Subject :2305CS101 – Database Management Systems

<table>
<tr><th>Sr.</th><th>Practical</th></tr>
<tr><td>1</td><td>1. What is SQL?
2. Introduction to Editor (SQL Server Management Studio).
3. Components of SQL (DDL, DML, DCL, DQL, TCL)</td></tr>
<tr><td>2</td><td>1. Introduction to Database, Table, Field, Row, Record.
2. Introduction to various data types INT, CHAR, VARCHAR, DATETIME, BIT, DECIMAL</td></tr>
<tr><td>3</td><td>Database Name: BANK_INFO
Part A:
Create following tables under BANK_INFO database.
<table>
<tr><th colspan="2">DEPOSIT</th></tr>
<tr><th>Column_Name</th><th>DataType</th></tr>
<tr><td>ACTNO</td><td>INT</td></tr>
<tr><td>CNAME</td><td>VARCHAR(50)</td></tr>
<tr><td>BNAME</td><td>VARCHAR(50)</td></tr>
<tr><td>AMOUNT</td><td>DECIMAL(8,2)</td></tr>
<tr><td>ADATE</td><td>DATETIME</td></tr>
</table>
<table>
<tr><th colspan="2">BRANCH</th></tr>
<tr><th>Column_Name</th><th>DataType</th></tr>
<tr><td>BNAME</td><td>VARCHAR(50)</td></tr>
<tr><td>CITY</td><td>VARCHAR(50)</td></tr>
</table>
<table>
<tr><th colspan="2">CUSTOMERS</th></tr>
<tr><th>Column_Name</th><th>DataType</th></tr>
<tr><td>CNAME</td><td>VARCHAR(50)</td></tr>
<tr><td>CITY</td><td>VARCHAR(50)</td></tr>
</table>
<table>
<tr><th colspan="2">BORROW</th></tr>
<tr><th>Column_Name</th><th>DataType</th></tr>
<tr><td>LOANNO</td><td>INT</td></tr>
<tr><td>CNAME</td><td>VARCHAR(50)</td></tr>
<tr><td>BNAME</td><td>VARCHAR(50)</td></tr>
<tr><td>AMOUNT</td><td>DECIMAL(8,2)</td></tr>
</table>
Insert the data into tables using Query as shown below.
DEPOSIT
<table>
<tr><th>ACTNO</th><th>CNAME</th><th>BNAME</th><th>AMOUNT</th><th>ADATE</th></tr>
<tr><td>101</td><td>ANIL</td><td>VRCE</td><td>1000.00</td><td>1-3-95</td></tr>
<tr><td>102</td><td>SUNIL</td><td>AJNI</td><td>5000.00</td><td>4-1-96</td></tr>
<tr><td>103</td><td>MEHUL</td><td>KAROLBAGH</td><td>3500.00</td><td>17-11-95</td></tr>
<tr><td>104</td><td>MADHURI</td><td>CHANDI</td><td>1200.00</td><td>17-12-95</td></tr>
<tr><td>105</td><td>PRMOD</td><td>M.G. ROAD</td><td>3000.00</td><td>27-3-96</td></tr>
<tr><td>106</td><td>SANDIP</td><td>ANDHERI</td><td>2000.00</td><td>31-3-96</td></tr>
<tr><td>107</td><td>SHIVANI</td><td>VIRAR</td><td>1000.00</td><td>5-9-95</td></tr>
<tr><td>108</td><td>KRANTI</td><td>NEHRU PLACE</td><td>5000.00</td><td>2-7-95</td></tr>
<tr><td>109</td><td>MINU</td><td>POWAI</td><td>7000.00</td><td>10-8-95</td></tr>
</table></td></tr>
</table>

BRANCH

BNAME	CITY
VRCE	NAGPUR
AJNI	NAGPUR
KAROLBAGH	DELHI
CHANDI	DELHI
DHARAMPETH	NAGPUR
M.G. ROAD	BANGLORE
ANDHERI	BOMBAY
VIRAR	BOMBAY
NEHRU PLACE	DELHI
POWAI	BOMBAY

CUSTOMERS

CNAME	CITY
ANIL	CALCUTTA
SUNIL	DELHI
MEHUL	BARODA
MANDAR	PATNA
MADHURI	NAGPUR
PRAMOD	NAGPUR
SANDIP	SURAT
SHIVANI	BOMBAY
KRANTI	BOMBAY
NAREN	BOMBAY

BORROW

LOANNO	CNAME	BNAME	AMOUNT
201	ANIL	VRCE	1000.00
206	MEHUL	AJNI	5000.00
311	SUNIL	DHARAMPETH	3000.00
321	MADHURI	ANDHERI	2000.00
375	PRMOD	VIRAR	8000.00
481	KRANTI	NEHRU PLACE	3000.00

From the above given tables perform the following queries (SELECT Operation):

1. Retrieve all data from table DEPOSIT.
2. Retrieve all data from table BORROW.
3. Retrieve all data from table CUSTOMERS.
4. Display Account No, Customer Name & Amount from DEPOSIT.
5. Display Loan No, Amount from BORROW.
6. Display loan details of all customers who belongs to ‘ANDHERI’ branch.
7. Give account no and amount of depositor, whose account no is equals to 106.
8. Give name of borrowers having amount greater than 5000.
9. Give name of customers who opened account after date '1-12-96'.
10. Display name of customers whose account no is less than 105.

Part B:

1. Display name of customer who belongs to either ‘NAGPUR’ Or ‘DELHI’. (**OR & IN**)
2. Display name of customers with branch whose amount is greater than 4000 and account no is less than 105.

3. Find all borrowers whose amount is greater than equals to 3000 & less than equals to 8000. (**AND & BETWEEN**)
4. Find all depositors who do not belongs to 'ANDHERI' branch.
5. Display Account No, Customer Name & Amount of such customers who belongs to 'AJNI', 'KAROLBAGH' Or 'M.G.ROAD' and Account No is less than 104.
6. Display all the details of first five customers.
7. Display all the details of first three depositors whose amount is greater than 1000.
8. Display Loan No, Customer Name of first five borrowers whose branch name does not belongs to 'ANDHERI'.
9. Retrieve all unique cities using DISTINCT. **(**Use **Customers Table)**
10. Retrieve all unique branches using DISTINCT. **(**Use **Branch Table)**
11. Retrieve all the records of customer table as per their city name in ascending order.
12. Retrieve all the records of deposit table as per their amount column in descending order.

