Lab-6 Trigger

Create Database with Name: **Person_LogInfo**

Create following table under Person_LogInfo database. (Using Design Mode)

Person		
Column_Name	**DataType**	**Constraints**
PersonID	Int	Primary Key
PersonName	Varchar (100)	Not Null
Salary	Decimal (8,2)	Not Null
JoiningDate	Datetime	Not Null
City	Varchar (100)	Not Null
Age	Int	Null
BirthDate	Datetime	Not Null

PersonLog		
Column_Name	**DataType**	**Constraints**
PLogID	Int	Primary Key, Auto increment
PersonID	Int	Not Null
PersonName	Varchar (250)	Not Null
Operation	Varchar (50)	Not Null
UpdateDate	Datetime	Not Null

From the above given tables perform the following queries:

Part – A

1. Create a trigger that fires on INSERT, UPDATE and DELETE operation on the Person table to display a message “Record is Affected.”
2. Create a trigger that fires on INSERT, UPDATE and DELETE operation on the Person table. For that, log all operations performed on the person table into PersonLog.
3. Create an INSTEAD OF trigger that fires on INSERT, UPDATE and DELETE operation on the Person table. For that, log all operations performed on the person table into PersonLog.
4. Create a trigger that fires on INSERT operation on the Person table to convert person name into uppercase whenever the record is inserted.

Part – B

5. Create a trigger that fires on INSERT operation on person table, which calculates the age and update that age in Person table.

Part – C

6. Create DELETE trigger on PersonLog table, when we delete any record of PersonLog table it prints ‘Record deleted successfully from PersonLog’.

Lab-7 Cursor

Create Database with Name: **Product_Info**

Create following table under Product _Info database. (Using Design Mode)

Products		
Column_Name	**DataType**	**Constraints**
Product_id	Int	Primary Key
Product_Name	Varchar (250)	Not Null
Price	Decimal (10,2)	Not Null

Products		
Product_id	**Product_Name**	**Price**
1	Smatphone	35000

2	Laptop	65000
3	Headphones	5500
4	Television	85000
5	Gaming Console	32000

From the above given tables perform the following queries:

Part - A

1. Create a cursor Product_Cursor to fetch all the rows from a products table.
2. Create a cursor Product_Cursor_Fetch to fetch the records in form of ProductID_ProductName. (Example: 1_Smartphone)
3. Create a cursor Product_CursorDelete that deletes all the data from the Products table.

Part – B

4. Create a cursor Product_CursorUpdate that retrieves all the data from the products table and increases the price by 10%.

Part – C

5. Create a cursor to insert details of Products into the NewProducts table if the product is "Laptop" (Note: Create NewProducts table first with same fields as Products table)

Lab-8
Exception Handling

Create Database with Name: **Customer_Info**

Create following table under Customer_Info database. (Using Design Mode)

Customers		
Column_Name	**DataType**	**Constraints**
Customer_id	Int	Primary Key
Customer _Name	Varchar (250)	Not Null
Email	Varchar (50)	Unique

Orders		
Column_Name	**DataType**	**Constraints**
Order_id	Int	Primary Key
Customer_id	Int	Foreign Key
Order_date	date	Not Null

From the above given tables perform the following queries:

Part – A

1. Handle Divide by Zero Error and Print message like: Error occurs that is - Divide by zero error.
2. Try to convert string to integer and handle the error using try…catch block.
3. Create a procedure that prints the sum of two numbers: take both numbers as integer & handle exception with all error functions if any one enters string value in numbers otherwise print result.
4. Handle a Primary Key Violation while inserting data into customers table and print the error details such as the error message, error number, severity, and state.
5. Throw custom exception using stored procedure which accepts Customer_id as input & that throws Error like no Customer_id is available in database.

Part – B

6. Handle a Foreign Key Violation while inserting data into Orders table and print appropriate error message.
7. Throw custom exception that throws error if the data is invalid.

Part – C

8. Create a procedure which prints the error message that "The Customer_id is already taken. Try another one".

Lab-9 Mongo DB

Create Database with Name: **EMPLOYEE_INFO**

employee

EID	ENAME	GENDER	JOININGDATE	SALARY	CITY
1	Nick	Male	01-JAN-13	4000	London
2	Julian	Female	01-OCT-14	3000	New York
3	Roy	Male	01-JUN-16	3500	London
4	Tom	Male	NULL	4500	London
5	Jerry	Male	01-FEB-13	2800	Sydney
6	Philip	Male	01-JAN-15	7000	New York
7	Sara	Female	01-AUG-17	4800	Sydney
8	Emily	Female	01-JAN-15	5500	New York
9	Michael	Male	NULL	6500	London
10	John	Male	01-JAN-15	8800	London

From the above given collection perform the following queries:

