MAHARSHIDAYANANDSARASWATI UNIVERSITYAJMER

DEPARTMENT OF COMPUTER SCIENCE

Syllabus

Bachelor of Computer Applications (BCA)

BCA Part-I (Session- 2023-24)
Semester- I & II

MaharshiDayanandSaraswatiUniversity
Ajmer

TEACHING AND EXAMINATION SCHEME

Bachelor of Computer Applications

W.E.F. 2023-2024 (CBCS)

Semester I

Category	Type	Code	Paper Name (Theory)	Lec	Tut	Max Marks		Credits (L+T)
						Sessional	Semester	
-	AEC	BCA-101	English / Hindi /Rajasthani *	1		30	70	**2**
CC	DCC	BCA-102	Computer Architecture	3	1	30	70	**6**
CC	DCC	BCA-103	C Programming	3	1	30	70	**4**
CC	DCC	BCA-104	Operating Systems	3	1	30	70	**4**

Category	Type	Code	Paper Name (Practical)	Prac Hrs	Max Marks	Credits (P)
AE	SEC	BCA-105	Lab-C Programming	3	50	**2**
AE	SEC	BCA-106	Lab-Operating Systems (Linux)	3	50	**2**

Total of Theory & Practical Marks& Credits	**500**	**20**

* Any one of the three options

Semester II

Category	Type	Code	Paper Name (Theory)	Lec	Tut	Max Marks		Credits (L+T)
						Sessional	Semester	
-	AEC	BCA-201	English Communication Skill/ Hindi Communication Skill / Rajasthani Communication Skill*	1		30	70	**2**
CC	DCC	BCA-202	Discrete Mathematics	3	1	30	70	**6**
CC	DCC	BCA-203	Data Structure Using C++	3	1	30	70	**4**
CC	DCC	BCA-204	Database Technology	3	1	30	70	**4**

Category	Type	Code	Paper Name (Practical)	Prac Hrs	Max Marks	Credits (P)
AE	SEC	BCA-205	Lab-Data Structure Using C++	3	50	**2**
AE	SEC	BCA-206	Lab-Database Technology (MySQL)	3	50	**2**

Total of Theory & Practical Marks& Credits	**500**	**20**

* Any one of the three options

Scheme of Examination
(For B. C. A. – Semester Scheme)

Theory:

Part A:

1. 10 Question of 2 mark each – 20 marks
2. Answer should not exceed more than 50 words
3. All questions are compulsory

Part B:

1. 10 Questions of 10 marks each – 50 marks
2. Answer should not exceed more than 400 words
3. at least three question from each unit be set and student will have to answer five question, selecting atleast one question from each unit

Sessional:

There will be sessional (internal assessment) of 30 marks conducted by the department.

Practical:

Practical exams shall be conducted by one internal and one external examiner of a batch of 20 students in a day.

Duration of Practical exam is 3 hours.

A Laboratory Exercise File should be prepared by each student for each practical paper and should be submitted during practical examinations.

Practical of 50 marks distribution is as under:

a. 30 marks for practical examination exercise for 3 questions
b. 10 marks for Viva-voce
c. 10 marks for Laboratory Exercise File

Eligibility:

10+2 with 50% marks in aggregate.

Scheme of Examination
(For B. C. A.)

Reg. 17 (b)
The examination shall consist of (a) Theory papers (b) Laboratory / Practical work (c) dissertation. Candidates will be required to pursue a regular, full time course of study at the University department in order to be eligible for appearing in the examination.

