

MANGALORE UNIVERSITY



State Education Policy – 2024 [SEP-2024]

CURRICULUM STRUCTURE

FOR

**BACHELOR OF COMPUTER APPLICATIONS
BCA-ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**

Semester II								
Sl. No	Course Code	Title of the Course	Category of Courses	Teaching Hours per Week	SE E	IA	Total Marks	Credits
1		Language-I	Lang	4	80	20	100	3
2		Language-II	Lang	4	80	20	100	3
3	BCA - AIML -2.1	Data Structure using C	Core	4	80	20	100	3
4	BCA- AIML -2.2	Database Management System	Core	4	80	20	100	3
5	BCA- AIML -2.3	Computer Architecture	Core	5	80	20	100	5
6	BCA - AIML -2.4	Data Structures Lab	Practical	4	40	10	50	2
7	BCA - AIML -2.5	Database Management System Lab	Practical	4	40	10	50	2
8		Constitution/ Values	Compulsory	2	40	10	50	2
Sub - Total				31	520	130	650	23

SEMESTER- II

Program Name	BCA-AIML	Semester	II
Course Title	Data Structure using C(Theory)		
Course Code:	BCA-AIML-2.1	No.of Credits	03
Contact hours	4 Hours per Week	Duration of SEA/Exam	3 Hours
Formative Assessment Marks	20	Summative Assessment Marks	80

Course Outcomes (COs):

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

- Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms
- Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs
- Write programs that use arrays, linked structures, stacks, queues, trees, and graphs
- Demonstrate different methods for traversing trees
- Compare alternative implementations of data structures with respect to performance
- Describe the concept of recursion, give examples of its use
- Discuss the computational efficiency of the principal algorithms for sorting, searching, and hashing

Unit	Description	Hours
1	Introduction and Overview: Definition, Elementary data organization, Data Structures, Data Structures operations, Abstract data types, algorithms complexity, time space tradeoff. Preliminaries: Mathematical notations and functions, Algorithmic notations, control structures, Arrays: Definition, Linear arrays, , Representation of Linear Arrays in Memory, Traversing Linear arrays, Inserting and deleting, Multi-dimensional arrays, Matrices and Sparse matrices.	13

	<p>Fundamentals of Algorithmic problem Solving: Important Problem Type Fundamentals of Data Structures, Fundamentals of the Analysis of Algorithm Efficiency, Analysis Framework, Measuring the input size, Units for measuring Running time, Orders of Growth, Worst-case, Best-case and Average- case efficiencies.</p> <p>Asymptotic Notations and Basic: Efficiency classes, Informal Introduction, O-notation, Ω-notation, θ-notation</p>	
2	<p>Linked list: Definition, Representation of Singly Linked List in memory, traversing a Singly linked list, searching in a Singly linked list, Memory allocation, Garbage collection, Insertion into a singly linked list, Deletion from a singly linked list; Doubly linked list, Header linked list, Circular linked list.</p> <p>Stacks: Definition, Array representation of stacks, linked representation of stacks, Arithmetic Expressions: Polish Notation, Conversion of infix expression to postfix expression, Evaluation of Postfix expression, Applications of Stacks, Recursion, Towers of Hanoi, Implementation of recursive procedures by stack.</p> <p>Queues: Definition, Array representation of queue, Linked list representation of queues. Types of queue: Simple queue, Circular queue, Double-ended queue, Priority queue, Operations on Queues, Applications of queues.</p>	13
3	<p>Binary Trees: Definitions, Tree Search, Traversal of Binary Tree, Tree Sort, Building a Binary Search Tree, Contiguous Representation of Binary Trees: Heaps, Lexicographic Search Trees: Tries, External Searching: B-Trees, Applications of Trees.</p> <p>Graphs: Mathematical Background, Computer Representation, Graph Traversal, Topological Sorting</p>	13
4	<p>Searching: Introduction and Notation, Sequential Search, Binary Search, Comparison of Methods. Sorting: Introduction and Notation, Insertion Sort, Selection Sort, Shell Sort, Divide and Conquer, Merge sort for Linked List, Quick sort for Contiguous List.</p> <p>Hashing: Sparse Tables, Choosing a Hash function, Collision Resolution with Open Addressing, Collision Resolution by Chaining.</p>	13

