 90 days.
28. List the no of years of experience of all the employees

Section D: Joins

29. List all employees with their department names (INNER JOIN).
30. Display all departments along with employees, including those departments without employees (LEFT JOIN).
31. Show employees and the projects they are working on (JOIN 3 tables: Employees, Employee_Project, Projects).
32. List projects along with total hours allocated by employees.
33. Write a query to find employees who are working on more than one project.
34. Show all projects handled by the 'Finance' department.

Section E: PL/SQL Programming

1. Write a procedure GetEmpInfo that takes emp_id as input and displays name, salary, and department.
2. Write a PL/SQL block that checks if an employee's salary is above 50,000.
   If yes, print "High Salary" ;Otherwise print "Standard Salary".
3. Write a PL/SQL program to display the top 10 rows in the Emp table based on their job and salary
4. Write a stored procedure GiveBonus that takes department ID and a designation as input, along with a bonus amount, and updates the salary of all employees in that department who have the specified designation by adding the bonus amount to their current salary.
5. Create a trigger to prevent inserting employees with a salary less than 30,000.
6. Create a trigger to avoid any transactions(insert, update, delete) on EMP table on Saturday & Sunday.

SEMESTER-V

COURSE 3: INTRODUCTION TO WEB PROGRAMMING

Theory **Credits: 3** **3 hrs/week**

Course Objectives

1. Understand the principles of web design and distinguish between web and desktop application architectures.Develop static web pages using HTML elements, attributes, and multimedia integration techniques.
2. Style web pages effectively using CSS, including layout control, responsive design, and UI enhancements.
3. Implement dynamic behaviors and form validations using JavaScript and the Document Object Model (DOM).
4. Explore client-side scripting techniques to build responsive, interactive interfaces.
5. Gain foundational knowledge of Content Management Systems (CMS) and apply practical skills in platforms such as WordPress.

Course Outcomes

At the End of the Course, The Students will be able to:

1. Design and structure HTML-based webpages incorporating text, images, tables, forms, and multimedia content.
2. Apply CSS styling rules to manage layout aesthetics, interactivity, and responsiveness across devices.
3. Use JavaScript for string manipulation, event handling, arrays, object operations, and basic validation.
4. Employ client-side scripting to enhance form functionality, create dialog interactions, and add animation via mouse and keyboard events.
5. Analyze different types of CMS and operate WordPress features like posts, pages, themes, plug-ins, and SEO tools to create and deploy basic websites.

Unit 1.HTML:

Introduction to web designing, difference between web applications and desktop applications, introduction to HTML, HTML structure, elements, attributes, headings, paragraphs, images, tables, lists, blocks, symbols, embedding multi-media components in HTML, HTML forms

Unit 2.CSS:

CSS home, introduction, syntax, CSS combinators, colors, background, borders, margins, padding, height/width, text, fonts, tables, lists, position, overflow, float, pseudo class, pseudo elements, opacity, tool tips, image gallery, CSS forms, CSS counters.

Unit 3.Java Script:
What is DHTML, JavaScript, basics, variables, operators, statements, string manipulations, mathematical functions, arrays, functions. objects, regular expressions, exception handling.

Unit 4. Client-Side Scripting:
Accessing HTML form elements using Java Script object model, basic data validations, data format validations, generating responsive messages, opening windows using java script, different kinds of dialog boxes, accessing status bar using java script, embedding basic animative features using different keyboard and mouse events.

Unit 5. Content Management Systems
Introduction to CMS: What is a CMS?, Types: traditional, headless, cloud-based

Popular CMS Platforms: WordPress, Joomla, Drupal, Shopify, When to choose each

Wordpress Basics: Introduction to word press, features, and advantages, wordpress (hosted access and local access), working with posts, managing pages, working with media, working with widgets, working with themes, extending wordpress with plug-ins, SEO and deployment.