Part C:

From the above given tables perform the following queries (UPDATE Operation):

1. Update deposit amount of all customers from 3000 to 5000.
2. Change branch name of ANIL from VRCE to C.G. ROAD. (Use **Borrow Table**)
3. Update Account No of SANDIP to 111 & Amount to 5000.
4. Give 10% Increment in Loan Amount.
5. Update deposit amount of all depositors to 5000 whose account no between 103 & 107.
6. Update amount of loan no 321 to *NULL*.
7. Display the name of borrowers whose amount is *NULL*.

4

Part A:

Create following table using query according to the definition.

Employee	
Column_Name	**DataType**
EmpNo	INT
EmpName	VARCHAR(25)
JoiningDate	DATETIME
Salary	DECIMAL (8,2)
City	VARCHAR(20)

Insert the following records in the Employee table.

EmpNo	EmpName	JoiningDate	Salary	City
101	Keyur	5-1-02	12000.00	Rajkot
102	Hardik	15-2-04	14000.00	Ahmedabad
103	Kajal	14-3-06	15000.00	Baroda
104	Bhoomi	23-6-05	12500.00	Ahmedabad
102	Harmit	15-2-04	14000.00	Rajkot

From the above given tables perform the following queries (DELETE Operation):

1. Delete all the records of Employee table having salary greater than and equals to 14000.
2. Delete all the Employees who belongs to 'RAJKOT' city.
3. Delete all the Employees who joined after 1-1-2007.
4. Delete all the records of Employee table. (Use **Truncate**)
5. Remove Employee table. (Use **Drop**)

Part B:

Create table and Insert the following records in the Account table.

Account

ACTNO	CNAME	BNAME	AMOUNT	ADATE
101	ANIL	VRCE	1000.00	1-3-95
102	SUNIL	AJNI	5000.00	4-1-96
103	MEHUL	KAROLBAGH	3500.00	17-11-95
104	MADHURI	CHANDI	1200.00	17-12-95
105	PRMOD	M.G. ROAD	3000.00	27-3-96
106	SANDIP	ANDHERI	2000.00	31-3-96
107	SHIVANI	VIRAR	1000.00	5-9-95
108	KRANTI	NEHRU PLACE	5000.00	2-7-95
109	MINU	POWAI	7000.00	10-8-95

1. Delete all the records of Account table having amount greater than and equals to 4000.
2. Delete all the accounts Bname is CHANDI.
3. Delete all the accounts having adate after 1-10-1995.
4. Delete all the records of Account table. (Use **Truncate**)
5. Remove Account table. (Use **Drop**)

Part C:

Create table and Insert the following records in the Account table.

Account

ACTNO	CNAME	BNAME	AMOUNT	ADATE
101	ANIL	VRCE	1000.00	1-3-95
102	SUNIL	AJNI	5000.00	4-1-96
103	MEHUL	KAROLBAGH	3500.00	17-11-95
104	MADHURI	CHANDI	1200.00	17-12-95
105	PRMOD	M.G. ROAD	3000.00	27-3-96
106	SANDIP	ANDHERI	2000.00	31-3-96
107	SHIVANI	VIRAR	1000.00	5-9-95
108	KRANTI	NEHRU PLACE	5000.00	2-7-95
109	MINU	POWAI	7000.00	10-8-95

1. Update the amount of Anil to 5000.
2. Update amount of actno 109 to *NULL*.
3. Retrieve all the records of account table as per their bname in descending order.
4. Retrieve all the records of account table whose amount is between 3000 and 5000.
5. Remove Account table. (Use **Drop**)

Subject :2305CS101 – Database Management Systems

5

Part A:

Create following table using query according to the definition.

Student	
Column_Name	**DataType**
Enrollment_No	VARCHAR(20)
Name	VARCHAR(25)
CPI	DECIMAL(5,2)
Birthdate	DATETIME

From the above given tables perform the following queries (ALTER Operation):

1. Add two more columns City VARCHAR (20) and Backlog INT.
2. Change the size of NAME column of student from VARCHAR (25) to VARCHAR (35).
3. Change the data type DECIMAL to INT in CPI Column.
4. Rename Column Enrollment No to ENO.
5. Delete Column City from the STUDENT table.
6. Change name of table STUDENT to STUDENT_MASTER.
7. Remove the table STUDENT_MASTER.

Part B:

Create following table using query according to the definition.

DEPOSIT	
Column_Name	**DataType**
ACTNO	INT
CNAME	VARCHAR(50)
BNAME	VARCHAR(50)
AMOUNT	DECIMAL(8,2)
ADATE	DATETIME

From the above given tables perform the following queries (ALTER Operation):

1. Add two more columns City VARCHAR (20) and Pincode INT.
2. Change the size of CNAME column from VARCHAR (50) to VARCHAR (35).
3. Change the data type DECIMAL to INT in amount Column.
4. Rename Column ActNo to ANO.
5. Delete Column City from the DEPOSIT table.

Part C:

1. Delete Column ADATE from the DEPOSIT table.
2. Rename Column CNAME to CustomerName.
3. Change name of table DEPOSIT to DEPOSIT_DETAIL.
4. Remove the table DEPOSIT_DETAIL.

6

Part-A:

Create following table using query according to the definition.

Student	
Column_Name	**DataType**
StuID	INT
FirstName	VARCHAR(25)
LastName	VARCHAR(25)
Website	VARCHAR(50)
City	VARCHAR(25)

Subject :2305CS101 – Database Management Systems

Insert the following records in the Student table.