Text Books:

1. Seymour Lipschutz, “Data Structures with C”, Schaum’ soutLines, Tata McGraw Hill, 2011.
2. Anany Levitin: “Introduction to The Design & Analysis of Algorithms”

Reference Books:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures, Computer Science Press, 1982.
2. Aaron M. Tenenbaum , Data structures using C, First Edition, Pearson Education
3. Kamathane, Introduction to Data structures, Pearson Education , 2004
4. Y. Kanitkar, Data Structures Using C, Third Edition, BPB
5. Padma Reddy: Data Structure Using C, Revised Edition 2003, Sai Ram Publications.
6. Sudipa Mukherjee, Data Structures using C – 1000 Problems and Solutions, McGraw Hill Education, 2007

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc.

Program Name	BCA-AIML	Semester	II
Course Title	Database Management System(Theory)		
Course Code:	BCA-AIML-2.2	No.of Credits	03
Contact hours	4 Hours per Week	Duration of SEA/Exam	3 Hours
Formative Assessment Marks	20	Summative Assessment Marks	80

Course Outcomes (COs):

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

- Demonstrate an understanding of the basic concepts of database systems.
- Design a relational database using ER modelling and normalization techniques.
- Write complex SQL queries to retrieve and manipulate data from databases.
- Develop PL/SQL programs to implement business logic in databases.
- Explain the concepts of transaction management, concurrency control, and database recovery.

Unit	Description	Hours
1	<p>Database and Database Users: DBMS Definition, Characteristics of the Database Approach, Advantages of Using a DBMS</p> <p>Database System concepts and architecture: Data Models, Schemas, and Instances, Three-schema architecture and Data Independence, Database Languages and Interfaces, The Database System Environment, Classification of Database Management Systems.</p> <p>Data Modeling Using the Entity-Relationship Model: High-Level Conceptual Data Models for Database Design, An example database application, Entity Types, Entity Sets, Attributes and Keys, Relationships, Relationship Types, sets, roles, and Structural Constraints, Weak Entity Types, ER Diagrams.</p>	13

2	<p>Relational Data Model, Relational Constraints: Relational Model Concepts, Relational model Constraints and Relational Database Schemas, Update Operations, transactions and Dealing with Constraint Violations.</p> <p>Disk Storage, basic file structures and Hashing: Secondary storage devices, Buffering of Blocks, Placing File Records on Disk, Operations on Files, Files of Unordered Records (Heap Files), Files of Ordered Records (Sorted Files), Hashing Techniques</p> <p>Functional dependencies and Normalization for Relational databases: Functional dependencies, Normal Forms based on primary keys, General definitions of second and third normal forms, Boyce-Codd Normal form.</p>	13
3	<p>Interactive SQL: Table fundamentals, oracle data types, CREATE TABLE command, inserting data into table, Viewing Data in the table, sorting data in a table, creating a table from a table, inserting data into a table from another table, Delete operations, Updating the contents of a table, Modifying the structure of tables, Renaming tables, destroying tables, displaying table structure.</p> <p>Data Constraints: Types of data constraints, IO constraints- The PRIMARY KEY constraint, The FOREIGN KEY constraint, The UNIQUE KEY constraint, Business Rule Constraints- NULL value concepts, NOT NULL constraints, CHECK constraint, DEFAULT VALUE concepts.</p> <p>Computations on table data: Arithmetic Operators, Logical Operators, Range Searching, Pattern Matching, Oracle Table – DUAL, Oracle Function- Types, Aggregate Function, Date Conversion Function. GROUPING DATA FROM TABLES IN SQL, Group By clause, Having clause, subqueries, JOINS, Using the UNION, INTERSECTION, MINUS clause SQL transaction commands COMMIT, ROLLBACK and SAVEPOINT. Security Management using SQL- Granting and Revoking Permissions</p>	13
4	<p>INTRODUCTION TO PL/SQL: Advantages of PL/SQL, The Generic PL/SQL Block, PL/SQL-The character set, Literals, PL/SQL datatypes, variables, Logical comparisons, Displaying User Messages on The VDU Screen, comments. Control Structure - Conditional Control, Iterative Control</p>	13