Text Book(s)

1. Web Programming Building Internet Applications, Chris Bates, Second Edition, Wiley
2. An Introduction to Web Design plus Programming, Paul S.WangSanda S. Katila, Thomson.

Reference Books

1. Head First HTML and CSS, Elisabeth Robson, Eric Freeman, O'Reilly Media Inc.
2. An Introduction to HTML and JavaScript: for Scientists and Engineers, David R. Brooks. Springer, 2007
1. Schaum's Easy Outline HTML, David Mercer, Mcgraw Hill Professional.
2. Wordpress for Beginners, Dr.Andy Williams

SEMESTER-V

COURSE 3: INTRODUCTION TO WEB PROGRAMMING

Practical **Credits: 1** **2 hrs/week**

List of Experiments:

1. Create an HTML document with the following formatting options:

   (a) Bold, (b) Italics, (c) Underline, (d) Headings (Using H1 to H6 heading styles), (e) Font (Type, Size and Color), (f) Background (Colored background/Image in background), (g) Paragraph, (h) Line Break, (i) Horizontal Rule, (j) Pre tag

2. Create an HTML document which consists of:

   (a) Ordered List (b) Unordered List (c) Nested List (d) Image

3. Create a Table with four rows and five columns. Place an image in one column.

4. Collect any ten images of your choice. Using table tag, align the images as follows:

Using table tag, align the images as follows:

5. Create a menu form using HTML.

6. Style the menu buttons using CSS.

7. Create a form using HTML which has the following types of controls:

   (a) Text Box (b) Option/radio buttons (c) Check boxes (d) Reset and Submit buttons

8. Embed a calendar object in your web page.

9. Create a form that accepts the information from the subscriber of a mailing system.

Word press:

10. Installation and configuration of word press

11. Access admin panel and manage posts
12. Access admin panel and manage pages
13. Add widgets and menus
14. Create users and assign roles
15. Create a site and add a theme to it

SEMESTER-V

COURSE 4: FUNDAMENTALS OF PYTHON PROGRAMMING

Theory **Credits: 3** **3 hrs/week**

Course Objectives

1. **Introduce the foundational concepts** of Python programming including its syntax, IDEs, and control structures.
2. **Develop proficiency in modular programming** using functions, lambda expressions, recursion, and Python's built-in modules and packages.
3. **Explore core data structures** like strings, lists, tuples, and dictionaries for effective data manipulation.
4. **Teach exception handling mechanisms** and the use of regular expressions for pattern matching and text processing.
5. **Enable students to interact with files and visualize data** using Python to build real-world applications involving persistent storage and data retrieval.

Course Outcomes

At the end of the Course, the students will be able to

1. **Write and execute structured Python programs** using variables, expressions, and flow control statements.
2. **Implement modular code** leveraging functions, argument types, recursion, and reusable libraries.
3. **Manipulate and organize data efficiently** using Python's string operations and complex data structures.
4. **Handle runtime errors and apply regular expressions** for robust and flexible program behaviour.
5. **Perform file operations and visualize data** through Python scripts to store, manage and visualize data effectively.

Unit 1. Basics of Python Programming:

Features of python, history of python, Python IDEs, Writing and Executing Python Program, literal constants, variables and identifiers, Data types, input operation- comments, Reserved words, Indentation, Operators and Expressions: Expressions in python, Operations in Strings, Other Data types, Type conversion, Decision control Statements, Iterative Statements, Nested loops, break, Continue, Pass Statements, else statement used with loops.

Unit 2. Strings and Collections:

Strings: Built-in String Methods and Functions

Lists: Access values in List, Updating values in Lists, Nested lists, Basic list operations, List Methods.

Tuple: Creating, Accessing, Updating and Deleting Elements in a tuple,Nested tuples.

