Gujarat University

M.Sc. Computer Science EFFECTIVE FROM ACADEMIC YEAR - 2016-2017				
	SEM-I	**Th.**	**Tu.**	**Pr.**
MSC-401	Enterprise Data Management & ERP	4	1	-
MSC-402	Advanced Java Technology	4	-	3
MSC-403	Advanced Data Structure	4	-	3
MSC-404	Advanced RDBMS	4	-	3
MSC-405	Object Oriented Analysis Design & UML	4	-	-
MSC-406	Computer Science Practical (Based on Advanced Java Technology, Advanced RDBMS, Advanced Data Structure)			

Unit	Computer Theory MSC – 401 4 Credit Total 100 Marks Internal 30 Marks External 70 Marks 4 hrs/Week	Computer Theory MSC – 402 4 Credit Total 100 Marks Internal 30 Marks External 70 Marks 4 hrs/Week	Computer Theory MSC – 403 4 Credit Total 100 Marks Internal 30 Marks External 70 Marks 4 hrs/Week	Computer Theory MSC – 404 4 Credit Total 100 Marks Internal 30 Marks External 70 Marks 4 hrs/Week	Computer Theory MSC – 405 4 Credit Total 100 Marks Internal 30 Marks External 70 Marks 4 hrs/Week	Computer Practical COM – 406 4 Credit Total 100 Marks Internal 30 Marks External 70 Marks 4 hrs/Week
I	Data Management & Information System	Java Web Architecture	Introduction	Transaction Management	Graphs	Consist of Practical Experiments including Advanced Java Programming:24 Experiments Advanced Data Structure:35 Experiments Advanced RDBMS: 9 Exercises
II	Introduction to ERP	Java Database Programming	Modeling Concepts	Concurrency Control	Trees	
III	ERP Manufacturing Perspective	Java Servlets	Class Modeling	Recovery System	Hashing	
IV	Products & Modules	Java Server Pages	State Modeling	Storage & File Structure	Heaps and Garbage Collection	
V	Benefits of ERP	Enterprise Java beans and Java Application Framework	Interaction Modeling	Indexing & Hashing	Algorithmic Patterns & Problem Solves	
VI	ERP Implementation Life Cycle		Analysis & Design	SQL Concepts		
VII	Business Intelligence		System Design	PL/SQL Concepts		
VIII			Class Design			

Paper: 401 **Subject**: Enterprise Data Management & ERP

Unit-1: Data Management and Information System

- Data Management
  - ✓ Hierarchy of Data
  - ✓ Data Modelling
  - ✓ Data Integrity
  - ✓ Data Quality
  - ✓ Metadata
  - ✓ Legacy System and Data Migration
- Information System
  - ✓ Overview of Information System
  - ✓ Overview of different types of information Systems : MIS,DSS, GDSS,ESS , GIS KSS
  - ✓ Impact of Information System on an organisation
  - ✓ An Introduction to Electronic Commerce and Mobile Commerce
  - ✓ Threats and security to e-commerce and m-commerce

Unit-2: Introduction to ERP

- Evolution of ERP and Reasons for the growth of ERP
- Scenario and Justification of ERP in India
- Various Modules Of ERP
- Advantage of ERP
- ERP for Small Business
- ERP for make to order companies
- Business Process Mapping for ERP Module Design
- Hardware Environment and its Selection for ERP Implementation

Unit-3: ERP Manufacturing Perspective

- MRP - Material Requirement Planning
- BOM - Bill Of Material
- MRP - Manufacturing Resource Planning
- DRP - Distributed Requirement Planning
- PDM - Product Data Management

Unit-4: ERP Products and Modules

- Introduction to ERP Products and modules
- Finance
- Plant Maintenance
- Quality Management
- Materials Management

Unit-5: Benefits of ERP

- Reduction of Lead-Time
- On-time Shipment
- Reduction in Cycle Time
- Improved Resource Utilization
- Better Customer Satisfaction
- Improved Supplier Performance
- Increased Flexibility
- Reduced Quality Costs
- Improved Information Accuracy and Design-making Capability

Unit-6: ERP Implementation Lifecycle

- Pre-evaluation Screening
- Package Evaluation
- Project Planning Phase
- Gap Analysis
- Reengineering
- Configuration
- Implementation Team Training
- Testing
- Going Live
- End-user Training
- Post-implementation (Maintenance mode)

Unit-7: Business Intelligence

- Introduction to BI
- Types of Business Rule
- Implementing Business Rule
- Business Re-engineering
- Overview of Data Warehousing and Data Mining
- Business Intelligence using Data Warehousing and Data Mining
- Business Intelligence Applications: Customer Relationship Management, Supply Chain Management.

