­­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 store the data in the procedure and LOGIN table. |
| drop procedure if exists addUser;  delimiter %  create procedure addUser(username varchar(20), password varchar(20), email varchar(20))  begin  insert into login values(username, password, 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(20))  BEGIN  DECLARE p BOOLEAN;  SELECT TRUE into p WHERE \_email IN (select emailid from login1);  IF p THEN  SELECT username, password from login1 WHERE emailid = \_email;  ELSE  INSERT INTO log1 (curr\_date, curr\_time, msg) VALUES (CURRENT\_DATE(), CURRENT\_TIME(), DEFAULT);  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(sid int)  BEGIN  if sid in(select studentid from student\_qualifications) then  select \* from student s1 join student\_qualifications sq on s1.id=sq.studentid where studentid=sid;  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(s\_fname varchar(20),s\_lname varchar(20),s\_phone varchar(20),s\_address varchar(20))  BEGIN  set @st=0;  set @stp=0;  set @sta=0;  select max(id)+1 into @st from student;  select max(id)+1 into @stp from student\_phone;  select max(id)+1 into @sta from student\_address;  insert into student(id,namefirst,namelast) VALUES(@st,s\_fname,s\_lname);  insert into student\_phone(id,p\_number,studentid,isActive) VALUES(@stp,s\_phone,@st,1);  insert into student\_address(id,studentid,address) VALUES(@sta,@st,s\_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 (siid int ,sq\_name varchar(20))  BEGIN  set @sid=0;  select max(id)+1 into @sid from student\_qualifications;  if siid in(select id from student) THEN  insert into student\_qualifications(id,studentid,name) values(@sid,siid,sq\_name);  select "Record inserted...";  ELSE  select "Student Not Found...";  end if;  end $  delimiter ; |
|  |