1. Eligibility for BCA is 10+2 with 50% marks in aggregate.
2. Examination:
    i. There shall be36 papers (24 theory including, 12 practical papers). Theory paper shall be of 100 marks of which 70 marks will be of examination of 3 hours duration, at the end of each semester as determined by the University and 30 marks shall be of internal assessment based on internal test and seminars. The practical shall be of 50 marks assessed by external examiner includinginternship of 50 marks, based on presentation and viva-voce.
    ii. To pass a semester a candidate shall have to score 36% marks in end-term examination, internal assessment and practical separately and also 40% marks in aggregate of all the papers prescribed for the examination.
    iii. Due paper(s) will be applicable if a candidate obtains 40% marks in aggregate and fails in not more than two (2) theory papers. The due paper(s) of I semester will be held along with the III semester, the due paper(s) of III semester will be held along with V semester, the due paper(s) of II semester will be held along with the IV semester and due paper(s) of the IV semester will be held along with the VI semester, the due paper(s) of V and VI semester will be held in the I and II semester respectively of the next year. The number of chances for the due papers will be as per university norms.
    iv. Wherever a candidate appears at for a due paper examination he/she will do so according to the syllabus in force.
    v. A candidate not appearing at any examination/absent in any paper of term end examination shall be deemed as fail.
3. A candidate for a pass in the examination shall be required to obtain:
    i. At least 40% marks in the aggregate of all the papers prescribed for the examination and
    ii. At least 40% marks in the practical(s) wherever prescribed at the examination, provided that if a candidate fails to secure at least 36% marks in each individual paper at the examination notwithstanding his/her having obtained the minimum percentage of marks required in the aggregate for that examination.

    The division will be awarded as per the University norms.
4. The grace marks shall be given up to 1% of the total aggregate marks of theory and practical of that semester in maximum one paper.
5. Candidates reappearing at an examination in a subsequent year shall be examined in accordance with the scheme and syllabi in force and shall be entitled to the award of the degree of year in which they clear the last failing/unclear paper.

Grade Points awarded in the mark sheet based on marks obtained in theory and practical as per University norms.

BCA-101 English / Hindi /Rajasthani*

The syllabus will be same as provided by the university

BCA-102 Computer Architecture

Unit 1:

Binary Systems and Combinational Logic, Digital Computers and Digital Systems, Binary Numbers, Number Base Conversion, Octal and Hexadecimal Numbers, subtraction using r's and r-1 complements, Binary Code, Binary Storage and Registers, Binary Logic, Integrated Circuits. Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Other Logic Operations, Digital Logic Gates, The map Method, Two – and Three – Variable Maps, Four – Variables Map .

Unit 2:

Arithmetic Circuits and Sequential Logic, Logic gates NAND and NOR Implementation, Other Two-Level Implementations, Don't Care Conditions. Introduction, Adders, Subtractors, Binary Parallel Adder, Decimal Adder, Magnitude Comparator, Decoders, Multiplexers, BOOTH algorithm for signed numbers with example. Sequential Logic: Introduction, different types of Flip – Flops, Triggering of Flip- Flops, Boolean algebra, combinational circuits, circuit simplification, sequential circuits

Machine Instruction: Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes.

Unit 3:

Assembly Language and Input /Output Organization and The Memory System, Basics of Assembly Language Program, Examples from Assembly Language Programming. Accessing I/O Devices, Interrupts, Enabling and Disabling interrupts, Handling multiple devices, Controlling input/output device behavior, Exceptions, DMA , Buses.

Basic Concepts, Semiconductor RAM Memories, Internal organization of memory chips, Static memories, Dynamic RAM, Synchronous D-RAM, Structure of larger memories, Read – Only Memories, Speed, Size, and Cost, Cache Memories, Virtual Memories, Memory Management Requirements.

BCA-103 C Programming

Unit 1:

Overview of C Language: Character set, C tokens, Identifiers, Keywords, Data types, Variables, Constants, Symbolic Constants , Operators in C, Hierarchy of Operators, Expressions

Managing Input and Output Operation: Formatted and Unformatted I/O Functions, Decision making, branching and looping: Decision Making Statements - if Statement, if–else statement, nesting of if-else statements, else–if ladder, switch statement, ternary operator

Looping - while, do-while, for loop, Nested loop, break, continue, and goto statements. Functions: Function Definition, prototyping, types of functions, passing arguments to functions, Nested Functions, Recursive functions.