Reference Books:

1. Enterprise Resource Planning - Alexis Leon, Tata McGraw Hill.
2. Enterprise Resource Planning – Diversified by Alexis Leon, TMH.
3. Enterprise Resource Planning - Ravi Shankar & S. Jaiswal , Galgotia.
4. Principles of Information Systems Managerial Approach By Ralph stair and George Reynolds Thomson Course Technology
5. Management Information System by W.S Jawadekar by TMH
6. Management Information System Text & Application by C.V.S Murthy by Himalaya Publishing House
7. Guide to Planning ERP Application, Annetta Clewwto and Dane Franklin, McGRaw- Hill, 1997
8. ERP: Make it Happen By Thomas wakace, Willey Publication
9. ERP : Tools ,Technioques and applications for Integrating the Supply Chain Second Edition by Carl a Ptak, Schragenheim by Wiley
10. Management Information Systems Managing the Digital Firm by Kenneth Laudon and Jane Laudon by PHI

Paper: 402 **Subject**: Advanced Java Technology

Unit-1: Java Web Architecture

- The Java Advantage for Web
- Java EE Web Architecture
- Java Web Application Server

Unit-2: Java Database Programming

- The 2-Tier Client Server Architecture
- Java Dabase Connectivity (JDBC) – API for Accessing Databases
- Database Drivers, Loading a Driver Class
- Connecting the Database Server
- CRUD operations with Statement Object, PreparedStatement Object, callable statement object
- The ResultSet Object

Unit-3: Java Servlets

- Server side programming with Java Servlet,
- HTTP and Servlet,
- Servlet API, life cycle, configuration and context,
- Request and Response objects,
- Getting Values from Forms and QueryStrings,
- Working with Databases, Working with HTTP Headers ,
- Using Hidden Fields,
- Session handling and event handling,
- ServletContext and ServletConfig,
- Initalization Parameters, Inter-Servlet Communication with Request
- Introduction to filters with writing simple filter application

Unit-4: Java Server Pages

- Overview of Java Server Pages (JSP)
- JSP Architecture
- JSP page life cycle
- JSP elements
- JSP components
- JSP bean tags
- Working with databases
- Writing a complete application

Unit-5: Enterprise Java Beans and Java Application Framework

- What is a java bean?
- Advantages of Java Beans
- Stateless Session Bean , Statefull Session Bean,
- Binding and looking up objects
- Singleton Beans, Overview of Message Driven Beans
- Local and Remote Lookups,
- Asynchronous EJB Methods
- Web Services

- The Java Web Application Frameworks
  - ✓ Action Based Framework – Overview of SPRING
  - ✓ Component Based Framework - JAVA SERVER FACES

Reference Books:

1. Professional Java Server Programming, a! Apress
2. Core Servlets and Javaserver Pages: Author Marty Hall , Larry Brown , Sun Micro System
3. Java Servlet & JSP Cookbook by Bruce W. Perry O;reilly
4. Mastering Enterprise JavaBeans and the Java 2 Platform, Enterprise Edition, by Ed Roman
5. Core Java, Volume II – Advanced Features, Eight Edition, Pearson
6. Unleashed Java 2 Platform, Sams Techmedia
7. Advanced Java , jambu Krishnamurthi, Comp-U Learn Inc
8. Mastering Enterprise Java Beans 3.0 , Rima patel , Wiely Publication
9. Java Server Pages for Beginners, Bayross and Shah, SPD
10. Java Servlet Programming, Jason Hunter, SPD (O'Reilly)

Paper: 403 **Subject**: Object Oriented Analysis Design & UML

Unit-1: Introduction

- About Object Orientated Technology,
- Development andOO Modeling History.

Unit-2: Modeling Concepts

- Modeling design Technique,
- Three models, Class Model, State model and Interaction model.

Unit-3: Class Modeling

- Object and class concepts, link and association
- Generalization and Inheritance
- Advanced class modeling- aggregation, Abstract class metadata, constraints.

Unit-4: State Modeling

- Event, state, Transition and conditions,
- state diagram, state diagram behavior,
- concurrency,
- Relation of Class and State models.

Unit-5: Interaction Modeling

- Use case Models,
- sequence models,
- activity models

Unit-6: Analysis and Design

- Development Life cycle, Development stages,
- Domain Analysis-Domain class model, domain state model, domain interaction model,
- Iterating and analysis.
- Application Interaction model, Application class model, Application state Model

Unit-7: System Design

- Estimating Performance, Making a reuse plan, breaking system into subsystem, identifying concurrency, allocation of subsystems,
- management of data storage, Handling Global resources,
- choosing a software control strategy, Handling boundary condition,
- Common Architectural style.

Unit-8: Class Design

- Overview of class design ,
- Designing algorithms,
- Refactoring, design optimization, Adjustment of Inheritance, Reification of Behavior.

Reference Books:

1. Oriented Modeling and Design wih UML ,second edition by michael Blaha and James Rambaugh
2. Object Oriented Modeling and Design, J. Rambaugh, M. Blaha et al, William Premerlani, Fredrick Eddy, William Lorensen, PHI
3. Applying UML & Patterns: An Introduction to Object Oriented Analysis and Design, Larman, Pearson Education.

