ANDHRA KESARI UNIVERSITY
Computer Science:MINOR
w.e.f 2023-24 onwards
COURSE STRUCTURE

Semester	Course	Title	No. Hrs./ Week	No. of Credits
II	1	Problem Solving using C - (T)	3	3
		Problem Solving using C- (P)	2	
III	2	Object Oriented Programming using Java- (T)	3	3
		Object Oriented Programming using Java- (P)	2	1
IV	3	Database Management System	3	3
		Database Management System	2	1
	4	Object Oriented Software Engineering	3	3
		Object Oriented Software Engineering	2	1

Computer Science Minor: III Semester

Course 2: Object Oriented Programming using Java Lab

Credits -1

List of Experiments

1. Write a Java program to print Fibonacci series using for loop.

2. Write a Java program to calculate multiplication of 2 matrices.

3. Create a class Rectangle. The class has attributes length and width. It should have methods that calculate the perimeter and area of the rectangle. It should have read Attributes method to read length and width from user.

4. Write a Java program that implements method overloading.

5. Write a Java program for sorting a given list of names in ascending order.

6. Write a Java program that displays the number of characters, lines and words in a text file.

7. Write a Java program to implement various types of inheritance

     i. Single   ii. Multi-Level   iii. Hierarchical   iv. Hybrid

8. Write a java program to implement runtime polymorphism.

9. Write a Java program which accepts withdraw amount from the user and throws an exception "In Sufficient Funds" when withdraw amount more than available amount.

10. Write a Java program to create three threads and that displays "good morning", for every one second, "hello" for every 2 seconds and "welcome" for every 3 seconds by using extending Thread class.

11. Write a Java program that creates three threads. First thread displays "OOPS", the second thread displays "Through" and the third thread Displays "JAVA" by using Runnable interface.

12. Implement a Java program for handling mouse events when the mouse entered, exited, clicked, pressed, released, dragged and moved in the client area.

13. Implement a Java program for handling key events when the key board is pressed, released, typed.

14. Write a Java swing program that reads two numbers from two separate text fields and display sum of two numbers in third text field when button "add" is pressed.

15. Write a Java program to design student registration form using Swing Controls. The form which having the following fields and button SAVE Form Fields are: Name, RNO, Mailid, Gender, Branch, Address.

Computer Science Minor: IV Semester
Course 3: Database Management Systems
Credits -**3**

Learning Objectives:

To familiarize with concepts of database design

Learning Outcomes: On successful completion of the course, students will be able to
1. Differentiate between database systems and file based systems
2. Design a database using ER model
3. Use relational model in database design
4. Use SQL commands for creating and manipulating data stored in databases.
5. Write PL/SQL programs to work with databases.

UNIT- I
Overview of Database Management System: Introduction to data, information, database, database management systems, file-based system, Drawbacks of file-Based System, database approach, Classification of Database Management Systems, advantages of database approach, Various Data Models, Components of Database Management System, three schema architecture of data base, costs and risks of database approach.

UNIT - II
Entity-Relationship Model: Introduction, the building blocks of an entity relationship diagram, classification of entity sets, attribute classification, relationship degree, relationship classification, reducing ER diagram to tables, enhanced entity-relationship model (EER model), generalization and specialization, **IS A** relationship and attribute inheritance, multiple inheritance, constraints on specialization and generalization, advantages of ER modeling.

UNIT - III
Relational Model: Introduction, CODD Rules, relational data model, concept of key, relational integrity, relational algebra, relational algebra operations, advantages of relational algebra, limitations of relational algebra, relational calculus, tuple relational calculus, domain relational Calculus (DRC), Functional dependencies and normal forms upto 3rd normal form.

UNIT - IV

Structured Query Language: Introduction, Commands in SQL, Data Types in SQL, Data Definition Language, Selection Operation, Projection Operation, Aggregate functions, Data Manipulation Language, Table Modification Commands, Join Operation, Set Operations, View, Sub Query.

UNIT - V

PL/SQL: Introduction, Shortcomings of SQL, Structure of PL/SQL, PL/SQL Language Elements,Data Types, Operators Precedence, Control Structure, Steps to Create a PL/SQL, Program, Iterative Control, Procedure, Function, Database Triggers, Types of Triggers.

Text Books:
1. Operating System Principles by Abraham Silberschatz, Peter Baer Galvin and GregGagne (7th Edition) Wiley India Edition.

Reference Books
1. Database Management Systems by Raghu Ramakrishnan, McGrawhill
2. Principles of Database Systems by J. D. Ullman
3. Fundamentals of Database Systems by R. Elmasri and S. Navathe
4. SQL: The Ultimate Beginners Guide by Steve Tale.

SUGGESTED CO-CURRICULAR ACTIVITIES & EVALUATION METHODS:

Unit 1: Activity: Seminar Presentation on Database Management Systems

   Evaluation Method: Depth of research, clarity of explanations, ability to addressquestions and engage the audience.