Unit 2:

Arrays: Declaring and Initializing, One Dimensional Arrays, Two Dimensional Arrays, Multi-Dimensional Arrays - Passing arrays to functions. Strings: Declaring and Initializing strings, Operations on strings, Arrays of strings, passing strings to functions. Storage Classes - Automatic, External, Static and Register Variables.

Unit 3:

Structures-Declaring and Initializing, Nested structure, Array of Structure, Passing Structures to functions, Unions, typedef, enum, Bit fields. Pointers – Declarations, Pointer arithmetic, Pointers and functions, Call by value, Call by reference, Pointers and Arrays, Arrays of Pointers, Pointers and Structures. Meaning of static and dynamic memory allocation, Memory allocation functions, infix, prefix, postfix

BCA-104 Operating Systems

Unit 1: Introduction to Operating Systems, goals of OS, operation of OS, resource allocater and related functions, classes of OS, batch processing, multi-processing, time sharing, distributed, real time systems, system calls, system programs

Process concept, interacting process, threads, fundamental of scheduling, scheduling criteria, long medium short term scheduling, scheduling algorithms , structure of concurrent system, critical section, critical region, inter-process communication, monitor and semaphores, implementation and uses.

Unit 2: Logical versus physical address, swapping, contiguous allocation, segmentation, paging, segmentation with paging, kernel memory allocation, page replacement algorithm, virtual memory, virtual memory with paging, demand paging, dead lock, characterization, methods for handling dead locks, prevention, avoidance, thrashing, allocation of frame, virtual memory using segmentation,

Files and Directories – File Concept, File types, File system Structure, file metsdata – Inodes, kernel support for file, system calls for file I/O operations – open, create, read, write, close, Iseek, dup2, file status information – sata family, file and record locking – fcntifuction, file permissions – chmod, fchmod, file ownership – chown, Ichown, fchown, links – soft links and hard links – symlink, link, unlink.

Unit 3

Directories – Creating, removing and changing Directories – mkdir, rmdir,chdir, obtaining current working directory – getcwd, Directory contents, Scanning Directories – opendir, readdir, closedir, rewinddir functions.

Process – process concept, Process environment – environment list, environment variables, getenv, setenv, system call interface for process management – fork, vfork, exit, wait, waitpid, exec family,
Shell programming with Bourne again shell (bash) – Introduction, shell respnsibilites, pipes and Redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shall variables, command substitutions, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples, interrupt processing, fictions, debugging shell scripts.

BCA-105 Lab-C Programming

Suggested Exercises for practice but not limited to the number of exercises mentioned here

1. Write a program to print sample strings like “hello world”, “Welcome to C Programming” with different formats using escape sequences.
2. Write a Program to read radius value from the keyboard and calculate the area of circle and print the result in both floating and exponential notation.
3. Write a Program to calculate simple interest.
4. Write a Program to convert temperature. (Fahrenheit –Centigrade and vice-versa)
5. Write a program to check equivalence of two numbers using conditional operator.
6. Write a program for computing the volume of sphere, cone and cylinder assume that dimensions are integer’s use type casting where ever necessary.
7. Write a Program to read marks of a student in six subjects and print whether pass or fail (using if-else).
8. Write a Program to calculate roots of quadratic equation (using if-else).
9. Write a Program to calculate electricity bill. Read starting and ending meter reading.
   The charges are as follows.
   No. of Units Consumed Rate in(Rs)
   1-100 1.50 per unit
   101-300 2.00 per unit for excess of 100 units
   301-500 2.50 per unit for excess of 300 units
   501-above 3.25 per unit for excess of 500 units
10. Write a Program to perform arithmetic operations using switch case.
11. Write a Program to display vowels and consonants using switch case.
12. Write a program to print prime numbers in the given range.
13. Write a program to display multiplication tables from 1 to 10 except 3 and 5.
14. Write a program to print the Fibonacci series for given ‘N’ value.
15. Write a program to read two numbers x and n then compute the sum of the Geometric Progression.
    1+x+ x2+x3+ - -- +xn
16. Write a program to print the following formats.
    1 *
    1 2 * *
    1 2 3 * * *
17. Write a program to store 10 elements in the 1-D array and print sum of the array.
18. Write a program to print minimum and maximum elements in the 1-D array.
19. Write a program to count no. of positive numbers, negative numbers and zeros in the array.
20. Write a program to search the given element by using linear search.
21. Write a program to sort the given elements using bubble sort technique.
22. Write a program to perform matrix addition, subtraction, multiplication, transpose
23. Write a program to perform various string manipulations using built-in functions.
24. Write a program to print the given strings in ascending order.
25. Write a program to find sum of two numbers using functions.
26. Write a program using functions without arguments, without return type to find product of two numbers, difference of two numbers, sum of two numbers
27. Write a program to find product of two numbers using functions with arguments, with return type.
28. Write a program to swap two numbers using a) Call By Value B) Call By Reference.
29. Write a program to calculate factorial, gcd using recursion and non-recursion functions.
30. Write program to perform arithmetic operations using pointer.
31. Write a program matrix addition using pointers.
32. Write a program to find total marks of individual student and average marks for 10 students using structures.
33. Write a program to create structure for an account holder in a bank with following
    Fields: name, account number, address, balance and display the details of five account holders.
34. Write a program to create structure called traveler and members of structure are train no, coach no, seat no, source ,destination , gender, age, name and departure date.

