CLASS: XII **SUBJECT: COMPUTER SCIENCE PRACTICALS (083)**

SL. NO	PROGRAM
9	Menu-driven program to create user defined functions to perform the following Text file (Story.txt) operations: a) To write data to a text file (Any Sample text data) b) To read the entire contents of the file and display it c) To print the statistics of the file. Read the text file and display the total no. of characters, Total no of words and total no of digits.
10	Menu-driven program to create user defined functions to perform the following Text file (Story.txt) operations: a) Read the contents of the text file line by line and display it.(File should be created before) b) Display all the lines starting with a letter‘s’. c) Display each word separated by a # symbol.
11	Menu-driven program to perform the following operations on a binary file ‘Player.dat’, which has the following structure: Player_id, Player_name, Player_game]. a) Create a function File_Create () to add records into the file. b) Create a function File_Read () to read the records c) Create a function Search_Data () to search for a particular Player using his Player_id. If found, print the Player details. Otherwise print appropriate message.
12	Menu-driven program to perform the following operations on a binary file ‘Employee.dat’, which has the structure as [empid, empname, salary, design]. a) Create a function File_Create () to add records into the file.

	b) Create a function File_Read () to read the records (employee data) from file. c) Create a function Update_Data () to update the salary of employees whose designation is 'Manager' by Rs.5000/- and display the updated data
13	Menu-driven program to perform the following operations on a CSV file 'Login.csv', which has the structure as [uid, passwd]: a) Create a function Create_csv () to add records to the file. b) Create a function Read_csv() to read and display all the records from the file. c) Create a function Search_csv() to search whether a user exists or not with the uid which is passed as an argument.
14	Menu-driven program to perform the following operations on a CSV file 'Address.csv', which has the structure [Name, Mob_no, email] a) Create a function Create_csv () to add records to the file. b) Create a function Read_csv() to read and display all the records from the file. c) Create a function Search_csv() to search for a mobile number, if found then write only those records to a new csv file 'NewAddress.csv'. If not found, print appropriate message.
15	Menu-driven program to implement Stack using a list data-structure, to perform the following operations: a) To Push an object containing Doc_ID and Doc_name of doctors who specialize in "ENT" to the stack. b) To Pop the objects from the stack and display them. c) To display the elements of the stack (after performing PUSH or POP)
16	Menu driven program to integrate MySQL with Python and perform the following operations: a) Create a database school, create a table student with fields as Rollno, Stu_Name, Stream and Percentage. Then insert 5 records into it. b) Display all the records from the table
17	Menu driven program to integrate MySQL with Python and perform the following operations: a) Create a database Company, create a table Product with fields as Pid, Pname, Qty and Price. Then insert 5 records into it. b) Display all the records from the table c) Read records from student table and search for a particular student based on the Pid.
18	Menu driven program to integrate MySQL with Python and perform the following operations: a) Create a database Company, create a table Employee with fields as Empid, Empname, designation and Salary. Then insert 5 records into it. b) Update the salary of an employee by 5000 whose designation as 'Manager' c) Display all the records from the table.

	d) Delete the records from employee based on their empid.
19	Menu driven program to integrate MySQL with Python and perform the following operations: a) Create a database Company, create a table create a table Client with fields as C_ID, ClientName, City. b) Read records from the table and display those records whose city is 'Delhi'. c) Delete the records from table based on their client id d) Display all the records.
	SQL PROGRAMS
20	Write queries (a) to (d) based on the table EMPLOYEE:

EMPID	NAME	DOB	DEPTID	DESIG	SALARY
120	Alisha	23-Jan-1978	D001	Manager	75000
123	Nitin	10-Oct-1977	D002	AO	59000
129	Navjot	12-Jul-1971	D003	Supervisor	40000
130	Jimmy	30-Dec-1980	D004	Sales Rep	
131	Faiz	06-Apr-1984	D001	Dep Manager	65000

(a) To display the average salary of all employees, department wise.
(b) To display name and designation of employees whose salary is more than 50000.
(c) To display the names of employees whose salary is not known, in alphabetical order
(d) To display the distinct designations from the table Employee.

21 Write SQL commands for the queries (a)-(d) based on the table CUSTOMER:

CUSTID	NAME	UNITPRICE	QTY	PID
101	REENA SONI	60,000	10	2102
102	MICHAEL PAUL	50,000	20	2106
103	MEETALI SINGH	70,000	15	2101
104	PARUL SOHAL	55,000	3	2103
105	RAJESH DESWAL	45,000	7	2104

(a) To display the name of the customers in reverse alphabetical order.
(b) To display the details of customers whose name starts with 'M'
(c) To increase the unit price of products by 1000 whose quantity is more than 50.

(d) To display the total count of rows in the table Customer.

22 Write SQL commands for the queries (a)-(d) based on the table PASSENGER:

NO	NAME	TRAVELDATE	EKG	PAXCODE	NOB
11	Ridhima Sen	2019-12-13	10	101	3
13	Gary Sandhu	2019-12-21	15	103	3
15	Ganesh V	2019-12-23	8	102	2
12	Salim Malik	2019-11-13	15	102	3
17	Geralene N	2019-12-10	11	104	2
14	Venkatesh	2019-11-28	8	105	4
16	Saed Bajwa	2019-11-06	20	101	3

(a) To display NO, NAME, TRAVELDATE from the table PASSENGER in descending order of NO.
(b) To display all the details of passengers whose number of bags (NOB) are more than 2.
(c) To display the PAXCODE and their number from PASSENGER table, grouped by the PAXCODE more than 1 in number.
(d) To display the PAXCODE, NAME, TRAVELDATE where Extra KG (EKG) is in the range of 10 and 20.

23 Write SQL commands for the queries (a)-(d) based on the table STUDENTS:

StudentID	Name	DOB	City	Sport	Email
1211	Anvi	2005-10-06	Chandigarh	Tennis	anvi@csmail.com
1212	Ananya	2005-09-11	Panchkula	TT	ana@csmail.com
1213	Bisman	2005-07-01	Mohali	Cricket	bisn@csmail.com
1214	Bani	2004-12-11	Chandigarh	TT	bani@csmail.com
1215	Brie	2005-03-16	Chandigarh	Squash	brie@csmail.com
1216	Diya	2006-07-11	Panchkula	Tennis	diya@csmail.com
1217	Fiza	2005-07-21	Mohali	Cricket	fiza@csmail.com
1218	Gauri	2004-12-30	Chandigarh	TT	Gari@csmail.com

(a) To display the StudentID, Name and Email of the students from Chandigarh who play TT.
(b) To display the count of students who are playing each of the sports from Students table.
(c) To display distinct City from the table Students.
(d) To display the details of students whose had born in the year 2005.

24 Write SQL commands for the queries (a)-(d) based on the table STATIONARY:

S_ID	ST_NAME	COMPANY	PRICE	STOCKDATE
DP01	DOTPEN	ABC	10	2011-03-31
PL02	PENCIL	XYZ	6	2010-01-01
ER05	ERASER	XYZ	7	2010-02-14
PL01	PENCIL	CAM	5	2009-01-09
GP02	GEL PEN	ABC	15	2009-03-19

a) To display details of all the Stationary items in the stationary table in descending order of Stock date.
b) To display the details of that stationary item whose company is XYZ and price is below 10.
c) To display the details of the stationary items whose name starts with 'P'.
d) To display the company name without repetition.