

CS355: Database Lab

Date: 21st Sep 2019
Submission Filename: `midsem.txt`

MidSem Lab Test: **Full Marks 50**
Duration: 3 hours

1 Task 1

Create a *University* database as per the following description. Use EXACTLY the same data types and table names mentioned below. Also define primary /foreign key constraints as specified below. Insert some relevant data to the tables (approx 6 to 10 rows in each table). 20 Points

Col Name	Type	Remarks
rollNo	char(8)	primary key
sname	varchar(30)	
deptID	char(2)	E.g. 'CS', 'EE', 'ME', etc
sex	char(1)	
sdob	date	

Table 1: student

Col Name	Type	Remarks
fID	smallint	primary key
fname	varchar(30)	
doj	date	
deptID	char(2)	E.g. 'CS', 'EE', 'ME', etc

Table 2: faculty

Col Name	Type	Remarks
courseID	char(5)	primary key
meetsAt	varchar(8)	E.g. "MoThFr" to indicate Mon, Thur and Fri
room	varchar(5)	
fID	smallint	foreign key to faculty.fID

Table 3: course

Col Name	Type	Remarks
rollNo	char(8)	foreign key to student.rollNo
courseID	char(5)	foreign key to course.courseID
		primary key is rollNo and courseID

Table 4: enrolled

2 Task 2

Write MySQL queries to perform the the followings. Unless specified explicitly try to write the query without using *views*. 10 × 3 = 30 Points

- Find all the details of the youngest enrolled student of course "CS355"
- Find the average age of CS students
- Find the room which is mostly used by "CS students"
- Find the room which is available on Wednesday
- Find the fID and name of the faculty who teaches most no. of courses.
- Find the fID and name of the faculties who have joined between 1st Jan 2012 and 30th Jun 2012 and assigned atleast one course.
- Find the fID of the faculty who does not teach to the student of his own department.
- Create a view facCourse that will show the courses offered by each faculty
- Find the list of students who are taught by all faculties (available in the faculty table)
- Create a procedure [name `procYoungFac`] that will take a *deptID* as one input parameter and will take another output parameter. The output parameter will store the followings-
 - 'No Result': If there is no faculty corresponding to that input deptID
 - Faculty name: if there is only one faculty who has joined recently corresponding to that input deptID
 - 'More than one faculty': If more than one faculty joined recently corresponding to that input deptID

3 Submission

Write all the relevant MySQL queries that you have used to perform *Task 1* and *Task 2*. Submit the queries using a [midsem.txt](#) file. Pls, submit the file using the following submission link only.

<http://172.16.1.252/~samrat/CS355/submission/>