BCA-106 Lab-Linux

Suggested Exercises for practice but not limited to the number of exercises mentioned here

1. Verify that you are in your home directory.
2. Make the directory “adir”.
3. List the files in the current directory to verify that the directory “adir” has been made correctly.
4. Change directory to “adir”.
5. Verify that you have succeeded in changing the “adir” directory.
6. Create the file “testfile” and put your name and address in it.
7. Verify that the file “testfile” exists.
8. Display the contents of the file “testfile” on screen.
9. Make a copy of the file “testfile” to another file named as “Secondfile”.
10. Verify that the files “testfile” and “Secondfile” both exists.
11. Display the contents of both files.
12. Delete the file “testfile”.
13. Verify that the “testfile” has been deleted.
14. Clear the screen.
15. Rename “Secondfile” to “thefile”.
16. Make a directory name as “Aryan”. And copy “thefile” in “Aryan” directory.
17. Remove “thefile” from the current directory.
18. Verify that the “thefile” has been removed.
19. Remove the directory “adir”.
20. Verify that “thefile” and “adir” are gone from your home directory.
21. Create hidden file named as “hiddenfile1”.
22. Create hidden directory named as “hiddenfolder”.
23. Display all files and also display hidden files and directories.

Exercise 2

1. Create a file named as “City” and write names of any 5 cities in it.
2. Display the contents of “City” file.
3. Append 5 cities in “City” file.
4. Display the contents of “City” file with line numbers.
5. Sort the contents of “City” file in ascending and descending order.
6. Sort the contents of “City” file and redirect sorted output to another file (No changes in original file).
7. Copy “City” file to “City1” file which is present in “Aryan” directory. (Using cp command)
8. Create alias of following commands:
    a. mkdir
    b. ls –l
    c. rmdir
    d. rm –r
9. Display count of lines, words and characters in “City” file.
10. Write a command to search a string in “City” file.
11. Write a command to display only top 3 lines of “City” file.
12. Write a command to display only last 5 lines from “City” file.
13. Display the calendar of current year.
14. Display the calendar of previous, current and next month.
15. Display the date in the following format:
    a. Date: dd/mm/yy
    b. Time: hours:minutes:seconds
    c. 22-February-2022
    d. Date: 22, Month: February, Year: 2022
    e. Date: dd/mm/yy
    f. Time: hours:minutes:seconds
    g. Tuesday-February-22-2022
16. Compress the “City” file.
17. Decompress the “City” file.
18. Calculate the given expression and display the output:
    a. 20 + 30
    b. 20 * 30

c. 10 % 5
d. 20 – 30
e. 40 / 2

19. Display the prime factor of following numbers:
    a. 10
    b. 6

Exercise 4

1. Write a shell script to input a three digit number and print sum of its digits.
2. Write a shell script to marks of a student in 3 subjects and display his percentage.

