Lab 1 Introduction to SSMS

Introduction to Editor (SQL Server Management Studio)
Installation Steps

Lab 2 Perform basic SQL commands like Create, Insert

Database Name: Branch_DIV_Rollno (Example: CSE_3A_101 or Bsc_Hons_101)
Note: Create all the tables under above database.

Create following tables and insert the data into tables using query as shown below.

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

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	PRAMOD	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

BRANCH	
Column_Name	**DataType**
BNAME	VARCHAR(50)
CITY	VARCHAR(50)

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	
Column_Name	**DataType**
CNAME	VARCHAR(50)
CITY	VARCHAR(50)

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

BORROW	
Column_Name	**DataType**
LOANNO	INT
CNAME	VARCHAR(50)
BNAME	VARCHAR(50)
AMOUNT	DECIMAL(8,2)

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	PRAMOD	VIRAR	8000.00
481	KRANTI	NEHRU PLACE	3000.00

Lab 3	**Perform SQL queries for Select with operators**

SELECT Operation

Part – A:

From the above given tables perform the following queries:

1. Retrieve all data from table DEPOSIT.
2. Retrieve all data from table BORROW.
3. Display Account No, Customer Name & Amount from DEPOSIT.
4. Display Loan No, Amount from BORROW.
5. Display loan details of all customers who belongs to 'ANDHERI' branch from borrow table.
6. Give account no and amount of depositor, whose account no is equals to 106 from deposit table.
7. Give name of borrowers having amount greater than 5000 from borrow table.
8. Give name of customers who opened account after date '1-12-95' from deposit table.
9. Display name of customers whose account no is less than 105 from deposit table.
10. Display name of customer who belongs to either 'NAGPUR' or 'DELHI' from customer table. (USE OR & IN)
11. Display name of customers with branch whose amount is greater than 4000 and account no is less than 105 from deposit table.
12. Find all borrowers whose amount is greater than equals to 3000 & less than equals to 8000 from borrow table. (USE AND & BETWEEN)
13. Find all depositors who do not belongs to 'ANDHERI' branch from deposit table.
14. 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 from deposit table.
15. Display all loan no, customer from borrow table does not belong to 'VIRAR' or 'AJNI' branch. (use NOT IN)
16. Display all the customer's name other than 'MINU' from deposit table (Use: NOT, <>, !=)
17. Display customer name from deposit table whose branch name is not available. (NULL)
18. Retrieve all unique branches using DISTINCT. (Use Branch Table)
19. Retrieve first 50% record from borrow table.
20. Retrieve first five account number from deposit table.

Part – B:

1. Display all the details of first five customers from deposit table.
2. Display all the details of first three depositors from deposit table whose amount is greater than 1000.
3. Display Loan No, Customer Name of first five borrowers whose branch name does not belongs to 'ANDHERI' from borrow table.
4. Select all details with account numbers not in the range 105 to 109 in deposit table.
5. Select all records from BORROW where the amount is greater than 1000 and less than or equal to 7000, and the loan number is between 250 and 600

Part – C:

1. Display all the detail of customer who deposited more than 5000 without using * from deposit table.
2. Retrieve all unique customer names with city. (Use Customer table)
3. Retrieve records from the BORROW table where the loan amount is greater than 3000 and the loan number is not a multiple of 3.
4. Retrieve records from the DEPOSIT table where amount is greater than 2000 also account number is between 100 and 110 and date is after '1-MAR-1995' or before '27-MAR-1996'.
5. Retrieve all odd/even value loan number from Borrow table.

Lab 4 | **Perform SQL queries for Select into and Update**

Select into Operation

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 and data.
2. Create table T20 from cricket with first two columns with no data.
3. Create table IPL From Cricket with No Data.
4. Select players who are either older than 30 and from 'Mumbai' or exactly 31 years old and not from 'Bombay', and insert them into a new table PLAYER.
5. Select players whose age is a prime number or their city belongs to India Country, and insert them into a new table PLAYER_INFO. (Consider Cricketer age between 18 to 55)
6. Select players whose age is a multiple of 5 and insert them into a new table PLAYER_DATA.
7. Insert the cricketer into IPL table whose city is 'Jharkhand'

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

Part – C:

Perform following queries on Employee table.