	<p>PL/SQL Transactions: Cursor-Types of Cursor, Cursor Attributes. Explicit cursor ,Explicit cursor Management, cursor for loop PL/SQL</p> <p>Database Objects: Procedures and Functions, Oracle Packages, Error Handling in PL/SQL, Triggers</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Ramez Elmasri and Shamkanth B.Navate, Fundamentals of Database Systems, 7th Edition, Pearson Education 2. Ivan Bayross, SQL/PL/SQL- the Programming language of Oracle, 2nd Revised edition (or 4th revised Ed), BPB Publications <p>Reference Books:</p> <ol style="list-style-type: none"> 1. An Introduction to Database Systems, Bipin Desai, Galgotia Publications, 2010. 2. Introduction to Database System, C J Date, Pearson, 1999. 3. Database Systems Concepts, Abraham Silberschatz, Henry Korth, S.Sudarshan, 6th Edition, McGraw Hill, 2010. 4. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw Hill, 2002 		

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc.

Program Name	BCA-AIML	Semester	II
Course Title	Computer Architecture(Theory)		
Course Code:	BCA-AIML-2.3	No.of Credits	05
Contact hours	5 Hours per Week	Duration of SEA/Exam	3 Hours
Formative Assessment Marks	20	Summative Assessment Marks	80

Course Outcomes (COs):

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

- Use number systems and complements
- Identify the importance of canonical forms in the minimization or other optimization of Boolean formulas in general and digital circuits.
- Minimize functions using any type of minimizing algorithms (Boolean algebra, Karnaugh map or Tabulation method).
- Analyze the design procedures of Combinational and Sequential circuits.
- Design the finite state machine using algorithmic state machine charts and perform simple projects with a few flip-flops.

Unit	Description	Hours
1	Digital computers and Digital system: Introduction to Number system, Decimal number, Binary number, Octal and Hexadecimal numbers, Number base conversion, Complements, Binary codes, Binary arithmetic's, Addition , Subtraction in the 1's and 2's complements system, Subtraction in the 9's and 10's complement system. Boolean Algebra: Basic definitions, Axiomatic definition of Boolean algebra, Basic theorems and properties of Boolean algebra, Venn diagram.	15
2	Digital logical gate: Boolean functions, Canonical and Standard forms, other logic operations, Digital logic gates, Universal gate. Simplification of Boolean function: The map method, Two and three variable maps, Four-variable maps, Don't care conditions,	15

	Product of sum Simplification, NAND implementation, NOR implementation. Implementation of EX-OR, EX-NOR using NAND and NOR gate	
3	Combinational Logic: Introduction, Design Procedure, Half adder, Full adder, half Subtractor, Full Subtractor, Binary parallel adder, BCD adder. Combinational logic with MSI and LSI: Code converter, Exclusive-OR and Equivalence functions. Magnitude comparator, Decoders, Encoders, Multiplexers, Demultiplexers	15
4	Sequential Logic: Introduction, Flip flops, RS-FF, D-FF, T-FF, and JK-FF, Triggering of flip-flops, Master slave Flip flop, state table, and State diagram. State equations, Flip Flop excitation tables, Sequential circuits design. Registers, Counters: Synchronous Counter Design using RS, JK, D & T flip flops. Ripple counters Introduction, Registers, Shift registers, Timing sequences, Bidirectional shift register.	15
<p>Text Book:</p> <p>1. M.Morris Mano, Digital Logic and Computer design, PHI, 2015</p> <p>Reference Books:</p> <p>1. Thomas L Floyd, Digital Fundamentals, 10th Edition, Pearson, 2011. 2. Thomas. C. Bartee, Digital Computer Fundamentals, 6th edition, TMH.</p>		

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe- Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc.