If-Else

1. Write a shell script to input two numbers and print largest of them.

Nested If Else

1. Write a shell script to accept the age of three students and find out the difference between the age of eldest and youngest student.

Switch Case

1. Write a shell script to accept an operator and two numbers from the user and display result accordingly.

Loops

1. Write a shell script to input a number and print its factorial.
2. Write a shell script to input a number and print its reverse.

Array

1. Write a shell script to input 10 numbers in an array and sort them in ascending order.
2. Write a shell script to input 10 numbers in an array and find the count of prime numbers.

String

1. Write a shell script to concatenate two strings.
2. Write a shell script to convert a string into upper case and lower case.

BCA-201 English Communication Skill/ Hindi Communication Skill / Rajasthani Communication Skill

The syllabus will be same as provided by the university

BCA-202Discrete Mathematics

Unit 1:

Sets: definition and types, set operations, partition of set, cardinality, recursive definition of set. Functions: concept, some special functions (polynomial, exponential & Logarithmic, absolute value, floor & ceiling, mod & div functions) properties of functions, cardinality of infinite set, countable and uncountable set, pigeon hole principle, composition of function

Relations: Boolean matrices, binary relation, adjacency matrix of relation, properties of relations, operations of relations, connectivity relation, transitive closure, Warshall Algorithm, equivalence relation, equivalence class

Unit 2:

Proof Methods: Vacuous, trivial, direct, indirect by contrapositive and contradiction, constructive & non-constructive proof, counterexample. The division algorithm, divisibility properties (prime numbers & composite numbers) principle of mathematical induction, the second principle of mathematical induction, fundamental theorem of arithmetic. Algorithm correctness: partial correctness, loop invariant, testing the partial correctness of linear and binary search, bubble and selection sorting

Unit 3:

Graph theory: Graphs, directed, undirected, simple, adjacency & incidence, degree of vertex, sub-graph, complete graph, cycle & wheel graph, bipartite & complete bipartite graph, weighed graph. Trees: spanning trees – Kruskal's Algo, finding spanning tree using depth first search, breadth first search, complexity of graph, minimum spanning tree.

BCA-203Data Structures Using C++

Unit 1:

Classes, Objects, Encapsulation, Polymorphism, Inheritance, Control Structures: Control and Looping Statements. Function: Function Prototyping, Call and Return by Reference, Inline Function, Default and Const Arguments, Friend Functions, Friend Classes, Inline Functions, Static Class Members, Static Data Members, Static Member Functions, The Scope Resolution Operator, Nested Classes, Passing Objects to Functions, Returning Objects, Object Assignment.

Arrays, Pointers, Function Overloading, Copy Constructors and Default Arguments, Operator Overloading, Inheritance:

Unit 2:

Definitions of Data Structure and Algorithm – Time and Space complexity- Algorithm notations.

Searching: (Linear and Binary), Concept of sorting, Sorting algorithms (Bubble Sort, quick sort, Selection Sort, merge sort). Recursion: Factorial, Fibonacci, Tower of Hanoi.

Linked Lists: Introduction to linked list and double linked list, Representation of linked lists in Memory, Traversing a linked list, Searching linked list, Insertion and deletion into linked list, Doubly linked lists, Traversing a doubly linked lists.

Unit 3:

Stacks and Queues: Primitives of stacks, Implementation of stacks using Array & Link List Introduction to queues, properties of queues

Trees: Definition & Basic concepts, linked tree representation, Introduction to Binary Tree, Traversing Binary Trees (Pre order, Post order and In-order), Concept of Binary search tree, algorithm of Searching, inserting and deleting in binary search trees.

Graph: Introduction to graphs, types of graphs, operation of Graph: adjacency Matrix, Graph Traversal: Breadth first search, Depth first search.

BCA-204 Database Technology

Unit 1:
Conventional file system, object of database systems, data abstraction, data definition language, data manipulation language, database administrator. Database design stages, database model, database system architecture.

