Nalongsone Danddank Student ID : 14958950 StarID: jf3893pd

Email: [nalongsone.danddank@my.metrostate.edu](mailto:nalongsone.danddank@my.metrostate.edu)

**Metropolitan State University**

**ICS 311 —Database Management Systems**

**Class Exercise – Advanced SQL**

**Exercise 1:**

Write a trigger, each time a row is Inserted in the table Line, Insert a row in Invoice table. Assume that cus\_code is 1001 and inv\_date is today’s date

* Invoice(inv\_number int, cus\_code varchar(5), inv\_date date)
* Line(inv\_number int, prod\_code varchar(5), line\_units int)

***Solution:***

*use nalongsonedanddank;*

*drop trigger if exists insert\_line\_invoice ;*

*delimiter $$*

*create trigger insert\_line\_invoice*

*after insert on line*

*for each row*

*begin*

*declare invNum int;*

*declare cusCode int;*

*declare InvDate date;*

*set invNum = new.inv\_number + 1 ;*

*insert into invoice(inv\_number, cus\_code, inv\_date) values(invNum, 10010, curdate());*

*end $$*

*delimiter ;*

*insert into line(inv\_number, prod\_code, line\_units) value(1006, 12322, 10);*

**Exercise 2:**

Write an AFTER Insert trigger for the following Employee table. Check Date of Birth and calculate age. Update AGE field with calculated value in Employee\_Info table

* Employee (Emp\_ID int, DOB date)
* Employee\_Info (Emp\_ID int, Fname char, Lname char, Age int)
* Set variable Current\_Date to CURDATE()
* Substract DOB from Current\_Date to find Age

*Hint: Consider using variables to calculate and store Age*

**Solution:**

*use nalongsonedanddank;*

*drop table if exists Employee;*

*create table Employee(Emp\_ID int primary key, DOB date);*

*drop table if exists Employee\_info;*

*create table Employee\_Info(Emp\_ID int primary key, Fname char(20), Lname char(20), Age int);*

*drop trigger if exists emp\_age;*

*delimiter $$*

*create trigger emp\_age*

*after insert on Employee*

*for each row*

*begin*

*declare count\_age integer;*

*set count\_age = timestampdiff(year, new.DOB, curdate());*

*update employee\_info E set Age = count\_age where E.Emp\_ID = new.Emp\_ID;*

*end $$*

*delimiter ;*

*insert into employee\_info(Emp\_ID, Fname, Lname, Age) value(1, 'Ping', 'Wang', 0);*

*insert into employee(Emp\_ID, DOB) value(1, '1986-12-14');*

*select \* from employee\_info;*

**Exercise 3:**

Consider Department table from University database. Create a procedure to Insert a new row, but increase the budget amount by x%. Percentage amount should be input parameter for the procedure. Assume your own values for dept\_name, building, budget.

department

(dept\_name varchar(20),

building varchar(15),

budget numeric(12,2),

primary key (dept\_name)

);

Write Call statement

***Solution:***

*use university;*

*drop procedure if exists department\_proc;*

*delimiter $$*

*create procedure department\_proc(in dname varchar(20), in \_building varchar(15), in \_budget float, in percentage float)*

*begin*

*declare inc\_budget float;*

*set inc\_budget = \_budget + \_budget\*(percentage/100);*

*insert into department(dept\_name, building, budget) value(dname, \_building, inc\_budget);*

*end $$*

*delimiter ;*

*call department\_proc('department name', 'CS', 10000, 10);*

**Exercise 4:**

Consider Department and Instructor tables from University database. Create a procedure to Insert a line in Instructor table. Check if dept\_name is Physics, increase salary by 20%. Input parameters are ID, Name, dept\_name, salary

Department (dept\_name varchar(20), building varchar(15), budget numeric(12,2))

Instructor (ID varchar(5), name varchar(20), dept\_name varchar(20), salary numeric(8,2)

*Solution:*

*use university;*

*drop procedure if exists dep\_inst\_proc;*

*delimiter $$*

*create procedure dep\_inst\_proc(in in\_id int, in in\_name varchar(20), in in\_dept\_name varchar(20), in in\_salary numeric(8,2))*

*begin*

*declare dname varchar(20);*

*declare new\_salary numeric(8,2);*

*select dept\_name into dname from department where dept\_name = 'Physics';*

*if dname = in\_dept\_name then*

*set new\_salary = in\_salary + in\_salary\*0.2;*

*else*

*set new\_salary = in\_salary;*

*end if;*

*insert into instructor(id,name,dept\_name,salary) value(in\_id, in\_name, in\_dept\_name, new\_salary);*

*end $$*

*delimiter ;*

*call dep\_inst\_proc(33333, 'Ping Wang','Physics', 90000);*

*call dep\_inst\_proc(33332, 'Pink Wang','History', 90000);*