**Class Exercise – Advanced SQL**

**Write code for all 4 exercises. Max (10) bonus points will be awarded for submitting answers for at least (1) trigger and (1) procedure exercise.**

**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)

**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*

**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

**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)