StuID	FirstName	LastName	Website	City
1011	Keyur	Patel	techonthenet.com	Rajkot
1022	Hardik	Shah	digminecraft.com	Ahmedabad
1033	Kajal	Trivedi	bigactivities.com	Baroda
1044	Bhoomi	Gajera	checkyourmath.com	Ahmedabad
1055	Harmit	Mitel	NULL	Rajkot
1066	Ashok	Jani	NULL	Baroda

From the above given tables perform the following queries (LIKE Operation):

1. Display the name of students whose name starts with 'k'.
2. Display the name of students whose name consists of five characters.
3. Retrieve the first name & last name of students whose city name ends with a & contains six characters.
4. Display all the students whose last name ends with 'tel'.
5. Display all the students whose first name starts with 'ha' & ends with't'.
6. Display all the students whose first name starts with 'k' and third character is 'y'.

Part-B:

1. Display the name of students having no website and name consists of five characters.
2. Display all the students whose last name consist of 'jer'.
3. Display all the students whose city name starts with either 'r' or 'b'.
4. Display all the name students having websites.
5. Display all the students whose name starts from alphabet A to H.
6. Display all the students whose name's second character is vowel.

Part-C:

1. Display the name of students having no website and name consists of minimum five characters.
2. Display all the students whose last name starts with 'Pat'.
3. Display all the students whose city name does not start starts with 'b'.
4. Display all the students whose name starts from alphabet A or H.
5. Display all the students whose name's second character is vowel and of and start with H.

7 **Math functions**

Part-A:

1. Display the result of 5 multiply by 30.
2. Find out the absolute value of -25, 25, -50 and 50.
3. Find smallest integer value that is greater than or equal to 25.2, 25.7 and -25.2.
4. Find largest integer value that is smaller than or equal to 25.2, 25.7 and -25.2.
5. Find out remainder of 5 divided 2 and 5 divided by 3.
6. Find out value of 3 raised to 2^{nd} power and 4 raised 3^{rd} power.
7. Find out the square root of 25, 30 and 50.
8. Find out the square of 5, 15, and 25.
9. Find out the value of PI.
10. Find out round value of 157.732 for 2, 0 and -2 decimal points.
11. Find out exponential value of 2 and 3.
12. Find out logarithm having base e of 10 and 2.
13. Find out logarithm having base b having value 10 of 5 and 100.
14. Find sine, cosine and tangent of 3.1415.
15. Find sign of -25, 0 and 25.

16. Generate random number using function.

Part-B:

Create and Insert the following records in the Employee table.

EmpNo	EmpName	JoiningDate	Salary	Commission	City
101	Keyur	5-1-02	12000.00	4500	Rajkot
102	Hardik	15-2-04	14000.00	2500	Ahmedabad
103	Kajal	14-3-06	15000.00	3000	Baroda
104	Bhoomi	23-6-05	12500.00	1000	Ahmedabad
102	Harmit	15-2-04	14000.00	2000	Rajkot

1. Display the result of Salary plus Commission.
2. Find smallest integer value that is greater than or equal to 55.2, 35.7 and -55.2.
3. Find largest integer value that is smaller than or equal to 55.2, 35.7 and -55.2.
4. Find out remainder of 55 divided 2 and 55 divided by 3.
5. Find out value of 23 raised to 2nd power and 14 raised 3rd power.

Part-C:

1. Find out the square root of 36, 49 and 81.
2. Find out the square of 3, 9, and 12.
3. Find out round value of 280.8952 for 2, 0 and -2 decimal points.
4. Find sine, cosine and tangent of 4.2014.
5. Find sign of -55, 0 and 95.

8

String functions

Part-A:

1. Find the length of following. (I) NULL (II) ‘ hello ’ (III) Blank
2. Display your name in lower & upper case.
3. Display first three characters of your name.
4. Display 3rd to 10th character of your name.
5. Write a query to convert ‘abc123efg’ to ‘abcXYZefg’ & ‘abcabcabc’ to ‘ab5ab5ab5’ using REPLACE.
6. Write a query to display ASCII code for ‘a’,’A’,’z’,’Z’, 0, 9.
7. Write a query to display character based on number 97, 65,122,90,48,57.
8. Write a query to remove spaces from left of a given string ‘ hello world ‘.
9. Write a query to remove spaces from right of a given string ‘ hello world ‘.
10. Write a query to display first 4 & Last 5 characters of ‘SQL Server’.
11. Write a query to convert a string ‘1234.56’ to number (Use cast and convert function).
12. Write a query to convert a float 10.58 to integer (Use cast and convert function).
13. Put 10 space before your name using function.
14. Combine two strings using + sign as well as CONCAT ().
15. Find reverse of “Darshan”.
16. Repeat your name 3 times.

Part-B: Perform following queries on Student table of practical no 6.

1. Find the length of FirstName and LastName columns.
2. Display FirstName and LastName columns in lower & upper case.
3. Display first three characters of FirstName column.

<table>
<tr><td></td><td>
4. Display 3rd to 10th character of Website column.
5. Write a query to display first 4 & Last 5 characters of Website column.

Part-C: Perform following queries on Student table of practical no 6.
1. Put 10 space before FirstName using function.
2. Combine FirstName and LastName columns using + sign as well as CONCAT ().
3. Combine all columns using + sign as well as CONCAT ().
4. Find reverse of FirstName column.
5. Repeat FirstName column 3 times.
</td></tr>
<tr><td>9</td><td>
Date Functions
Part-A:

