

# GUJARAT TECHNOLOGICAL UNIVERSITY

## MASTERS IN COMPUTER APPLICATION

### Year – 2 (Semester – III) (W.E.F. JULY 2018)

**Subject Name: Database Management Systems (DBMS)**

**Subject Code: 4639303\_Practical List**

**A. List of Practical Related to DBMS:**

**Part I: RDBMS (Module Weightage: 100%)**

**Tools: MySQL**

**Topics:**

|    |   |
|----|---|
| 1  | Introduction to Database: MySQL, Installation, Data Types,  |
| 2  | Manage Database: Create Database, Drop Database, Select Database (MySQL> prompt)  |
| 3  | Create Table:<br>The Create Table Command,<br>Creating a table from a table (with data, without data, with all columns, with selected columns)<br>Drop Table<br>Alter Table |
| 4  | Study DML Commands ( Select, insert, update, delete )   |
| 5  | Sorting Data, Handling Null values ( IS NULL)   |
| 6  | Join  |
| 7  | Like Clause , REGEXP  |
| 8  | Transaction Control statements: Commit, Rollback  |
| 9  | Advanced Concepts: View, Index, Sequences   |
| 10 | Database Export / Import  |
| 11 | Study single row functions: String functions, Numeric Functions, Date Functions   |
| 12 | Study aggregate / group functions   |
| 13 | Study subquery concepts   |
| 14 | Constraints: Primary Key, Foreign Key, Check, Default, Not Null, Unique   |
| 15 | Set Operators   |
| 16 | Compound Statement Handling:<br><br>Syntax, Variables, flow of control, condition handling, Cursor Management,<br>Create procedure and Function                             |
| 17 | Create Triggers   |
| 18 | Data dictionary   |
| 19 | Security / privileges (Desirable)   |

**References:**

- 1) Steve Suehring, Tim Converse, Joyce Park, PHP 6 and MySQL Bible, Wiley
- 2) Andrea Tarr, PHP and MySQL 24-Hour Trainer, Wiley

**Web references:**

- 1) <https://dev.mysql.com>

## **Part II: NoSQL Database (Desirable)**

Tools: MongoDB

|   |   |
|---|---|
| 1 | Introduction, Installation  |
| 2 | Create Database, Drop Database                                    |
| 3 | Create Collection, show collection                                |
| 4 | Insert document, Query Document, Update document, delete document |
| 5 | Projection  |
| 6 | Limiting rows   |
| 7 | Export and Import   |

<https://docs.mongodb.com/manual/mongo/>

## **Part III: Practical's to perform: (SQL and PL/SQL)**

### **SET 1**

DEPARTMENT (dept\_no, dept\_name, location)

1. Create the Simple DEPARTMENT Table.
2. Display structure of department table.
3. Insert below records into Department Table

| Dept_no | Dept_name  | Location |
|---------|------------|----------|
| 10      | Account    | NY       |
| 20      | HR         | NY       |
| 30      | Production | DL       |
| 40      | Sales      | NY       |
| 50      | EDP        | MU       |
| 60      | TRG        |          |
| 110     | RND        | AH       |

4. Display all records of Department table
5. Display all department belonging to location 'NY'
6. Display details of Department 10
7. List all department names starting with 'A'
8. List all departments whose number is between 1 and 100
9. Delete 'TRG' department
10. Change department name 'EDP' to 'IT'

### **SET 2**

EMPLOYEE (emp\_id, emp\_name, birth\_date, gender, dept\_no, address, designation, salary, experience, email)

DEPARTMENT (dept\_no, dept\_name, location)

**Do as directed:**

11. Create the EMP Table with all necessary constraints such as  
In EMP TABLE: Employee id should be primary key, Department no should be

Foreign key, employee age (birth\_date) should be greater than 18 years, salary should be greater than zero, email should have (@ and dot) sign in address, designation of employee can be “manager”, “clerk”, “leader”, “analyst”, “designer”, “coder”, “tester”.

