

Date:

Duration: 120 Minutes

Max Marks: 40

- Number in bracket at the end of each question represents marks for that question.
- Execute the SQL script shared with you before attempting the questions.
- Create a Text/SQL file on your Desktop and Name it with your full Roll Number.  
Put answer to below questions numbering from 1 to 17 in the file as per the question number.
- Keep saving the answer file regularly.

(1) Create a table with name **cities** in your schema with following design. (2)

COLUMN NAME	DATA TYPE	LENGTH	CONSTRAINT
-----			
city_id	integer	6	PRIMARY_KEY
city_name	varchar	100	NOT NULL
city_state	varchar	50	NOT NULL

**NOTE: PLEASE EXECUTE THE SHARED SQL SCRIPT FIRST BEFORE ATTEMPTING QUESTIONS BELOW**

(2) Write a query to get details of all students who are older than **18 years**. (1)

(3) Write a query to get all courses whose **courseName** ends with "**ing**". (1)

(4) Write a query to calculate **average age** of all students living in each city. (1)

(5) Display cities and number of students who live in those cities. (1)

(6) Write a query to add a **Foreign Key** constraint in **student** table on column **courseId** using column **courseId** in **course** table. (1)

(7) Using appropriate **join**, write a query to find the number of students who opted for course '**Networking**'. (2)

(8) Using **subquery**, write a query to get details of students who opted for course '**Programming**'. (2)

- (9) Write a query to add a new column in **course** table with name **courseCredit** with decimal value and default value as 4.0 (Use appropriate SQL Datatype) (2)
- (10) Write a query to **update** the **studentName** to '**Michael**' in **students** table whose **studentId** is **4**. (1)
- (11) Write a query to **delete** record from table **student** whose **studentName** is '**Neha**' (1)
- (12) Write a query to create **view vw\_studDetails** with following columns **studentId**, **studentName**, **studentCourse(name of course)**. The view should be sorted according to **studentId** (2)
- (13) Write a procedure **proc\_age** using table **student** to display '**JUNIOR Student**' if **studentAge** is less than 18, '**SENIOR Student**' if **studentAge** is equal to **18**, '**GRADUATE Student**' if **studentAge** is greater than 18. Take **studentId** as input parameter. (2)
- (14) Write **a before delete trigger** on **course** table. If you **delete a row** from **course** table then it will change the value of **courseId** to **null** in the respective entries in **student** table. (2)
- (15) Write a function to find whether a **input string is palindrome or not**, return the result. (2)

**NOTE: CREATE A DATABASE AND CONNECT TO IT USING FOLLOWING COMMAND BEFORE ATTEMPTING QUESTIONS BELOW -**

**use mydba**

**MongoDB**

- (16) Write mongodb query to create a new "CDAC" database with collection "Employee". (2)
- (17) Write mongodb query to print top 5 records in the collection "Employee" in a nice readable format. (2)