Entity-Relationship Model, entity, entity set, attributes, tuples, domains, keys, super and candidate key, overview of hierarchical, network and relational models, comparison of network, hierarchical and relational models, file organization

Unit 2:
Relational Model: Storage organization for relations, relational algebra, set operators, relational operators, decomposition of relation schemes, functional dependencies, normalization up to BCNF.

Unit 3:
MySQL Overview: Introduction, connecting to and disconnecting from the server , Entering queries , Creating and using a database , Creating and selecting a database , creating a table , loading data into a table , Retrieving information from a table , selecting all data , selecting particular rows , selecting particular columns , sorting rows , date calculations , working with NULL values , pattern matching , counting rows , using more than one tables.

Connecting to a MySQL database, querying the database, Retrieving and displaying the results, modifying data, deleting data.

BCA-205 Lab-C++ Programming

Suggested Exercises for practice but not limited to the number of exercises mentioned here

1. Write a program for Basic/Simple logic building in C++
2. Write a program to implement the concept of class and object
3. Write a program to implement the concept of Inline function.
4. Write a program to implement function overloading
5. Write a program to implement the concept of friend function
6. Write a program to implement the concept of static data member.
7. Write a program to implement static member function.
8. Write a program using the concept of constructor.
9. Write a program using the concept of constructor.
10. Write a program to implement operator overloading
11. Write a program to implement single inheritance
12. Write a program to Implement Multiple inheritance
13. Write a program to Implement Virtual function
14. Write a program to implement Virtual Base class
15. Write a program to create, read & write sequential file
16. Write a program to create, read & write random access file
17. Write a program to enter five records into a file and display them.
18. Write a program to implement function template
19. Write a program to implement class template
20. Write a program for exception handling

Function Overloading

1. Create a function max & overload it-
    i. void max(int a, int b);
    ii. void max(int a, int b, int c);
    iii. void max(int a, intb,intc,int d);
2. Write a C++ program to define overloaded functions to find volume of cube & volume of Cylinder.

Classes and Objects

1. Create a class named as Student which is containing the following private members
   Rollno, Name, Address, Course, Phone no. Create constructors in the class-:
    1. Student() : Default Constructor assigning -1 and undefined to the members
    2. Student(int, char 1, char()char [] , long)

   Add the following methods in the class:
    1. intgetRollno()
    2. char* getName()
    3. char *getAddress()
    4. char *getCourse()
    5. long getPhone()
    6. void setRollno(int)
    7. void setName(char nm[50])
    8. void setAddress(char add[50])
    9. void setCourse(char c[50])
    10. void setPhone(long)
    11. void displayDetails()

   Create Array of 5 objects and display their details also create following methods to test the functionality of the class-:
2. Input five records of the Student and display the details in proper format.
3. Input Rollno from the user and find that record in the list of five, if found display the details else display an error message.
4. Input course name and count the students studying in that particular course.
5. Sort the student list of five in ascending order on the basis ofRollno.

6. Sort the student list of five in ascending order on the basis of name.

Static members

1. Create a class "Student" containing name, rollno, marks of the student. Also create functions to accept and display the values. Then create some instances of this class. When this program is executed it displays the total number of instances created of the "Student" class.

Inheritance

1. Assume that a bank maintains two kinds of accounts for customers, one called as savings account and the other as current account.
   a. The savings account provides compound interest and withdrawal facilities but no checkbook facility.
   b. The current account provides checkbook facility but no interest. Current account holders should also maintain a minimum balance of Rs.1000 and if the balance falls below this level, a service charge is imposed.
   c. Create a class account that stores customer name, account number and type of account. From this derive the classes: - cur_acct and sav_acct to make them more specific to their requirements.
   d. Include necessary member functions in order to achieve the following tasks: -
      i. Accept deposit from a customer and update the balance.
      ii. Display the balance.
      iii. Complete and deposit interest.
      iv. Permit withdrawal and update the balance.
      v. Check for the minimum balance, impose penalty necessary, and upgrade the balance. Do not use any constructors.
      vi. Use member functions to initialize the class members.

