

GRAPHIC ERA HILL UNIVERSITY, DEHRADUN

SEMESTER V

Name of Department:- Computer Science and Engineering

1.	Subject Code:	TCS 509	Course Title:	Machine Learning
2.	Contact Hours:	L: 3	T: -	P: 2
3.	Semester: V			

4. Pre-requisite: TCS201, TCS421

5. Course Outcomes: After completion of the course students will be able to

1. Have a good understanding of the fundamental issues and challenges of machine learning: data, model selection, model complexity, etc.
2. Distinguish the strengths and weaknesses of many popular machine learning approaches.
3. Analyze the underlying relationships within and across Machine Learning algorithms and the paradigms of supervised and un-supervised learning.
4. Utilize the structure and design concepts of neural networks applications to solve real life problems
5. Plan and execute successful machine learning and big data projects, including selecting an adequate process for the specific task and avoiding the machine learning pitfalls.
6. Evaluate the issues raised by current research in the field of machine learning

6. Detailed Syllabus

UNI T	CONTENT	Contact Hrs
Unit – I	Review of Statistical Concepts: Mean, Median, Mode, Outliers, Range, Average Deviation, Absolute Deviation, Squared Deviation, Standard Deviation, Total Sum of Squares. Introduction to Machine Learning: What is Machine Learning, Introduction to ML's three approaches: Supervised, Unsupervised and Reinforcement Learning. Introduction to Python: Basic Operations, Lists, Tuples, Dictionaries, Flow Control, Strings, File handling, Numpy, Scikit-learn	10
Unit - II	Introduction to Exploratory Data Analysis Introduction to Exploratory Data Analysis (EDA) – Steps in EDA, Data Types: Numerical Data – Discrete data, continuous data – Categorical data Data Transformation Transformation Techniques: Performing data deduplication - replacing values – Discretization and binning. Introduction to Missing data, handling missing data Data Visualization using Matplotlib, Seaborn	
Unit – III	Supervised Learning Algorithms: Linear Regression, Logistic Regression, Decision Trees, Random Forest, Support Vector Machine, K-Nearest Neighbours, CN2 Algorithm, Naive Bayes	10
Unit -IV	Clustering: K-means, Silhouette Scores, Hierarchical Clustering, Fuzzy c-means, DBScan Dimensionality Reduction: Low Variance Filter, High Correlation Filter, Backward Feature Elimination, Forward Feature Selection, Principle Component Analysis, Projection Methods.	8

Unit V	Model Evaluation and Selection: Cross-validation, model evaluation metrics, model selection, and hyperparameter tuning. Hyperparameter Optimization Techniques Manual Search, Random Search, Grid Search Case study in Python for Hyperparameter Tuning	
Total		49

Text and Reference Books

1. "Machine Learning For Dummies", John Paul Mueller and Luca Massaron
2. "A Course in Machine Learning", Hal Daumé III.
3. "Programming Collective Intelligence: Building Smart Web 2.0 Applications", Toby Segaran
4. "Building Machine Learning Systems with Python", WilliRichert and Luis Pedro Coelho
5. "Learning scikit-learn: Machine Learning in Python", Raúl Garreta and Guillermo Moncecchi
6. "Machine Learning in Action", Peter Harrington

GRAPHIC ERA HILL UNIVERSITY, DEHRADUN

SEMESTER V

1. Name of Department: - Computer Science and Engineering

2. Subject Code: Course . . . **TCS 502** Title: **Operating Systems**

3. Contact Hours: L: **3** . T: **0** P: **0**

4. Examination Duration (Hrs): **Theory** **3** **Practical** **0**

5. Relative Weight: **CIE** **25** **MSE** **25** **SEE** **50**

6. Credits: **3**

7. Semester: **4**

8. Category of Course: **DC**

9. Pre-requisite: **TCS 301, TCS 302, TCS 404**

9. Course Outcome **:	After completion of the course the students will be able to: CO1 Understand the concept and design issues associated with an operating system. CO2: Identify the problems related to process management, synchronization and apply learned methods to solve basic problems. CO3. Explain the basics of memory management and the use of virtual memory in modern operating systems. CO4. Understand the concept deadlock avoidance, prevention, and detections techniques. CO5: Implementation of process management, memory management and file management using system calls. CO6: Analyze the data structures and algorithms used for developing an operating system.
------------------------------	--