Dictionaries: Creating a dictionary, Accessing values, Modifying an Entry, Deleting items, Built-in Dictionary Functions and Methods

Unit 3. Functions and Modules:
Function Definition, Function Call, Variable Scope and lifetime, The return statement, Required Arguments, Keyword Arguments, Default Arguments and Variable Length Arguments, Lambda Functions, Recursive Functions.

Importing Libraries, Modules, Packages in python, Standard library modules- Globals(), Locals(), and Reload(), Function Redefinition.

Unit 4. Exception Handling:
Errors and Exceptions, Handling Exceptions, Multiple Except blocks, Multiple Exceptions in a single block, Except Block without Exception, The else clause, Built-in and user-defined Exceptions, The finally block, Re-raising Exception, Assertions in python

Unit 5. File Handling & Data Visualization:
Types of files in Python, Opening and Closing files, Reading and Writing files: write() and writelines() methods- append() method, read() and readlines() methods, Splitting words, File Positions.

Introduction to matplotlib: Various plots - bar, line, histogram, box plot, customization, styling

Textbooks:

1. Python Programming using Problem Solving Approach, Reema Thareja, Oxford University Press 2020
2. Exploring Python, Budd T A, McGraw-Hill Education, 1st Edition, 2011.

Reference Book:

1. Python: The Complete Reference, Martin C. Brown, McGraw-Hill, 2018
2. Fundamentals of Python, Kenneth A. Lambert. (2019), First Programs,2nd Edition, CENGAGE Publication.

Activities:

Outcome: Write and execute structured Python programs using variables, expressions, and flow control statements.

Activity: Create a calculator program that uses variables, arithmetic expressions, and flow control (if-else) to perform basic operations (add, subtract, multiply, divide) based on user input.

Evaluation Method: Code walkthrough and output validation. Use a checklist to assess on a 10-point scale to check the:

- Correct use of variables and expressions
- Proper flow control logic
- Accurate results for different inputs

Outcome: Implement modular code leveraging functions, argument types, recursion, and reusable libraries.

Activity: Build a factorial calculator using both iterative and recursive functions. Include parameterized functions and import the math library for comparison.

Evaluation Method: Code review and oral explanation. Assess on 10-point scale based on:

- Function structure and argument usage
- Recursion logic
- Use of reusable libraries

Outcome: Manipulate and organize data efficiently using Python's string operations and complex data structures.

Activity: Develop a contact manager that stores names and phone numbers using dictionaries and lists. Include string formatting and search functionality.

Evaluation Method: Practical demo with test cases. Evaluate based on:

- Use of string methods (split, join, format)
- Data structure selection and manipulation
- Search and retrieval accuracy

Outcome: Handle runtime errors and apply regular expressions for robust and flexible program behaviour.

Activity: Create a form validator that checks email and phone number formats using regular expressions. Include try-except blocks to handle invalid inputs.

Evaluation Method: Scenario-based testing. Assess based on:

- Regex accuracy for pattern matching
- Robust error handling
- Program stability with edge cases

Outcome: Perform file operations and visualize data through Python scripts to store, manage and visualize data effectively.

Activity: Build a student grade logger that reads from a file, stores data in a file, and visualizes the data.

Evaluation Method: Lab test with sample data. Evaluate on a 10-point scale:

- File read/write operations
- Visualization of data stored in files.

