**/\* 8.2.1\*/**

**CREATE OR REPLACE TRIGGER TR\_Trans**

**AFTER INSERT ON TRANSACTION**

**FOR EACH ROW**

**BEGIN**

**INSERT INTO Transaction\_changes**

**(   Cust\_code,**

**Amount**

**)**

**VALUES**

**(**

**:new.cust\_code,**

**:new.amount**

**);**

**END TR\_TRANS;**

**/\*8.2.2\*/**

**CREATE OR REPLACE TRIGGER record\_fare**

**/\* triggering event \*/**

**AFTER INSERT OR UPDATE OF fare ON PASSENG**

**/\* executes for each row \*/**

**FOR EACH ROW**

**DECLARE**

**BEGIN**

**/\* :new refers to the newly updated values \*/**

**/\* :old refers to the old values \*/**

**IF( :new.fare != :old.fare )**

**THEN**

**INSERT INTO newfare VALUES( :new.fare, SYSDATE );**

**END IF;**

**END**

**/\* 8.2.3\*/**

**CREATE OR REPLACE TRIGGER check\_salary**

**/\* firing or triggering event \*/**

**BEFORE INSERT OR UPDATE OF salary ON INFORM**

**/\* will check for each record \*/**

**FOR EACH ROW**

**/\* triggering constraint \*/**

**WHEN (new.organization != 'SELF')**

**DECLARE**

**BEGIN**

**/\* :new refers to the updated record \*/**

**/\* :old refers to the old record \*/**

**IF(  :new.salary > 125000)**

**THEN**

**raise\_application\_error( -20000, 'salary out of range');**

**ELSIF( :new.salary < :old.salary )**

**THEN**

**raise\_application\_error( -20002, 'Decrease in salary');**

**END IF;**

**END;**

**/\*8.2.4\*/**

**CREATE OR REPLACE TRIGGER INSURE1**

**AFTER  UPDATE OF AMOUNT**

**ON INSURE**

**FOR EACH ROW**

**BEGIN**

**Dbms\_output.put\_line('old Amount: '||:old.amount||'     '||'new Amount:  '||:new.amount);**

**END;**

**/\* 8.2.5\*/**

**Create trigger tr\_flight**

**Before insert on flight**

**For each row**

**when ( new.flight\_no != 0)**

**DECLARE**

**flight\_count       number(2);**

**Begin**

**SELECT count(\*) into flight\_count**

**from flight;**

**if ( flight\_count >= 6)**

**THEN**

**raise\_application\_error( -20002, 'cannot accept any new flights');**

**end IF;**

**END;**

**/\*8.2.6\*/**

**Create trigger no\_delete**

**Before delete on booking**

**For each row**

**Begin**

**raise\_application\_error(-20008, 'Delete Not Allowed');**

**End;**

**/\*8.2.7\*/**

**Create or replace trigger check\_station**

**Before insert or update OF str\_pt, dest\_pt on train**

**For each row**

**Begin**

**if ( :new.str\_pt = :new.dest\_pt )**

**then**

**raise\_application\_error( -20000, 'source and destination cannot be same');**

**end if;**

**end;**

**/\*8.2.8\*/**

**CREATE TRIGGER check\_insu**

**BEFORE INSERT OR UPDATE OF insu\_amt ON insu\_vehicle**

**FOR EACH ROW**

**WHEN (new.insu\_amt < 5000 )**

**BEGIN**

**raise\_application\_error( -20020, 'Amount less than 5000');**

**END;**

**/\*8.2.9\*/**

**CREATE TRIGGER LICENSE\_AUDIT**

**BEFORE  UPDATE ON LICENSE**

**FOR EACH ROW**

**WHEN  (new.owner\_code=1306)**

**BEGIN**

**RAISE\_APPLICATION\_ERROR(-20000,' DO NOT UPDATE since new.owner\_code equal to 1306');**

**END;**

**/\*8.2.10\*/**

**Create trigger max\_credit**

**Before insert or update of teacherno, credits on course**

**For each row**

**Declare**

**tot\_credit    number(3);**

**Begin**

**select sum(credits) into tot\_credit**

**from course**

**WHERE teacherno = :new.teacherno**

**GROUP BY teacherno;**

**IF (tot\_credit  + :new.credits >= 10)**

**THEN**

**raise\_application\_error( -20555, 'Credit limits for a teacher exceeds');**

**End if;**

**END max\_credit;**

**/\*8.2.11\*/**

**CREATE TRIGGER NO\_OF\_fundagency**

**BEFORE INSERT OR UPDATE ON SPONSRES**

**FOR EACH ROW**

**DECLARE**

**mcount   number(2);**

**BEGIN**

**/\* checking the number of  projects for a teacher \*/**

**select count(\*) into mcount**

**from sponsres where teacherno = :new.teacherno;**

**IF ( mcount >1)**

**THEN**

**raise\_application\_error( -20500, 'Sponsored by more than one agency');**

**END IF;**

**END;**

**/\*8.2.12\*/**

**Create trigger spons\_check**

**before insert  on sponsres**

**for each row**

**WHEN (new.fundagency != 'DEF' OR new. fundagency != 'LIC' OR new. fundagency !='DS')**

**BEGIN**

**RAISE\_APPLICATION\_ERROR(-20000,'Not an existing funding agencency');**

**END;**

**/\*8.2.13\*/**

**CREATE TRIGGER TRIGGER\_CALLS**

**/\* trigger firing condition \*/**

**BEFORE INSERT OR UPDATE OF AREA\_CODE ON CALLS\_INFO**

**FOR EACH ROW**

**DECLARE**

**mcode NUMBER(6);**

**BEGIN**

**/\* retrieving the area code from the rate\_info table \*/**

**SELECT AREA\_CODE INTO mcode FROM   RATE\_INFO**

**WHERE AREA\_CODE = :NEW.AREA\_CODE;**

**EXCEPTION**

**/\* if the area code doesn't exist then raise error \*/**

**WHEN NO\_DATA\_FOUND THEN**

**RAISE\_APPLICATION\_ERROR( -20002, 'AREA\_CODE:   '||TO\_CHAR(:NEW.AREA\_CODE) ||' NOT EXISTING');**

**END;**

**/\*8.2.14\*/**

**Create trigger cust\_delete**

**After delete on cust\_details**

**For each row**

**When ( old.tel\_no != 0 )**

**Begin**

**/\* inserts the deleted record in logtable \*/**

**INSERT into logtable**

**values( :old.tel\_no, :old.cust\_name, :old.acode, sysdate);**

**END;**

**/\*8.2.15\*/**

**Create trigger del\_40000**

**Before delete on inventory**

**For each row**

**Begin**

**IF(:old.unit\_price < 40000 )**

**Then**

**raise\_application\_error( -20008, 'Unit price < 40000');**

**end if;**

**end;**