***Oefening 1:***

**CREATE OR REPLACE TRIGGER** bus\_jobs  
 **BEFORE UPDATE OF** min\_salary, max\_salary  
 **ON** jobs  
**DECLARE  
BEGIN  
 IF** *USER* **IN** ('STUDENT', 'BEZOEKER') **THEN** *RAISE\_APPLICATION\_ERROR*(-20000, 'Je hebt onvoldoende rechten om deze actie uit te voeren!');  
 **ELSE** DBMS\_OUTPUT.PUT\_LINE('Ben je zeker dat je het min of max salaris wil aanpassen? Indien niet voer dan een ROLLBACK uit!');  
 **END IF**;  
**END**;

***Oefening 2:***

**CREATE OR REPLACE TRIGGER** aiuds\_jh  
 **AFTER INSERT OR UPDATE OR DELETE  
 ON** JOB\_HISTORY  
**DECLARE** v\_actie **VARCHAR2**(15);  
**BEGIN  
 IF INSERTING THEN** v\_actie := 'INSERT';  
 **ELSIF DELETING THEN** v\_actie := 'DELETE';  
 **ELSE** v\_actie := 'UPDATE';  
 **END IF**;  
  
 **INSERT INTO** log\_history  
 **VALUES** (*USER*, *SYSDATE*, *SYSTIMESTAMP*, v\_actie);  
**END**;

***Oefening 3:***

**CREATE OR REPLACE TRIGGER** biuds\_employees  
 **BEFORE UPDATE OF** SALARY **OR INSERT OR DELETE  
 ON** EMPLOYEES  
**DECLARE  
BEGIN  
 IF UPDATING THEN  
 IF** *TO\_CHAR* (*SYSDATE*, 'HH24:MI') **NOT BETWEEN** '09:00' **AND** '17:00' **OR** *TO\_CHAR* (*SYSDATE*, 'DY') = 'MON' **THEN** *RAISE\_APPLICATION\_ERROR* (-20205, 'Je mag de maandlonen enkel aanpassen op maandag tussen 09:00 en 17:00!');  
 **END IF**;  
 **ELSE  
 IF** *TO\_CHAR* (*SYSDATE*, 'DY') **IN** ('SAT', 'SUN') **THEN** *RAISE\_APPLICATION\_ERROR* (-20205, 'Je mag geen employees in het weekend toevoegen of verwijderen!');  
 **END IF**;  
 **END IF**;  
**END**;

***Oefening 4:***

**CREATE OR REPLACE TRIGGER** aur\_emp  
 **AFTER UPDATE OF** salary  
 **ON** EMPLOYEES  
 **FOR EACH ROW  
BEGIN  
 IF** :**NEW**.SALARY < :**OLD**.SALARY **THEN** *RAISE\_APPLICATION\_ERROR*(-20000, 'Loonsverlaging kan niet!');  
 **ELSIF** :**NEW**.SALARY > :**OLD**.SALARY \* 1.05 **THEN** *RAISE\_APPLICATION\_ERROR*(-20000, 'Een loonsverhoging met meer dan 5% kan niet!');  
 **END IF**;  
**END**;

***Oefening 5:***

**CREATE OR REPLACE TRIGGER** aur2\_emp  
 **AFTER UPDATE OF** salary  
 **ON** EMPLOYEES  
 **FOR EACH ROW  
DECLARE** v\_datum EMPLOYEES.HIRE\_DATE%**TYPE**;  
**BEGIN  
 SELECT** HIRE\_DATE  
 **INTO** v\_datum  
 **FROM** EMPLOYEES;  
  
 **IF** *TO\_CHAR*(v\_datum, 'DD/MM/YY') < '01/01/95' **THEN  
 IF** :**NEW**.SALARY < :**OLD**.SALARY **THEN** *RAISE\_APPLICATION\_ERROR*(-20000, 'Loonsverlaging kan niet!');  
 **ELSIF** :**NEW**.SALARY > :**OLD**.SALARY \* 1.05 **THEN** *RAISE\_APPLICATION\_ERROR*(-20000, 'Een loonsverhoging met meer dan 5% kan niet!');  
 **END IF**;  
 **END IF**;  
**END**;