Paper: 404 **Subject**: Advanced Relational Database Management System

Unit-1: Transaction Management

- Transaction concepts, properties of transactions,
- Serializability of transactions, testing for Serializability,
- System recovery, Two- Phase Commit protocol,
- Recovery and Atomicity, Log-based recovery,
- Concurrent executions of transactions and related problems,

Unit-2: Concurrency Control

- Locking mechanism,
- solution to concurrency related problems,
- Multiple Granularity
- Deadlock Handling

Unit-3: Recovery System

- Failure Classification
- Recovery and Atomicity
- Logbased Recovery
- Buffer Management
- Deferred Database Modification
- Immediate Database Modification
- Checkpoints
- Shadow Paging

Unit-4: Storage and File Structure

- RAID
- Storage Access
- File Organization
- Organization of Records in Files

Unit-5: Indexing & Hashing

- Basic Concepts
- Ordered Indices
- B+-Tree Index Files
- B-Tree Index Files
- Overview of Static & Dynamic Hashing
- Comparison of Ordered Indexing & Hashing

Unit-6: SQL Concepts

- Basics of SQL, DDL,DML,DCL, structure – creation, alteration,
- Defining constraints – Primary key, foreign key, unique, notnull, check, IN operator,
- aggregate functions, Built-in functions –numeric, date, string functions, set operations
- Sub-queries, correlated sub-queries, join, Exist, Any, All , view and its types.,
- Transaction control commands.

Unit-7: PL/SQL Concepts

- Cursors,
- Stored Procedures,
- Stored Function,
- Database Triggers

Reference Books:

1. An introduction to Database Systems, C J Date, Addition-Wesley.
2. Database System Concepts, Abraham Silberschatz, Henry F. Korth & S. Sudarshan, McGraw Hill.
3. Understanding SQL by Martin Gruber, BPB
4. SQL- PL/SQL by Ivan bayross

Paper: 405 **Subject**: Advanced Data Structure

Unit-1: Graphs

- Basics
- Traversals and search – Depth-First, Breadth-First, Branch and Bound
- Applications- Topological Sort
- Shortest Path Algorithm
- Minimum Cost Spanning trees – Prim's and Kruskal's algorithm
- Dijkstra's algorithm
- Critical Path Analysis

Unit-2: Trees

- Basic Concepts
- Binary Trees
- N-ary Trees
- Tree Traversals
- Search Trees
- Algorithms like Binary, AVL, B-Tree, B+ Tree
- Huffman trees and Data compression including Huffman coding

Unit-3: Hashing

- Basic Idea – Keys and Hash Functions including Collision avoidance
- Hashing Methods
- Division Method
  - ✓ Division Method
  - ✓ Middle Square Method
  - ✓ Multiplication Method
- Fibonacci Hashing
- Hash Function Implementations
  - ✓ Integral Keys
  - ✓ Floating Point Keys
  - ✓ Character String Keys
  - ✓ Hashing Containers
  - ✓ Using Associations
- Abstract Hash Tables

Unit-4: Heaps and Garbage Collection

- Basic Concepts
- Binary, Leftist, Binomial Queues
- Recent Applications
- Basic concepts of Garbage Collection
- Reference Counting Garbage Collections
- Mark-and-Sweep Garbage Collections
- Stop-and-Copy Garbage Collections
- Mark-and-Compact Garbage Collections

Unit-5: Algorithmic Patterns and Problem Solves

- Brute-Force and Greedy Algorithms
- Divide and Conquer Algorithm
- Backtracking Algorithm- Depth-First, Branch & Bound
- Bottom-Up Algorithms

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, PHI.
2. Data Structures & Algorithms , Alfred V.Aho., Jeffery D. Ullman, Addison-Wesley, Oracle Press
3. An Introduction to Data Structures with Applications. by Jean-Paul Tremblay & Paul G. Sorenson Publisher-Tata McGraw Hill.

Paper:406 **Subject:** Advanced Relational Database Management System

Assignment 1

Table Name : Account

Field Name	Type	Size	Description
Acc_no	Number	3	Account Number
Acc_name	Character	30	Account Holder Name
Acc_dob	Date		Account Holder Birth Date
Acc_op_date	Date		Account Opening Date
Acc_balance	Number	11,2	Account Current Balance
Acc_type	Character	1	Account Type [Saving[S]/Current [C]]
Acc_status	Character	1	Account Status [Open[O] /Close [C]]
Acc_chk_flag	Number	1	Account Check Book Flag [1-Yes, 0-No]
Acc_gender	Character	1	Account holder gender [F- Female , M-Male]

Enter Given data into Account :

Acc_no	Name	Dob	Op_date	Balance	Type	Status	Chk flag	gender
1	Sachin	10-May-1975	10-May-1996	5,30,000	S	O	1	M
2	Dravid	15-Jan-1970	25-Jan-1999	1,49,333	S	O	1	M
3	Sonali	12-Dec-1969	20-Dec-1995	1,30,30,333	C	O	1	F
4	Rani	18-Feb-1978	23-Jan-1999	4,000	C	O	1	F
5	Ashwariya	15-Jan-1975	30-Dec-2000	-3,000	S	C	1	F
6	Kajol	12-Dec-1979	28-Jan-2001	2,00,000	C	C	0	F
7	Kapil	12-Dec-1950	19-Jan-2003	80,000,00	S	O	1	M
8	Dhoni	29-Jan-1985	20-Dec-2005	80,000	S	O	1	M
9	Pathan	23-Oct-1975	21-Dec-2004	45,000	C	C	0	M
10	Stafi	28-Mar-1974	21-Jan-2003	3,00,000,00	S	O	1	F