1. Create DEPT table with necessary constraint such as  
Department no should be primary key, department name should be unique.
12. After creation of above tables, modify Employee table by adding the constraints as ‘Male’ or ‘Female’ in gender field and display the structure.
13. Insert proper data (at least 5 appropriate records) in all the tables.
14. Describe the structure of table created
15. List all records of each table in ascending order.
16. Delete the department whose location is Ahmedabad.
17. Display female employee list
18. Display Department wise employee Names
19. Find the names of the employee who has salary less than 5000 and greater than 2000.
20. Display the names and the designation of all female employee in descending order.
21. Display the names of all the employees whose names start with ‘A’ ends with ‘A’.
22. Find the name of employee and salary for those who had obtained minimum salary.
23. Add 10% raise in salary of all employees whose department is ‘IT’.
24. Count total number of employees of ‘IT’ department.
25. List all employees who born in the current month.
26. Print the record of employee and dept table as “Employee works in department ‘MBA’.
27. List names of employees who are fresher’s (less than 1 year of experience).
28. List department wise names of employees who has more than 5 years of experience.
29. Create Sequence to generate department ID
30. List department having no employees

### **SET 3**

**STUDENT (rollno, name, class, birthdate)**

**COURSE (courseno, coursename, max\_marks, pass\_marks)**

**SC (rollno, courseno, marks)**

1. Create the above three tables along with key constraints.
2. Write an Insert script for insertion of rows with substitution variables and insert appropriate data.
3. Add a constraint that the marks entered should strictly be between 0 and 100.
4. While creating SC table, composite key constraint was forgotten. Add the composite key now.
5. Display details of student who takes ‘Database Management System’ course.
6. Display the names of students who have scored more than 70% in Computer Networks and have not failed in any subject.
7. Display the average marks obtained by each student.
8. Select all courses where passing marks are more than 30% of average maximum mark.
9. Display details of students who are born in 1980 or 1982.
10. Create a view that displays student courseno and its corresponding marks.

### **SET 4**

Create the database COMPANY and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

**EMPLOYEE (emp\_id, emp\_name, birth\_date, gender, dept\_no, address, designation, salary, experience, email)**

**DEPART (dept\_no, dept\_name, total\_employees, location)**

**PROJECT (proj\_id, type\_of\_project, status, start\_date, emp\_id)**

Insert proper data (at least 5 appropriate records) in all the tables.

**Do as directed:**

1. Delete the department whose total number of employees less than 1.
2. Display the names and the designation of all female employee in descending order.
3. Display the names of all the employees who names starts with 'A' ends with 'A'.
4. Find the name of employee and salary for those who had obtain minimum salary.
5. Add 10% raise in salary of all employees whose department is 'CIVIL'.
6. Count total number of employees of 'MCA' department.
7. List all employees who born in the current month.
8. Print the record of employee and dept table as "Employee works in department 'CE'.
9. List names of employees who are fresher's(less than 1 year of experience).
10. List department wise names of employees who has more than 5 years of experience.
11. Write a function which will display total number of projects based on status (pass status as parameter).
12. Write a procedure that will display list of projects which is going to start today.
13. Write a trigger which do not allow insertion/updation/deletion into Project table if status type is 'pending'

**SET 5**

Create the database STUD and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

HOSTEL (HNO, HNAME, HADDR, TOTAL\_CAPACITY, WARDEN)

ROOM (HNO, RNO, RTYPE, LOCATION, NO\_OF\_STUDENTS, STATUS)

CHARGES (HNO, RTYPE, CHARGES)

STUDENT (SID, SNAME, MOBILE-NO, GENDER, FACULTY, DEPT, CLASS, HNO, RNO)

FEES (SID, FDATE, FAMOUNT)

The STATUS field tells us whether the room is occupied or vacant. The charges represent the term fees to be paid half yearly. A student can pay either the annual fees at one time or the half yearly fees twice a year.

Insert proper data (at least 5 appropriate records) in all the tables.

**Do as directed:**

1. Display the total number of rooms that are presently vacant.
2. Display number of students of each faculty and department wise staying in each hostel.
3. Display hostels, which have at least one single-seated room.
4. Display the warden name and hostel address of students of Computer Science department.
5. Display those hostel details where single seated or double-seated rooms are vacant.
6. Display details of hostels occupied by medical students.
7. Display hostels, which are totally occupied to its fullest capacity.
8. List details about students who are staying in the double-seated rooms of Chanakya Hostel.