SEMESTER-V

COURSE 4: FUNDAMENTALS OF PYTHON PROGRAMMING

Practical **Credits: 1** **2 hrs/week**

List of Experiments

1. Display a welcome message using print().
2. Declare and use identifiers belonging to strings, integers, floats, and booleans.
3. Accept user input (name, age, height, student status) and display each value with its type using type().
4. Perform operations like .upper(), .find(), .replace() on strings.
5. Write a program to reverse the string, count vowels and words.
6. Write a program for slicing, sorting, and list comprehension.
7. Program to apply list methods: append(), extend(), insert(), remove(), pop(), sort().
8. Create tuples to store student (name, age, course) data and perform
    a. Accessing elements using indexing and slicing.
    b. Demonstrate immutability by attempting to modify a tuple.
    c. Create and navigate nested tuples.
9. Create a dictionary with student roll number as keys and names/marks as values.
    a. Accessing, adding, updating, and deleting key-value pairs.
    b. Iterating through keys, values, and items.
10. Write a program to demonstrate variable length arguments.
11. Write a program to illustrate lambda and recursive functions.
12. Write a program to demonstrate Globals(), Locals(), and Reload() functions.
13. Demonstrate exception handling and assertions in Python.
14. Write a Python program to copy the contents of one file into another in reverse order.
15. Working with Datasets using Numpy and Pandas
16. Use matplotlib to generate line, bar, pie, and scatter graphs for the data stored in files.

SEMESTER-VI

COURSE 5: CYBER SECURITY

Theory **Credits: 3** **3 hrs/week**

Course Objectives

1. Understand the fundamental concepts of cybersecurity, cybercrimes, and cyber laws.
2. Identify vulnerabilities, threats, and security measures in modern information systems.
3. Explore emerging technologies and their applications in enhancing cybersecurity and digital trust.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Explain key concepts of cybersecurity, types of cybercrimes, and the role of national/international policies.
2. Identify and assess system vulnerabilities and apply basic security management practices.
3. Implement data protection, secure system design, and apply cybersecurity tools and technologies.
4. Analyze the impact of emerging technologies such as AI, blockchain, and quantum computing in the cybersecurity domain.

Unit 1. Introduction Security and Cyber Crimes

Cyberspace and Cybersecurity, What is Information Security? Organization and Governance of the Internet and Cybersecurity, Information Security Models

What is a Cybercrime?, Classification of Cybercrimes , Forms of Cybercrimes , Cyber Scams and Frauds , Sources of Threats: Threat Actors and their Motivations , Tools and Methods Used in Cyberattacks/Cybercrimes , What is a Cyberattack? , Responding to Cyberattacks and the Cyber Kill Chain, National Cybersecurity Policy , Online Code of Conduct and Computer Ethics

Unit 2. Cybersecurity Vulnerabilities and Security Management

Types of Vulnerabilities, Project OWASP, Vulnerabilities Assessment, Common Vulnerabilities and Exposures (CVE). Overview of Cybersecurity Management, Information Classification Process, Security Policies, Security Procedures and Guidelines, Security Controls, Security Organization, Incident Response, Business Continuity and Disaster Recovery

Unit 3. Developing Secure Information Systems

Introduction, Securing Information Assets, Data Security and Protection, Application Security, Security Architecture and Design, Security Issues in Hardware, Mobile Devices and Internet of Things, Techniques/Methods for Data Security and Protection, Issues Related to Digital File Sharing

Unit 4. Cybersecurity Technologies:
Introduction, Securing Networks, Web Applications, Services and Servers, Email Security, Antivirus Technologies and Solutions, Identity and Access Management, Authentication, Cryptography, How Do Digital Money, Cryptocurrency and NFTs Work?, Digital Signatures.

Unit 5: Cyber Laws and Cyber Security applications
Need for Cyber Laws and Regulations, Role of International Law and Governments, Challenges for Law-makers and Law Enforcement Agencies, Cybersecurity Regulations

Web 3.0, Harnessing Artificial Intelligence for Cybersecurity, Blockchain for Cybersecurity, Quantum Computing and Cybersecurity, Combating Advanced Persistent Threats, Digital Trust and Identity Management, 5G Networks and Cybersecurity.

Text Books:

1. Introduction to Cybersecurity: Concepts, Principles, Technologies and Practices, Ajay Singh, 2023, Publisher: Universities Press, ISBN 9789393330314

Reference Book:

2. The Art of Invisibility: The World's Most Famous Hacker Teaches You How to Be Safe in the Age of Big Brother and Big Data. – Kevin Mitnick, 2017, Publisher: Little, Brown and Company, ISBN 978-0316380508

Activities

Outcome: Explain key concepts of cybersecurity, types of cybercrimes, and the role of national/international policies.

