

AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course No: CSE4126

Course Title: Distributed Database Systems Lab

Spring 2021 | Lab Final Examination | Set A | Marks 30 | Time: 60 Minutes

1. Answer the following questions. [10]

- a. Why are OUT Parameters used in functions even though functions already have return types? [1]
- b. What is the purpose of the DUAL table in SQLplus? Write down 3 examples of how it can be used. [3]
- c. What type of exception do we use when we do not want to specify the type of error we want to handle? [1]
- d. Explain when the following predefined exceptions are raised: [3]
 - i. DUP_VAL_ON_INDEX
 - ii. CASE_NOT_FOUND
 - iii. STORAGE_ERROR
- e. What is the difference between IN parameter and OUT parameter? [1]
- f. What is the purpose of using aliasing? [1]

2. Assume that, we have a table named "BOOK" that has four attributes :- [5]

- *bid* (type: int),
- *bookName* (varchar2),
- *price* (int)
- *copies* (int)

Also assume that, there are already three rows inserted in "BOOK" table as follows –

bid	bookName	price	copies
1	War and Peace	450	10
2	Macbeth	250	5
3	Harry Potter	350	20

Create five triggers (trig1, trig2, trig3, trig4, trig5) so that when we run the following queries they will work as expected:

insert into book values (4,'A brief history of time',550,20);

This will trigger trig1, trig2. Only trig2 will be triggered after insertion.

update book set copies=copies-1 where bookname='Macbeth';

This will trigger trig2, trig4.

delete from book where price<300;

This will trigger trig3, trig4. Only trig3 will be triggered before deletion.

delete from book where bid=1;

This will trigger trig4.

update book set price=300 where bookname='Harry Potter';

This will trigger trig2, trig4, trig5. Only trig5 will be triggered before insertion.

Note that, inside triggers you have to print the trigger name. For example – when the trigger trig1 is executed it will print “trig1 activated”, for trig2 trigger - print “trig2 activated” and so on.

3. Consider that you have two database tables named “Person” and “Salary”, which belong to the HR department of a company. Suppose, you have an .sql file which shows the description of the table as follows:

[10]

```
CREATE TABLE Person (pID int, pName VARCHAR2(50), dateOfBirth date,
basicSalary int,
PRIMARY KEY (pID));
```

```
CREATE TABLE Salary (pID int, totalSalary number(7), age number(2),
FOREIGN KEY (pID) REFERENCES Person(pid));
```

```
insert into Person values (1,'Ikhtiar','10-Jan-1980',25000);
```

```
insert into Person values (2,'Uddin','10-Feb-1980',35000);
```

```
insert into Person values (3,'Mohammad','10-Mar-1980',45000);
```

```
insert into Person values (4,'Bakhtiyar','10-Apr-1980',55000);
```

```
insert into Person values (5,'Khilji','10-May-1980',65000);
```

Notice that no data has been inserted into the **Salary** table yet. This is because all the data for this table needs to be derived/calculated from the **Person** table. Remember that, total salary means the summation of basic salary + house rent (which is 50% of basic salary) + medical allowance (which is 20% of basic salary) + travel allowance (which is 20% of basic salary) + utilities (which is Tk. 5000). Also remember that, simply subtracting the date of birth from the current date gives your age in days.

Your tasks are as follows, to be written in PL/SQL:

- You need to write a **function**, called “AgeCalculator”, that calculates the age of the Person. It takes in the **date of birth** as input from the main block and returns the **age**.
- You need to write a **procedure**, called “SalaryCalculator”, that calculates the total Salary of a Person. It takes in **pID** and **age** as input from the calling block, calculates the total Salary using a formula, and inserts a row into the **Salary** table, with the required data.
- pID must be taken as user input in the main block.
- The function and the procedure need to be in a **package** called “Employees”

4. Suppose, you have a table named STUDENTS which contains the data of all students in a particular school. The school has 2 branches i.e. sites (site1, site2). Now, write a **procedure** that updates the class of the students when run at the starting of each school year, i.e. when they are promoted to the next class. The STUDENTS table looks as follows:

[5]

STUDENTS

student_id (int)	name (varchar2(50))	currentClass (int)
1	Rafiq	2
1501	Shafiq	9

Assume that, all the students pass every year. The data is distributed in the 2 sites as follows:

Student ID Range	Stored At (Site)
1-1500	site1
1501-3000	site2

Make sure that students at both sites get promoted.