Assignment

Sept23/ DBT/126

Database Technologies

Diploma in Advance Computing

September 2023

**Procedure**

|  |
| --- |
| 1. Create a LOGIN table (username, password, and email). Write a procedure (named ***addUser***) to pass the username, password, and email-ID through the procedure and store the data in the LOGIN table. |
| drop procedure if exists adduser;  delimiter $  create procedure adduser(in x varchar(20),in y varchar(20),in z varchar(20))  BEGIN  insert into login values(x,y,z);  End $  delimiter ; |
|  |
| 1. Create a LOG table having following columns (id (auto\_increment), curr\_date, curr\_time, and message). Write a procedure (named ***checkUser***) to pass the email-ID as an input, check whether passed email-ID is available in LOGIN table or not available. If the email-ID is available then display the username and his password. If the email-ID is not available then, insert (curr\_date, curr\_time, and message) in LOG table. |
| drop procedure if exists p1;  delimiter $  create procedure p1()  BEGIN  create table log(id int auto\_increment primary key,curr\_date date,curr\_time time,message varchar(20));  End $  delimiter ;  drop procedure if exists checkuser;  delimiter $  create procedure checkuser(in \_email varchar(20))  BEGIN  declare x bool;  select true into x from login where email=\_email;  if x THEN  select username,pwd from login;  ELSE  insert into log (curr\_date,curr\_time,message)values(curdate(),curtime(),\_email);  END IF;  End $  delimiter ; |
|  |
| 1. Write a procedure(named getQualification) that takes studentID as a parameter. If studentID is present in the student table, then print his student details along with STUDENT\_QUALIFICATION details and if the studentID is not present display message “Student not found…” (Use: STUDENT, and STUDENT\_QUALIFICATION tables) |
| drop procedure if exists getQualification;  delimiter $  create procedure getQualification(in studentID int)  begin  declare x bool;  select true into x from student where id=studentID;  if x then  select s.\*,sq.\* from student s join student\_qualifications sq on s.id=sq.studentID;      else  select " student not found ";    end if;  end $  delimiter ; |
|  |
| 1. Write a procedure (named addStudent) that inserts a new student with his phone number and his address into the STUDENT, PHONE, and ADDRESS table. |
| DROP PROCEDURE IF EXISTS addstuddent;  delimiter $  CREATE PROCEDURE addstudent( id INT , namefirst varchar(20), namelast varchar(50), dob date, emailid varchar(20) , spID int , number1 varchar(20),isActive bool,aID int,address varchar(20))  BEGIN  insert into student values ( id,namefirst,namelast,dob, emailid);  insert into student\_phone values ( spid,id,number1,isactive);  insert into student\_address values ( aid,id,address);  end $  delimiter ; |
|  |
| 1. Write a procedure (named addQualification) that takes studentID, and qualification details as a parameter. If studentID is present in the STUDENT table, then insert the qualification in STUDENT\_QUALIFICATION table and return a message “Record inserted” or else print ‘Student not found’. (hint: using OUT parameter) (Use: STUDENT, and STUDENT\_QUALIFICATION tables) |
| DROP PROCEDURE IF EXISTS addQualification;  delimiter $  create procedure addQualification(sID int,studentid int,name varchar(20),college varchar(20),university varchar(40),marks int,year2 varchar(20))  BEGIN  declare x bool;  select true into x from student s where s.ID=studentID;  if x then  insert into student\_qualifications values (sid,studentid,name,college,university,marks,year2);  else  select "Invalid Student id";  END IF;  end $  delimiter ; |
|  |