Activities:

- Seminar on types of cybercrimes and real-world case studies.
- Group discussion on cybersecurity policies (India's IT Act, GDPR, etc.).
- Quiz on definitions, cyber laws, and key terms.

Evaluation Methods:

- Written quiz/test (MCQs and short answers).
- Group presentation assessment.
- Evaluation of seminar participation and content depth.

Outcome: Identify and assess system vulnerabilities and apply basic security management practices.

Activities:

- Lab activity: Use of vulnerability scanning tools (e.g., Nessus, OpenVAS).
- Case study analysis on system breaches.
- Risk assessment report for a sample IT infrastructure.

Evaluation Methods:

- Practical lab test with hands-on vulnerability identification.

- Report evaluation based on thoroughness and recommendations.
- Oral questioning/viva to assess understanding of system vulnerabilities.

Outcome: Implement data protection, secure system design, and apply cybersecurity tools and technologies.

Activities:

- Demonstration and use of encryption techniques (e.g., AES, RSA).
- Firewall and antivirus configuration in the lab.
- Mini project: Design a secure login system or simulate a secure network.

Evaluation Methods:

- Practical exam on encryption and tool usage.
- Project evaluation (design, functionality, security coverage).
- Peer review and demonstration of secure system designs.

Outcome: Analyze the impact of emerging technologies such as AI, blockchain, and quantum computing in the cybersecurity domain.

Activities:

- Debate or panel discussion on AI in Cyber Defense.
- Research paper/presentation on Blockchain for data integrity.
- Seminar on threats posed by quantum computing to encryption.

Evaluation Methods:

- Rubric-based assessment of research presentation.
- Critical analysis writing assignment.
- Participation and insight evaluation during debates/seminars.

SEMESTER-VI

COURSE 5: CYBER SECURITY

Practical **Credits: 1** **2 hrs/week**

Practicals

1. Use command-line tools to find host/domain name and IP address
2. Check assigned IP address of Ethernet adapter using appropriate command
3. Verify network connectivity using ping or similar command
4. Use netstat to find current network connections
5. Find all IP addresses of a domain using DNS lookup (nslookup, dig)
6. View and manipulate routing tables using route or ip route
7. Download a website for offline viewing using HTTRACK
8. Monitor live network traffic using tcpdump or wireshark
9. Identify packets containing HTTP GET requests using Wireshark
10. Analyze number of HTTP requests and time gaps between them in Wireshark
11. Demonstrate SQL Injection in a controlled environment
12. Explore OSSEC: list components, agents, and key features
13. Explain the role of HIDS and LIDS in intrusion detection
14. Analyze mail/web logs using pflogsumm, Webalizer, and AWStats
15. List firewall rules using iptables and monitor packet filtering

SEMESTER-VI

COURSE 6: DATA VISUALIZATION USING TABLEAU & POWER BI

Theory **Credits: 3** **3 hrs/week**

Course Objectives:

1. **Understand the fundamentals of** Tableau's interface, data connectivity, and basic visualization options.
2. **Use appropriate** chart types and interactive filters to represent univariate and multivariate data.
3. **Design** interactive dashboards and geographic maps using Tableau features and custom settings.
4. **Analyze and transform** raw data from multiple sources using Power BI's data tools.
5. **Evaluate and construct** optimized data models using relationships, hierarchies, and DAX calculations.

Course Outcomes

At the end of the Course, the students will be able to

1. create a new workbook, connect to various data sources, and apply formatting and dual-axis charts using Tableau's visualization design tools.
2. construct bar charts, scatter plots, box plots, and apply filters, quick filters, and parameters to allow viewer interaction with data visualizations.
3. build dashboards integrating maps, charts, and interactive elements such as filter actions and URL links, customized to specific analytical requirements.
4. import, transform, and clean data using Group By, Unpivot, Merge, and Append queries and evaluate the readiness of data for visualization.
5. assess data modeling approaches, build star schemas, create calculated measures using DAX (like YTD, QTD), and generate insights through drilldown and drill-through reports.

