| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |

The value of Grade is calculated as follows:

| | |
|---|---|
| If % >=90 | Grade A |
| If % >=80 & <90 | Grade B |
| If % >=70 & <80 | Grade C |
| If % >=60 & <70 | Grade D |

Otherwise, students will be declared fail.

i) Calculate Grade using if function
ii) Sort the data according to total marks
iii) Apply filter to display the marks of the students having more than 65% marks.
iv) Enter the S.No. of a student and find out the Grade of the student using VLOOKUP.
v) Extract all records where name
   a) Begins with "A"
   b) Contains "A"
   c) Ends with "A"



## GE3a: Database Management Systems

**Course Objective**

The course introduces the students to the fundamentals of database management system and its applications. Emphasis is given on the popular relational database system. Students will learn about the importance of database structure and its designing using Entity Relationship diagram and formal approach using normalization. Basic concepts of file indexing and transaction processing will be taught. The course would give students hands-on practice of structured query language to create, manipulate and implement a relational database.

**Course Learning Outcomes**

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

1. Use relational database management software to create and manipulate the database.
2. Create conceptual data models using entity relationship diagrams for modeling real-life situations and map it to corresponding relational database schema.
3. Use the concept of functional dependencies to remove redundancy and update anomalies.
4. Apply normalization theory to get a normalized database scheme to get anomalies free database.
5. Write queries in relational algebra.
6. Implement relational databases and formulate queries for data retrieval and data update problems using SQL .
7. Learn the importance of index structures and concurrent execution of transactions in database systems.

**Syllabus**

**Unit 1 Introduction to Database:** Database, characteristics of database approach, data models, database management system, three-schema architecture, components of DBMS, data independence, and file system approach vs database system approach.

**Unit 2 Entity Relationship Modeling:** Conceptual data modeling - motivation, entities, entity types, attributes, relationships, relationship types, constraints on relationship, Entity Relationship diagram as conceptual data model.

**Unit 3 Relational Data Model:** Data anomalies, Relational Data Model - Characteristics of a relation, schema-instance distinction, types of keys, relational integrity constraints. Relational algebra operators like selection, projection, cartesian product, join and write simple queries using them.

**Unit 4 Structured Query Language (SQL):** DDL to create database and tables, table constraints, DML, Querying in SQL to retrieve data from the database, aggregation functions group by and having clauses, generate and query views.

**Unit 5 Database Design:** Mapping an Entity Relationship diagram to corresponding relational database scheme, functional dependencies and Normal forms, 1NF, 2NF, and 3NF decompositions and desirable properties of them.

**Unit 6 Basics of File indexing and introduction to Transaction Processing:** Need of file indexes, types of indexes, introduction to file organizations, basics of concurrent execution of transactions.

**References**


1. Elmasri, R., Navathe, B. S., *Fundamentals of Database Systems*, 7th Edition, Pearson Education, 2016.
2. Murach J., *Murach's MySQL,* 3th Edition, Pearson, 2019.


**Additional References**


(i) Connolly T. M., Begg C. E. *Database Systems: A Practical Approach to Design, Implementation, and Management*, 6th edition, Pearson, 2019.

(ii) Ramakrishnan R., Gehrke J. *Database Management Systems*, 3rd Edition, McGraw-Hill, 2014.

(iii) Silberschatz A., Korth H.F., Sudarshan S. *Database System Concepts*, 7th Edition, McGraw Hill, 2019.


**Suggested Practical List**

**Create and use the following student-course database schema for a college to answer the given queries using the standalone SQL editor.**

| **STUDENT** | **Roll No** | **StudentName** | **CourseID** | **DOB** |
|---|---|---|---|---|
| | Char(6) | Varchar(20) | Varchar(10) | Date |

| **COURSE** | **CID** | **CourseName** | **Course Type** | **Teacher-in-charge** | **TotalSeats** | **Duration** |
|---|---|---|---|---|---|---|
| | Char(6) | Varchar (20) | Char (8) | Varchar (15) | Unsigned int | Unsigned int |

| **ADMISSION** | **Roll No** | **CID** | **DateOfAdmission** |
|---|---|---|---|
| | Char(6) | Char(6) | **Date** |

Here Rollno (ADMISSION) and SID (ADMISSION) are foreign keys. Note that course type may have two values viz. Fulltime and Parttime and a student may enroll in any number of courses

1. Retrieve names of students enrolled in any course.
2. Retrieve names of students enrolled in at least one part time course.
3. Retrieve students' names starting with letter ‘A’.
4. Retrieve students' details studying in courses ‘computer science’ or ‘chemistry’.
5. Retrieve students’ names whose roll no either starts with ‘X’ or ‘Z’ and ends with ‘9’
6. Find course details with more than N students enrolled where N is to be input by the user
7. Update student table for modifying a student name.
8. Find course names in which more than five students have enrolled
9. Find the name of youngest student enrolled in course ‘BSc(P)CS’
10. Find the name of most popular society (on the basis of enrolled students)
11. Find the name of two popular part time courses (on the basis of enrolled students)
12. Find the student names who are admitted to full time courses only.
13. Find course names in which more than 30 students took admission
14. Find names of all students who took admission to any course and course names in which at least one student has enrolled
15. Find course names such that its teacher-in-charge has a name with ‘Gupta’ in it and the course is full time.
16. Find the course names in which the number of enrolled students is only 10% of its total seats.
17. Display the vacant seats for each course
18. Increment Total Seats of each course by 10%
19. Add enrollment fees paid (‘yes’/’No’) field in the enrollment table.
20. Update date of admission of all the courses by 1 year.
21. Create a view to keep track of course names with the total number of students enrolled in it.
22. Count the number of courses with more than 5 students enrolled for each type of course.
23. Add column Mobile number in student table with default value ‘9999999999’
24. Find the total number of students whose age is > 18 years.
25. Find names of students who are born in 2001 and are admitted to at least one part time course.
26. Count all courses having ‘science’ in the name and starting with the word ‘BSc’.

**Students are also encouraged to implement the database given in the textbook and do the related queries.**

## GE3b : Java Programming

### Course Objective

This course is designed to develop understanding of object-oriented programming concepts like Classes, Objects, Inheritance and Polymorphism using Java. The course provides