Program Name	BCA-AIML	Semester	II
Course Title	Data Structures Lab		
Course Code:	BCA-AIML-2.4	No.of Credits	02
Contact hours	4 Hours per Week	Duration of SEA/Exam	3 Hours
Formative Assessment Marks	10	Summative Assessment Marks	40

PART-A

1. Program to sort the given list using selection sort technique.
2. Program to sort the given list using insertion sort technique
3. Program to solve Tower of Hanoi using Recursion
4. Program to reverse String using Stack
5. Program to search an element using recursive binary search technique.
6. Program to implement Stack operations using arrays.
7. Program to implement Queue operations using arrays.
8. Program to implement dynamic array. Find smallest and largest element.

PART-B

1. Program to sort the given list using merge sort technique.
2. Program to implement circular queue using array.
3. Program to sort the given list using quick sort technique.
4. Program to implement Stack operations using linked list.
5. Program to implement Queue operations using linked list.
6. Program to evaluate postfix expression.
7. Program to perform insert node at the end, delete a given node and display contents of single linked list.
8. Menu driven program for the following operations on Binary Search Tree(BST) of Integers
 - (a) Create a BST of N Integers
 - (b) Traverse the BST in Inorder, Preorder and Post Order.

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	PART-A Writing:5 Marks Execution:3Marks	15 Marks
Program-2	PART-B Writing:8 Marks Execution:4Marks	20 Marks
Practical Record		05 Marks
Total		40Marks

Program Name	BCA-AIML	Semester	II
Course Title	Database Management System Lab		
Course Code:	BCA-AIML-2.5	No.of Credits	02
Contact hours	4 Hours per Week	Duration of SEA/Exam	3 Hours
Formative Assessment Marks	10	Summative Assessment Marks	40

PART-A

1. Create a table EMPLOYEE using SQL command to store details of employees such as EMPNO, NAME, DESIGNATION, DEPARTMENT, GENDER and SALARY. Specify Primary Key and NOT NULL constraints on the table. Allow only 'M' or 'F' for the column GENDER. DEPARTMENT can be SALES, ACCOUNTS, IT. Choose DESIGNATION as CLERK, ANALYST, MANAGER, ACCOUNTANT and SUPERVISOR that depends on department

Write the following SQL queries:

- a) Display EMPNO, NAME and DESIGNATION of all employees whose name ends with RAJ.
 - b) Display the details of all female employees who is earning salary within the range 20000 to 40000 in SALES or IT departments.
 - c) List the different DEPARTMENTS with the DESIGNATIONS in that department.
 - d) Display the department name, total, average, maximum, minimum salary of the DEPARTMENT only if the total salary given in that department is more than 30000.
 - e) List the departments which have more than 2 employees.
2. Create a table CLIENT to store CLIENT_NO, NAME, ADDRESS, STATE, BAL_DUE. Client no must start with 'C'. Apply the suitable structure for the columns. Specify Primary Key and NOT NULL constraints on the table. Insert 10 records.

Write the following SQL queries:

- a) From the table CLIENT, create a new table CLIENT1 that contains only CLIENT_NO and NAME, BAL_DUE from specified STATE. Accept the state during run time.

- b) Create a new table CLIENT2 that has the same structure as CLIENT but with no records. Display the structure and records.
 - c) Add a new column by name PENALTY number (10, 2) to the CLIENT.
 - d) Assign Penalty as 10% of BAL_DUE for the clients C1002, C1005, C1009 and for others 8%. Display Records.
 - e) Change the name of CLIENT1 as NEW_CLIENT.
 - f) Delete the table CLIENT2.
3. Create a table BOOK using SQL command to store Accession No, TITLE, AUTHOR, PUBLISHER, YEAR, PRICE. Apply the suitable structure for the columns. Specify Primary Key and NOT NULL constraints on the table. Insert 10 records.