1. Write a query to display the current date & time. Label the column Today_Date.
2. Write a query to find new date after 365 day with reference to today.
3. Display the current date in a format that appears as may 5 1994 12:00AM.
4. Display the current date in a format that appears as 03 Jan 1995.
5. Display the current date in a format that appears as Jan 04, 96.
6. Write a query to find out total number of months between 31-Dec-08 and 31-Mar-09.
7. Write a query to find out total number of years between 25-Jan-12 and 14-Sep-10.
8. Write a query to find out total number of hours between 25-Jan-12 7:00 and 26-Jan-12 10:30.
9. Write a query to extract Day, Month, Year from given date 12-May-16.
10. Write a query that adds 5 years to current date.
11. Write a query to subtract 2 months from current date.
12. Extract month from current date using datename () and datepart () function.
13. Write a query to find out last date of current month.
14. Calculate your age in years and months.
Part-B:
Create a table Emp_detail and insert the following records in the table.
<table>
<tr><th>EmpNo</th><th>EmpName</th><th>JoiningDate</th><th>Salary</th><th>City</th></tr>
<tr><td>101</td><td>Keyur</td><td>15-1-02</td><td>12000.00</td><td>Rajkot</td></tr>
<tr><td>102</td><td>Hardik</td><td>15-2-04</td><td>14000.00</td><td>Ahmedabad</td></tr>
<tr><td>103</td><td>Kajal</td><td>14-3-06</td><td>15000.00</td><td>Baroda</td></tr>
<tr><td>104</td><td>Bhoomi</td><td>23-6-05</td><td>12500.00</td><td>Ahmedabad</td></tr>
<tr><td>102</td><td>Harmit</td><td>15-2-04</td><td>14000.00</td><td>Rajkot</td></tr>
<tr><td>105</td><td>Jay</td><td>12-3-07</td><td>12000.00</td><td>Surat</td></tr>
</table>
1. Write a query to find new date after 365 day with reference to JoiningDate.
2. Display the JoiningDate in a format that appears as may 5 1994 12:00AM.
3. Display the JoiningDate in a format that appears as 03 Jan 1995.
4. Display the JoiningDate in a format that appears as Jan 04, 96.
5. Write a query to find out total number of months between JoiningDate and 31-Mar-09.
6. Write a query to find out total number of years between JoiningDate and 14-Sep-10.
Part-C:
1. Write a query to extract Day, Month, Year from JoiningDate.
2. Write a query that adds 5 years to JoiningDate.
3. Write a query to subtract 2 months from JoiningDate.
4. Extract month from JoiningDate using datename () and datepart () function.
</td></tr>
</table>

	5. Calculate your age in years and months.
10	**SET Operators** **Part-A:** **Create below two tables as per following data.** **Computer** RollNo | Name 101 | Ajay 109 | Haresh 115 | Manish **Electrical** RollNo | Name 105 | Ajay 107 | Mahesh 115 | Manish 1. Display name of students who is either in Computer or in Electrical. 2. Display name of students who is either in Computer or in Electrical including duplicate data. 3. Display name of students who is in both Computer and Electrical. 4. Display name of students who are in Computer but not in Electrical. 5. Display name of students who are in Electrical but not in Computer. 6. Display all the details of students who are either in Computer or in Electrical. 7. Display all the details of students who are in both Computer and Electrical. **Part-B:** **Create below two tables as per following data.** **Employee** EID | Name 1 | Ajay 9 | Haresh 5 | Manish **Customer** CID | Name 5 | Ajay 7 | Mahesh 5 | Manish 1. Display name of persons who is either Employee or Customer. 2. Display name of persons who is either Employee or Customer including duplicate data. 3. Display name of persons who is both Employee as well as Customer. 4. Display name of persons who are Employee but not Customer. 5. Display name of persons who are Customer but not Employee. **Part-C: Perform all the queries of Part-B but display ID and Name columns instead of Name only.**
11	**Part-A:** **Create table as per following.** **Cricket** Name | City | Age Sachin Tendulkar | Mumbai | 30 Rahul Dravid | Bombay | 35 M. S. Dhoni | Jharkhand | 31 Suresh Raina | Gujarat | 30 1. Create table Worldcup from cricket with all the columns. 2. Create table T20 from cricket with first two columns with no data. 3. Create table IPL From Cricket with No Data 4. Insert the Data into IPL from Cricket Whose Second Character Should Be 'A' And String Should Have At least 7 Characters in Cricket Name Field. 5. Delete All the Rows From IPL. 6. Delete the Detail of Cricketer Whose City is Jharkhand. 7. Rename the Table IPL to IPL2018. 8. Destroy table T20 with All the Data.

Part-B:

Create table as per following.

Employee		
Name	City	Age
Jay Patel	Rajkot	30
Rahul Dave	Baroda	35
Jeet Patel	Surat	31
Vijay Raval	Rajkot	30

1. Create table Employee_detail from Employee with all the columns and data.
2. Create table Employee_data from Employee with first two columns with no data.
3. Create table Employee_info from Employee with no Data
4. Insert the Data into Employee_info from Employee Whose Second Character Should Be 'A' And String Should Have At least 7 Characters in Name Field.
5. Insert the Data into Employee_info from Employee Whose age is more than 32.

Part-C: Perform following queries on Employee table.

1. Delete all the Rows from Employee_info.
2. Delete the Detail of employee from Employee Whose City is Rajkot.
3. Rename the Table Employee to New_Employee.
4. Delete all the records from New_Employee table.
5. Destroy table New_Employee with all the Data.

Subject :2305CS101 – Database Management Systems

12

Part-A:

Create table and inset records as per below.

Employee					
EID	EName	Department	Salary	JoiningDate	City
101	Rahul	Admin	56000	1-Jan-90	Rajkot
102	Hardik	IT	18000	25-Sep-90	Ahmedabad
103	Bhavin	HR	25000	14-May-91	Baroda
104	Bhoomi	Admin	39000	8-Feb-91	Rajkot
105	Rohit	IT	17000	23-Jul-90	Jamnagar
106	Priya	IT	9000	18-Oct-90	Ahmedabad
107	Neha	HR	34000	25-Dec-91	Rajkot

1. Display the Highest, Lowest, Total, and Average salary of all employees. Label the columns Maximum, Minimum, Total_Sal and Average_Sal, respectively.
2. Find total number of employees of EMPLOYEE table.
3. Give maximum salary from IT department.
4. Count total number of cities of employee without duplication.
5. Display city with the total number of employees belonging to each city.
6. Display city having more than one employee.
7. Give total salary of each department of EMPLOYEE table.
8. Give average salary of each department of EMPLOYEE table without displaying the respective department name.
9. Give minimum salary of employee who belongs to Ahmedabad.
10. List the departments having total salaries more than 50000 and located in city Rajkot.

Part-B:

1. Count the number of employees living in Rajkot.
2. Display the difference between the highest and lowest salaries. Label the column DIFFERENCE.
3. Display the total number of employees hired before 1st January, 1991.
4. Display total salary of each department with total salary exceeding 35000 and sort the list by total salary.
5. List out department names in which more than two employees.