Virtual Functions

1. Create a class named as Student containing roll number, name, address, date of birth which is an object of class Date containing day, month and year.Class must contain appropriate constructors, destructors and members methods.
   a. Derive a class named as "InternalExam" from Student class which contains maximum_marks and marks_obtained in the three subjects, create method to calculate total.
   b. Derive a class named as "ExternalExam" from Student class which contains maximum_marks and marks_obtained in the three subjects, create methods to calculate total.
   c. Derive a class named as "FinalExam" from "InternalExam" and "ExternalExam" which will contain marks_obtained and max_marks of the three same subjects, create constructors and destructors, call method of the super classes, create methods for calculations and print a performance statement of the student.

Operator overloading

1. Create a class "String", which contains following data members:-char "str and int size. Create default and parameterized constructors. Allocate memory dynamically to str. Create two objects of this class. Accept data into the str. concatenate the strings in str of both the objects and store it in the third object of the same class. Overload binary addition operator.

File handling

1. Accept two file names from the user at command line. Copy only the upper case and lower case letters of first file into the second file.

BCA-206 Lab-MySQL

Suggested Exercises

Exercise-1 (Select)

1. List all the information about employees.
2. Display the names of all employees along with their salary.
3. List the names of all employees who are working in department number 20.
4. Display the details of those employees who have joined before the end of Sept 2020.
5. List the employee name and employee number who is 'Manager'.
6. List the employee detail that does not belong to department number 10 and 20.
7. List employee name and salary whose salary varies from 5000 to 10000.
8. List employee detail that is clerk and earns more than 5000.
9. List name and job of all employees who are not clerk.
10. List the Employee's name and Date of joining in descending order of Date of joining. The column title should be Date of Joining.
11. List employees having TT or LL in their names.
12. List employees who are working as salesman and having name of four characters.
13. List employees having S at the end of their name.
14. List employees who are not managed by anyone.
15. List employees who are earning commission more than their salary. Only those records should be displayed where commission is given (also sort the output in descending order).

Exercise-2 (Grouping and Having)

1. List department number and total number of employees in each department.
2. List different job names available in the EMP table.

Exercise -3 (Joins)

1. List all the employee details who are working in NEW YORK.
2. List the information of all managers of all departments.

Exercise-4 (Sub Queries)

1. List employee who are earning salary same as MARTIN.
2. Find the employees who work in the department same as department of SCOTT.
3. List employees who are earning more than the average salary of department number 20. Also ensure that records should not contain records of department number 20.

Exercise-5 (Data Definition Language)

1. Create table emp, dept

Dept table

Field Type	Type	Constraint
Deptno	Int(2)	Primary key
Dname	Varchar(20)	Not null , unique
Loc	Varchar(20)	

Emp table

Field type	Type	Constraints
Empno	Int(4)	Primary key
Ename	Varchar(20)	Not null
Job	Varchar(25)	Not null
Deptno	Int(2)	Foreign key
Mgr	Int(4)	

Comm	Int(3)	
Sal	Int(7)	Default (1000)
Hiredate	Date	

2. Modify Ename column of emp table, set its width to varchar(30).
3. Add a default constraint on deptno column of EMP table to set default value to 10.
4. Modify the structure of EMP table and add NOT NULL constraint on job column.
5. Modify the structure of EMP table and add unique constraint on Ename column.
6. Modify the structure of EMP table and remove unique constraint on Ename column.

Exercise-6 (Data Manipulation Language)

1. Insert Records into EMP, dept and salgrade table.
2. Update all employees of accounting department. Increase salary by 10%; (subquery).
3. Change all the Managers to Executive.
4. Delete the records of employee who is earning the lowest salary in the company.
5. Update all employees of EMP table, set the commission to 500 for those employees who are not getting any commission and also working as Manager.

Exercise-7 (Functions)

1. Calculate the square root of given number (10000).
2. Find the rounding value of 123.456 up to 2, 0 and -2 places of decimal.