Unit 2: Activity: Case Study on EER model

   Evaluation Method: Identification of inheritance relationships, effective use of generalization and specialization, and adherence to constraints.

Unit 3: Activity: Exercise on Normalization: Assign students a set of unnormalized tables andhave them normalize the tables to third normal form

Evaluation Method: Normalized table designs, identification of functional dependencies,adherence to normalization rules, and elimination of anomalies.

Unit 4: Activity: Competition on SQL Query Writing

   Evaluation Method: Query correctness, efficiency, proper use of SQL commands, abilityto handle complex scenarios, and creativity in query formulation.

Unit 5: Activity: Peer Review of PL/SQL code

   Evaluation Method: Peer evaluation of code quality, adherence to coding standards,proper use of language elements, and logic.

List of Experiments:

1. Draw ER diagram for hospital administration

2. Creation of college database and establish relationships between tables

3. Relational database schema of a company is given in the following figure.

Relational Database Schema - COMPANY
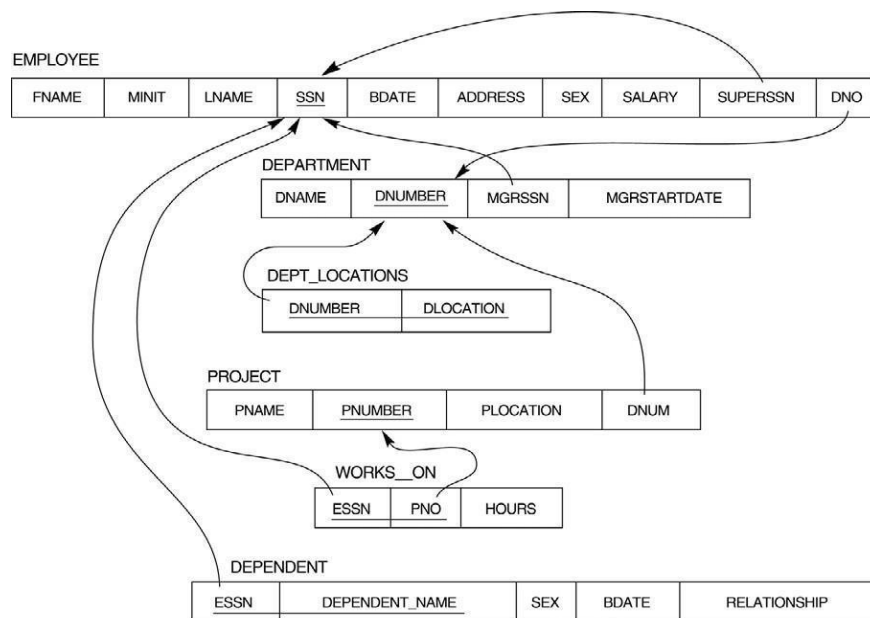
Questions to be performed on above schema

1. Create above tables with relevant Primary Key, Foreign Key and other constraints

2. Populate the tables with data

3. Display all the details of all employees working in the company.

4. Display ssn, lname, fname, address of employees who work in department no 7.

5. Retrieve the Birthdate and Address of the employee whose name is 'Franklin T. Wong'

6. Retrieve the name and salary of every employee

7. Retrieve all distinct salary values

8. Retrieve all employee names whose address is in 'Bellaire'

9. Retrieve all employees who were born during the 1950s

10. Retrieve all employees in department 5 whose salary is between 50,000 and 60,000(inclusive)

11. Retrieve the names of all employees who do not have supervisors
12. Retrieve SSN and department name for all employees
13. Retrieve the name and address of all employees who work for the 'Research' department
14. For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birth date.
15. For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.
16. Retrieve all combinations of Employee Name and Department Name
17. Make a list of all project numbers for projects that involve an employee whose last nameis 'Narayan' either as a worker or as a manager of the department that controls the project.
18. Increase the salary of all employees working on the 'Product X' project by 15%. Retrieve employee name and increased salary of these employees.
19. Retrieve a list of employees and the project name each works in, ordered by the employee's department, and within each department ordered alphabetically by employeefirst name.
20. Select the names of employees whose salary does not match with salary of any employeein department 10.

21. Retrieve the employee numbers of all employees who work on project located in Bellaire, Houston, or Stafford.
22. Find the sum of the salaries of all employees, the maximum salary, the minimum salary,and the average salary. Display with proper headings.
23. Find the sum of the salaries and number of employees of all employees of the 'Marketing' department, as well as the maximum salary, the minimum salary, and the average salary in this department.
24. Select the names of employees whose salary is greater than the average salary of all employees in department 10.
25. Delete all dependents of employee whose ssn is '123456789'.
26. Perform a query using alter command to drop/add field and a constraint in Employee table.

Course Objective:

To introduce Object-oriented software engineering (OOSE) - which is a popular technical approach to analyzing, designing an application, system, or business by applying the object- oriented paradigm and visual modeling.