*** Describe the specific knowledge, skills or competencies the students are expected to acquire or demonstrate.*

10. **Details of the Course:**

Sl. No.	Contents	Contact Hours
----------------	-----------------	----------------------

1	Introduction to Operating Systems, UNIX: What operating systems do; Operating System structure; Operating System Services; Operating System Classification; User - Operating System interface; System calls; Types of system calls; Operating System structure; Unix command: Command Structure, Internal and External commands, filters; vi editor.	8
2	Process Management: Process concept; Operations on processes; Multithreading models; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling. Process Synchronization: Inter-process communication; Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization.	10
3	Deadlocks: Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance - resource trajectories, safe and unsafe states, bankers' algorithm; Deadlock detection and recovery from deadlock.	8
4	Memory Management: Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Page replacement; Allocation of frames; Thrashing	10
5	File System Management: File System: File concept; Access methods; Directory structure; Protection. File system structure; Directory implementation; Allocation methods; Free space management. Secondary Storage Structures: Mass storage structures; Disk structure; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Access matrix.	8
	Total	44

11. Suggested Books:

SL. No.	Name of Authors/Books/Publishers	Edition	Year of Publication / Reprint
Textbooks			
1.	Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 7th edition, Wiley India, 2006.	7 st	2006
2.	Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 7th edition, Wiley India, 2006.	7 th	2006
3.	Unix concepts and applications – Sumitabha Das	1 st	2005
Reference Books			

1.	Andrew S Tanenbaum: Operating Systems: Design and Implementation, 3rd edition, Prentice Hall, 2006	3 rd	2006
2.	Stuart E. Madnick, John Donovan: Operating Systems, Tata McGraw Hill, 2008		2008

12.	Mode of Evaluation	Test / Quiz / Assignment / Mid Term Exam / End Term Exam
-----	---------------------------	--

GRAPHIC ERA HILL UNIVERSITY, DEHRADUN

SEMESTER V

Name of Department: - Computer Science and Engineering

1.	Subject Code:	TCS-503	Course Title:	Database Management System
2.	Contact Hours:	L: 3	T: 0	P: 0
3.	Examination Duration (Hrs):	Theory 3	Practical 0	
4.	Relative Weight:	CIE 25	MSE 25	SEE 50
5.	Credits:	3		
6.	Semester:	5th		
7.	Category of Course:	DC		
8.	Pre-requisite:	TCS 302, TCS 404		

9. Course Outcome**:	<p>After completion of the course the students will be able to:</p> <p>CO1: Understand the different issues involved in the design and implementation of a database system.</p> <p>CO2: Study the physical and logical database designs, database modeling, relational, hierarchical, and network models.</p> <p>CO3: Understand and use data manipulation language to query, update, and manage a database.</p> <p>CO4: Develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency.</p> <p>CO5: Design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.</p> <p>CO6: Evaluate a business situation and designing & building a database application</p>
-----------------------------	--

*** Describe the specific knowledge, skills or competencies the students are expected to acquire or demonstrate.*

10. Details of the Course:

Sl. No.	Contents	Contact Hours
1	<p>Unit 1:</p> <p>Introduction: An overview of DBMS; Advantages of using DBMS approach; Database systems vs File Systems, Database system concepts and architecture</p> <p>1 Data models, schemas, and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures; Classification of Database Management systems.</p>	9
2	<p>Unit 2:</p> <p>Entity-Relationship Model: Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets,</p>	9

	Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two.	
3	<p>Relational Model and Relational Algebra: Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra and Calculus Operations from Set Theory; Binary Relational Operations: JOIN and DIVISION; Additional Relational Operations. Examples of Queries in Relational Algebra; Relational Database Design Using ER- to-Relational Mapping.</p> <p>SQL – 1: SQL Data Definition and Data Types; Specifying basic constraints in SQL; Schema change statements in SQL; Basic queries in SQL; More complex SQL Queries.</p> <p>Insert, Delete and Update statements in SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; Additional features of SQL; Database programming issues and techniques; Embedded SQL, Dynamic SQL; Database stored procedures.</p>	11
4	Optimization of SQL Queries through Indexes, Concepts of NoSQL.	
4	<p>Unit 4: Database Design – 1: Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form</p> <p>Properties of Relational Decompositions; Algorithms for Relational Database Schema Design; Multivalued Dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form; Inclusion Dependencies; Other Dependencies and Normal Forms</p>	9
5	<p>Unit 5:</p> <p>Transaction Management: The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery; 2PL, Serializability and Recoverability; Lock Management; Log Files; Checkpointing; Recovering from a System Crash; Media Recovery</p>	10
Total		48