Write the following SQL queries:

- a) List the details of publishers having 'a' as the second character in their names.
 - b) Display Accession No., TITLE, PUBLISHER and YEAR of the books published by the specified author before 2010 in the descending order of YEAR. Accept author during run time.
 - c) Modify the size of TITLE to increase the size 5 characters more.
 - d) Display the details of all books other than Microsoft press publishers.
 - e) Remove the records of the books published before 1990.
4. Create a table SALES with columns SNO, SNAME, MANAGER_NAME, JOIN_DATE, DATE_BIRTH, SALARY, SALES_AMOUNT and COMMISSION. Minimum age for joining the company must be 18 Yrs. Default value for Commission should be 0. Apply the suitable structure for the columns. Specify Primary Key and NOT NULL constraints on the table. Insert 10 records with data except commission. Manager of Manager can be NULL.

Write the following SQL queries:

- a) Display the details of Sales Persons whose salary is more than Average salary in the company.
 - b) Update commission as 20% of Sales Amount.
 - c) Display SNO, SNAME, MANAGER_NAME, SALARY, COMMISSION, MANAGER_SALARY of the sales persons getting sum of salary and commission more than salary of manager.(Self join)
 - d) Display the records of employees who finished the service of 10years.
5. Create a table Sales_Details with the columns SNO, MONTH, TARGET and QTY_SOLD to store the Sales Details of one year. Specify the composite primary key to the columns SNO and MONTH. TARGET and SALES must be positive numbers.

Write the following SQL queries:

- a. Display the total sales by each sales person considering only those months sales where target was reached.
 - b. If a commission of RS.50 provided for each item after reaching target, calculate and display the total commission for each sales person.
 - c. Display the SNO of those who never reached the target.
 - d. Display the SNO, MONTH and QTY_SOLD of the sales persons with SNO S0001 or S0003
6. Create a table Bank with the columns ACNO, ACT_NAME, ACT_TYPE and BAL. Specify the Primary Key. Initial BAL must be greater than 500.

Write a PL/SQL program to perform debit operation by providing acct_no and amount required. The amount must be greater than 100 and less than 20000 for one transaction. If the account exist and BAL-amount>100 Bank table must be updated, otherwise “NO SUFFICIENT BALANCE” message should be displayed. If account number is not present then display “NO SUCH ACCOUNT” message to the user.

7. Create a table STOCK_DETAIL with the columns PNO, PNAME and QTY_AVL to store stock details of computer accessories. Specify Primary Key and NOT NULL constraints on the table. QTY_AVL should be positive number.

Write a PL/SQL Program to define a user defined exception named “LOW_STOCK” to validate the transaction. The program facilitates the user to purchase the product by providing product number and quantity required. It should display an error message “NO SUFFICIENT STOCK” when the user tries to purchase a product with quantity more than QTY_AVL, Otherwise the STOCK_DETAIL table should be updated for valid transaction.

8. Write a PL/SQL cursor program to calculate electricity bill of several domestic customers. Accept Input RR No, name of the customer, previous meter reading, and current meter reading from the table. The rates of electricity consumption are as follows - For the first 30 units Rs. 2.5 per unit, for the next 70 units Rs. 3.5 per unit, for the next 100 units Rs. 4.5 per unit, for the next 100 units Rs. 6 per unit and for units above 300 Rs. 8 per unit. A fixed amount of Rs. 150 is also charged. 5% tax to be paid on the sum of bill amount & fixed amount. Use Data validation to see that current reading is more than previous reading. Assume the records of 5 customer details.

Create the output which contains the RR number, name of the Customer, previous meter reading, and current meter reading, Units Consumed, Total Bill in the following format.

```
=====
RR No. Customer Current Meter Previous Meter Units Total
      Name Reading Reading Consumed Bill
=====
```

PART-B

1. Create the following tables by identifying primary and foreign keys. Specify the not null property for mandatory keys.

SUPPLIERS (SUPPLIER_NO, SNAME, SADDRESS, SCITY)

COMPUTER_ITEMS (ITEM_NO, SUPPLIER_NO, ITEM_NAME, IQANTITY)

Consider three suppliers. A supplier can supply more than one type of items.