Unit 1. Introduction to Tableau:
What is Tableau? - Opening Existing Workbooks - Creating New Workbooks.

Basic Visualization Design: Choosing Color Options - Setting Mark Size - Choosing Shapes - Text Tables and Mark Labels - Formatting Options - Evaluating Multiple Measures - Shared Axis Charts - Measure Names and Measure Values - Dual Axis Charts.

Connecting to Data: Connecting to Various Data Sources - The Data Source Page - Customizing Your View of the Data: Changing Data Type - Modifying Dimension / Measure Assignment - Hiding -Renaming and Combining Fields - Splitting Fields - Saving and Sharing Metadata Extracting Data -Data Blending - Moving from Test to Production Database.

Unit 2. Chart Types & Viewer

Top 10 Chart Types (Uni-variate/Bi-Variate& Multi-variate Charts): Bar Chart - Line/Area Chart - Pie Chart - Text Table / Crosstab - Scatter Plot - Bubble Chart - Bullet Graph - Box Plot - Tree Map - Word Cloud.

Interacting with the Viewer: Filtering Data - Basic Filtering -Quick Filters - Parameters - Creating a Parameter - Displaying a Parameter - Using a Parameter in a Worksheet - Filter Actions - Highlight Actions - URL Actions.

Unit 3. Tableau Maps & Dashboards

Tableau Maps: Geographic Hierarchies and Ambiguity - Custom Geocoding - Background Maps and Layers - Navigating Maps and Selecting Marks - Map Options - Web Map Services - Mapping and Mark Types - Custom Background Images - Generating Your Own Coordinate System

Creating Dashboards and Stories: Creating a Simple Dashboard - Setting Dashboard - Size - Adding Sheets - Layout Container - Blank Text - Image - Webpage - Setting Dashboard and Element - Sizes - Dashboard Actions - Highlight Action - Filter Action - URL Action.

Unit 4. Introduction to Power BI

Power Bi: Getting Data from Existing Systems - Data Sources of Power Bi - Natural Language Queries - Getting data from web - Importing Data from Northwind ODATA feed T3_IMF - Functions & list Dates in Power Bi - Group By and unpivot in Power Bi - Merging Queries in Power Bi- Append Query in Power Bi - Charts in Power Bi

Unit 5. Data Modelling in Power BI

Power Pivot Model: Explain what a Data Model is, Create Relationships between Tables in the Model, Create and use a Star Schema Data Modeling in Power Bi

Power BI Environment: Adding Calculations and Measures - User Graphs, Dash boards - Dashboard for Corona Cases analysis.

Hierarchies with Drilldown - Hierarchies with Drill Through into Power BI - Create Reports using Aggregation Functions and calculate a (scalar) value such as Count, Sum, Average, Create Reports Using MTD QTD YTD I - Create Reports Using filter functions in DAX.

Textbooks:

1. Tableau 9 - The Official Guide, George Peck, McGraw Hill, 2016
2. Beginning Power BI: A Practical Guide to Self Service Data Analytics with Excel 2016 and Power BI Desktop, Dan Clark, O'Reilley, Second Edition

Reference Books:

1. Tableau Data Visualization Cookbook, Ashutosh Nandeshwar, Packt Publishing Ltd, 2013
2. The Excel User's Guide to DAX Power Query, Power BI & Power Pivot in Excel 2010-2016, Rob Collie &Avi Singh, Holy Macro! Books, 2016

3. Tableau Your Data! Fast and Easy Visual Analysis with Tableau Software, second Edition, Daniel G. Murray, John Wiley & Sons

Activities:

CO1: Create a new workbook, connect to various data sources, and apply formatting and dual-axis charts using Tableau's visualization design tools.