11. Suggested Books:

SL. No.	Name of Authors/Books/Publishers	Edition	Year of Publication / Reprint
Textbooks			
1.	McGraw-Hill. Date K., Swamynathan S. An Introduction to Database Systems. Eight Edition. Pearson.	2nd	2012
2.	Elmasri R. and Navathe S.B., Fundamentals of Database Systems.	2 nd	2012

3.	Fifth Edition.Pearson. Singh S.K., Database Systems-Concepts, Designs and Application. 2nd Edition. Pearson	2 nd	2011
4.	Date, C.J. Introduction to Database Systems (Vol I & II) 8th Edition. Addison- Wesley.	8 th	2004
	Reference Books		
1.	Silberschatz A. Korth H. F. Sudarshan S., Database System Concepts. Sixth Edition	1 st	2014

12.	Mode of Evaluation	Test / Quiz / Assignment / Mid Term Exam / End Term Exam
-----	---------------------------	--

GRAPHIC ERA HILL UNIVERSITY, DEHRADUN

SEMESTER V

PCS 502 Operating Systems Lab

. Course Outcome:	After completion of the course the students will be able to:
	<p>CO1. Implement concept of system calls for process management.</p> <p>CO2. Analyze and Implement various algorithms like FCFS, Priority and Round Robin for CPU scheduling.</p> <p>CO3. Simulate working of page replacement policies like FIFO, LRU.</p> <p>CO4. Compare various algorithms for communication between processes like pipe, named pipe, message queue and shared memory.</p>

Details of the Course:

Sl. No.	List of problems for which student should develop program and execute in the Laboratory	Contact Hours
1	Problem Statement 1: Demonstration of FORK() System Call	3
2	Problem Statement 2: Parent Process Computes the SUM OF EVEN and Child Process Computes the sum of ODD NUMBERS using fork	3
3	Problem Statement 3: Demonstration of WAIT() System Call	3
4	Problem Statement 4: Implementation of ORPHAN PROCESS & ZOMBIE PROCESS	3
5	Problem Statement 5: Implementation of PIPE	3
6	Problem Statement 6: Implementation of FIFO	3
7	Problem Statement 7: Implementation of MESSAGE QUEUE	3

8	Problem Statement 8: Implementation of SHARED MEMORY	3
9	Problem Statement 9: Implementation of FIRST COME FIRST SERVED SCHEDULING ALGO	3
10	Problem Statement 10: Implementation of SHORTEST JOB FIRST SCHEDULING ALGO	3
11	Problem Statement 11: Implementation of PRIORITY SCHEDULING ALGO	3
12	Problem Statement 12: Implementation of First comes first serve page replacement policy	3
13	Problem Statement 13: Implementation of Least recent used page replacement policy	3
14	Problem Statement 14: Demonstration of execl() where child process executes "ls" COMMAND and Parent process executes "date" COMMAND	3
15	Problem Statement 15: Implementation of COMMAND ls wc USING PIPES.	3
	Total	45

Text Books:

Authors Name	Title	Edition	Publisher, Country	Year
Abraham Silberschatz, Peter Baer Galvin, Greg Gagne:	Operating System Principles	7 th Edition	Wiley India	2006
Sumitabha Das	Unix concepts and applications	4 th Edition	McGraw Hill Education	2017

Reference Books:

Authors Name	Title	Edition	Publisher, Country	Year

Andrew S Tanenbaum	Operating Systems: Design and Implementation,	3 rd Edition	Prentice Hall,	2006
Stuart E. Madnick, John Donovan:	Operating Systems,	1 st Edition	Tata McGraw Hill,	2008