Write the SQL queries for the following

- a. List *ITEM* and *SUPPLIER* details in alphabetical order of city name and in each city decreasing order of *QUANTITY*.
 - b. List the name ,city,and address of the suppliers who are supplying keyboard.
 - c. List the supplier name, items supplied by the suppliers ‘Cats’ and ‘Electrotech’.
 - d. Find the items having quantity less than 5 and insert the details of supplier and item of these, into another table *NEWORDER*.
-
2. Create the following tables identifying Primary and Foreign keys. Specify the not null property for mandatory keys.
EMPLOYEE_MASTER (*EMP_ID*, *EMP_NAME*, *EMAIL_ID*, *EMP_ADDRS*, *PHONE*)
ATTENDANCE (*EMP_ID*, *MONTH*, *WOM*, *MHRS*, *THRS*, *WHRS*, *TRHRS*, *FHRS*, *SHRS*, *SUHRS*).
 (Valid values for *WOM* ≤ 5, *MONTH* can be 1-12). Apply appropriate constraints. Consider 3 employees. And attendance records for at least two months.

Write the SQL queries for the following

- a) Display *EMP_ID*, *EMP_NAME* and *EMAIL_ID* of all employees who are working on every Sunday of 2nd and 4th week in a month.
 - b) Display total hours worked by each employee in each month with *EMP_ID*.
 - c) Display the names of the employees who never attended the duty so far(Attendances not given so far).
 - d) Display the employee name, month, week, total hours worked for employees who have total no. of hours more than 20 hrs. a week.
3. Create the following tables by identifying primary and foreign keys, specify the not null property for mandatory keys.

PRODUCT_DETAIL				
P_NO	PRODUCTNAME	QTYAVAILABLE	PRICE	PROFIT %
P0001	Monitor	10	3000	20
P0002	Pen Drives	50	650	5
P0003	CD Drive	100	10	3
P0004	Key Board	25	600	10
PURCHASED_DETAIL				
CUSTNO	P_NO	QTYSOLD		
C1	P0003	2		
C2	P0002	4		
C3	P0002	10		
C4	P0001	3		
C1	P0004	2		
C2	P0003	2		
C4	P0004	1		

Write the following SQL queries:

- a) Display total amount spent by C2.
 - b) Display the names of product for which either QtyAvailable is less than 30 or total QtySold is less than 5(USE UNION).
 - c) Display the name of products and quantity purchased by C4.
 - d) How much Profit does the shopkeeper gets on C1's purchase?
 - e) How many 'Pen Drives' have been sold?
4. Create table *STUDENT_PROFILE* includes Rollno, name, class, *ECCC*(Extra-Co curricular he belongs to such as *SPORTs*, *NSS* etc.) and another table

MARKS_REPORT includes Rollno, Internal_Test, Marks1, Marks2, Marks3 and ECCC_marks.

Constraints

- Internal_Test can be either 1 or 2.
- Each mark can be 0-100. Absence in the test can be entered as -1.
- Consider atleast 3 classes.

Apply suitable data type and constraints to each column.

Insert 5 students marks report in the both the tests.

Write the following SQL queries:

- Find number of students failed class- wise.
- Display the complete details of the students secured distinction(Percentage \geq 70) in I BCA.
- Display class and highest total marks in second internals in each class.
- Display the student name with rollno and class of those who passed in I internals and failed in II internals.(use SET operator)

5. Write a PL/SQL program to compute the selling price of books depending on the book code and category. Use Open, Fetch and Close.The Book_detail table contains columns: Book Code, Author, Title, Category and Price.Insert 10 records.
The selling price=Price-Discount.

The discount is calculated as follows:

Book Code	Category	Discount Percentage
A	Novels	10% of Price
	Technology	12.5% of Price
B	Commerce	18% of Price
	Science	19% of Price
C	Songs	25% of Price
	Sports	24% of Price
D	All	28% of Price

Print the result in tabular form with proper alignment

Book Code	category	title	author	price	discount %	discount amount	sell price
-----------	----------	-------	--------	-------	------------	-----------------	------------

=====

6. Write a PL/SQL program to display employee pay bill (using Cursor For loop) Use a **Procedure** to receive basic pay and to compute DA, HRA, Tax, PF, Gross Pay and Net Pay(Use OUT). Base table contains the following columns empnum, empname, basic pay. Insert 3 records.