*******Check the Query For given Table Structure Data.*******

1 **Write command to create above table**
2 **Insert 10 relevant records**
3 **Display all information of all account holders**
4 **Display Name and Birth Date of all account holders.**
5 **Display Name and Balance of all account holders.**
6 **Display Name and Balance of current account holder**
7 **Display all information of Close account holder in ascending order of Name.**
8 **Display Name of account holder who has a check Book.**
9 **Display information of female account holder.**
10 **Display information of female open before 1999.**
11 **Display name of all account holders in upper case.**
12 **Display name of all account holders in proper case and in descending order of birth date.**
13 **Display name of account holders who has open his account in 'January'**
14 **Display all information of all current account holders in ascending order of balance.**
15 **Display information of account holder whose balance is negative**
16 **Display Total Number of Male and Female Account holder**
17 **Display Total Current Account**
18 **Display Total Male Current Account**
19 **Display Name and Birth year of account holder who has a check book.**
20 **Display Name of account holder and status whose birth in January**
21 **Display Name of account holder whose balance is less than 50,000.**
22 **Display Name of account holder and age as on today.**
23 **Display name information of account holder whose name is started with 'S'**
24 **Display information of male current account holder whose age is greater than 25.**
25 **Display total number account holder.**

26 Display the name of account holder whose balance is maximum
27 Display the name of account holder
28 Display the name of account holder whose balance is greater than 50,000 and less than 1,00,000.
29 Display close account information.
30 Display information of account open in December
31 Display Maximum balance amount.
32 Display Minimum balance amount.
33 Display First Three Character of Name
34 Display Last three Character of Name
35 Display the Account holder information whose name is greater than 5 character.
36 Display the Account holder information whose name contains 'D' character.
37 Change the name of table account to acc_master.
38 Add new column address and city to acc_master table.
39 Update the value city = 'Bhavnagar' for all member.
40 Update the related address for account holder.
41 Delete the information of Dhoni
42 Update the balance of all account holders with (balance+ 5%interest).
43 Delete all close account.
45 Insert the information of two new account holder (Johan and Ronaldo).
46 Display the information in order of balance
47 Display the information where age is 25 as on today.
48 Display the information of male, open, saving account.
49 Deduct the balance Rs. 100 for current account for balance greater than 50,000.
50 Deduct the balance Rs. 200 check remuneration for account holder who has check book.

Assignment 2

Create given table and give the sql statement for the following. (select with from, distinct, where, and, or, between, not, in, not in, like clauses, order by clause and delete).

EMPLOYEE:

Column Name	Data Type	Size
emp_id	number	5
emp_name	varchar2	10
emp_street	varchar2	10
emp_city	varchar2	10
emp_desg	varchar2	10
emp_sal	number	8,2
emp_date_join	date	

Note: Insert values according to queries.

1. **Display all information of all employees without duplicate.**
2. **Display emp_id, emp_name whose designation is manager.**
3. **Display emp_id, emp_name whose salary same or greater than 10,000.**
4. **Display all information of all employees whose joining date is "4th July 1990".**
5. **Display employee information with designation is clerk with salary grater than 10,000.**
6. **Display all information of all employees whose id is 101 or 102.**
7. **Display emp_id, emp_name whose salary between 20,000 and 35,000.**
8. **Display all information of all employees who live in 'bombay', 'delhi','surat' or 'ahmedabad'.**
9. **Display all information of all employees who are not live in 'surat', 'jamnagar' or 'ahmedabad'.**
10. **Display all information of all employees in descending order of their names.**
11. **Display all information of all employees in ascending order of their id.**
12. **Display name of all employee starting with 'a'.**
13. **Display name of all employee whose 3rd char is's' or 'm'.**
14. **Display name of all employee whose name is exactly 3 char long.**
15. **Display name of all employee containing 'am' as substring.**
16. **Display emp_id, emp_name, emp_address (in emp_address both emp_street, emp_city must be there).**

Ex: emp_id emp_name emp_address
1 xyz vakilvadi, Ahmedabad

17. **Delete employee having id 2.**

Assignment 3

Issue Date: 07/09/2012

Create given table and give the sql statement for the following. (select with from, distinct, where, and, or, between, not, in, not in, like clauses, order by clause and delete).

HOTEL (HNO, NAME (not null), ADDRESS, TOTAL_ROOM)
ROOM (HNO, RNO, RTYPE (not null), LOCATION)
CHARGES (HNO, RTYPE, CHARGES)

1. **Create tables using the above schema along with necessary constraints (Primary OR Composite key, foreign key, not null, Unique constraints).**

2. **Insert four necessary records in each table.**
3. **Add a column to the ROOM table, which allow us to store STATUS whether the room is occupied or vacant.**
4. **Add a check constraint to the room table so that the room type allows the following values only – 's' for single,'d' for double-seater.**
5. **Sort all hotels in descending order by their address.**
6. **Display the total number of rooms that are vacant presently.**
7. **Display the hotel name and address having total rooms > 50.**
8. **Display the hotel name having the greatest charges on double-seater room.**
9. **Display hotels, which are totally occupied to its fullest capacity.**
10. **Create a simple view with HOTEL names and their ADDRESS only.**

Assignment 4

Issue Date: 07/09/2012

Create given table and give the sql statement for the following. (select with from, distinct, where, and, or, between, not, in, not in, like clauses, order by clause and delete).