GRAPHIC ERA HILL UNIVERSITY, DEHRADUN

SEMESTER V

Name of Department: - Computer Science and Engineering

1.	Subject Code:	PCS-503	Course Title:	Database Management System Lab	
2.	Contact Hours:	L: 0	T: 1	P: 2	
3.	Examination Duration (Hrs):	Theory 0	Practical 3		
4.	Relative Weight:	CIE 25	MSE 25	SEE 50	
5.	Credits:	2			
6.	Semester:	5th			
7.	Category of Course:	DC			
8.	Pre-requisite:	TCS 302, TCS 404			

9. Course Outcome**:	After completion of the course the students will be able to: CO1: Students get practical knowledge on designing and analysis of conceptual model and mapping of conceptual model to relational database systems. CO2: Design and implement SQL queries using DDL and DML concepts for updation and managing a database. CO3: Design and implement advance SQL queries such as relational constraints, joins, set operations, aggregate functions, and views. CO4: Design and implement queries using optimization techniques. CO5: Application of transaction control language (TCL), data control language (DCL) in SQL to evaluate practical implications of DBA such as transaction, recovery, and security.
-----------------------------	--

*** Describe the specific knowledge, skills or competencies the students are expected to acquire or demonstrate.*

10. Details of the Course:

Sl. No.	List of problems for which student should develop program and execute in the Laboratory	Contact Hours
1.	Week 1: (Store all your data in a file named `db.sql` and email it to yourself for future reference and use in subsequent labs.) <ul style="list-style-type: none"> • Create a Scenario based ER-Models with the entities. (Hospital Details like: Wards, Patients, Doctor, Bills etc) • Convert this ER-model into table with all the entities. (Minimum five Entities). 	

	<ul style="list-style-type: none"> • Insert random data in each column of all the tables. • Update the table by applying some conditions.(For example: using alter command) • Apply the `DELETE` and `DROP` command, and then review the results. 	
2.	<p>Week 2:</p> <ul style="list-style-type: none"> • Create a user and provide the GRANT privileges to the user on the database then REVOKE the given privileges. • Insert any five records in the previous schema and apply the rollback. Also check the results. • Add default, check, unique and not null constraints to the schema. • Insert NULL values and check the results. • Add duplicate value and try to make a column as primary key, check what happen to the table. 	
3.	<p>Week 3: (If the Employee table is not present in the `db.sql` file, please create it.)</p> <ul style="list-style-type: none"> • Create an Employee table with the following attributes and constraints: Employee Table - (Employee Id. (Primary key), Name, Department, Age (check >18), Salary, City). • Display the total number of employees. • Retrieve all information of employees whose age is 22. • Fetch the employee id, name, and department, whose salary >= 50000. • Print the name of the employees and label the column as "Full Name" for those employees whose department name is 'Finance' and age is 22. • Print the department names from the employee table without having the duplicates. <p>Perform similar queries until the end of the lab session.</p>	
4.	<p>Week 4:</p> <ul style="list-style-type: none"> • Find out the maximum and minimum salary from the employee table. • Show the total salary and average salary of all the employees. • Show all the details of the employees who have the same salary. • Display the employees name from lowest salary to the highest salary. • Display the employee name and salary (department-wise) for employees, whose salary is greater than or equal to 10,000 and age is greater than 25. 	

5.	<p>Week 5:</p> <ul style="list-style-type: none"> Fetch the information of employees who belong to the city "Delhi" or "Pune." Print the name and department of employees whose ID is in the range from 2001 to 2005. Show the names of employees who belong to the same city (use the IN operator). Check whether the all employee is belongs to the same city or not. (use ALL operator) Check whether the all employee is belongs to the same city or not. (use ANY operator) Check whether the all employee is belongs to the same city or not. (use Exists operator) 	
6.	<p>Week 6:</p> <ul style="list-style-type: none"> Show the record of employees who are working in the 'CSE' department. Fetch the names of employees whose names start with the letters 'ay'. Fetch the information of employees, including their names and departments, whose names end with the letters 'sh'. Display the employee names and their departments of employees, whose city name starts with 'D' or ends with 'h'. Print all records of employees whose salary is greater than 15,000 and whose name starts with 'h'. Print the names of employees whose names consist of exactly three letters. Print the names of employees along with their city for those whose names have at least five letters. 	
7.	<p>Week 7:</p> <ul style="list-style-type: none"> Create two tables named as employee and department with the given constraints and attributes: Employee table - (Employee Id.(Primary key), Department ID, Name, Age (check >18), Salary, City) Department table - (Department Id, and Department name) Display the details of employees along with their corresponding department names. Print the names of employees who are not assigned to any department. Print the employee names and department names for employees whose salary is greater than 25,000. (Using left join). Display the names of employees along with their department names for those who are not assigned to any department. 	

