**Journal Questions (Computer Science)**

* Dictionaries

1. WAP create a dictionary called phonedict with name as keys and telephone number as values. Write a menu driven program to search, display, add, modify and delete a record from the dictionary.
2. WAP that interactively creates a dictionary called FLIGHT whose keys are Flight Number and vales as another dictionary whose keys are name of the flight and destination respectively. Ask the user to enter the flight number and use the dictionary to tell the details. Make a list of flights whose destination is “ Mumbai “. Also make a list of all the flight names in alphabetical order.
3. Create a dictionary called STUDENT whose keys are names of students and value is another dictionary whose keys are stream and marks, where the mark is a list which consists of 5 subjects. Write a script in Python that prints the name, stream, total and average marks of each student in a dictionary.

* Tuples

1. WA menu driven program to find the largest element in a nested tuple, sum of diagonals ( L🡪R ) ( R🡪L ) and the sum of rows and columns.
2. WAP to create a tuple and split the tuple into two parts. The first part should contain all the even numbers and the second should contain all the odd numbers.
3. WAP to create a tuple in which you enter the details such as book\_id , name and book\_type. Create a new tuple which stores all the book types of the type “comedy”.

* Files (Using Functions)

1. WAP using binary files to create, modify , add , delete , search a record(Using dictionaries). Create a separate function for each option.
2. WAP to count the words “ to “ and “ the “ present in a text file “poem.txt”. Create a file and find the same.
3. WAP to count the number of uppercases alphabets present in a text file “article.txt”.
4. WAP to copy the words starting with vowels and write in another file. Source file is “poem.txt” and destination is “article.txt”
5. WAP to replace a word with another word (replace ()).
6. WA menu driven program to create a file “student.dat” using a class called school which have the following attributes:

\_\_init\_\_() : to input the data for the data members

display() : to display the data members

Using the above class, create functions to perform the following: create() , append() , display() , search(), modify() , insert()

* Classes And Inheritance

1. WA Menu driven program using a class called bank having the following data members:

List as a class variable , acc\_no , acc\_name and amount

Methods:

A constructor function which is used to input the data members.

Disp() method which receives acc\_no as a parameter and displays the details pertaining to that account number

Withdraw() method which receives acc\_no as a parameter and inputs the amount to be withdrawn and hence deduct the total amount in the account

Deposit() method which receives acc\_no as a parameter and inputs the amount to be deposited and hence increase the total amount in the account.

1. Declare a class to represent books in a library, include the following attributes:  
   Book Number , Book name , Publisher , Price , No. Of Copies and the number of copies issued.  
     
   Methods:  
   A constructor method which assigns initial values by inputting the values for all the data members.  
     
   issue() : To issue a book after checking for its availability ( Check by book number passed as an argument ).  
     
   return(): To return a book (check by book number as an argument).  
     
   display(): To display book information ( check by book number passed as argument):
2. WAP to illustrate the example of Single Inheritance
3. WAP to illustrate the example of Multiple Inheritance
4. WAP to illustrate the example of Multilevel Inheritance.

* Data Structures

1. **Array**  
    A) One Dimensional Array  
   i. WA Menu driven program to allow the user to input a list and then search for an element in a list using the searching techniques according to the user’s specification.  
     
   ii. WA menu driven program to allow the user to input a list and then sort the list using the different sorting techniques according to the user’s specification.  
     
   iii. WAP to swap the successive elements in a list.  
     
   iv. WA Menu driven program to either insert or delete an element in an ordered and an unordered list.  
     
   v. WAP to create a sorted list using the ‘Bisect’ method.  
     
   vi. WAP to merge two user inputted lists and sort them as they are merged.   
     
   B) Two Dimensional Array.  
   i. WA Menu driven program to perform the following functions:  
    a) Create a Matrix of order m x n.   
    b) Display the matrix.  
    c) Transpose the entered matrix.  
    d) To find the sum of diagonals   
    e) Print the upper and lower triangles of the matrix.  
    f) Print the largest element in the matrix.  
    g) To display the sum of rows and columns in the matrix.  
    h) To find the sum of two matrices.  
    i) To find the product of two matrices.  
     
   ii. Create a 2D array for a mark sheet for a class containing ‘n’ students. The mark sheet consists of Student No., Student name, marks in 5 subjects, total, average. Display the mark sheet , the toppers in each subject, the failures in each subject and the first 3 rankers of the class.
2. **Stack**i. WA Menu Driven program that illustrates the Stack implementation consisting of Push, Pop and Display. (You may use predefined functions).  
     
   ii. WA Menu driven program to illustrate the stack implementation consisting of Push, Pop, Display and seek (which gives the position of the Top Pointer) without using pre-defined functions.  
     
   iii. WA Menu Driven Program to implement a stack for these book details (book\_no and book\_name).
3. **Queue**i. WA Menu Driven Program to illustrate the Queue Implementation consisting of Insert, Delete and Display. (You May use predefined functions)  
     
   ii. WA Menu Driven Program to illustrate the Queue implementation consisting of Insert, Delete, Display and Seek ( Which gives the positions of the front and rear pointer) without using pre-defined functions.  
     
   iii. WA Menu Driven program to implement a queue for the employee details ( Emp\_id , Emp\_name ).

