**MySQL Workbench Class Exercises, Summer 2021**

**create database TriggerExercises;**

**use TriggerExercises;**

**---------------------------------------------------------**

**Trigger to create Log File**

**Create table MyLog (message varchar(70));**

**create table Mycustomer(Fname varchar (20), Lname varchar(20),salary integer);**

**Delimiter $$**

**create trigger add\_customer after insert on Mycustomer**

**for each row**

**begin**

**insert into Mylog values(concat('customer has been added by ',current\_user(), ' ', new.Lname, ' on ',current\_date()));**

**end;**

**$$**

**insert into Mycustomer values('Keith','Jackson',2330);**

**insert into Mycustomer values('Mehdi', 'Owrang', 2100);**

**select \* from Mycustomer;**

**select \* from Mylog;**

**=================================================================**

**Delimiter $$**

**create trigger update\_customer after update on Mycustomer**

**for each row**

**begin**

**insert into Mylog values(concat('customer has been updated by ',current\_user(),' ', new.Lname , ' on ',current\_date()));**

**end;**

**$$**

**update Mycustomer set salary=salary + 0.10 \* salary where Lname='Owrang';**

**select \* from Mylog;**

**=================================================================**

**Delimiter $$**

**create trigger delete\_customer after delete on Mycustomer**

**for each row**

**begin**

**insert into Mylog values(concat('customer has been deleted by ',current\_user(),' ', old.Lname, ' on ',current\_date()));**

**end;**

**$$**

**delete from Mycustomer where Lname='Jackson';**

**select \* from Mycustomer;**

**select \* from Mylog;**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Trigger to create Summary Table**

**create table Student(id integer, name varchar(20), major varchar(20), gpa double);**

**create table Registration(name varchar(20), courseNo varchar(10), semester varchar(15), grade varchar(4));**

**create table major\_gpa\_summary(major varchar(15),mingpa double, maxgpa double, avggpa double);**

**Delimiter $$**

**create trigger major\_gpa\_insert after insert on Student**

**for each row**

**begin**

**delete from major\_gpa\_summary;**

**insert major\_gpa\_summary**

**select major, min(gpa),max(gpa),avg(gpa) from student group by major;**

**end;**

**$$**

**insert into Student value(1234,"Bob","CS", 3.8);**

**insert into Student value(1255,"Mary","Math", 4.0);**

**insert into student values(2244,"James jones","Art",3.7);**

**insert into student values(1991,"Ali Baba","CS",3.9);**

**insert into student values(4444,"Anna","Math",4.0);**

**select \* from Student;**

**select \* from major\_gpa\_summary;**

**===========================================================**

**Delimiter $$**

**create trigger major\_gpa\_update after update on Student**

**for each row**

**begin**

**delete from major\_gpa\_summary;**

**insert major\_gpa\_summary**

**select major, min(gpa),max(gpa),avg(gpa) from student group by major;**

**end;**

**$$**

**Update Student set gpa=2.0 where id=1234;**

**select \* from major\_gpa\_summary;**

**=============================================================**

**Delimiter $$**

**create trigger major\_gpa\_delete after delete on Student**

**for each row**

**begin**

**delete from major\_gpa\_summary;**

**insert major\_gpa\_summary**

**select major, min(gpa),max(gpa),avg(gpa) from student group by major;**

**end;**

**$$**

**Delete from Student where gpa <=2.0;**

**select \* from student;**

**select \* from major\_gpa\_summary;**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Enforcing Referential Integrity rule**

**Delete Parent, we need to delete children**

**Student is Parent, registration is Child.**

**insert into registration values("Bob","CSC434","Spring 2020","A");**

**Insert into registration values("Ali Baba","CSC493","Fall 2019","B");**

**This version of MySQL doesn't yet support 'multiple triggers with the same action time and event for one table'.**

**major\_gpa\_delete after delete on Student:**

**need to drop the major\_gpa\_delete trigger.**

**drop trigger major\_gpa\_delete;**

**Delimiter $$**

**create trigger student\_delete after delete on Student**

**for each row**

**begin**

**delete from registration where name=old.name;**

**end;**

**$$**

**delete from Student where name='Bob';**

**select \* from student;**

**select \* from registration;**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Trigger for Enforcing Business Rules:**

**Account deposit must be at least $500.**

**create table Account(AccountType varchar(10), AccountNo varchar(10), Customer varchar(20), Balance decimal(12,2));**

**Delimiter $$**

**create trigger checkdeposit\_account before insert on Account**

**for each row**

**begin**

**if new.balance < 500 then**

**SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Deposit into account can not be < 500';**

**end if;**

**end;**

**$$**

**Activating Trigger**

**insert into Account values('checking','12345','Mary Smith',2300.50);**

**insert into Account values('saving','22123','Joe Jones', 300.00);**

**insert into Account values('checking','23231','Jack Brown',1300.50);**

**insert into Account values('saving','22123','Joe Jones', 450.00);**

**select \* from Account;**

**============================================================**

**Attribute Domain Checking**

**create table AUMajors(majors varchar(20));**

**Create Domain Table:**

**insert into AUMajors values("Art");**

**insert into AUMajors values("Biology");**

**insert into AUMajors values("Engineering");**

**insert into AUMajors values("Math");**

**insert into AUMajors values("Medicine");**

**insert into AUMajors values("Nursing");**

**insert into AUMajors values("Music");**

**select \* from AUMajors;**

**// can not use multiple triggers with the same timing and event on the same table. (before insert on Student)**

**Delimiter $$**

**create trigger MajorCheck before insert on Student**

**for each row**

**Begin**

**declare temp Int; set temp=0;**

**select count(\*) into temp from AUMajors**

**where majors = new.major;**

**if temp=0 then**

**SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Invalid major';**

**end if;**

**end;**

**$$**

**Activation:**

**insert into student values(3333,"Mary Jackson","Biology",3.6);**

**insert into student values(4141,"Isabella","Medicine",3.9);**

**insert into student values(4321,"Anna","Physics",4.0);**

**insert into student values(7676,"BiBe","Dancing",3.6);**

**select \* from Student;**

**================================================================**