	<ul style="list-style-type: none"> Print the employee names and their corresponding department names for employees with a salary greater than 25,000. (Using right join). Display the names of departments along with the names of employees who are older than 30 years. 	
8.	<p>Week 8:</p> <ul style="list-style-type: none"> Create the table to keep track of customer records and their order. Customer table - (Name as Not null, Customer_id as primary key, Age, Address) Order table - (Customer_id, order_id, date). Apply the full join and the full outer join to the schema and review the results. Display the name of the city as "destination" for customers who have placed orders. Apply the cross join and check the results. Display the customer names and order IDs for customers who have placed orders from the same city. 	
9.	<p>Week 9:</p> <ul style="list-style-type: none"> Create the Student table, Register table and Program table. Student table - (Roll no. as primary key, Name as not null, city) Program table - (Program ID as primary key, Program Name as not null, Program Fee not less than 10000, Department) Register table - (Program ID and Roll no. as primary composite key) Display the details of students who are registered in the "MCA" program. Display the list of all students, who are registered in at least one program. Display the details of programs that have fees greater than the average fee. Display the names of students who are registered in a program having fees less than 30000. Display the details of students who have not registered in any course. Display the names of programs in which a maximum number of students are registered. Display the names of programs in which a minimum number of students are registered. 	
10.	<p>Week 10:</p> <ul style="list-style-type: none"> Find out the second minimum salary of an employee. 	

	<ul style="list-style-type: none"> • Find out the second minimum salary of an employee without using limit, dense range, and order by clause. • Find out the third maximum salary of an employee. • Find out the third maximum salary of an employee without using limit, dense range, and order by clause. • Display the names and salaries of employees who earn more than the average salary of their department. • Fetch the list of the employee who belongs to the same department but earns less than the second employee. • Display the names of employees who are older than their colleagues in the same department. 	
11.	<p>Week 11:</p> <ul style="list-style-type: none"> • Create a row level trigger for the customers table that would fire for INSERT or UPDATE or DELETE operations performed on the EMPLOYEE table. This trigger will display the salary difference between the old values and new values. • Add a new employee with the salary value inserted and check the result. • Try to update the existing employee salary and see what happens. • Delete a record of employees and check what happens. • Convert employee name into uppercase whenever an employee record is inserted or updated. 	
12.	<p>Week 12:</p> <p>Case study 1: (General Hospital)</p> <p>A hospital relies on a database to manage its operations effectively. This database helps keep track of various aspects, including different wards like the General Ward, Emergency Ward, and Specific Ward. Each ward contains patients who are admitted based on their General Practitioner's (GP) recommendation and the approval of a consultant from the hospital. When a patient is admitted, the hospital records essential personal details such as their name, age, gender, address, and contact information. This information is crucial for medical and administrative purposes. Additionally, the hospital maintains a separate register to record all medical tests and treatments for each patient, ensuring that their medical history is thoroughly documented. Patients may undergo multiple tests during their stay, and the database is designed to link each patient with these test results.</p>	