1. Insert the Data into Employee_info from Employee whose CITY is Rajkot
2. Insert the Data into Employee_info from Employee whose age is more than 32.

Update Operation

Part – A:

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

1. Update deposit amount of all customers from 3000 to 5000. (Use **Deposit Table**)
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. (Use **Deposit Table**)
4. Update amount of KRANTI to 7000. (Use **Deposit Table**)
5. Update branch name from ANDHERI to ANDHERI WEST. (Use **Branch Table**)
6. Update branch name of MEHUL to NEHRU PALACE. (Use **Deposit Table**)
7. Update deposit amount of all depositors to 5000 whose account no between 103 & 107. (Use **Deposit Table**)
8. Update ADATE of ANIL to 1-4-95. (Use **Deposit Table**)
9. Update the amount of MINU to 10000. (Use **Deposit Table**)
10. Update deposit amount of PRAMOD to 5000 and ADATE to 1-4-96 (Use **Deposit Table**)

Part – B:

1. Give 10% Increment in Loan Amount. (Use **Borrow Table**)
2. Customer deposits additional 20% amount to their account, update the same. (Use **Deposit Table**)
3. Increase Amount by 1000 in all the account. (Use **Deposit Table**)
4. Update the BORROW table to set the amount to 7000 and the branch name to 'CENTRAL' where the customer name is 'MEHUL' and the loan number is even.
5. Update the DEPOSIT table to set the date to '2022-05-15' and the amount to 2500 for all accounts in 'VRCE' and with an account number less than 105.

Part – C:

1. Update amount of loan no 321 to *NULL.* (Use **Borrow Table**)
2. Update branch name of KRANTI to NULL (Use **Borrow Table**)
3. Display the name of borrowers whose Loan number is *NULL*. (Use **Borrow Table**)
4. Display the Borrowers whose having branch. (Use **Borrow Table**)
5. Update the Loan Amount to 5000, Branch to VRCE & Customer Name to Darshan whose loan no is 481. (Use **Borrow Table**)
6. Update the Deposit table and set the date to 01-01-2021 for all the depositor whose amount is less than 2000.
7. Update the Deposit table and set the date to NULL & Branch name to 'ANDHERI whose Account No is 110.

Lab 5 Perform SQL queries for Alter, Rename, Delete, Truncate, and Drop

Alter, Rename Operation

Part – A:

Use Deposit table of lab-1.

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, RENAME Operation):

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

Part – B:

1. Rename Column ADATE to AOPENDATE OF DEPOSIT_DETAIL table.
2. Delete Column AOPENDATE from the DEPOSIT_DETAIL table.
3. Rename Column CNAME to CustomerName.
4. Add Column country.

Part – C:

Create following table using query according to the definition.

STUDENT_DETAIL	
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, RENAME Operation):

1. Add two more columns City VARCHAR (20) (Not null) and Backlog INT (Null).
2. Add column department VARCHAR (20) Not Null.
3. Change the size of NAME column of student_detail from VARCHAR (25) to VARCHAR (35).
4. Change the data type DECIMAL to INT in CPI Column.
5. Delete Column City from the student_detail table.
6. Rename Column Enrollment_No to ENO.
7. Change name of table student_detail to STUDENT_MASTER.

DELETE, Truncate, Drop Operation

Part – A:

Use Deposit_Detail table (Altered table of DEPOSIT)

DEPOSIT_DETAIL	
Column_Name	**DataType**
ANO	INT
CustomerName	VARCHAR(35)
BNAME	VARCHAR(50)
AMOUNT	INT
PINCODE	INT

1. Delete all the records of DEPOSIT_DETAIL table having amount less than and equals to 4000.
2. Delete all the accounts CHANDI BRANCH.
3. Delete all the accounts having account number (ANO) is greater than 102 and less than 105.
4. Delete all the accounts whose branch is 'AJNI' or 'POWAI'
5. Delete all the accounts whose account number is NULL.
6. Delete all the remaining records using Delete command.
7. Delete all the records of Deposit_Detail table. (Use **Truncate**)
8. Remove Deposit_Detail table. (Use **Drop**)

Part – B:

Create following table using query according to the definition.