ITEM (ITEMNO, ITEMNAME (not null), ITEM_PRICE, QTY_ON_HAND)
INVOICE (INVNO, INVDATE)
INV_ITEM (INVNO, ITEMNO, QTY)

1. **Create tables using the above table schema along with necessary constraints (Primary OR Composite key, foreign key, not null, Unique constraints).**

2. **Insert four necessary records in each table.**
3. **Add a check constraint to the ITEM table so that item price allow greater than 0 values only.**
4. **Display items with price of at least Rs. 100.**
5. **Display invoices dates in 'February 02, 2010' format.**
6. **Find average, highest and lowest item price.**
7. **Find invoices with 'key board' in their item name.**
8. **Find the items that are cheaper than motherboard.**
9. **Find the items with top three prices.**
10. **Display all item quantity and item price for invoices.**

Assignment: 5

Write the sql statement for the following.

1.Create table emp1 having fields named id, name, city, salary from the source table named employee with data.

2.Create table emp2 having fields named id, name, city, salary from the source table named employee with only structure.

3.Insert data into emp2 table from employee table whose emp_id is 101.

4.Update the name of employee having id 103 to mitesh.

5.Enter a new field called branch (number (10)) in the emp2 table.

6.Drop the column city from the emp2 table.

7.Change the data type of field branch of emp2 table to varchar2.

8.Insert following records into the emp2 table.

ID	NAME	SALARY	BRANCH
102	Chhaya	20000	Maninagar
103	Ivan	16000	Vadaj
104	Mamta	20000	Maninagar
105	Anil	35000	Vadaj
106	Mehul	12000	Vasna
107	Ishan	23000	Maninagar

9.Find out number of records (No. Of Record) in epm2 table.

10. Find out number of branches appearing in emp2 table.

11. Find out number of employee working in Vadaj branch.

12. Find out sum of salary of all employees (change salary column name to total_salary in output).

13. Find out average of salary of all employees (avg_sal_allemp).

14. Find out sum of salary of all employees at each branch.

15. Find out sum of salary of all employees (sum_salary) and maximum salary (max_salary) at each branch.

16.Find out average of salary of all employees (avg_salary) and minimum salary (min_salary) at each branch.

17. Find out how many employees are there in each branch.

18. Find out branch name(s) having more than one employee.

19. Change table name from emp2 to employee2.

Assignment-6

Create the following tables with given constraints.

Table1: salespeople

Column Name	Data Type	Size	Default	Attribute
Snum	varchar2	5		Primary key, start with 's'
sname	varchar2	10		Not null
City	varchar2	12	ahmedabad	
Comm.	number	5,2		Must be >0

Table2: customer

Column Name	Data Type	Size	Default	Attribute
Cnum	varchar2	5		Primary key, start with 'c'
cname	varchar2	10		Not null
City	varchar2	12	surat	
Rating	number	5		Can not be 0
Snum	varchar2	5		Foreign Key references snum of salespeople table

Table3: order

Column Name	Data Type	Size	Default	Attribute
Onum	varchar2	5		Primary key, start with 'o'
amount	number	9,3		
Odate	date			
Cnum	varchar2	5		Foreign Key references cnum of customer table
Snum	varchar2	5		Foreign Key references snum of salespeople table

1. **Find all customers with orders on oct 3rd**
2. **Find names and numbers of all customers with rating equal to the maximum for their city.**
3. **Find salespeople number, name and city that have multiple customers.**
4. **Find salespeople who serve only one customer.**
5. **Extract rows of all salespeople with more than one current order.**
6. **Find salespeople with customers located in their cities.**
7. **Delete all sales persons who have at least one customer with a rating of 100 from salespeople table.**
8. **Find the lowest order for each day and delete the sales persons who product it from salespeople's table.**
9. **Find the lowest order for each day and delete the sales persons who product it from salespeople's table but Peel should not be deleted.**
10. **Increase the commission by 20% of all salespeople with total current orders above $3000.**

Write the answer for the following (based on view).

1. **How to delete duplicate row from table. Give example.**
2. **Create view called V_salespeople containing columns sname, city.**
3. **Insert data into V_salespeople view.**
4. **Create view called V_salespeoplepn containing columns snum, sname, city.**
5. **Can we insert, update or delete record from V-salespeoplepn view? Why?**

Assignment-7

Flight(flightId, company_name, flightFrom, flightTo, flightFare, capacity)
Passenger(pId, Name,Address, City, BirthDate, Gender, ContactNo)
Flight_Scheduled(Transid, flightid, departuredate)
Flight_Passenger(Transid, pId)

Passenger Id must start with 'P'.
Flightfare cannot be NULL.

