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(25), emailid varchar(45))  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 loguser;  delimiter $  create procedure loguser()  begin  create table log(id int primary key auto\_increment,curr\_date date, curr\_time time, message varchar(25));  end $  delimiter ;  drop procedure if exists checkUser;  delimiter $  create procedure checkUser(in x varchar(25))  begin  declare flag bool;  select true into flag from login where emailid = x;  if flag then  select username, password from login where emailid = x;  else  insert into log(curr\_date, curr\_time, message) values (CURRENT\_DATE, CURRENT\_TIME, "not exist");  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 \_id int)  begin  declare flag bool;  select true into flag from student where id = \_id;  if flag then  select s.\*, sq.\* from student s inner join student\_qualifications sq on s.id = sq.studentid where s.id = \_id;  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(25), \_namelast varchar(25), \_number int, \_address varchar(45))  begin  declare exit handler for 1062 select "duplicate entry";  insert into student(id, namefirst, namelast) values(\_id, \_namefirst, \_namelast);  insert into student\_phone(ID, Studentid, number, isActive) values(\_id, \_id, \_number, true);  insert into student\_address(ID, Studentid, address) values(\_id, \_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(\_id int, \_sid int, \_name varchar(25), \_college varchar(25), \_university varchar(25), \_marks varchar(25), \_year int, out z varchar(25))  begin  declare flag bool;  select true into flag from student where id = \_id;  if flag then  insert into student\_qualifications(id, studentid, name, college, university, marks, year) values(\_id, \_sid, \_name, \_college, \_university, \_marks, \_year);  set z = "record inserted";  else  set z = "student not found";  end if ;  end $  delimiter ; |
|  |