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

Insert the following records in the EMPLOYEE_MASTER 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
105	Harmit	15-2-04	14000.00	Rajkot
106	Mitesh	25-9-01	5000.00	Jamnagar
107	Meera	Null	7000.00	Morbi
108	Kishan	6-2-03	10000.00	NULL

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

1. Delete all the records of Employee_MASTER 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 the records of Employees whose joining date is null and Name is not null.
5. Delete the records of Employees whose salary is 50% of 20000.
6. Delete the records of Employees whose City Name is not empty.
7. Delete all the records of Employee_MASTER table. (Use **Truncate**)
8. Remove Employee_MASTER table. (Use **Drop**)

Part – C:

1. Summarize Delete, Truncate and Drop

Lab 6 Perform SQL queries for Like operator

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)
Address	VARCHAR(100)

Insert the following records in the STUDENT table.

StuID	FirstName	LastName	Website	City	Address
1011	Keyur	Patel	techonthenet.com	Rajkot	A-303 'Vasant Kunj', Rajkot
1022	Hardik	Shah	digminecraft.com	Ahmedabad	"Ram Krupa", Raiya Road
1033	Kajal	Trivedi	bigactivities.com	Baroda	Raj bhavan plot, near garden
1044	Bhoomi	Gajera	checkyourmath.com	Ahmedabad	"Jig's Home", Narol
1055	Harmit	Mitel	@me.darshan.com	Rajkot	B-55, Raj Residency
1066	Ashok	Jani	NULL	Baroda	A502, Club House Building

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'.

7. Display the name of students having no website and name consists of five characters.
8. Display all the students whose last name consist of ‘jer’.
9. Display all the students whose city name starts with either ‘r’ or ‘b’.
10. Display all the name students having websites.
11. Display all the students whose name starts from alphabet A to H.
12. Display all the students whose name’s second character is vowel.
13. Display the name of students having no website and name consists of minimum five characters.
14. Display all the students whose last name starts with ‘Pat’.
15. Display all the students whose city name does not starts with ‘b’.
16. Display all the students whose student ID ends with digit.
17. Display all the students whose address does not contain any digit.
18. Find students whose first name starts with 'B', last name ends with 'A', and their website contains either 'math' or 'science'. Ensure that their city does not start with 'B'.
19. Retrieve students who have 'Shah' in their last name and whose city ends with 'd'. Additionally, their website should be either null or contain 'com'.
20. Select students whose first and second character is a vowel. Their city should start with 'R' and they must have a website containing '.com'.

Part – B:

1. Display all the students whose name’s second character is vowel and of and start with H.
2. Display all the students whose last name does not ends with ‘a’.
3. Display all the students whose first name starts with consonant.
4. Retrieve student details whose first name starts with 'K', last name ends with 'tel', and either their website contains 'tech' or they live in a city starting with 'R'.
5. Retrieve students whose address contains a hyphen '-' and whose city starts with either 'R' or 'B'. They must have a website that ends with '.com' and their first name should not start with 'A'.

Part – C:

1. Display all the students whose address contains single quote or double quote.
2. Find students whose city does not contain the letter 'S' and their address contains either single or double quotes. Their last name should start with 'P' and they must have a website that contains 'on'.

Lab 7 — Perform SQL queries for Aggerate function and group by (without having)

Part – A:

Create table and inset records as per below.

EMP

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

1. Display the Highest, Lowest, Label the columns Maximum, Minimum respectively.
2. Display Total, and Average salary of all employees. Label the columns Total_Sal and Average_Sal, respectively.
3. Find total number of employees of EMPLOYEE table.
4. Find highest salary from Rajkot city.

5. Give maximum salary from IT department.
6. Count employee whose joining date is after 8-feb-91.
7. Display average salary of Admin department.
8. Display total salary of HR department.
9. Count total number of cities of employee without duplication.
10. Count unique departments.
11. Give minimum salary of employee who belongs to Ahmedabad.
12. Find city wise highest salary.
13. Find department wise lowest salary.
14. Display city with the total number of employees belonging to each city.
15. Give total salary of each department of EMP table.
16. Give average salary of each department of EMP table without displaying the respective department name.
17. Count the number of employees for each department in every city.
18. Calculate the total salary distributed to male and female employees.
19. Give city wise maximum and minimum salary of female employees.
20. Calculate department, city, and gender wise average salary.