Activities:

- Guided tutorial on connecting to sample datasets.
- Group discussion: Dimensions vs. Measures.
- Hands-on task: Clean and prepare data using Tableau tools (split, rename, blend).

Evaluation Methods:

- Quiz on Tableau terminology and interface.
- Practical test: Import a dataset and perform preparation steps.
- Viva or oral review of the interface and steps taken.

CO2: Construct bar charts, scatter plots, box plots, and apply filters, quick filters, and parameters to allow viewer interaction with data visualizations.

Activities:

- Create a set of charts using a given dataset.
- Lab activity: Add parameters and filter actions to make visuals interactive.
- Peer demo: Walkthrough of created interactive visuals.

Evaluation Methods:

- Lab assessment: Submission of a Tableau workbook with charts and interactivity.
- Observation checklist during practical sessions.
- Rubric-based evaluation of visual quality and interactivity.

CO3: Build dashboards integrating maps, charts, and interactive elements such as filter actions and URL links, customized to specific analytical requirements.

Activities:

- Project: Create a dashboard with at least one map, chart, and action.
- Group assignment: Compare dashboards created by peers.
- Use case study: Build a dashboard from a real-world dataset (e.g., sales by region).

Evaluation Methods:

- Final dashboard project submission with rubric for creativity, accuracy, and interactivity.
- Peer evaluation using feedback forms.
- Instructor assessment of dashboard structure and functionality.

CO4: Import, transform, and clean data using Group By, Unpivot, Merge, and Append queries and evaluate the readiness of data for visualization.

Activities:

- Guided Power BI lab: Load and transform data using Power Query Editor.
- Individual worksheet: Identify transformation needs from messy data.
- Case analysis: Suggest cleaning steps for sample raw datasets.

Evaluation Methods:

- Practical test on using transformation tools (Group By, Merge, etc.).
- Submission of a cleaned dataset with documentation.
- Checklist-based evaluation of transformation accuracy.

CO5: Assess data modeling approaches, build star schemas, create calculated measures using DAX (like YTD, QTD), and generate insights through drilldown and drill-through reports.

Activities:

- Data modeling exercise: Create a star schema using sample sales data.
- DAX lab: Write measures for aggregations (MTD, YTD).
- Dashboard development with drillthrough functionality.

Evaluation Methods:

- Rubric-based evaluation of data model quality and relationships.
- Quiz on DAX functions and schema concepts.
- Demo presentation of final Power BI dashboard with drilldowns.

SEMESTER-VI

COURSE 6: DATA VISUALIZATION USING TABLEAU & POWER BI

Practical **Credits: 1** **2 hrs/week**

Practicals

1. Install and Explore Tableau
2. Blend Data from Multiple Sources in Tableau
3. Create Univariate Charts in Tableau
4. Create Dual Axis and Shared Axis Charts in Tableau
5. Create Bivariate Charts in Tableau
6. Create Multivariate Charts in Tableau
7. Create Word Cloud and Bubble Chart in Tableau
8. Build Interactive Dashboards in Tableau
9. Create Maps and Geographic Visualizations in Tableau
10. Import Data from Web and External Sources in Power BI
11. Use Natural Language Queries and Date Functions in Power BI
12. Transform Data with Group By, Unpivot, Merge, and Append in Power BI
13. Create Charts, Reports, and Dashboards in Power BI
14. Apply Advanced Reporting Features in Power BI
15. Demonstrate Hierarchies with Drill down.
16. Demonstrate Hierarchies with Drill through.
17. Generate reports using Aggregation functions to calculate a (scalar) value such as Count, Sum, Average,Minimum, Maximum for all rows in a column or table as defined by the expression.
18. Generate reports Using MTD QTD YTD in Power BI.
19. Generate reports Using filter functions in DAX.