* **Boolean Algebra**i. Assume a car race field having 4 tracks. A signal comes on ( ie; Output of 1 ) under the following conditions. Develop a truth table for the race track.  
   a) When exactly two cars on the track  
   b) when 4 cars are on the track.  
   c) when exactly 3 cars are on the track.  
  based on the Truth Table : (i) Get the SOP expression . (ii) Simplify using K-Map and (iii) Implement the circuit.  
    
  ii. 4 Combination circuit having 4 inputs and 1 output produced Zero when:  
   a) All inputs are 0.  
   b) None of the inputs are 0.  
   c) Odd number of 0 inputs.   
   Draw the truth table. Obtain the POS expression. Minimize using K-Map and Implement the circuit.
* **SQL**

**1) CARDEN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ccode | CarName | Make | Color | Capacity | Charges |
| 501 | A-Star | Suzuki | RED | 3 | 14 |
| 503 | Indigo | Tata | SILVER | 3 | 12 |
| 502 | Innova | Toyota | WHITE | 7 | 15 |
| 509 | SX4 | Suzuki | SILVER | 4 | 14 |
| 510 | C Class | Mercedes | RED | 4 | 35 |

CUSTOMER

|  |  |  |
| --- | --- | --- |
| Code | CName | CCode |
| 1001 | Hemant Sahu | 501 |
| 1002 | Raj Lal | 509 |
| 1003 | Feroza Shah | 503 |
| 1004 | Ketan Dhal | 502 |

WRITE SQL COMMANDS FOR THE FOLLOWING:

1. To display the names of all the silver colored cars.
2. To display name of car, make and capacity of cars in descending order of their sitting capacity.
3. To display the highest charges at which a vehicle can be hired from CARDEN.
4. To display the customer name and the corresponding name of the Cars hired by them.

GIVE THE OUTPUT FOR THE SQL QUERIES:

1. SELECT COUNT(DISTINCT Make) FROM CARDEN ;
2. SELECT MAX(Charges),MIN(Charges) FROM CARDEN;
3. SELECT COUNT(\*), Make FROM CARDEN;
4. SELECT CarName FROM CARDEN WHERE Capacity = 4 ;

2)

PRODUCTS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PId | PName | Qty | Price | Company | SupCode |
| 101 | Digital Camera 14X | 120 | 12000 | Renix | S01 |
| 102 | Digital Pad 11i | 100 | 22000 | Digipop | S02 |
| 104 | Pen Drive 16GB | 500 | 1100 | StoreKing | S01 |
| 106 | LED Screen 32” | 70 | 28000 | Dispexperts | S02 |
| 105 | CAR GPS System | 60 | 12000 | MoveOn | S03 |

SUPPLIERS

|  |  |  |
| --- | --- | --- |
| SupCode | Sname | City |
| **S01** | **Get All Inc.** | **Kolkata** |
| **S03** | **Easy Market Corp.** | **Delhi** |
| **S02** | **Digi Busy Group.** | **Chennai** |

WRITE SQL COMMANDS FOR THE FOLLOWING:

1. To display the details of all the products in ascending order of product names (PName)
2. To display product name and price of all those products whose price is in the range of 10000 and 150000 (Both Values Inclusive)
3. To Display the number of products, which are supplied by each supplier ie; the expected output should be:   
   S01 2   
   S02 2  
   S03 1
4. To Display the price, product name and Qty of those products which have Qty more than 100.
5. To display the names of those suppliers who are either from Delhi or Chennai
6. To display the name of the companies and the name of the products in descending order of company names.

OBTAIN THE OUTPUT OF THE FOLLOWING SQL COMMANDS

1. SELECT DISTINCT(SupCode) FROM PRODUCTS;
2. SELECT MAX(Price), MIN(Price) FROM PRODUCTS;
3. SELECT Price\*Qty AMOUNT FROM PRODUCTS WHERE PId = 104;
4. SELECT PName , SName FROM PRODUCTS P , SUPPLIERS S WHERE P.SupCode = S.SupCode AND Qty>100;

3)

SHOPPE

|  |  |  |
| --- | --- | --- |
| Id | SName | Area |
| S001 | ABC Computeronics | CP |
| S002 | All Infotech Media | GK II |
| S003 | Tech Shoppe | CP |
| S004 | Geeks Techno Soft | NEHRU PLACE |
| S005 | HighTech Tech Store | NEHRU PLACE |

ACCESSORIES

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Name | Price | Id |
| A01 | Motherboard | 12000 | S01 |
| A02 | Hard Disk | 5000 | S01 |
| A03 | Keyboard | 500 | S02 |
| A04 | Mouse | 300 | S01 |
| A05 | Motherboard | 13000 | S02 |
| A06 | Keyboard | 400 | S03 |
| A07 | LCD | 6000 | S04 |
| T08 | LCD | 5500 | S05 |
| T09 | Mouse | 350 | S05 |
| T10 | Hard Disk | 4500 | S03 |

WRITE THE SQL QUERIES :

1. To Display Name , And Price of all the accessories in ascending order of their price.
2. To display Id and SName of all SHOPPE located in NEHRU PLACE
3. To display Minimum and Maximum Price of each name of accessories
4. To display name, price of all accessories and their respective SName where they are available.

WRITE THE OUTPUT OF THE FOLLOWING SQL COMMANDS:

1. SELECT DISTINCT(Name) FROM ACCESSORIES WHERE Price >= 5000;
2. SELECT Area, COUNT(\*) FROM SHOPPE GROUP BY Area;
3. SELECT COUNT(DISTINCT Area) FROM SHOPPE;
4. SELECT Name, Price\*0.05 DISCOUNT FROM ACCESSORIES WHERE SNo IN ( ‘S02’ , ‘S03’) ;