1. **Display all the flight details which are flying from 12-Jun-2012 to 15-Jun-2012**
2. **Display all Air India flights which flied carrying more than 30 passengers.**
3. **Display total males and females travelling in flightid 101 on 12th June 2012.**
4. **Display all the passengers with starting with name 'm' and flying to Mumbai.**
5. **List all the Flights having the same company.**
6. **Change the Flight Date with is flying from Ahmedabad to Mumbai.**
8. **Find the age of all passengers.**
9. **Find the number of male and female passengers.**
10. **Display the flight details that are not flying today.**

Assignment-8

Airlines(Aid, Aname,Total_dom, Total_international)
Flight(flightId, company_name, flightFrom, flightTo, flightFare, capacity, Aid)
Passenger(pId, Name,Address, City, BirthDate, Gender, ContactNo)
Flight_Scheduled(Transid, flightid, departuredate)
Flight_Passenger(Transid, pId)

Apply the following Constraints.
1. **Passenger Id must start with 'P'.**
2. **Flightfare cannot be NULL.**

Implement the following SQL Queries.
1. **Display all the flight details which are flying from 12-Jun-2012 to 15-Jun-2012.**
2. **Display all Air India flights which flied carrying more than 30 passengers.**
3. **Display total males and females travelling in flightid 101 on 12th June 2012.**

Create following PL/SQL Blocks.
1. **Create a procedure to display list of passengers whose residence city is always the place from which he/she has boarded flight. For example, A person lives in Mumbai and till today he has travelled only on flights flying from Mumbai to other cities.**

2. **Write the trigger that checks that flight should not carry passengers more than its capacity.**

Assignment-9

Train(TrainId, Train_name, TrainFrom, TrainTo, departuretime, arrivaltime)
Train_fare(Trainid, class, fare)
Passenger(pId, Name,Address,City, BirthDate, ContactNo, email_id)
Train_Passenger(TrainId, pId)
Apply the following Constraints.
1. **Passenger Id must start with 'P' and Train Fare should be Default 0.**
2. **Train Class can be either 1AC,2AC,3AC,Sleeper, Gene**ral.

Implement the following SQL Queries.
1. **Display all the Train details which are travelling from 12-Jun-2012 to 15-Jun-2012.**
2. **Display all Train details which are not travelling currently.**
3. **Display all the passenger details which are travelling from Delhi to Mumbai.**

Create following PL/SQL Blocks.
1. Create a SQL/PLSQL Block that displays all the passenger details of a particular train. Display in proper format:
PassengerName TrainName TrainFrom TrainTo TrainFare
Naman RAJDHANI Delhi Mumbai 4000
TOTAL FARE: 4000
2. Write the trigger that keeps a track of birth date of every passenger. Whenever a passenger record is inserted and if the birth month is the current month then message should be displayed that 'Naman's birthday is in current Month' and if the birth date is current date than message should be displayed that 'Happy Birthday Naman . You are 22 years old'. Note: Also calculate the age of the passenger and then display it.

PL/SQL Programs

1. Write a PL/SQL block to find the given numbers is ODD or EVEN.
2. Write a PL/SQL block to find the factorial of given number.
3. Write a PL/SQL block to find the maximum number from given three numbers.
4. Write a PL/SQL block to find the sum of first 100 natural Nos.
5. Write a PL/SQL block to display the pyramid using any character.
6. Make a block to insert the row in table. (CREATE student TABLE with all possible fields.)
7. Write a PL/SQL block to display the information of given student on following table. (using variable) Stud(sno,sname,address,city)
8. Write a PL/SQL block to display the information of given student on following table. (using record type variable)Student(sno,sname,address,city).
9. Create a block which delete the records of student whose total marks is less than 200 out of 500. Deletion is done after confirmation from user.
10 Retrieve the details of the employees on the basis of the department name. Write exception handler if no row are retrieves; otherwise display the employee getting the maximum salary.
11 Give an increment of 15% to those who have joined in any year but the month of current date.
12 Write PL/SQL block that find the area of patient 'Sector 15'. If the patient count < 5 in the area the exception is raise.
13 Write a PL/SQL block for preparing a Net salary of given employee on following table.

EMP(eno,ename,address,city) SALARY(eno,basic,da,hra,it)
NET_SALARY(eno,total_allowance,total_deduction,netpay) Notes : DA = 59% of basic, H.R.A. = 500, I.T. = 2% of basic

a. Total_Allowance = basic + D.A. + H.R.A. b. Total_Deduction = I.T.
c. Netpay = Total_Allowance – Total_Deduction

Procedure Programs

1. Write a procedure to check the given year is leap or not.
2. Write a procedure to display the following type of Multiplication Table as per given number.

   5 * 1 = 5
   5 * 2 = 10 “ “ = “ ““
   = “
   5 * 10 = 50
3. Write a procedure to display this kind of output on screen.