Part - A

1. Display employees whose gender is Male.
2. Display employees who belong to London city.
3. Display employees whose salary is greater than 3500.
4. Display employees whose joining date is before 2015-01-01.
5. Display employees whose EID is greater than or equal to 7.
6. Display employees whose city is Landon or New York (use:IN)
7. Display employees who do not belongs to Landon or New York (use: NOT IN)
8. Display the EID of those employee who lives in city London.
9. Display first 2 employee names who lives in New york.
10. Display next 2 employee after skipping first 2 whose city is London.
11. Display Male employees who lives Sydney.
12. Display EID, ENAME, CITY and SALARY of those employees who belongs to London or Sydney.
13. Display ENAME, SALARY, and CITY of those employee whose salary is more than 7000.
14. Display documents whose name start with E.
15. Display documents whose name starts with S or M in your collection.
16. Display documents where city starts with A to M in your collection.
17. Display documents where city name ends in 'ney'.
18. Display employee info whose name contains n. (Both uppercase(N) and lowercase(n))
19. Display employee info whose name starts with E and having 5 characters.
20. Display employee whose name start with S and ends in a.
21. Display EID, ENAME, CITY and SALARY whose name starts with 'Phi'.
22. Display ENAME, JOININGDATE and CITY whose city contains 'dne' as three letters in city name.
23. Display ENAME, JOININGDATE and CITY who does not belongs to city London or Sydney.
24. Delete the documents whose city is New York.
25. Update ENAME of Nick to 'Naysa' and GENDER to 'Female'.

Part – B

Create Database with Name: **STUDENT_INFO**

student

ROLLNO	SNAME	DEPARTMENT	FEES	SEM	GENDER	CITY
101	Vina	CE	15000	3	Female	Rajkot

102	Krisha	EC	8000	5	Female	Ahmedabad
103	Priti	Civil	12000	7	Female	Baroda
104	Mitul	CE	15000	3	Male	Rajkot
105	Keshav	CE	15000	3	Male	Jamnagar
106	Zarna	Civil	12000	5	Female	Ahmedabad
107	Nima	EE	9000	5	Female	Rajkot
108	Dhruv	Mechanical	10000	5	Male	Rajkot
109	Krish	Mechanical	10000	7	Male	Baroda
110	Zeel	EE	9000	3	Female	Jamnagar

From the above given collection perform the following queries:

26. Display Female students.
27. Display students who belong to Rajkot city.
28. Display students studying in 7th sem.
29. Display students not studying in 3rd sem.
30. Display students whose roll no is greater than 107.
31. Display students whose city is Jamnagar or Baroda (use:IN)
32. Display students whose fees is less than 9000.
33. Display the roll no of those students who belongs to Mechanical department.
34. Display first 2 students names who lives in Baroda.
35. Display Male students who studying in 3rd sem.
36. Display sname and city and fees of those students whose roll no is less than 105.
37. Display documents where sname start with K.
38. Display documents where sname starts with Z or D in your collection.
39. Display documents where city starts with A to R in your collection.
40. Display students' info whose name start with P and ends in i.
41. Display students' info whose department name starts with 'C'.
42. Display name, sem, fees, and department whose city contains 'med' as three letters somewhere in city name.
43. Display name, sem, fees, and department who does not belongs to city Rajkot or Baroda.
44. Delete the documents whose city is Jamnagar.
45. Update sname of Krish to 'fenny' and gender to 'Female'.

Part – C

46. Display next 2 students after skipping first 2 whose city is Ahmedabad.
47. Display rollno, sname, fees, and department of those students who is from Baroda and belongs to CE department.
48. Display documents where city name ends in 'oda'.
49. Display students' info whose name contains v. (Both uppercase(V) and lowercase(v))
50. Display students' info whose name starts with V and having 4 characters.

Lab-10 Mongo DB

From the above given collection perform the following queries:

Part – A (Use employee collection of Lab 9)

1. Display distinct city.
2. Display city wise number of persons.
3. Display sum of salary in your collection.
4. Display average of salary in your document.
5. Display maximum and minimum salary of your document.

	6. Display city wise total salary in your collection. 7. Display gender wise maximum salary in your collection. 8. Display city wise maximum and minimum salary. 9. Display count of persons lives in Sydney city in your collection. 10. Display average salary of New York city. **Part – B (Use student collection of Lab 9)** 11. Display distinct department. 12. Display city wise number of students. 13. Display sum of fees in your collection. 14. Display average of fees in your document. 15. Display maximum and minimum fees of your document. **Part – C (Use student collection of Lab 9)** 16. Display department wise total fees in your collection. 17. Display gender wise maximum fees in your collection. 18. Display department maximum and minimum fees. 19. Display count of persons lives in Rajkot city in your collection. 20. Display department wise number of students.