Part-C:

1. Count the number of employees living in Rajkot or Baroda.
2. Display the difference between the highest and lowest salaries. Label the column DIFFERENCE.
3. Display the total number of employees hired before 1st January, 1991 in IT department.
4. Display total salary of each department with total salary exceeding 35000 and sort the list by total salary in descending order.

13

Part-A:

JOINS (Create below tables as per following data)

Student		
Rno	Name	Branch
101	Raju	CE
102	Amit	CE
103	Sanjay	ME
104	Neha	EC
105	Meera	EE
106	Mahesh	ME

Result	
Rno	SPI
101	8.8
102	9.2
103	7.6
104	8.2
105	7.0
107	8.9

Employee		
EmployeeNo	Name	ManagerNo
E01	Tarun	NULL
E02	Rohan	E02
E03	Priya	E01
E04	Milan	E03
E05	Jay	E01
E06	Anjana	E04

1. Combine information from student and result table using cross join or Cartesian product.
2. Display Rno, Name, Branch and SPI of all students.

3. Display Rno, Name, Branch and SPI of CE branch's student only.
4. Display Rno, Name, Branch and SPI of other than EC branch's student only.
5. Display average result of each branch.
6. Display average result of each branch and sort them in ascending order by SPI.
7. Display average result of CE and ME branch.
8. Perform the left outer join on Student and Result tables.
9. Perform the right outer join on Student and Result tables.
10. Perform the full outer join on Student and Result tables.
11. Retrieve the names of employee along with their manager name from the Employee table.

Part-B:

Person

PersonID	PersonName	DepartmentID	Salary	JoiningDate	City
101	Rahul Tripathi	2	56000	01-01-2000	Rajkot
102	Hardik Pandya	3	18000	25-09-2001	Ahmedabad
103	Bhavin Kanani	4	25000	14-05-2000	Baroda
104	Bhoomi Vaishnav	1	39000	08-02-2005	Rajkot
105	Rohit Topiya	2	17000	23-07-2001	Jamnagar
106	Priya Menpara	NULL	9000	18-10-2000	Ahmedabad
107	Neha Sharma	2	34000	25-12-2002	Rajkot
108	Nayan Goswami	3	25000	01-07-2001	Rajkot
109	Mehul Bhundiya	4	13500	09-01-2005	Baroda
110	Mohit Maru	5	14000	25-05-2000	Jamnagar

Department

DepartmentID	DepartmentName	DepartmentCode	Location
1	Admin	Adm	A-Block
2	Computer	CE	C-Block
3	Civil	CI	G-Block
4	Electrical	EE	E-Block
5	Mechanical	ME	B-Block

From the above given table perform the following queries:

1. Find all persons with their department name & code.
2. Give department wise maximum & minimum salary with department name.
3. Find all departments whose total salary is exceeding 100000.
4. Retrieve person name, salary & department name who belongs to Jamnagar city.
5. Find all persons who does not belongs to any department.
6. Find department wise person counts.
7. Find average salary of person who belongs to Ahmedabad city.
8. Produce Output Like: <PersonName> earns <Salary> from department <DepartmentName> monthly. (In Single Column)
9. List all departments who have no persons.
10. Find city & department wise total, average & maximum salaries.

Part - C

1. Display Unique city names.
2. List out department names in which more than two persons.
3. Combine person name's first three characters with city name's last three characters in single column.
4. Give 10% increment in Computer department employee's salary.
5. Display all the person name's who's joining dates difference with current date is more than 365 days.

Subject :2305CS101 – Database Management Systems

14

Part-A: Create table as per following data.

City

CityID **(Primary Key)**	Name **(Unique Key)**	Pincode **(Not Null)**	Remakrs
1	Rajkot	360005	Good
2	Surat	335009	Very Good
3	Baroda	390001	Awesome
4	Jamnagar	361003	Smart
5	Junagadh	362229	Historic
6	Morvi	363641	Ceramic

Village

VID **(Primary Key)**	Name **(Not Null)**	CityID **(Foreign Key)**
101	Raiya	1
102	Madhapar	1
103	Dodka	3
104	Falla	4
105	Bhesan	5
106	Dhoraji	5

1. Display all the villages of Rajkot city.
2. Display city along with their villages & pin code.
3. Display the city having more than one village.
4. Display the city having no village.
5. Count the total number of villages in each city.
6. Count the number of cities having more than one village.

Create below table with following constraints

1. Do not allow SPI more than 10
2. Do not allow Bklog less than 0.
3. Enter the default value as 'General' in branch to all new records IF no other value is specified.

Student

Rno(PK)	Name	Branch	SPI	Bklog
101	Raju	CE	8.80	0
102	Amit	CE	2.20	3
103	Sanjay	ME	1.50	6
104	Neha	EC	7.65	0
105	Meera	EE	5.52	2
106	Mahesh		4.50	3

- ► Try to update SPI of Raju from 8.80 to 12.
- ► Try to update Bklog of Neha from 0 to -1.

Part-B: Create table as per following schema with proper validation and try to insert data which violate your validation.

1. Emp(Eid, Ename, Did, Cid, Salary, Experience)
   Dept(Did, Dname)
   City(Cid, Cname)

Part-C: Create table as per following schema with proper validation and try to insert data which violate your validation.

1. Emp(Eid, Ename, Did, Cid, Salary, Experience)
   Dept(Did, Dname)
   City(Cid, Cname, Did))
   District(Did, Dname, Sid)
   State(Sid, Sname, Cid)
   Country(Cid, Cname)
2. Insert 5 records in each table.
3. Display employeename, departmentname, Salary, Experience, City, District, State and country of all employees.

Subject :2305CS101 – Database Management Systems

15 **Sub Queries**

Student			
Rno	Name	City	DID
101	Raju	Rajkot	10
102	Amit	Ahmedabad	20
103	Sanjay	Baroda	40
104	Neha	Rajkot	20
105	Meera	Ahmedabad	30
106	Mahesh	Baroda	10

Academic		
Rno	SPI	Bklog
101	8.8	0
102	9.2	2
103	7.6	1
104	8.2	4
105	7.0	2
106	8.9	3

Department	
DID	DName
10	Computer
20	Electrical
30	Mechanical
40	Civil

Part-A:

1. Display details of students who are from computer department.
2. Displays name of students whose SPI is more than 8.
3. Display details of students of computer department who belongs to Rajkot city.
4. Find total number of students of electrical department.