9. Display the total number of students staying in each room type of each hostel.
10. Display details about students who have paid fees in the month of Nov. 2017.
11. For those hostels where total capacity is more than 300, display details of students studying in Science faculty.
12. Display hostel details where there are at least 10 vacant rooms.
13. Display details of students who have still not paid fees.
14. Display those hostels where single-seated room is the costliest.
15. Write a trigger which do not allow to insert or update student record if mobile\_no length is less than 10 digits.
16. Write a PL/SQL block which will count total number of student's gender wise.  
Male Students: 999 students  
Female Students: 999 students

## **SET 6**

Create the database HOSPITAL and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

DOCTOR (DNO, DNAME, SPECIALIZATION, CLINIC\_ADDR)

MEDICINE (MNO, MNAME, TYPE, CONTENT, MANUFACTURER)

DISEASE (DISEASE\_NAME, SYMPTOM1, SYMPTOM2, SYMPTOM3)

TREATMENT (TNO, DNO, DISEASE\_NAME, MNO, DOSAGE, AVG\_CURE\_TIME)

Insert proper data (at least 5 appropriate records) in all the tables.

### **Do as directed:**

1. Display records of each table in ascending order.
2. Count total number of doctors which has not given any treatment.
3. Display all Chennai doctors who treat cancer.
4. Remove disease "polio" from disease table as well as treatment table.
5. Delete all those treatment related to liver of Dr.Shah.
6. Create index on dno, Disease name in the treatment table.
7. Display details of doctors who treat migraines.
8. What is the maximum dosage of "penicillin" prescribe by the doctor for the treatment of any disease?
9. Display total number of disease treated by every doctor.
10. Which doctor have no treatment for "depression"?
11. Create a view which contains the treatment and doctors details. Make sure that no body is allowed to modify any detail in the view.
12. Write a PL/SQL block to print the following report ( Symptoms wise print total number of medicine given )

| Medicine | Symptom1 | Symptom2 | Symptom3 | Total |
|----------|----------|----------|----------|-------|
| M1       | 999      | 999      | 999      | 9999  |
| M2       | 999      | 999      | 999      | 9999  |
| M3       | 999      | 999      | 999      | 9999  |

13. Write a trigger which does not allow to insert or update treatment table if AVG\_CURE\_TIME is less than 1.

## **SET 7**

Create the database SHOPPING and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

CUSTOMER (cno, cust\_name, cust\_phone, location, gender)

ITEM (itemno, itemname, color, weight, expire\_date, price, shop\_name)

CUST\_ITEM (cno, itemno, quantity\_purchased, date\_purchase)

Insert proper data (at least 5 appropriate records) in all the tables.

### **Do as directed:**

- Delete the items whose price is more than 50000.
- Find the names of the customer who is located in same location as that of other customer.
- Display the names of items which is black, white & brown in color.
- Display the names of all the items whose names lies between 'p' and 's'.
- Find the item which is having less weight.
- Add one month more to those items whose item no =40.
- Count total number of items which is going to expire in next month
- List all customers whose phone number starts with '99'.
- Display total value (qty\*price) for all items.
- List customer details who has purchased maximum number of items
- Display total price item wise.
- List name of items, customer details and qty purchased.
- Write a PL/SQL procedure which will display records in the following format

Today's Date: \_\_\_\_\_

Shop name: \_\_\_\_\_

| Item number | Item name | Expire date | Quantity | Price | Total Rs. |
|-------------|-----------|-------------|----------|-------|-----------|
|             |           |             |          |       |           |
|             |           |             |          |       |           |

Grand Total Rs. \_\_\_\_\_

14. Write a trigger which do not allow insertion / updation / deletion of Item details on Sunday.

## **SET 8**

Create the database THEATRE and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

**SCREEN (SCREEN\_ID, LOCATION, SEATING\_CAP)**

**MOVIE (MOVIE\_ID, MOVIE\_NAME, DATE\_OF\_RELEASE)**

**CURRENT (SCREEN\_ID, MOVIE\_ID, DATE\_OF\_ARRIVAL, DATE\_OF\_CLOSURE)**

### **Check Constraints:**

Value of screen\_id must start with letters 'S'.

Attribute location can be any one of 'FF', 'SF', or 'TF'.