***Oefening 6:***

**CREATE OR REPLACE TRIGGER** aiur\_emp

**BEFORE INSERT OR UPDATE OF** FIRST\_NAME, LAST\_NAME, JOB\_ID

**ON** EMPLOYEES

**FOR EACH ROW**

**DECLARE**

**BEGIN**

--Puntje 1

:**NEW**.JOB\_ID := *UPPER*(:**NEW**.JOB\_ID);

:**NEW**.FIRST\_NAME := *INITCAP*(:**NEW**.FIRST\_NAME);

:**NEW**.LAST\_NAME := *INITCAP*(:**NEW**.LAST\_NAME);

--Puntje 2

**IF INSERTING THEN**

**IF** :**NEW**.HIRE\_DATE < *SYSDATE* **THEN**

*RAISE\_APPLICATION\_ERROR*(-20000, 'Je kan geen datum in het verleden instellen!');

**END IF**;

**END IF**;

--Puntje 3

**IF UPDATING THEN**

**IF** *SUBSTR*(:**NEW**.JOB\_ID, 4, 3) = 'MAN' **OR** *SUBSTR*(:**NEW**.JOB\_ID, 4, 3) = 'MGR' **THEN**

**IF** :**OLD**.JOB\_ID != 'AD\_PRES' **AND** :**OLD**.JOB\_ID != 'AD\_VP' **THEN**

**UPDATE** EMPLOYEES **SET** SALARY = :**OLD**.SALARY \* 1.05 **WHERE** EMPLOYEE\_ID = :**NEW**.EMPLOYEE\_ID;

**END IF**;

**END IF**;

**END IF**;

**END**;

***Oefening 7:***

**CREATE OR REPLACE TRIGGER** aidr\_emp

**BEFORE INSERT OR DELETE**

**ON** EMPLOYEES

**FOR EACH ROW**

**DECLARE**

v\_first\_name EMPLOYEES.FIRST\_NAME%**TYPE**;

v\_last\_name EMPLOYEES.LAST\_NAME%**TYPE**;

**BEGIN**

--Puntje 1

**IF INSERTING THEN**

**IF** :**NEW**.MANAGER\_ID **IS NULL THEN**

**SELECT** MANAGER\_ID

**INTO** :**NEW**.MANAGER\_ID

**FROM** DEPARTMENTS

**WHERE** DEPARTMENT\_ID = :**NEW**.DEPARTMENT\_ID;

**SELECT** FIRST\_NAME, LAST\_NAME

**INTO** v\_first\_name, v\_last\_name

**FROM** EMPLOYEES

**WHERE** EMPLOYEE\_ID = :NEW.MANAGER\_ID;

DBMS\_OUTPUT.PUT\_LINE('de chef wordt' || v\_last\_name || ' ' || v\_first\_name);

**END IF**;

--Puntje 2

**IF** :**NEW**.HIRE\_DATE **IS NULL THEN**

:**NEW**.HIRE\_DATE := *NEXT\_DAY*(*SYSDATE*, 'MONDAY');

**END IF**;

--Puntje 3

**IF** :**NEW**.SALARY **IS NULL THEN**

:**NEW**.SALARY := 1000;

DBMS\_OUTPUT.PUT\_LINE('het salaris van employee ' || :**NEW**.LAST\_NAME || ' wordt ' || :**NEW**.SALARY);

**END IF**;

**ELSIF DELETING THEN**

--Puntje 4

DBMS\_OUTPUT.PUT\_LINE('dit is per maand een besparing van ' || :**OLD**.SALARY || ' euro');

**END IF**;

**END**;