Assignment

March23/ DBT/126

Database Technologies

Diploma in Advance Computing

March 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(\_name varchar(40),\_pwd varchar(40),\_email varchar(40))  BEGIN  create table if not exists login(username varchar(40),password varchar(40),email varchar(40));  insert into login values(\_name,\_pwd,\_email);  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(\_email varchar(40))  BEGIN  declare count int default 0;  create table if not exists login(username varchar(40),password varchar(40),email varchar(40));  create table if not exists log(id int auto\_increment, curr\_time time, curr\_date date,message varchar(150));  select count(email) into count from login where email=\_email;  if count >= 1 THEN  select username,password from login;  ELSE  select "User not found";  insert into log(curr\_date,curr\_time,message) values(curtime(),curdate(),concat(\_email," was searched "));  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(\_id int)  BEGIN  declare count int default 0;  select count(id) into count from student where id=\_id;  if count > 0 then  select \* from student where id=1;  select \* from student\_qualifications where studentid=1;  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(\_first varchar(45),\_last varchar(45),\_dob date, \_email varchar(128),\_number varchar(10),\_address varchar(128))  BEGIN  declare \_id int default 0;  declare \_id\_p int default 0;  declare \_id\_a int default 0;  select max(id)+1 into \_id from student;  select max(id)+1 into \_id\_p from student\_phone;  select max(id)+1 into \_id\_a from student\_address;  insert into student values(\_id,\_first,\_last,\_dob,\_email);  if \_id != 0 then  insert into student\_phone values(\_id\_p,\_id,\_number,1);  insert into student\_address values(\_id\_a,\_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(\_id int,\_name varchar(128),\_college varchar(128),\_uni varchar(128),\_marks varchar(45),\_year int)  BEGIN  declare c int default 0;  declare \_id\_q int default 0;  select count(id) into c from student where id=\_id;  select max(id)+1 into \_id\_q from student\_qualifications;  if c > 0 then  insert into student\_qualifications values(\_id\_q,\_id,\_name,\_college,\_uni,\_marks,\_year);  select "Record inserted";  ELSE  select "Student not found";  end if;  end $  delimiter ; |
|  |