­­Assignment

**PRN NO - 230945920032\_230945920091**

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(10),EmailID varchar(30))**  **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 cheakUser;**  **delimiter $**  **create procedure cheakUser(EID varchar(30))**  **Begin**  **declare flag bool;**  **select true into flag from login where EmailID=EID;**  **IF flag then**  **select name,password from login where EmailID=EID ;**  **else**  **insert into LOG (curr\_date, curr\_time, message) values (CURRENT\_DATE(),CURRENT\_TIME(),"Record not found");**  **select ("Record 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 getQualification;**  **delimiter $**  **create procedure getQualification(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 q on s.id=q.studentID where s.ID=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(namefirst varchar(20),namelast varchar(20), DOB date, emailID varchar(128), number VARCHAR(20), address varchar(128))**  **Begin**  **declare z int;**  **declare y,x int;**  **select max(ID)+1 into z from student;**  **insert into student values (z,namefirst ,namelast , DOB, emailID);**  **select max(ID)+1 into y from student\_phone;**  **insert into student\_phone values(y,z,number,1);**  **select max(ID)+1 into x from student\_address;**  **insert into student\_address values(x,z,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, name varchar(128), college varchar(128), university varchar(128), marks varchar(45), year int )**  **Begin**  **declare flag bool;**  **declare z int;**  **select true into flag from student where id=sid;**  **if flag then**  **select max(ID)+1 into z from student\_qualifications;**  **insert into student\_qualifications values ( z, sid, name, college, university, marks, year);**  **else**  **SELECT 'student not found';**  **end if;**  **end $**  **delimiter ;** |
|  |