	<p>records. Each patient is assigned a leading consultant who oversees their treatment, but they may also be examined by other doctors if needed.</p> <p>The database also tracks the connections between patients, consultants, and doctors. Consultants and doctors might specialize in different medical fields and can treat patients from various wards, adding flexibility to the care provided. Overall, this database ensures that patient information, medical records, and hospital operations are managed efficiently. It supports the hospital in delivering high-quality care, streamlining administrative tasks, and addressing the specialized needs of patients and medical staff.</p> <p>Based on the details provided in the case study, address the following requirements:</p> <p>Create an ER diagram based on the hospital's database system case study. Include entities like patients, wards, consultants, and doctors with relevant attributes such as Patient ID, Ward ID, and Consultant ID. Also, none of the entities in the template are marked as weak; if you wish to change that, you may. You will need to specify two things:</p> <ol style="list-style-type: none"> Specify all attributes and keys for each entity. Clearly define relationships, such as patients being associated with wards, consultants, and doctors, and include connections between patients and their medical tests. Define all relationships and constraints, including primary keys, cardinality, and participation constraints. Show how a patient can undergo multiple tests and be treated by various doctors. <p>Note: Model most constraints from the description. If some constraints can't be represented, provide comments explaining the limitations.</p>	
13.	<p>Week 13:</p> <p>Case Study 2: (Tracking the Employee Record)</p> <p>An organization has implemented a detailed database system to manage and track its employees and departmental activities. The organization is divided into various departments, each with a unique identification number and name. Each department is managed by a designated manager, who is responsible for overseeing the operations within that department.</p> <p>Additionally, some departments may be located in different geographic locations, reflecting the organization's diverse operational reach. The database maintains comprehensive records for each employee, including their name, identification number, birth date, address, gender, and salary. Employees are assigned to specific departments, and the system tracks the date on which a manager was appointed to each department, ensuring that managerial changes are recorded accurately. Beyond departmental assignments, the database captures supervisory relationships where employees may be directly supervised by others. This helps in understanding the hierarchical structure within the organization. Moreover, every project undertaken by the organization is managed by a specific</p>	

	<p>department, although employees from various departments may be assigned to these projects based on their expertise and the project's requirements.</p> <p>For each project, the database includes details such as the project name, project number, and location. Additionally, it records the hours spent by employees on each project, providing insights into individual contributions and project progress. This tracking helps in managing project resources effectively and ensures that employees' time and efforts are properly accounted for. Overall, this database system facilitates efficient management of employee information, departmental structures, and project assignments. It allows the organization to monitor departmental performance, manage employee roles and responsibilities, and track the progress of various projects with precision. This comprehensive approach ensures that all aspects of employee and project management are well-coordinated and effectively managed.</p> <p>Based on the details provided in the case study, address the following requirements:</p> <p>Create an ER diagram representing the organization's database system. Include key entities such as Departments, Employees, and Projects. For each Department, capture attributes like Department ID, Department Name, Manager ID, and Location. For Employees, include attributes such as Employee ID, Name, Birth Date, Address, Gender, Salary, and Department ID. Projects should have attributes like Project ID, Project Name, and Project Location.</p> <ol style="list-style-type: none"> Specify all attributes and primary keys for each entity. Clearly define relationships between entities, such as employees being assigned to departments, departments managing projects, and supervisory relationships among employees. Define the relationships and constraints, including primary keys, cardinality, and participation constraints. For example, a department can manage multiple projects, and employees can work on multiple projects while reporting to one or more supervisors. <p>Note: Include comments to address any constraints from the case study that cannot be fully represented in the ER diagram. Ensure the diagram accurately reflects the management of employee information, departmental structures, and project assignments as described.</p> <p>Based on the provided case study, perform the following queries:</p> <ol style="list-style-type: none"> Write an SQL query to identify the department(s) with the highest average salary among its employees. Write an SQL query to list all employees who are directly supervised by more than one manager. Write an SQL query to find the project(s) with the highest total hours spent by employees. 	
--	--	--

	<p>d. Write an SQL query to find all employees who have never been assigned to any project.</p> <p>e. Write an SQL query to list each department along with the total number of projects managed by the department and the total number of employees assigned to it.</p>	
14.	<p>Week 14:</p> <p>Case Study 2: (Trainee Record in a Institution)</p> <p>A training institute requires a sophisticated database to effectively track the progress of trainees in their various training programs. Trainees enroll in different programs, such as Java Developer, Full Stack Developer, and Data Scientist. Each of these programs consists of several courses offered by the institute, each with its own unique code, title, and number of credit hours. Courses are overseen by an instructor, who may also be involved in teaching the course. In addition to the main instructor, each course has one or more teaching assistants who help facilitate the course. An instructor can manage and teach multiple courses across different programs, reflecting their expertise and versatility. The database captures detailed information about each course, including its unique code, title, credit hours, the instructor responsible for managing and teaching the course, the teaching assistants assigned, and the department to which the course belongs. Each course may have prerequisites, which means that some courses must be completed before others can be taken. This requirement ensures that trainees follow a structured learning path. Additionally, some training programs have mandatory courses that all trainees must complete to successfully finish the program.</p> <p>For trainees, the database records essential details such as their ID numbers, names, addresses, the training program they are enrolled in, and their academic performance. This includes information on the courses they have taken and the grades they received. By maintaining these records, the database helps monitor each trainee's progress and ensures they meet the necessary requirements for their chosen program. This database system plays a crucial role in managing the training institute's operations. It provides a clear view of course offerings, tracks the performance of trainees, and ensures that the educational requirements and prerequisites are properly enforced. This comprehensive approach allows the institute to deliver a well-organized training experience and supports both the instructors and trainees in achieving their educational goals.</p> <p>Based on the details provided in the case study, address the following requirements:</p> <p>Develop an ER diagram to model the training institute's database system, incorporating all key entities and their attributes. The primary entities to include are Trainees, Programs, Courses, Instructors, and Teaching Assistants. For each entity, specify attributes and primary keys: Trainees should have Trainee_ID, Name, Address, Program_ID, and Performance</p>	