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.

Part – C:

1. Count the number of employees living in Rajkot or Baroda.
2. Display the total number of employees hired before 1st January, 1991 in IT department.
3. Find the Joining Date wise Total Salaries.
4. Find the Maximum salary department & city wise in which city name starts with 'R'.

Lab 8 — Perform SQL queries for Group by with having and Order by

Table: SALES_DATA

Region	Product	Sales_Amount	Year
North America	Watch	1500	2023
Europe	Mobile	1200	2023
Asia	Watch	1800	2023
North America	TV	900	2024
Europe	Watch	2000	2024
Asia	Mobile	1000	2024
North America	Mobile	1600	2023
Europe	TV	1500	2023
Asia	TV	1100	2024
North America	Watch	1700	2024

Part – A:

1. Display Total Sales Amount by Region.
2. Display Average Sales Amount by Product
3. Display Maximum Sales Amount by Year
4. Display Minimum Sales Amount by Region and Year
5. Count of Products Sold by Region
6. Display Sales Amount by Year and Product
7. Display Regions with Total Sales Greater Than 5000

8. Display Products with Average Sales Less Than 10000
9. Display Years with Maximum Sales Exceeding 500
10. Display Regions with at Least 3 Distinct Products Sold.
11. Display Years with Minimum Sales Less Than 1000
12. Display Total Sales Amount by Region for Year 2023, Sorted by Total Amount
13. Find the Region Where 'Mobile' Had the Lowest Total Sales Across All Years.
14. Find the Product with the Highest Sales Across All Regions in 2023.
15. Find Regions Where 'TV' Sales in 2023 Were Greater Than 1000.

Part – B:

1. Display Count of Orders by Year and Region, Sorted by Year and Region
2. Display Regions with Maximum Sales Amount Exceeding 1000 in Any Year, Sorted by Region
3. Display Years with Total Sales Amount Less Than 10000, Sorted by Year Descending
4. Display Top 3 Regions by Total Sales Amount in Year 2024
5. Find the Year with the Lowest Total Sales Across All Regions.

Part – C:

1. Display Products with Average Sales Amount Between 1000 and 2000, Ordered by Product Name
2. Display Years with More Than 1 Orders from Each Region
3. Display Regions with Average Sales Amount Above 1500 in Year 2023 sort by amount in descending.
4. Find out region wise duplicate product.
5. Find out year wise duplicate product.

Lab 9 — Implement SQL In-built functions (Math, String, and Date Functions)

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 2nd power and 4 raised 3rd 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 EMP_MASTER table.

EmpNo	EmpName	JoiningDate	Salary	Commission	City	Dept Code
101	Keyur	5-1-02	12000.00	4500	Rajkot	3@g
102	Hardik	15-2-04	14000.00	2500	Ahmedabad	3@
103	Kajal	14-3-06	15000.00	3000	Baroda	3-GD
104	Bhoomi	23-6-05	12500.00	1000	Ahmedabad	1A3D
105	Harmit	15-2-04	14000.00	2000	Rajkot	312A

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. Retrieve the details of employees whose total earnings (Salary + Commission) are greater than 15000.
2. Find the details of employees whose commission is more than 25% of their salary.
3. List the employees who joined before 2005 and whose total earnings (Salary + Commission) are greater than 15000.
4. Find employees whose total earnings (Salary + Commission) are at least double their salary.

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 EMP_MASTER table.

1. Find the length of EMP Name and City columns.
2. Display EMP Name and City columns in lower & upper case.
3. Display first three characters of EMP Name column.
4. Display 3rd to 10th character of city column.
5. Write a query to display first 4 & Last 5 characters of EMP Name column.

Part – C: Perform following queries on EMP_MASTER table.

1. Put 10 space before EMP Name using function.
2. Combine EMP Name and city columns using + sign as well as CONCAT ().
3. Combine all columns using + sign as well as CONCAT ().
4. Combine the result as < EMP Name > Lives in <City>.
5. Combine the result as ‘EMP no of < EMP Name> is <EmpNo> .
6. Retrieve the names of all employee where the third character of the Name is a vowel.
7. Concatenate the name and city of students who have a name that ends with the letter 'r' and a city that starts with 'R'.

