

2.)

```
CREATE FUNCTION scalarfunction (  
    v      IN  NUMBER,  
    mina   IN  NUMBER,  
    maxa   IN  NUMBER,  
    newmina IN  NUMBER,  
    newmaxa IN  NUMBER  
) RETURN NUMBER IS  
    z NUMBER;  
BEGIN  
    z := ( v - mina ) / ( maxa - mina ) * ( newmaxa - newmina ) + newMinA;  
    RETURN z;  
END;
```

```
SET SERVEROUTPUT ON FORMAT WRAPPED;
```

```
CREATE PROCEDURE skalarp (  
    maxan IN  NUMBER,  
    minan IN  NUMBER  
) IS  
    maxa   NUMBER;  
    mina   NUMBER;  
    current1 NUMBER;  
    current2 NUMBER;  
BEGIN  
    SELECT  
        MAX(value)  
    INTO maxa  
    FROM  
        value_table;
```

```

SELECT
    MIN(value)
INTO mina
FROM
    value_table;
FOR current1 IN (
    SELECT
        value
    FROM
        value_table
) LOOP
    current2 := scalarfunction(current1.value, mina, maxa, minan, maxan);
    UPDATE value_table
    SET
        value = current2
    WHERE
        value = current1.value;
END LOOP;
END;

```

```

begin
    skalarP(10,4);
end;

```

3.)

```

create type angestellte_t as object
(
    name VARCHAR2(50),
    geburtsdatum DATE,
    berufsbezeichnung VARCHAR2(50),
    monatsgehalt NUMBER,
    geschlecht VARCHAR2(10),

```

```

        angestelltennr VARCHAR2(10)
    );

CREATE TABLE angestellte of angestellte_t
(angestelltennr not null,
primary key (angestelltennr));

insert into angestellte values ('Jens Jensen', '01.01.01', 'Mechaniker', 2000, 'männlich',
'0000000001');

insert into angestellte values ('Mustermann, Max', '02.02.02', 'Informatiker', 3500, 'männlich',
'0000000002');

insert into angestellte values ('Sonja Jensen', '01.01.84', 'Sekretärin', 1500, 'weiblich', '0000000003');


create type arbeiter_t as object
(name varchar2(50),
vorname varchar2(50),
geburtsmonat VARCHAR2(5),
stundenlohn number);

create table arbeiter of arbeiter_t;

insert into arbeiter values ('Hemmerich', 'Max', '01.80', 20);
insert into arbeiter values ('Reichelt', 'Sonja', '02.65', 12);


create table berufscode
(code varchar2(4) NUMBER generated by default on null as IDENTITY,
bezeichnung varchar2(50),
primary key (code));

create table zuordnungstab
(personalnr varchar2(10) NUMBER generated by default on null as IDENTITY,
system number,
arbeiter_angestelltennr varchar2(100),

```

```
primary key (personalnr));
```

```
create table personal  
(personalnr varchar2(10),  
name varchar2(50),  
vorname varchar2(50),  
"alter" VARCHAR2(5),  
geschlecht varchar2(1),  
berufscore varchar2(4),  
jahreseinkommen number,  
primary key (personalnr));
```

```
create or replace FUNCTION berufscore_bestimmen (  
    inputberuf    IN  VARCHAR2  
) RETURN NUMBER IS  
    outputnumber NUMBER;  
BEGIN  
    BEGIN  
        SELECT  
            code  
        INTO outputnumber  
        FROM  
            berufscore  
        WHERE  
            bezeichnung = inputberuf;  
  
    EXCEPTION  
        WHEN no_data_found THEN  
            INSERT INTO berufscore ( bezeichnung ) VALUES ( inputberuf );  
  
        SELECT
```

```

        code
    INTO outputnumber
    FROM
        berufscore
    WHERE
        bezeichnung = inputberuf;
END;

RETURN outputnumber;
END;

create or replace FUNCTION geschlecht_bestimmen (
    inputsex  IN NUMBER,
    inputname IN VARCHAR2,
) RETURN NUMBER IS
    outputsex NUMBER;
BEGIN
    IF inputsex <> 0 THEN
        BEGIN
            SELECT
                geschlechtercode
            INTO outputsex
            FROM
                geschlecht
            WHERE
                name = inputname;

        EXCEPTION

            WHEN no_data_found THEN
                INSERT INTO geschlecht (
                    name,
                    geschlechtercode

```

```

        ) VALUES (
            inputname,
            inputsex
        );
        outputsex := inputsex;
END;

IF outputsex != 0 AND outputsex != inputsex THEN
    UPDATE geschlecht
    SET
        geschlechtercode = 0
    WHERE
        name = inputname;
END IF;
/* output =input*/

outputsex := inputsex;
RETURN outputsex;
ELSE
    BEGIN
        SELECT
            geschlechtercode
        INTO outputsex
        FROM
            geschlecht
        WHERE
            name = inputname;

    EXCEPTION
        WHEN no_data_found THEN

            outputsex := 0;

