CREATE OR REPLACE TRIGGER ACTION\_Insert\_After AFTER INSERT ON ACTION

FOR EACH ROW

DECLARE Notification\_Owner INTEGER := 0;

BEGIN

SELECT CreatorId INTO Notification\_Owner

FROM CREATOR

WHERE CREATOR.EffectedId = :NEW.DestinationId;

INSERT INTO CREATOR\_NOTIFICATION

VALUES

(Notification\_Owner,:NEW.ActionId,:NEW.ActionType,TO\_DATE(SYSDATE ,'DD/MM/YYYY'));

INSERT INTO EFFECTED

VALUES(:NEW.DestinationId);

END;

/

CREATE OR REPLACE TRIGGER LINKEDIN\_USER\_insert BEFORE INSERT ON LINKEDIN\_USER

FOR EACH ROW

BEGIN

INSERT INTO CREATOR

VALUES(:NEW.CreatorId,Effected\_Seq.NEXTVAL);

END;

/

CREATE OR REPLACE TRIGGER CREATOR\_insert BEFORE INSERT ON CREATOR

FOR EACH ROW

BEGIN

INSERT INTO EFFECTED

VALUES(Effected\_Seq.CURRVAL);

END;

/

CREATE OR REPLACE TRIGGER COMMUNITY\_insert AFTER INSERT ON COMMUNITY

FOR EACH ROW

BEGIN

INSERT INTO CREATOR

VALUES(:NEW.CreatorId,Effected\_Seq.NEXTVAL);

IF :NEW.CommunityType = 'GROUP' THEN

INSERT INTO LINKEDIN\_GROUP

VALUES(:NEW.CommunityId,:NEW.UserId);

ELSE

INSERT INTO COMPANY

VALUES(:NEW.CommunityId,:NEW.UserId,NULL);

END IF;

END;

/