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 hours between 25-Jan-12 7:00 and 26-Jan-12 10:30.
8. Write a query to extract Day, Month, Year from given date 12-May-16.
9. Write a query that adds 5 years to current date.
10. Write a query to subtract 2 months from current date.
11. Extract month from current date using datename () and datepart () function.
12. Write a query to find out last date of current month.
13. Calculate your age in years and months.

Part – B: Perform following queries on EMP_MASTER table.

1. Write a query to find new date after 365 days with reference to JoiningDate.
2. Write a query to find out total number of months between JoiningDate and 31-Mar-09.
3. Write a query to find out total number of years between JoiningDate and 14-Sep-10.

Part – C: Perform following queries on EMP_MASTER table.

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.
5. Select employee who joined between the 1st and 15th of any month in any year.
6. Find employee whose JoiningDate is the last day of any month.
7. List employee whose JoiningDate is during a leap year.

Lab 10 Implement SQL View

Part – A: Views (First create a view then display all views)

STUDENT_INFO

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

1. Create a view Personal with all columns.
2. Create a view Student_Details having columns Name, Branch & SPI.
3. Create a view AcademicData having columns RNo, Name, Branch.
4. Create a view Student_ bklog 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 AcademicData 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 AcademicData view.
9. Create view for the student whose name starts with M and ends with a, having SPI more than 6 and backlog less than 2.
10. Create the view for the students whose name contains vowel.

Part – B:

1. Create a view that displays information of all students whose SPI is above 8.5
2. Create a view that displays 0 backlog students.
3. Create a view Computerview that displays CE branch data only.

Part – C:

1. Create a view Result_EC that displays the name and SPI of students with SPI less than 5 of branch EC.
2. Update the result of student MAHESH to 4.90 in Result_EC view.
3. Create a view Stu_Bklog with RNo, Name and Bklog columns in which name starts with 'M' and having bklogs more than 5.
4. Drop Computerview form the database.

Lab 11 Implement SQL Joins

Create below tables as per following data

STU_INFO

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

RESULT

Rno(FK)	SPI
101	8.8
102	9.2
103	7.6
104	8.2
105	7.0
107	8.9

EMPLOYEE_MASTER

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

Part – A:

1. Combine information from student and result table using cross join or Cartesian product.
2. Perform inner join on Student and Result tables.
3. Perform the left outer join on Student and Result tables.
4. Perform the right outer join on Student and Result tables.
5. Perform the full outer join on Student and Result tables.
6. Display Rno, Name, Branch and SPI of all students.
7. Display Rno, Name, Branch and SPI of CE branch's student only.
8. Display Rno, Name, Branch and SPI of other than EC branch's student only.
9. Display average result of each branch.
10. Display average result of CE and ME branch.
11. Display Maximum and Minimum SPI of each branch.
12. Display branch wise student's count in descending order.

Part – B:

1. Display average result of each branch and sort them in ascending order by SPI.
2. Display highest SPI from each branch and sort them in descending order.

Part – C:

1. Retrieve the names of employee along with their manager's name from the Employee table.

Lab 12 Implement Complex Joins

Create following table (Using Design Mode)

PERSON

Column_Name		
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

DEPT		
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

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

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. Combine information from Person and Department table using cross join or Cartesian product.
2. Find all persons with their department name
3. Find all persons with their department name & code.
4. Find all persons with their department code and location.
5. Find the detail of the person who belongs to Mechanical department.
6. Final person's name, department code and salary who lives in Ahmedabad city.
7. Find the person's name whose department is in C-Block.
8. Retrieve person name, salary & department name who belongs to Jamnagar city.
9. Retrieve person's detail who joined the Civil department after 1-Aug-2001.
10. Display all the person's name with the department whose joining date difference with the current date is more than 365 days.
11. Find department wise person counts.
12. Give department wise maximum & minimum salary with department name.
13. Find city wise total, average, maximum and minimum salary.
14. Find the average salary of a person who belongs to Ahmedabad city.
15. Produce Output Like: <PersonName> lives in <City> and works in <DepartmentName> Department. (In single column)

Part – B:

1. Produce Output Like: <PersonName> earns <Salary> from <DepartmentName> department monthly. (In single column)
2. Find city & department wise total, average & maximum salaries.
3. Find all persons who do not belong to any department.

