Assignment

Feb20/ DBT/128

Database Technologies

Diploma in Advance Computing

February 2020

**Trigger**

|  |
| --- |
| 1. Write a trigger (named insertStudent) that saves the message "Record inserted successfully" in LOG(current date, current time, and message columns) table as soon as you insert the record in STUDENT table. |
| drop trigger if exists insertStudent;  delimiter $$  create trigger insertStudent before insert on student for each row  begin    insert into log(curr\_date, curr\_time, message) values (current\_date, current\_time, 'Record Inserted');  end$$  delimiter ; |
|  |
| 1. Write a trigger (named insertDuplicate) on STUDENT table, that as when we INSERT a record in STUDENT table the same record should get duplicated (INSERTED) in STUDENT\_LOG table. (Create STUDENT\_LOG table, having the same structure as STUDENT table). |
| Create table student\_log as select \* from student where 1=2;  drop trigger if exists insertDuplicate;  delimiter $$  create trigger insertDuplicate before insert on student for each row  begin    insert into student\_log (id, namefirst, namelast, dob, emailid) values (new.id, new.namefirst, new.namelast, new.dob, new.emailid);  end$$  delimiter ; |
|  |
| 1. Write a trigger(named updateStudent) on STUDENT table, that as soon as we UPDATE student emailID column data in STUDENT table, the update record should get inserted in STUDENT\_LOG table. |
| drop trigger if exists updateStudent;  delimiter $$  create trigger updateStudent before update on student for each row  begin  insert into student \_log (id, namefirst, namelast, dob, emailid) values (old.id, old.namefirst, old.namelast, old.dob, old.emailid);  end$$  delimiter ; |
|  |
| 1. Write a trigger (named deleteStudent) on STUDENT table, that as soon as we DELETE any record from STUDENT table, then that record should get inserted into STUDENT\_LOG table. |
| drop trigger if exists deleteStudent;  delimiter $$  create trigger deleteStudent before delete on student for each row  begin  insert into student \_log (id, namefirst, namelast, dob, emailid) values (old.id, old.namefirst, old.namelast, old.dob, old.emailid);  end$$  delimiter ; |
|  |
| 1. Write a trigger (named insertValidation) on STUDENT table, that if today is Sunday then, no record should get inserted in STUDENT table. |
| drop trigger if exists insertValidation;  delimiter $$  create trigger insertValidation after insert on student for each row  begin  if date\_format (now(), '%W') = 'Sunday' then  signal sqlstate '42000' set message\_text = 'Record cannot be inserted';  end if;  end$$  delimiter ; |
|  |