Allowances are computed as follows.

Basic Pay	DA	HRA
<=20000	35% of Basic	8% of Basic
>20000 & <=30000	38%	9%
>30000 & <=40000	40%	10%
>40000	45%	10%

Gross=Basic+DA+HRA

PF=12% of Gross or Rs. 2000 whichever is minimum.

PT=Rs. 100 upto Gross is 25,000 else Rs. 200.

Net=Gross-(PF+PT)

Print Pay slip as follows.

```

=====PAYSLIP=====
Empno      :10011      Empname : Raj
Basic Pay  :20000      P.F.   : 3432
DA         :7000      P.T.   : 200
H.R.A.    :1600
Gross     :28600      Net Pay : 24968
*****
=====PAYSLIP=====
Empno      :10012      Empname : Rani
Basic Pay  :30000      P.F.   : 5292
DA         :11400     P.T.   : 200
H.R.A.    :2700
Gross     :44100      Net Pay : 38608
*****

```

7. Given the following tables:

ITEM_MASTER(itemno, name, stock, unit_price) [Apply the Primary key and check constraint for stock and price as >0] [Insert 5 records]

ITEM_TRANS(itemno, quantity and trans_date)

Create a **package** PCK_ITEM includes a function CHK_ITEM and a procedure PROC_ITEM.

Function CHK_ITEM gets one argument itemno and is used to check whether the parameter itemno exists in ITEM_MASTER and should return 1 if exist. Otherwise 0 and displays proper message.

Procedure PROC_ITEM gets two arguments itemno and quantity, and is used to perform the following if item exists. If required quantity is not available, give appropriate message. If available , insert a record of this transaction to ITEM_TRANS and modify the stock in ITEM_MASTER.

Write a PL/SQL program to accept ITEM_NO and Quantity needed of required item. Use Package to do the transaction process(Transaction date can be current date).

OUTPUT to be shown as follows:

```
Enter value for accept_itemno: 1
old 5:          X:=&accept_itemno;
new 5:          X:=1;
Enter value for quantity: 3
old 6:          M:=&quantity;
new 6:          M:=3;
Item :aa  Quantity :3 Price :15 Total Amount :45
```

8. Consider the following tables

LIBRARY(Accession no, Title, Author, Publication, Status). Status can be **A** for available and **I** for Issued. Insert 3 records with status '**A**' for all initially.

ISSUE(Rollno, Accession no, Borrowdate, returndate).

OUTDATED(Accession no, Title, Author, Publication, tdate),

Write the following Trigger programs.

- i. Whenever the book is to be issued, Insert a new record to ISSUE without having return date. When the record is **inserted** to ISSUE table, trigger TRIG_ISSUE to be executed to update status in LIBRARY as '**I**'.
- ii. Whenever book is returned, update return date of that record as todays date in ISSUE table. When the record is **updated** to ISSUE table, trigger TRIG_ISSUE to be executed to update status in LIBRARY as '**A**'.
- iii. Whenever the book is **deleted** by accepting Accession no. for status '**A**' (at SQL >), trigger TRIG_OUTDATE has to be executed to insert a record to OUTDATED.

Write a PL/SQL program to accept Rollno, Accession no and transaction(**B** for Borrow &**R** for Return). Check for the existence of given Accession no and proceed as follows.

- If does not exist, display the message ‘Given accession no. is not available’
 - If exist and transaction is B, check the status as ‘A’, then insert to ISSUE, and display the message with accno, author, title, publication and roll no to whom it is issued.
 - If exist and transaction is R, then update return date as current system date in ISSUE by accepting Rollno and Accession no(for the record having return date empty).
- If searched record is not available, raise the predefined exception.

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	PART-A Writing:5 Marks Execution:3Marks	15 Marks
Program-2	PART-B Writing:8 Marks Execution:4Marks	20 Marks
Practical Record		05 Marks
Total		40 Marks