**Worksheet on Triggers – 1**

*Kindly save all the statements that you create for this exercise in same folder. Example Make a Folder Trigger 1 and store the files A.sql, B.sql etc*

**Steps:** (Any confusion refer to Chapter 9 on Triggers)

1. Design the following table such that when the trigger executes it should store user information and user activities in this table.

|  |  |  |
| --- | --- | --- |
| Tablename: **LOG\_CONTACTS** | | |
| ***Columns*** | ***Data type*** | ***Constraints*** |
| LOG\_ID | NUMBER(5) | PRIMARY KEY |
| LOG\_DT | DATE | NOT NULL |
| OPERATION\_DONE | VARCHAR2(50) | NOT NULL |
| USERNAME | VARCHAR2(50) | NOT NULL |
| RECORD\_ID | NUMBER(5) | NOT NULL |

1. Create the sequence to automatically generate the values for the primary key of the above table starting from 1
2. Create a trigger which will keep track of **two operations** i.e. when record is added or edited on your **CONTACTS** table. When these events are encountered, insert into the above table the relevant information. They are
   * + LOG\_ID is to be generated.
     + LOG\_DT current system date
     + OPERATION\_DONE a text describing which operation was done.
     + USERNAME who performed this operation and
     + RECORD\_ID stores the CONTACT\_ID value of the record affected i.e. newly added or edited in the **CONTACTS** table.

This is to be inserted **before** the record is added or edited.

1. Test this trigger by adding or editing some records of the CONTACTS table.

Note\*\*: SET SERVEROUTPUT ON is not required while testing as this trigger does not display any message to the user.

1. Write a PL/SQL program to display all the records stored in the **LOG\_CONTACTS** table. This way any time you need to see log records, you can execute this piece of code.