Part-B:

5. Display name of student who is having maximum SPI.
6. Display details of students having more than 1 backlog.
7. Display name of student who is having second highest SPI.

Part-C:

8. Display name of students who are either from computer department or from mechanical department.
9. Display name of students who are in same department as 102 studying in.
10. Display name of students whose SPI is more than 9 and who is from electrical department.

16 **Create table and insert data as per below.**

Student	
Column_Name	**DataType**
StuID	Int
Name	Varchar (100)
EnrollmentNo	Varchar (12)
Division	Varchar (50)
Sem	Int
BirthDate	Datetime
Email	Varchar (100)
ContactNo	Varchar (50)

StuID	Name	EnrollmentNo	Division	Sem.	BirthDate	Email	ContactNo
101	Naimish Patel	090200107051	BCX-3	3	1992-12-06	naimishp49@gmail.com	8866205253
102	Firoz A. S.	090200107090	BCY-3	3	1994-05-03	Firoz.me@gmail.com	8885999922
103	Krunal Vyas	090243107101	BCZ-5	5	1984-03-01	Krunal.vyas@gmail.com	9990888877
104	Vijay Patel	090200107102	BCX-5	5	1985-02-15	Vijay.patel123@gmail.com	8787878787
105	Vimal Trivedi	090200107103	BCY-3	3	1988-01-20	Maulik123@gmail.com	8789564512

From the above given table perform the following queries:

Part - A

1. Display Name of Student who belongs to either Semester 3 or 5. (Use IN & OR)
2. Find Student Name & Enrollment No in which Student Id between 103 to 105.
3. Find Student Name & Enrollment No with their Email Who belongs to 5th Semester.

4. Display All the Details of first three students.
5. Display Name & Enrollment no of first 30% Students who's contact number ends with 7.
6. Display Unique Semesters.
7. Retrieve All the Students who have no Enrollment.
8. Find All Students whose Name does not start with 'V'.
9. Find All Students in which Email Contains '3@G' & Only Six Characters.
10. Find Out All the Students whose First Name Starts with F And third character must be R.

Part - B

1. Find All the Student Details whose Contact No contains 877.
2. Display Student Name in Which Student belongs to Semester 3 & Contact Number Does Not Contains 8 & 9.
3. Find Students who born after date 01-01-1990.
4. Update Division to BCX-5 & Semester 5 whose Student Id Is 102.
5. Change the Student's Name to Firoz Sherasiya in which Email is Firoz.Me@Gmail.Com & Contact No is 888

Part - C

1. Add one more Column City Varchar (50) in Student Table.
2. Remove All BCX-3 Division Students.
3. Change Column Name Email to EmailAddress.
4. Remove All the Data from Student Table Using Truncate.
5. Write an SQL query to clone a new table Student_New from Student table with all data.

17 **Create table and insert data as per below.**

Employee	
Column_Name	**DataType**
EID	Int
EName	Varchar (100)
Gender	Varchar (10)
JoiningDate	Datetime
Salary	Decimal (8,2)
City	Varchar (100)

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 table perform the following queries:

Part - A

1. Display all the employees whose name starts with "m" and 4th character is "h".
2. Find the value of 3 raised to 5. Label the column as output.
3. Write a query to subtract 20 days from the current date.
4. Write a query to display name of employees whose name starts with "j" and contains "n" in their name.
5. Display 2nd to 9th character of the given string "SQL Programming".

Subject :2305CS101 – Database Management Systems

Part - B

1. Display name of the employees whose city name ends with “ney” &contains six characters.
2. Write a query to convert value 15 to string.
3. Add department column with varchar (20) to Employee table.
4. Set the value of department to Marketing who belongs to London city.
5. Display all the employees whose name ends with either “n” or “y”.

Part - C

1. Find smallest integer value that is greater than or equal to 63.1, 63.8 and -63.2.
2. Display all employees whose joining date is not available.
3. Display name of the employees in capital letters and city in small letters.
4. Change the data type of Ename from varchar (30) to char (30).
5. Display city wise maximum salary.

18 **Consider the same Employee table of practical no 17 and perform the following queries:**

Part - A

1. Produce output like <Ename> works at <city> and earns <salary>.
2. Find total number of employees whose salary is more than 5000.
3. Write a query to display first 4 & last 3 characters of all the employees.
4. List the city having total salaries more than 15000 and employees joined after 1st January, 2014.
5. Write a query to replace “u” with “oo” in Ename.

Part - B

1. Display city with average salaries and total number of employees belongs to each city.
2. Display total salaries paid to female employees.
3. Display name of the employees and their experience in years.
4. Remove column department from employee table.
5. Update the value of city from syndey to null.

Part - C

1. Retrieve all Employee name, Salary & Joining date.
2. Find out combine length of Ename & Gender.
3. Find the difference between highest & lowest salary.
4. Rename a column from Ename to FirstName.
5. Rename a table from Employee to EmpMaster.

19 **Create table and insert data as per below.**

Person

Column_Name	DataType	Constraints
PersonID	Int	Primary Key
PersonName	Varchar (100)	Not Null
DepartmentID	Int	Foreign Key, Null
Salary	Decimal (8,2)	Not Null
JoiningDate	Datetime	Not Null
City	Varchar (100)	Not Null

PersonID	PersonName	DepartmentID	Salary	JoiningDate	City
101	Rahul Tripathi	2	56000	01-01-2000	Rajkot
102	Hardik Pandya	3	18000	25-09-2001	Ahmedabad
103	Bhavin Kanani	4	25000	14-05-2000	Baroda
104	Bhoomi Vaishnav	1	39000	08-02-2005	Rajkot
105	Rohit Topiya	2	17000	23-07-2001	Jamnagar
106	Priya Menpara	*NULL*	9000	18-10-2000	Ahmedabad

107	Neha Sharma	2	34000	25-12-2002	Rajkot
108	Nayan Goswami	3	25000	01-07-2001	Rajkot
109	Mehul Bhundiya	4	13500	09-01-2005	Baroda
110	Mohit Maru	5	14000	25-05-2000	Jamnagar

Department		
Column_Name	**DataType**	**Constraints**
DepartmentID	Int	Primary Key
DepartmentName	Varchar (100)	Not Null, Unique
DepartmentCode	Varchar (50)	Not Null, Unique
Location	Varchar (50)	Not Null

DepartmentID	DepartmentName	DepartmentCode	Location
1	Admin	Adm	A-Block
2	Computer	CE	C-Block
3	Civil	CI	G-Block
4	Electrical	EE	E-Block
5	Mechanical	ME	B-Block

From the above given table perform the following queries:

Part - A

1. Find all persons with their department name & code.
2. Give department wise maximum & minimum salary with department name.
3. Find all departments whose total salary is exceeding 100000.
4. Retrieve person name, salary & department name who belongs to Jamnagar city.
5. Find all persons who does not belongs to any department.