	<p>details. Courses should include Course_Code, Title, Credit_Hours, Prerequisites, and links to Instructors and Teaching Assistants.</p> <p>a. Define the relationships between entities, such as Trainees being enrolled in Programs, Courses assigned to Programs, and Instructors and Teaching Assistants managing and teaching Courses. Include relationships for course prerequisites and mandatory courses within Programs.</p> <p>b. Clearly specify key constraints like primary keys for each entity and cardinality and participation constraints for relationships. For instance, a Course may have multiple Teaching Assistants and prerequisites, while a Trainee can be enrolled in multiple Courses.</p> <p>Note: Add comments to highlight any constraints from the case study that are challenging to represent in the ER diagram. Ensure the diagram effectively captures the management of trainees' progress, course requirements, and instructor roles as outlined.</p> <p>Based on the provided case study, perform the following queries:</p> <p>a. Write an SQL query to show the names and IDs of trainees who have completed all the required courses for their program.</p> <p>b. Write an SQL query to get the names of instructors and the courses they teach, including any courses that have prerequisites.</p> <p>c. Write an SQL query to find the average grade for each course and list the courses where the average grade is below than 70%.</p> <p>d. Write an SQL query to find the trainees who have signed up for the most courses, no matter which program they are in.</p> <p>e. Write an SQL query to list all courses and the total hours all trainees have spent on each course.</p>	
	Total	40

11. Suggested Books:

S. No.	Name of Authors/Books/Publishers	Edition	Year of Publication / Reprint
Text Books			
1.	RamezElmasri, Shamkant, B. Navathe, "Database Systems", Pearson Education, 7Th Edition.	7 th	2015
3.	M L Gillenson, "Introduction to Database Management", Wiley Student Edition	2 nd	2012
Reference Books			

1.	Mary Beth Roeser: [1Oracle® Database] SQL Language Reference 12c Release 1 (12.1) E41329-25	2 nd	2017
----	---	-----------------	------

12.	Mode of Evaluation	Test / Quiz / Assignment / Mid Term Exam / End Term Exam
-----	---------------------------	--

GRAPHIC ERA HILL UNIVERSITY, DEHRADUN

SEMESTER V

Name of Department: - Computer Science and Engineering

1.	Subject Code:	TCS 552	Course Title:	Cloud-Based Application Development and Management
2.	Contact Hours:	L: 3	T: 0	P: 0
3.	Examination Duration (Hrs):	Theory 3	Practical 0	
4.	Relative Weight:	CIE 25	MSE 25	SEE 50
5.	Credits:	3		
6.	Semester:	5		
7.	Category of Course:	DE		
8.	Prerequisite:	TCS-451		

9. Course Outcome**:	<p>After completion of the course the students will be able to:</p> <p>CO1: Recognize the cloud based application development platforms and economic benefits.</p> <p>CO2: Analyze the use case of various cloud service provider's applications and platforms.</p> <p>CO3: Apply the advanced cloud computing application's concepts.</p> <p>CO4: Analyze the use case of cloud-based application deployment and management concepts.</p> <p>CO5: Explore the use case of various cloud platforms, offered services and security aspects.</p> <p>CO6: Develop and deploy the cloud based server-side application using Node.js and the front-end using React.</p>
-----------------------------	--