### **Do as directed:**

1. Get the name of movie which has run the longest in the multiplex so far.
  2. Get the average duration of a movie on screen number 'S4'.
  3. Get the details of movie that closed on date 24-november-2004.
  4. Movie 'star wars III' was released in the 7<sup>th</sup> week of 2005. Find out the date of its release considering that a movie releases only on Friday.
  5. Get the full outer join of the relations screen and current.
  6. Write a PL/SQL function which will count total number of day's horror movie last longer.
  7. Write a PL/SQL procedure that will display movie which is going to release today.
  8. Write a trigger which will not allow to insert/update in current table if Date\_of\_arrival is less than date\_of\_closure.
- 

## **SET 9**

Create the database EXAM and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

**APPLICANT (AID, ANAME, ADDR, ABIRTH\_DT)**

**ENTRANCE\_TEST (ETID, ETNAME, MAX\_SCORE, CUT\_SCORE)**

**ETEST\_CENTRE (ETCID, LOCATION, INCHARGE, CAPACITY)**

**ETEST\_DETAILS (AID, ETID, ETCID, ETEST\_DT, SCORE)**

(This database is for a common entrance test which is being conducted at a number of centers and can be taken by an applicant on any day except holidays)

### **Do as directed:**

1. Modify the APPLICANT table so that every applicant id has an 'A' before its value. E.g. if value is '1123', it should become 'A1123'.
2. Display test center details where no tests were conducted.

3. Display details about applicants who have the same score as that of Ajaykumar in 'ORACLE FUNDAMENTALS'.
  4. Display details of applicants who appeared for all tests.
  5. Display those tests where no applicant has failed.
  6. Display details of entrance test centers which had full attendance between 1<sup>st</sup> Oct 15 and 15<sup>th</sup> Oct 16.
  7. Display details of the applicants who scored more than the cut score in the tests they appeared in.
  8. Display average and maximum score test wise of tests conducted at Mumbai.
  9. Display the number of applicants who have appeared for each test, test center wise.
  10. Display details about test centers where no tests have been conducted.
  11. For tests, which have been conducted between 2-3-17 and 23-4-17, show details of the tests as well as the test centre.
  12. How many applicants appeared in the 'ORACLE FUNDAMENTALS' test at Chennai in the month of February?
  13. Display details about applicants who appeared for tests in the same month as the month in which they were born.
  14. Display the details about APPLICANTS who have scored the highest in each test, test centre wise.
  15. Design a read only view, which has details about applicants and the tests that he has appeared for.
  16. Write a procedure which will print maximum score centre wise.
  17. Write a procedure which will print details of entrance test.
- 

Centre name: \_\_\_\_\_ candidate id: \_\_\_\_\_ date: \_\_\_\_ score: \_\_\_\_\_

---

15. Write a trigger which do not allow insertion / updation / deletion of Entrance test details on Sunday.

### **SET 10**

Create the database BUS TRANSPORT and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

**CATEGORY (CAT\_CODE, CATDESC)**

**ROUTEMASTER (ROUTENO, ORIGIN, DESTINATION, FARE, DISTANCE, CAPACITY, DAY, CAT\_CODE)**

**TICKETHEADER (TICKETNO, DATEOFISSUE, DATEOFTRAVEL, BOARDPLACE, ROUTENO)**



## **TICKET DETAILS (TICKETNO, NAME, SEX, AGE, FARE)**

ADD THE FOLLOWING CONSTRAINTS:

1. DELUXE, SUPERDELUXE, SUPERFAST AND NORMAL ARE THE CATDESC
2. ORIGIN AND DESTINATION CANNOT BE SAME,
3. CAPACITY SHOULD BE >0 AND <=60

Do as directed:

1. Display the total number of people traveled on each ticket group by ticket no 23.
2. Give the total collection of fare for each route.
3. Give the number of months between issue date and travel date of each ticket issued.
4. Count number of person boarding from the same place and same route.
5. Display count of person who has traveled in each category.
6. Write a trigger which allow to insert or update the bus capacity only greater than zero and less than 60.
7. Write a Procedure which will print tour details, a driver is going to take it. ( pass route\_no as parameter)

Route No: \_\_\_\_\_ Vehicle number: \_\_\_\_\_

Transport Details

| Source | Destination | Start date | Total days | Capacity |
|--------|-------------|------------|------------|----------|
| Xxx    | Xxx         | Xxx        | 999        | 999      |

## **SET 11**

Create the database BUS TRANSPORT and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