Course Outcomes:

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

1. Understand and apply the fundamental principles of Object-Oriented Programming (OOP) concepts and Unified Modeling Language (UML) basics, in the development of software solutions.
2. Analyze and specify software requirements, develop use cases and scenarios, apply object-oriented analysis and design (OOAD) principles
3. Familiar with the concept of test-driven development (TDD) and its practicalimplementation
4. Analyze and Evaluate Software Maintenance and Evolution Strategies
5. Apply Advanced Object-Oriented Software Engineering Concepts

UNIT-I

Introduction to Object-Oriented Programming: Overview of software engineering, Introduction to Object-Oriented Programming (OOP) concepts (classes, objects, inheritance, polymorphism), Unified Modelling Language (UML) basics, Introduction to software development process and software development life cycle (SDLC).

UNIT-II

Requirements Analysis and Design: Requirements analysis and specification, Use cases and scenarios, Object-oriented analysis and design (OOAD), Design patterns, UML modelling techniques (class diagrams, sequence diagrams, state machine diagrams, activity diagrams)

UNIT-III

Software Construction and Testing: Software construction basics, Object-oriented design principles, Object-oriented programming languages (Java, C++, Python), Software testing basics (unit testing, integration testing, system testing), Test-driven development (TDD)

UNIT-IV

Software Maintenance and Evolution: Software maintenance basics, refactoring techniques Software version control, Code review and inspection, Software evolution and reengineering

UNIT-V

Advanced Topics in Object-Oriented Software Engineering: Model-driven engineering (MDE), Aspect-oriented programming (AOP), Component-based software engineering (CBSE), Service-oriented architecture (SOA), Agile software development and Scrum methodologies.

Text Book(s)
1. An Introduction to Object-Oriented Analysis and Design and the Unified Process, 3rd Edition, Craig Larman, Prentice-Hall.
2. Programming in Java by Sachin Malhotra, Oxford University Press

Reference Books
1. Requirements engineering: processes and techniques, G.Kotonya and, I.Sommerville, 1998, Wiley
2. Design Patterns, E.Gamma, R. Helm, R. Johnson, and J. Vlissides
3. The Unified Modeling Language Reference Manual, J. Rumbaugh, I.Jacobson and G. Booch, Addison Wesley

SUGGESTED CO-CURRICULAR ACTIVITIES & EVALUATION METHODS:

Unit 1: Activity: Group Activity: Design and implement a small OOP project
   Evaluation Method: Presentation evaluation rubric, Project evaluation based on OOP principles.

Unit 2: Activity: Use Case Scenario Presentation & Peer Activity: Review and provide feedback on each other's use case diagrams
   Evaluation Method: Presentation evaluation rubric, Peer feedback assessment.

Unit 3: Activity: Poster Presentation: Illustrate TDD principles and benefits
   Evaluation Method: Poster presentation evaluation

Unit 4: Activity: Peer Activity: Analyze and discuss different maintenance strategies
   Evaluation Method: Peer discussion participation evaluation

Unit 5: Activity: Seminar on Design Patterns
   Evaluation Method: Depth of research, clarity of explanations, ability to addressquestions and engage the audience.

Suggested Software Tools: StarUML/UMLGraph/Topcased/Umberollo/ArgoUML/ Eclipse IDE, Visual Paradigm for UML/Rational Software Architect/Any other Open Source Tool

List of Experiments:

Select domain of interest (e.g. College Management System) and identify multi-tier software application to work on (e.g. Online Fee Collection). Analyze, design and develop this application using OOSE approach:

1. Develop an IEEE standard SRS document. Also develop risk management and project plan (Gantt chart).

2. Understanding of System modeling: Data model i.e. ER – Diagram and draw the ER Diagram with generalization, specialization and aggregation of specified problem statement

3. Understanding of System modeling: Functional modeling: DFD level 0 i.e. Context Diagram and draw it

4. Understanding of System modeling: Functional modeling: DFD level 1 and DFD level 2 and draw it.

5. Identify use cases and develop the use case model.

6. Identify the business activities and develop an UML Activity diagram.

7. Identity the conceptual classes and develop a domain model with UML Class diagram.

8. Using the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams.

9. Draw the state chart diagram.

10. Identify the user interface, domain objects, and technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation.

11. Implement the technical services layer.

12. Implement the domain objects layer.

13. Implement the user interface layer.

14. Draw component and deployment diagrams.

ANDHRA KESARI UNIVERSITY-ONGOLE, PRAKASAM DISTRICT
Minor Programme from the Year 2023-24 Onwards
Programme-B.Sc. Honours Computer Science -Question Paper model,
Second Year-Semester-III & IV

Time: 3 Hours   Total Marks: 75

PART –A
Answer any Five of the following

5X5=25 Marks

1.

2.

3.

4.

5.

6.

7.

8.

9.

10

PART –B
Answer the following   **5x10=50 Marks**

11.

12.

13.

14.

15.

16.

17.

18.

19.

20.