­­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 login;  delimiter $  CREATE PROCEDURE login(name varchar(30) ,password varchar(30),emailid varchar(30))  BEGIN  CREATE table t1(name varchar(30) ,password varchar(30),emailid varchar(30));  insert into t1 VALUES(name,password,emailid);  end $  delimiter ;  drop PROCEDURE if EXISTS pro2;  delimiter $  CREATE PROCEDURE pro2(id int ,CURR\_DATE DATE ,CURRE\_time TIME,message varchar(30))  BEGIN  CREATE table t3(id int PRIMARY key auto\_increment,CURR\_DATE DATE ,CURRE\_time TIME,message varchar(30));  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 checkuser;  delimiter $  CREATE procedure checkuser(\_emailid varchar(30), message varchar(20))  BEGIN  declare x BOOLEAN;  SELECT true into x from t1 WHERE emailid=\_emailid;  if (x=true)  THEN  SELECT name,password from t1 WHERE emailid=\_emailid;  ELSE  insert into t3(CURR\_DATE,CURRE\_time,message) VALUES(CURRENT\_DATE() , current\_time(), message);  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(studentid int )  BEGIN  declare x BOOL;  SELECT true into x from student WHERE id=studentid;  if(x=true)  THEN  SELECT \* from student s join student\_qualifications sq on s.id=sq.studentid where s.id=studentid;  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 addStudent;  delimiter $  create procedure addStudent(id int ,namefirst varchar(20) ,  namelast varchar(30),dob varchar(30),emailid varchar(30),num varchar(30),address varchar(30))  BEGIN  declare x BOOL;  insert into student(id,namefirst,namelast,dob,emailid) VALUES(id,namefirst,namelast,dob,emailid);  insert into student\_phone(id,studentid,number) VALUES(60,id,num);  insert into student\_address(id,studentid,address) VALUES(67,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(200), college varchar(200), university varchar (200), marks varchar (200), years int)  BEGIN  declare x bool ;  SELECT true into x from student where id = studentid;  if(x=true)  then  insert into student\_qualifications VALUES(sid,studentid,name, college, university, marks, years);  ELSE  SELECT 'student not found';  end if;  end $  delimiter ; |
|  |