### **TRAIN MASTER:**

| FIELD NAME     | DATA TYPE     | CONSTRAINTS   |
|----------------|---------------|---|
| TRAIN NUMBER   | VARCHAR2(6)   | PRIMARY KEY AND LAST TWO CHARS SHOULD BE 'DN' OR 'UP' |
| TRAIN NAME     | VARCHAR2(25)  | NOT NULL  |
| ARRIVAL TIME   | DATE          | NOT NULL  |
| DEPARTURE TIME | DATE          | NOT NULL  |
| NO OF HOURS    | NUMBER(5,2)   | NOT NULL  |
| SOURCE STATION | VARCHAR2(25)  | NOT NULL  |
| END STATION    | VHARCHAR2(25) | NOT NULL  |

### **PASSENGER DETAILS:**

| FIELD NAME    | DATA TYPE   | CONSTRAINTS                           |
|---------------|-------------|---------------------------------------|
| TICKET NUMBER | NUMBER(5)   |                                       |
| TRAIN NUMBER  | VARCHAR2(6) | FOREIGN KEY REFERENCE TO TRAIN_MASTER |

|                |              |  |
|----------------|--------------|--|
|                |              | THIS RELATED RECORD SHOULD BE DELETED IF MASTER RECORD IS DELETED. |
| SEAT NUMBER    | NUMBER(2)    | NOT NULL   |
| PASSENGER NAME | VARCHAR2(35) | NOT NULL   |
| AGE            | NUMBER(2)    | NOT NULL   |
| GENDER         | CHAR(1)      | SHOULD BE 'M' FOR MALE OR 'F' FOR FEMALE                           |
| TRAVEL DATE    | DATE         |  |
| CLASS          | VARCHAR2(4)  | SHOULD BE IN (IA, IIA, IIIA, IC, II)                               |

### **TRAIN SEAT MASTER:**

| FIELD NAME   | DATA TYPE   | CONSTRAINTS   |
|--------------|-------------|---|
| TRAIN NUMBER | VARCHAR2(6) | FOREIGN KEY REFERENCE TO TRAIN_MASTER<br>THIS RELATED RECORD SHOULD BE DELETED IF MASTER RECORD IS DELETED. |
| CLASS        | VARCHAR2(4) | SHOULD BE IN (IA, IIA, IIIA, IC, II)  |
| TOTAL SEATS  | NUMBER(2)   | SHOULD BE >= 25 AND <= 90   |

### **TRAIN DAY MASTER:**

| FIELD NAME   | DATA TYPE   | CONSTRAINTS   |
|--------------|-------------|---|
| TRAIN NUMBER | VARCHAR2(6) | FOREIGN KEY REFERENCE TO TRAIN_MASTER<br>THIS RELATED RECORD SHOULD BE DELETED IF MASTER RECORD IS DELETED. |
| DAY          | VARCHAR2(3) | VALUE SHOULD BE IN 'MON' ... TO ... 'SUN'   |

Do as directed:

1. Give all the train names starting from “Bombay” and going to “Ahmedabad” on Tuesday and Wednesday.
2. List all trains which are available on Sunday.
3. Give classwise seat availability on 10-June-2018 for train 9012DN.
4. List total seats classwise for train running on Thursday.
5. List train names which have no sleeper class.
6. List train number which run on Monday during 8:00: am to 1:00pm.
7. Write a procedure which will print all train details going from Baroda to Bangalore.
8. Write a function which will print arrival time and departure time for a given train. ( pass train no as a parameter)
9. Write a trigger which do not allow to insert or update passenger record if age is greater than 100.

## **SET 12**

CUSTOMER(cid, fname, lname, city, country, phone)

ORDER (oid, oDate, oNumber, cid, oTotalAmount)

1. List the number of customers in each country. Only include countries with more than 100 customers.
2. List the number of customers in each country, except China, sorted high to low. Only include countries with 5 or more customers.
3. List all customers with average orders between Rs.5000 and Rs.6500.
4. Create a trigger that executes whenever country is updated in CUSTOMER table.
5. Create a function to return customer with maximum orders.
6. Create a procedure to display month names of dates of ORDER table. The month names should be unique.