   1
   2 3
   3 4 5
   4 5 6 7
   5 6 7 8 9
   . . . . . 90 91
4. Write a procedure to convert given octal number to decimal number.
5. Write a procedure that take Employee Number and return all the information related to the employee. Display the following format using table – Emp(eno,ename,city,salary)

Employee Number	Employee Name	City	Salary

6. Create a procedure to generate an Invoice with the following information. Invoice number, customer name, qty, amount, total bill and date of invoice etc.
7. Create a procedure to convert the entire invoice from unpaid to paid for given customer no. For a given customer check his total pending amount if it equal to the amount if it equal to the amount given in the parameter. Convert the unpaid to pay the parameter also generate the error message.
8. Examine the following rule in each sub:
   - ™ Passing marks in each subject = 40%
   - ™ Fail in 1 sub = Grace
   - ™ Fail in 2 sub = ATKT
   - ™ Fail in >= 3 sub = Fail
   - ™ If pass in all subject then
     - 9 >=70% = Division
     - 9 >=60 and >70 =”First”
     - 9 <60 = “Second”

   Write a PL/SQL procedure to find out the status and update the field status.

Function Programs

1. Write a function to check whether the given number is prime or not.
2. Write a function to find the sum of digits of accepted no.
3. Write a function to display first 25 Fibonacci nos.
4. Write a function to display the reverse string of a given string.
5. Write a function that take Employee Number and return the salary on following table. Emp(eno,ename,city,salary)
6. Write a function to count the total number of student having ‘PASS’ on following table. Student(sno,sname,grade)
7. Write a function to assign the grade to all the student using following table. Stud(sno,sname,sub1,sub2,sub3,grade) Note :
   i. if percentage >= 70 then ‘Distinction’ else ii. if percentage >= 60 then ‘First’ else
   iii. if percentage >= 50 then ‘Second’ else iv. otherwise ‘Fail’.

Package Programs

1. Write a package that has a function that check the given string is palindrome or not.
2. Write a package that has a procedure to find 1 + 1/2 + 1/3 + …….. + 1/n.
3. Write a package that has a function to check given number is not negative and a procedure to convert the given number into word. For example 25 = Twenty Five.
4. Write a package that has two object on following table.
   I. The function is used to calculate the Net Salary.
   II. The procedure is used to display the Pay Slip in following format.
   Emp(eno,ename,basic)
   Salary(eno,da,hra,it,gross_sal,net_sal)
   Notes : D.A. = 59% of basic, H.R.A. = 500, I.T. = 2% of basic
   Gross_salary = Basic + D.A. + H.R.A. , Net_salary = Gross_salary – I.T.
5. Create a package,which contains following functions and procedure:
   - ™ Function, which return the department having maximum employees.
   - ™ Procedure, which gives the details of managers of each department.
   - ™ Procedure, which prepare the report of employee overtime when employee id is passed.
6. Create one package which stores the procedures of insert, delete & display of the following table. Write one function which return the Names of employee whose birth date is between the two passing date.

   Table : employee master

   ===== ==========

   Empcd, Emp name, Emp address, Birth date, Salary

Trigger Programs

1. Write a trigger on insert to convert the name into capital letters.
2. Write a trigger to check the pin code is exactly six digits or not.
3. Write a trigger to check the mark is not zero or negative.
4. Write a trigger that check the student_id must be start with ‘M’.
5. Create trigger that is fired before the DML statement’s execution on the employee table. The trigger checks the day based on sysdate. If the day is Sunday, the trigger does not allow the DML statement’s execution and raises an exception. Write appropriate message in the exception section.
6. Write a trigger that is fired after an insert statement is executed for the student table.

The trigger writes the new student’s ID, name and sysdate in a new table student_info.

Paper:406 **Subject:** Advanced Data Structure

Experiment 1(Graph)

1. WAP to perform topological sort on dag using depth first search.
2. WAP to generate minimum spanning tree in a connected, undirected weighted graph using Kuruskal's algorithm with disjoint set data structures.
3. WAP to generate minimum spanning tree in a connected, undirected weighted graph using Prims's algorithm with disjoint set data structures.
4. WAP to find single-source shortest path in a weighted dag using topological sort
5. WAP to implement Dijkstra's algorithm for single-source shortest path in a weighted directed graph using fibonacci heap
6. Implement DFS & BFS search and traversal on a graph

Experiment 2(Tree)

1. WAP for Binary Search Tree to implement following operations:
   a. Insertion
   b. Deletion
      i. Delete node with only child
      ii. Delete node with both children
   c. Finding an element
   d. Finding Min element
   e. Finding Max element
   f. Left child of the given node
   g. Right child of the given node
   h. Finding the number of nodes, leaves nodes, full nodes, ancestors, descendants.