4. Find all departments whose total salary is exceeding 100000.

Part – C:

1. List all departments who have no person.
2. List out department names in which more than two persons are working.
3. Give a 10% increment in the computer department employee's salary. (Use Update)

Lab 13 Implement Advanced level Joins

Create following table (Using Design Mode)

Author		
Column Name	**Data Type**	**Constraints**
AuthorID	INT	Primary Key
AuthorName	VARCHAR(100)	NOT NULL
Country	VARCHAR(50)	NULL

Publisher		
Column Name	**Data Type**	**Constraints**
PublisherID	INT	Primary Key
PublisherName	VARCHAR(100)	NOT NULL, UNIQUE
City	VARCHAR(50)	NOT NULL

Book		
Column Name	**Data Type**	**Constraints**
BookID	INT	Primary Key
Title	VARCHAR(200)	NOT NULL
AuthorID	INT	Foreign Key, AUTHOR(AuthorID), NOT NULL
PublisherID	INT	Foreign Key, PUBLISHER(PublisherID), NOT NULL
Price	DECIMAL(8,2)	NOT NULL
PublicationYear	INT	NOT NULL

AuthorID	AuthorName	Country
1	Chetan Bhagat	India
2	Arundhati Roy	India
3	Amish Tripathi	India
4	Ruskin Bond	India
5	Jhumpa Lahiri	India
6	Paulo Coelho	Brazil
7	Sudha Murty	India

PublisherID	PublisherName	City
1	Rupa Publications	New Delhi
2	Penguin India	Gurugram
3	HarperCollins India	Noida
4	Aleph Book Company	New Delhi

BookID	Title	AuthorID	PublisherID	Price	PublicationYear
101	Five Point Someone	1	1	250.00	2004
102	The God of Small Things	2	2	350.00	1997
103	Immortals of Meluha	3	3	300.00	2010
104	The Blue Umbrella	4	1	180.00	1980
105	The Lowland	5	2	400.00	2013
106	Revolution 2020	1	1	275.00	2011
107	Sita: Warrior of Mithila	3	3	320.00	2017
108	The Room on the Roof	4	4	200.00	1956

From the above given table perform the following queries:

Part – A:

1. List all books with their authors.
2. List all books with their publishers.
3. List all books with their authors and publishers.
4. List all books published after 2010 with their authors and publisher and price.
5. List all authors and the number of books they have written.
6. List all publishers and the total price of books they have published.
7. List authors who have not written any books.
8. Display total number of Books and Average Price of every Author.
9. lists each publisher along with the total number of books they have published, sorted from highest to lowest.
10. Display number of books published each year.

Part – B:

1. List the publishers whose total book prices exceed 500, ordered by the total price.
2. List most expensive book for each author, sort it with the highest price.

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

1. Emp_info(Eid, Ename, Did, Cid, Salary, Experience)
   Dept_info(Did, Dname)
   City_info(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.

Lab 14 Perform SQL queries for Set operator and, Subqueries

Sub Queries

STUDENT_DATA

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

DEPARTMENT

DID	DName
10	Computer
20	Electrical
30	Mechanical
40	Civil

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

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

Part – B:

1. Display name of students who are either from computer department or from mechanical department.
2. Display name of students who are in same department as 102 studying in.

Part – C:

1. Display name of students whose SPI is more than 9 and who is from electrical department.
2. Display name of student who is having second highest SPI.
3. Display city names whose students SPI is 9.2
4. Find the names of students who have more than the average number of backlogs across all students.
5. Display the names of students who are in the same department as the student with the highest SPI.

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.

EMP_DATA	
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:

1. Perform all the queries of Part-B but display ID and Name columns instead of Name only.

Lab 15 Perform SQL queries to implement constraints

Part – A:

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

STU_MASTER				
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

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

1. Emp_details(Eid, Ename, Did, Cid, Salary, Experience)
   Dept_details(Did, Dname)
   City_details(Cid, Cname)

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

1. Emp_info(Eid, Ename, Did, Cid, Salary, Experience)
   Dept_info(Did, Dname)
   City_info(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.