## **SET 13**

**DISTRIBUTOR (dno, dname, daddress, dphone)**

**ITEM (itemno, itemname, colour, weight)**

**DIST\_ITEM (dno, itemno, qty)**

1. Add a column CONTACT\_PERSON to the DISTRIBUTOR table with the not null constraint.
2. Create a view LONDON\_DIST on DIST\_ITEM which contains only those records where distributors are from London. Make sure that this condition is checked for every DML against this view.
3. Display the details of all those items that have never been supplied.
4. Delete all those items that have been supplied only once.
5. List the names of distributors who have an 'A' and also a 'B' somewhere in their names.
6. Count the number of items having the same colour but not having weight between 20 and 100.
7. Display all those distributors who have supplied more than 1000 parts of the same type.
8. Display the average weight of items of the same colour provided at least three items have that colour.
9. Display the position where a distributor name has an 'OH' in its spelling somewhere after the fourth character.
10. Count the number of distributors who have a phone connection and are supplying item number 'I100'.
11. Create a view on the tables in such a way that the view contains the distributor name, item name and the quantity supplied.
12. List the name, address and phone number of distributors who have the same three digits in their number as 'Mr. Talkative'.
13. List all distributor names who supply either item I1 or I7 or the quantity supplied is more than 100.
14. Display the data of the top three heaviest ITEMS.

## **SET 14**

- a) Write a PLSQL block which will print Employee list (Empno and Name)  
EMP (empno, empnm, empadd, salary, date\_birth, joindt, deptno)
- b) Write a function that returns total number of incomplete jobs, using table  
JOB (jobid, type\_of\_job, status)

- c) Write a function which displays the number of items whose weight fall between a given ranges for a particular color using table ITEM (itemno, name, color, weight)
- d) Write a procedure to display top five highest paid workers who are specialized in "PAINTING" using table WORKER (workerid, name, wage\_per\_hour, specialized\_in, manager\_id)

### **SET 15**

EMP (empno, empnm, empadd, salary, date\_birth, joindt, deptno)  
DEPT (deptno, deptnm)

Write a PL/SQL block (table above EMP-DEPT table) which takes as input Department name and displays all the employees of this department who has been working since last five years

### **SET 16**

EMPMAS (empno, name, pfno, empbasic, deptno, designation)  
DEPT (DNO, DNAME)

Rules: HRA = 15% of basic  
DA = 50% of basic  
Medical = 100  
PF = 8.33% of basic

Print Salary slip. Design your own format

### **SET 17**

**Consider the Bank schema as**

**ACCOUNT (AC\_NO, NAME, AC\_TYPE, BALANCE\_AMT, BALANCE\_DATE)**  
**TRANSACTION (AC\_NO, DATE, TR\_TYPE, AMOUNT, PREV\_BALANCE, REMARK)**

Note: 1. AC\_type may be S for saving or C for current, 2. TR\_type may be D for deposit or W for withdrawal.

- a. Write a procedure to print the Bank Transaction details by passing from and to dates.

### **SET 18**

**BRANCH (branch\_no, area, city)**

**MEMBERS (mno, name branch\_no, salary, manager\_no)**

Note: Manager can be from one of the members.

1. Write a procedure which list the name of members who earns more than that of his managers.
2. Write a procedure which gives details of employee having maximum salary branch wise.

### **SET 19**

Employee (eid, fname, lname, salary)

1. Use a Cursor for Loop inside a function to calculate and return total paid salary to all employees by the company.

2. Modify the function created above to become a procedure and display the total paid salary from the procedure itself. Instead of calculating for all employees, calculate only for those employees whose name starts from a character passed as parameter to the procedure and hence to the cursor.

## **SET 20**

Consider the DUAL and data dictionary tables/views to solve the following Queries.

1. Find out the names of all the tables, views and constraints associated with current tables in the system.
2. Write a query to add 15 days to the current date.
3. Write a query to Add and subtract 5 months from the current month.
4. Find out the ASCII equivalent of character 'M'.
5. Find out the character equivalent of ASCII 67, 65 and 84.
6. Write a query to find the last day of the month.
7. Find out how many days are left in the current month.
8. Write a query to calculate the Date difference between current date and 20/05/2015.
9. Write a query to Calculate the number of months between current date and 03/03/2016.
10. Find out the second occurrence of 'or' from third position in the string 'corporate floor '.
11. Find out log to the base 3 of 81.
12. Convert the string 'gujarat technological university' so that first character of each word is in capital.
13. Convert the string 'jack and jae' Into 'black and blue'.
14. Round off the date 27-July-2016 to the current year.
15. Find out the user name and user id of currently logged on user.