```

```

        END;

    END IF;

    RETURN outputsex;

END;

CREATE OR REPLACE TRIGGER angestellten_trigger AFTER
    INSERT ON angestellte

DECLARE

    angestellter angestellte_t;

    bcode        NUMBER;

    gcode        NUMBER;

    name_idx      NUMBER;

    name1         VARCHAR2(50);

    vorname       VARCHAR2(50);

    id            NUMBER;

BEGIN

    FOR angestellter IN (

        SELECT

            name,

            ( to_char(sysdate, 'YYYY') - to_char(geburtsdatum, 'YYYY') ) AS jahre,

            berufsbezeichnung,

            ( monatsgehalt * 12 ) AS jahresgehalt,

            geschlecht,

            angestelltennr

        FROM

            angestellte

    ) LOOP

        name_idx := instr(angestellter.name, ' ', 1);

        IF name_idx <> 0 THEN

            name1 := trim(substr(angestellter.name, 1, name_idx - 1));

```

```
vorname := trim(substr(angestellter.name, name_idx + 2, length(angestellter.name)));
```

```
ELSE
```

```
name_idx := instr(angestellter.name, ' ', 1);
```

```
vorname := trim(substr(angestellter.name, 1, name_idx - 1));
```

```
name1 := trim(substr(angestellter.name, name_idx + 1, length(angestellter.name)));
```

```
END IF;
```

```
bcode := berufscore_bestimmen(angestellter.berufsbezeichnung);
```

```
IF angestellter.geschlecht = 'männlich' THEN
```

```
gcode := 2;
```

```
ELSE
```

```
IF angestellter.geschlecht = 'weiblich' THEN
```

```
gcode := 1;
```

```
ELSE
```

```
gcode := 0;
```

```
END IF;
```

```
END IF;
```

```
gcode := geschlecht_bestimmen(gcode, vorname);
```

```
BEGIN
```

```
SELECT
```

```
personalnr
```

```
INTO id
```

```
FROM
```

```
zuordnungstab
```

```
WHERE
```

```
arbeiter_angestelltennr = angestellter.angestelltennr;
```

```
UPDATE personal
```


SET

```
name = name1,  
vorname = vorname,  
"alter" = angestellter.jahre,  
geschlecht = gcode,  
berufscore = bcode,  
jahreseinkommen = angestellter.jahresgehalt
```

WHERE

```
personalnr = id;
```

EXCEPTION

```
WHEN no_data_found THEN  
    INSERT INTO zuordnungstab (  
        system,  
        arbeiter_angestelltennr  
    ) VALUES (  
        1,  
        angestellter.angestelltennr  
    );
```

SELECT

```
    personalnr
```

INTO id

FROM

```
    zuordnungstab
```

WHERE

```
    arbeiter_angestelltennr = angestellter.angestelltennr;
```

INSERT INTO personal VALUES (

```
    id,
```

```
    name1,
```

```
    vorname,
```

```
        angestellter.jahre,  
        gcode,  
        bcode,  
        angestellter.jahresgehalt  
    );
```

```
END;
```

```
DELETE FROM angestellte
```

```
WHERE
```

```
    angestelltennr = angestellter.angestelltennr;
```

```
END LOOP;
```

```
END;
```

```
CREATE OR REPLACE TRIGGER arbeiter_trigger AFTER
```

```
    INSERT ON arbeiter
```

```
DECLARE
```

```
    arbeiter_ arbeiter_t;
```

```
    bcode    NUMBER;
```

```
    gcode    NUMBER;
```

```
    id       NUMBER;
```

```
    arbeiternr VARCHAR2(100);
```

```
    jahre    NUMBER;
```

```
BEGIN
```

```
    FOR arbeiter_ IN (
```

```
        SELECT
```

```
            name,
```

```
            vorname,
```

```
            geburtsmonat,
```

```
            ( stundenlohn * 40 * 4 * 12 ) AS jahresgehalt
```

```
        FROM
```

```

        arbeiter
    ) LOOP

        bcode := berufscode_bestimmen('Arbeiter');
        gcode := geschlecht_bestimmen(0, arbeiter_.vorname);
        arbeiternr := arbeiter_.vorname || arbeiter_.geburtsmonat;
        jahre := 100 + to_char(sysdate, 'YY') - trim(substr(arbeiter_.geburtsmonat, 4, 5));

    BEGIN

        SELECT
            personalnr
        INTO id
        FROM
            zuordnungstab
        WHERE
            arbeiter_angestelltennr = arbeiternr;

    UPDATE personal
    SET
        name = arbeiter_.name,
        vorname = arbeiter_.vorname,
        "alter" = jahre,
        geschlecht = gcode,
        berufscode = bcode,
        jahreseinkommen = arbeiter_.jahresgehalt
    WHERE
        personalnr = id;

    EXCEPTION

        WHEN no_data_found THEN
            INSERT INTO zuordnungstab (
                system,
                arbeiter_angestelltennr

```

```
) VALUES (  
    2,  
    arbeiternr  
);
```

```
SELECT  
    personalnr  
INTO id  
FROM  
    zuordnungstab  
WHERE  
    arbeiternr = arbeiternr;
```

```
INSERT INTO personal VALUES (  
    id,  
    arbeiternr.name,  
    arbeiternr.vorname,  
    jahre,  
    gcode,  
    bcode,  
    arbeiternr.jahresgehalt  
);
```

```
END;
```

```
DELETE FROM arbeiternr arbeiternr;
```

```
END LOOP;  
END;
```