2. WAP for AVL Tree to implement following operations: (For nodes as integers)
   I. Insertion: Test program for all cases (LL, RR, RL,

LR rotation)

II. Deletion: Test Program for all cases (R0, R1, R-1, L0, L1, L-1)

III. Display: using set notation

3. WAP using function which computes the balance factor of any given node in a BST
4. WAP to transform BST into AVL trees and also count the number rotations performed.
5. WAP to find whether the given BST is AVL tree or not.
6. WAP to implement insertion, deletion, display and search operation in m-way B tree (i.e. a non-leaf node can have atmost m children) for the given data as integers (Test the program for m=3, 5, 7).
7. WAP to implement insertion, deletion, display and search operation in m-way B tree (i.e. a non-leaf node can have atmost m children) for the given data as strings (Test the program for m=3, 5, 7).
8. WAP to implement insertion, deletion, display and search operation in m-way B tree (i.e. a non-leaf node can have atmost m children) for the given data as Student structures (as given above), with key as student_ roll_no . (Test the program for m=3, 5, 7)
9. WAP to implement insertion, deletion, display and search operation in m-way B tree (i.e. a non-leaf node can have atmost m children) for the given data as Faculty structures (as given above), with key as faculty_ID. (Test the program for m=3, 5, 7).
10. Write a program to implement Huffman coding for the sentence inputted by the user

Experiment 3(Hashing)

1. WAP to implement Hashing methods on an inputted string
2. WAP to store k keys into an array of size n at the location computed using a hash function, loc = key % n, where k<=n and k takes values from [1 to m], m>n. To handle the collisions use the following collision resolution techniques,
    a. Linear probing
    b. Quadratic probing
    c. Random probing
    d. Double hashing/rehashing
    e. Chaining
3. Implement the above program I using hash function from Division methods
4. Implement the above program I using hash function from Truncation methods.
5. Implement the above program I using hash function from Folding methods.
6. Implement the above program I using hash function from Digit analysis methods.

Experiment 4(Heaps)

1. WAP to implement insertion, deletion and display operation in Min-Max Heap for the given data as integers
2. WAP to implement Make_Heap, Insertion, Find_Min, Extract_Min, Union, Decrease_Key and Delete_Key operations in Binomial Heap for the given data as strings
3. WAP to implement Make_Heap, Insertion, Find_Min, Extract_Min, Union, Decrease_Key and Delete_Key operations in Fibonacci Heap for the given data as Student structures (contains student_name, student_roll_no, total_marks), with key as student_ roll_no.
4. Implement the above program (I) of Min-Max heap for Employee structures (contains employee_name, emp_no, emp_salary), with key as emp_no.
5. Implement the above program (II) of Binomial Heap for Faculty structures (contains faculty_name, faculty_ID, subject_codes, class_names), with key as faculty_ID.
6. Implement the above program (III) of Fibonacci Heap for strings.

Experiment 5(Algorithmic Patterns)

1. Write a program to demonstrate Brute Force algorithm
2. Write a program to demonstrate Greedy algorithm
3. Write a program to demonstrate divide & conquer algorithm by using the quick sort algorithm
4. Write a program for coin change backtracking solution using recursive function.
5. Write a program for coin change backtracking solution using recursive backtracking function.
6. Write a program for Generalized Fibonacci Numbers to fulfill Bottom up Method
7. Write a program to demonstrate Bottom-up algorithm

Paper:406 **Subject:** Advanced Java Technology

Experiment 1(Java Database Connectivity)

1. Write a database application that uses any JDBC driver.
2. Write a java program to perform JDBC Operation like insert, Update, Delete and view the student's records.

Experiment 2(Applet & Swing)

1. Write an applet that draws a circle. The dimension of the applet should be 500*300 pixels. The circle should be centered in the applet and have a radius of 100 pixels. Display your name centered in a circle. (using DrwaOval() method)
2. Write a program in java for implementing calculator in an applet.
3. Draw ten red circles in a vertical column in the center of the applet.
4. Write an applet that contains three check boxes and 30*30 pixel canvas. The three checkboxes should be labeled "Red", "Green", "Blue". The selection of the checkboxes determines the color of the canvas. For example, if the user selects both "Red" and "Blue", the canvas should be purple.
5. Write a program to show the features of swing component.
6. Develop and application/applet with a Menu File and two menu items color and font. The submenu of the menu item color will contain different colors which when selected should change the background of the applet. The submenu of the menu item font

should contain the list of fonts. Create a Text Field in the center of the container. When the font is selected from the font list of menu, the Text Field text should be appeared in that font.

7. Create a split pane which divides the frame into two parts. The first part possesses a list and on selecting an item in a list, the item should be displayed in the other portion
8. Swing example login form validation
9. Create a client/server application where the client requests for a particular file on the server. If the file exists on the server, then write the contents of the file to the client.
10. Write a program to present a set of choice for user to select stationary products and display the price of products after selection from the list.

Experiment 3(Java Servlets)

1. Write a simple servlet that just generates plain text
2. Write a program which displays a cookie id
3. Write a program of calling one servlet by another servlet
4. Write a program to show validation of user using servlet
5. Write a program for implementing calculator using servlet.

Experiment 4(Java Server Pages)

1. Write a program to insert data into table using JSP
2. Write a program to show validation of user using JSP
3. Write a program to display message on browser using JSP
4. Develop a JSP application to authenticate user login as per registration details. If login succeeds forward user to index page otherwise show login failure message.

Experiment 5(Java Beans)

1. Write a program to demonstrate use of beans
2. Write a program in java implementing Java Beans
3. Develop a room reservation system application using enterprise Java Beans