­­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(username varchar(20),password varchar(20),emailid varchar(40))  BEGIN  insert into login values(username, password, emailid);  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(40))  BEGIN  declare flag bool;  select true into flag from login where emailid = \_emailid;  if flag THEN  select username, password from login where emailid = \_emailid;  ELSE  insert into log1 (curr\_date,curr\_time,message) values(CURRENT\_DATE(),CURRENT\_TIME(),'emailid not found!');  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 getQualifications;  delimiter $  create procedure getQualifications(\_Studentid int)  BEGIN  declare flag bool;  select true into flag from student where id = \_Studentid;    if flag THEN  select \* 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 addStudent;  delimiter $  create procedure addStudent(\_id int, \_namefirst varchar(30), \_namelast varchar(30), \_dob date, \_emailid varchar(30), \_number varchar(30), \_address varchar(50))  BEGIN  declare flag bool;  select true into flag from student s where s.id = \_id;    if flag THEN  select "Student already present ...";  ELSE  insert into student (id,namefirst, namelast, dob, emailid) values(\_id, \_namefirst, \_namelast, \_dob, \_emailid);  insert into student\_phone (id,studentid, number) values(100,\_id, \_number);  insert into student\_address (id,studentid, address) values(100,\_id, \_address);  end if ;  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(\_studentid int, \_name varchar(20), \_college varchar(20), \_university varchar(20), \_marks int, \_year varchar(20), out message varchar(20))  BEGIN  declare a BOOLEAN;  declare \_id int;  select true into a from student where id = \_studentid;    IF a THEN  select max(id)+1 into \_id from student\_qualifications;  insert into student\_qualifications values(\_id, \_studentid, \_name, \_college, \_university, \_marks, \_year);  set message := "Record inserted ";  else  set message := "Student not found ";  END IF ;  END $  delimiter ; |
|  |