Part - B

1. Find department wise person counts.
2. Find average salary of person who belongs to Ahmedabad city.
3. Produce Output Like: <PersonName> earns <Salary> from department <DepartmentName> monthly. (In Single Column)
4. List all departments who have no persons.
5. Find city & department wise total, average & maximum salaries.

Part - C

1. Display Unique city names.
2. List out department names in which more than two persons.
3. Combine person name's first three characters with city name's last three characters in single column.
4. Give 10% increment in Computer department employee's salary.
5. Display all the person name's who's joining dates difference with current date is more than 365 days.

20

Part-A:

Views (First create a view then display all views)

Student				
RNo(PK)	Name	Branch	SPI	Bklog
101	Raju	CE	8.80	0
102	Amit	CE	2.20	3
103	Sanjay	ME	1.50	6
104	Neha	EC	7.65	1
105	Meera	EE	5.52	2
106	Mahesh	EC	4.50	3

1. Create a view Personal with all columns.
2. Create a view Student_Details having columns Name, Branch & SPI.
3. Create a view Academic having columns RNo, Name, Branch.
4. Create a view Student_Data having all columns but students whose bklog more than 2.
5. Create a view Student_Pattern having RNo, Name & Branch columns in which Name consists of four letters.
6. Insert a new record to Academic view. (107, Meet, ME)
7. Update the branch of Amit from CE to ME in Student_Details view.
8. Delete a student whose roll number is 104 from Academic view.

Part-B:

Create table and insert records as per below.

Student		
Column_Name	**DataType**	**Constraints**
Rno	Int	Primary Key
Name	Varchar (50)	Not Null
Branch	Varchar (50)	Not Null
SPI	Decimal (4,2)	Not Null
Bklog	Int	Not Null

RNo	Name	Branch	SPI	Bklog
101	Raju	CE	8.80	0
102	Amit	CE	2.20	3
103	Sanjay	ME	1.50	6
104	Neha	EC	7.65	1
105	Meera	EE	5.52	2
106	Mahesh	EC	4.50	3

From the above given table perform the following queries:

1. Create a view Personal with all columns.
2. Create a view Student_Details having columns Name, Branch & SPI.
3. Create a view Academic having columns RNo, Name, Branch.
4. Create a view Student_Data having all columns but students whose bklogs are more than 2.
5. Create a view Student_Pattern having RNo, Name & Branch columns in which Name consists of four letters.
6. Insert a new record to Academic view. (107, Meet, ME). Remaining all columns must be null.
7. Update the branch of Amit from CE to ME in Student_Details view.

Part – C

1. Delete a student whose roll number is 104 from Academic view.
2. Create a view that displays information of all students whose spi is above 8.5.
3. Create a view that displays 0 backlog students.
4. Create a view Computer that displays CE branch data only.
5. Create a view Result_EC that displays the name and SPI of students with SPI less than 5 of branch EC.
6. Update the result of student Sanjay to 4.90 in Result_EC view.
7. Create a view Stu_Bklog with RNo, Name and Bklog columns in which name starts with 'M' and having bklogs more than 5.
8. Drop Computer view form the database.

21

Create table and insert records as per below.

Person		
Column_Name	**DataType**	**Constraints**

WorkerID	Int	Primary Key, Auto Increment
FirstName	Varchar (100)	Not Null
LastName	Varchar (100)	Not Null
Salary	Decimal (8,2)	Not Null
JoiningDate	Datetime	Not Null
DepartmentID	Int	Foreign Key, Null
DesignationID	Int	Foreign Key, Null

Department		
Column_Name	**DataType**	**Constraints**
DepartmentID	Int	Primary Key
DepartmentName	Varchar (100)	Not Null, Unique

Designation		
Column_Name	**DataType**	**Constraints**
DesignationID	Int	Primary Key
DesignationName	Varchar (100)	Not Null, Unique

WorkerID	FirstName	LastName	Salary	JoiningDate	DepartmentID	DesignationID
101	Rahul	Anshu	56000	01-01-1990	1	12
102	Hardik	Hinsu	18000	25-09-1990	2	11
103	Bhavin	Kamani	25000	14-05-1991	*NULL*	11
104	Bhoomi	Patel	39000	20-02-2014	1	13
105	Rohit	Rajgor	17000	23-07-1990	2	15
106	Priya	Mehta	25000	18-10-1990	2	*NULL*
107	Neha	Trivedi	18000	20-02-2014	3	15

DepartmentID	DepartmentName
1	Admin
2	IT
3	HR
4	Account

DesignationID	DesignationName
11	Jobber
12	Welder
13	Clerk
14	Manager
15	CEO

From the above given tables create Stored Procedures:

Part – A

1. Department, Designation & Person Table's INSERT, UPDATE & DELETE Procedures.
2. Department, Designation & Person Table's SELECTBYPRIMARYKEY
3. Department, Designation & Person Table's (If foreign key is available then do write join and take columns on select list)
4. Create a Procedure that shows details of the first 3 persons.

Part – B

1. Create a Procedure that takes the department name as input and returns a table with all workers working in that department.
2. Create Procedure that takes department name & designation name as input and returns a table with worker's first name, salary, joining date & department name.
3. Create a Procedure that takes the first name as an input parameter and display all the details of the worker with their department & designation name.
4. Create Procedure which displays department wise maximum, minimum & total salaries.
5. Create Procedure which displays designation wise average & total salaries.