*** Describe the specific knowledge, skills or competencies the students are expected to acquire or demonstrate.*

10. Details of the Course:

SL. NO.	Contents	Contact Hours
1	Unit 1: Fundamental of Cloud Based Applications Cloud Data centers, Software stack, Virtualization, software defined networks and storage, cloud storage, and programming models, Clouds Based Application development motivating factors, benefits, challenges, service models, SLAs and security. Concepts behind data center design and management, Economic and technological benefits of the cloud paradigm.	9

2	Unit 2: Cloud Platforms in Industry	9
	<p>Amazon Web Services: Compute Services, Storage Services, Communication Services.</p> <p>Google App Engine: Architecture, Core Concepts, Application Life Cycle, Cost Model, Observations.</p> <p>Microsoft Azure: Azure Core Concepts, SQL Azure, Azure Compute and Storage, Azure Database and Networking, Monitoring and Managing Azure Solutions.</p> <p>IBM Cloud (Kyndryl), Salesforce, Heroku, Alibaba Cloud, Oracle Cloud, Tencent Cloud, OVHcloud, DigitalOcean, and Linode (Akamai).</p> <p>Case study on available Cloud Platforms in Industry.</p>	
3	<p>Unit 3: Advanced Cloud Computing</p> <p>Energy Efficiency in Clouds, Green Cloud Computing Architecture, Market based Management of Clouds, Market-Oriented Cloud Computing, Reference Model for MOCC.</p> <p>Federated Clouds/Intercloud: Definition, Characterization, Cloud Federation Stack, Technologies for Cloud Federation.</p> <p>Third Party Cloud Services, MetaCDN, Spot Cloud, Cloud Authentication Protocols, Cloud Security Threats with Cloud Apps.</p> <p>Virtualized CPU, memory and I/O resources, network (SDN) and storage (SDS), Key role of virtualization to enable the cloud. Cloud storage concepts like data distribution, durability, consistency and redundancy.</p> <p>Case study on Advanced Cloud Computing services.</p>	9
4	<p>Unit 4: Cloud Management</p> <p>Fundamentals of Cloud Management, Management Services, Cloud properties, Multi-tier Application Deployment in Clouds, Challenges, Requirements, Service Level Agreements (SLAs), Billing & Accounting.</p> <p>Cloud Policy and Governance: Risk Management and Regulatory Practices.</p> <p>Cloud Analytics and Cost Metrics.</p> <p>Case study on Cloud Management Services, Distributed file systems, NoSQL databases, object storage using HDFS, CephFS, HBASE, MongoDB, Cassandra, DynamoDB, S3, and Swift.</p>	9
5	<p>Unit 5: Cloud Based Secured Applications Development</p> <p>Current trends in cloud computing i.e. IoT, Big Data, Machine Learning. Cloud Infrastructure Security, Network level security, Host level security, Application level security, Access management and control.</p> <p>MapReduce, Spark and GraphLab programming models, Develop and deploy the cloud based server-side application using Node.js and the front-end using React, Case Study on Open Source and Commercial Clouds applications: Amazon EC2, Amazon S3, Amazon Redshift, GitHub Repository, AWS IoT Core, AWS IoT Device Defender, AWS IoT Device Management, AWS IoT FleetWise, AWS IoT SiteWise, AWS IoT Events, AWS IoT TwinMaker, AWS IoT Analytics, Azure IoT Hub, Azure IoT Central, Azure Digital Twins, Azure IoT Edge, Azure Percept, Azure Sphere, and Azure RTOS.</p> <p>Design and Deploy a Restaurant Application to Cloud.</p>	9
	Total	45 Hrs.

11. Suggested Books:

SL. No.	Name of Authors/Books/Publishers	Edition	Year of Publication / Reprint
	Textbooks		
1.	Mastering Cloud Computing by Rajkumar Buyya, Vecchiola & Selvi (Published by McGraw Hill Education Pvt. Ltd),	1 st	2013.
2	Cloud Management & Security by Imad. M. Abbadi (WILEY Publication	3 rd	2014.
	Reference Books		
1.	Cloud Computing – A Hands-On Approach by Arshdeep Bahga, Vijay Madisetti.	1 st	2014

12.	Mode of Evaluation	Test / Quiz / Assignment / Mid Term Exam / End Term Exam
------------	---------------------------	--