Part – C

1. Create Procedure that Accepts Department Name and Returns Person Count.
2. Create Procedure that accepts Department Name & Designation as a parameter with given test cases and returns a table with FirstName, LastName, Salary, JoiningDate, DepartmentName & Designation.

Department Name	Designation
IT	*NULL*
NULL	Jobber
IT	Jobber
NULL	*NULL*

3. Create Procedure that returns DepartmentID, DepartmentName & Count of all person belongs to that department. i.e. 1 | Admin | 2

22 **Create table and insert records as per below**

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.

Part – B

1. 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.
2. Create a trigger that fires on INSERT operation on the Person table to convert person name into uppercase whenever the record is inserted.

Part – C

1. Create a trigger that fires on INSERT operation on person table, which calculates the age and update that age in Person table.
2. Create DELETE trigger on PersonLog table, when we delete any record of PersonLog table it prints 'Record deleted successfully from PersonLog'.

23 **Create table and insert records as per below**

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)
   Create a cursor Product_CursorDelete that deletes all the data from the Products table.

Part – B

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

Part – C

1. 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)

24

Perform the following queries.

Scalar valued functions

Part – A

1. Write a function to print "hello world".
2. Write a function which returns addition of two numbers.
3. Write a function to print a cube of a given number.
4. Write a function to check whether the given number is ODD or EVEN.
5. Write a function which returns a table with details of a person whose first name starts with B.
6. Write a function which returns a table with unique first names from the person table.

Part – B

1. Write a function to compare two integers and return the comparison result. (Using Case statement)
2. Write a function to print number from 1 to N. (Using while loop)
3. Write a function to print the sum of even numbers between 1 to 20.

Part – C

1. Write a function to check whether a given number is prime or not.
2. Write a function which accepts two parameters start date & end date, and returns a difference in days.
3. Write a function which accepts two parameters year & month in integer and returns total days in a given month & year.
4. Write a function which accepts departmentID as a parameter & returns a detail of the persons.

25

Create Database with Name: **BANK_INFO**

Deposite

ACTNO	CNAME	BNAME	AMOUNT	ADATE
101	ANIL	VRCE	1000.00	1-3-95
102	SUNIL	AJNI	5000.00	4-1-96
103	MEHUL	KAROLBAGH	3500.00	17-11-95
104	MADHURI	CHANDI	1200.00	17-12-95
105	PRMOD	M.G. ROAD	3000.00	27-3-96
106	SANDIP	ANDHERI	2000.00	31-3-96
107	SHIVANI	VIRAR	1000.00	5-9-95
108	KRANTI	NEHRU PLACE	5000.00	2-7-95

From the above given collection perform the following queries:

Part - A

1. Retrieve/Display every document of Deposite collection.
2. Retrieve/Display every document of Deposite collection. (Use: pretty())
3. Display only one document of Deposite collection. (Use: findOne())
4. Insert following document to Deposite collection. (Use: insertOne())

109	KIRTI	VIRAR	3000.00	3-5-97

5. Insert following documents to your collection. (Use: insertMany())

110	MITALI	ANDHERI	4500.00	4-9-95
111	RAJIV	NEHRU PLACE	7000.00	2-10-98

6. Display documents with CNAME, BNAME and AMOUNT fields.
7. Display every document of Deposite collection on ascending order by CNAME.
8. Display every document of Deposite collection on descending order by BNAME.
9. Display every document of Deposite collection on ascending order by ACTNO and descending order by AMOUNT.
10. Display only two documents of Deposite collection.
11. Display 3rd document of Deposite collection.
12. Display 6th and 7th documents of Deposite collection.
13. Display the count of documents in Deposite collection.

Part- B

1. Insert following document to Deposite collection. (Use: insertOne())

112	MANISH	ANDHERI	8000.00	9-5-98

2. Display 9th document of Deposite collection.
3. Display 11th and 12th documents of Deposite collection.

Part – C

1. Display every document of Deposite collection on ascending order by AMOUNT and descending order by BNAME.
2. Display only five documents of Deposite collection.
3. Delete the collection Deposite.
4. Drop BANK_INFO database.

26 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

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:

1. Display Female students.
2. Display students who belong to Rajkot city.
3. Display students studying in 7th sem.
4. Display students not studying in 3rd sem.
5. Display students whose roll no is greater than 107.
6. Display students whose city is Jamnagar or Baroda (use:IN)
7. Display students whose fees is less than 9000.
8. Display the roll no of those students who belongs to Mechanical department.
9. Display first 2 students names who lives in Baroda.
10. Display Male students who studying in 3rd sem.
11. Display sname and city and fees of those students whose roll no is less than 105.
12. Display documents where sname start with K.
13. Display documents where sname starts with Z or D in your collection.
14. Display documents where city starts with A to R in your collection.
15. Display students' info whose name start with P and ends in i.
16. Display students' info whose department name starts with 'C'.
17. Display name, sem, fees, and department whose city contains 'med' as three letters somewhere in city name.

	18. Display name, sem, fees, and department who does not belongs to city Rajkot or Baroda. 19. Delete the documents whose city is Jamnagar. 20. Update sname of Krish to 'fenny' and gender to 'Female'. **Part – C** 1. Display next 2 students after skipping first 2 whose city is Ahmedabad. 2. Display rollno, sname, fees, and department of those students who is from Baroda and belongs to CE department. 3. Display documents where city name ends in 'oda'. 4. Display students' info whose name contains v. (Both uppercase(V) and lowercase(v)) 5. Display students' info whose name starts with V and having 4 characters.
27	**From the above given collection perform the following queries:** **Part – A (Use employee collection of Lab 26)** 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 26)** 1. Display distinct department. 2. Display city wise number of students. 3. Display sum of fees in your collection. 4. Display average of fees in your document. 5. Display maximum and minimum fees of your document. **Part – C (Use student collection of Lab 26)** 1. Display department wise total fees in your collection. 2. Display gender wise maximum fees in your collection. 3. Display department maximum and minimum fees. 4. Display count of persons lives in Rajkot city in your collection. 5. Display department wise number of students.