--DROP FUNCTION

DROP FUNCTION idClub;

DROP FUNCTION idNegara;

DROP PROCEDURE newStadium;

--DROP TRIGGER

DROP TRIGGER newClub;

DROP TRIGGER newNegara;

DROP TRIGGER delHtrans;

DROP TRIGGER delDtrans;

--TUGAS

--No. 1

CREATE OR REPLACE FUNCTION idClub (

club VARCHAR2

) RETURN VARCHAR2

IS

ada NUMBER;

cari NUMBER;

spasi NUMBER;

kode NUMBER;

gen VARCHAR2(6);

BEGIN

kode:= 1;

cari:= 1;

spasi:= 0;

WHILE INSTR(club, ' ', 1, cari) != 0

LOOP

spasi:= INSTR(club, ' ', 1, cari);

cari:= cari + 1;

END LOOP;

IF spasi = 0 THEN

gen:= UPPER(SUBSTR(club, 1, 2));

ELSE

gen:= UPPER(SUBSTR(club, 1, 1) || SUBSTR(club, spasi + 1, 1));

END IF;

SELECT COUNT(id\_club) INTO ada

FROM club

WHERE SUBSTR(id\_club, 1, 2) = gen;

IF ada > 0 THEN

SELECT MAX(SUBSTR(id\_club, 3)) + 1 INTO kode

FROM club

WHERE SUBSTR(id\_club, 1, 2) = gen;

END IF;

IF kode > 0 AND kode < 10 THEN

gen:= gen || '00' || kode;

ELSIF kode >= 10 AND kode < 100 THEN

gen:= gen || '0' || kode;

ELSIF kode >= 100 AND kode < 1000 THEN

gen:= gen || kode;

END IF;

RETURN gen;

END;

/

SHOW ERR;

CREATE OR REPLACE PROCEDURE newStadium (

club VARCHAR2

)

IS

tahun DATE;

d2 NUMBER;

d3 NUMBER;

kode NUMBER;

id VARCHAR2(6);

nama VARCHAR2(100);

nama\_std VARCHAR2(100);

kapasitas NUMBER;

BEGIN

SELECT MAX(SUBSTR(id\_stadium, 4)) + 1 INTO kode

FROM stadium;

IF kode > 0 AND kode < 10 THEN

id:= 'STD' || '00' || kode;

ELSIF kode >= 10 AND kode < 100 THEN

id:= 'STD' || '0' || kode;

ELSIF kode >= 100 AND kode < 1000 THEN

id:= 'STD' || kode;

END IF;

SELECT nama\_club, tahun\_berdiri INTO nama, tahun

FROM club

WHERE id\_club = club;

IF LENGTH(nama || ' Stadium') > 20 THEN

nama\_std:= SUBSTR(nama, 1, 12) || ' Stadium';

END IF;

d2:= TO\_NUMBER(SUBSTR(TO\_CHAR(tahun, 'YYYY'), 2, 1));

d3:= TO\_NUMBER(SUBSTR(TO\_CHAR(tahun, 'YYYY'), 3, 1));

kapasitas:= POWER(d2, 3) + (d3\*100) + (LENGTH(nama)+50);

INSERT INTO stadium VALUES (id, nama\_std, kapasitas, club, 'ENG');

END;

/

SHOW ERR;

CREATE OR REPLACE TRIGGER newClub

AFTER INSERT ON club

DECLARE

idx NUMBER;

last\_inserted NUMBER;

last\_id VARCHAR2(6);

new\_id VARCHAR2(6);

new\_nama VARCHAR2(100);

new\_tahun DATE;

new\_liga VARCHAR2(6);

cek\_kode NUMBER;

cek\_id VARCHAR2(6);

BEGIN

idx:= 0;

FOR i IN (SELECT \*

FROM club)

LOOP

idx:= idx + 1;

END LOOP;

last\_inserted:= 0;

FOR i IN (SELECT \*

FROM club)

LOOP

last\_inserted:= last\_inserted + 1;

IF last\_inserted = idx THEN

last\_id:= i.id\_club;

new\_id:= i.id\_club;

new\_nama:= i.nama\_club;

new\_tahun:= i.tahun\_berdiri;

new\_liga:= i.id\_liga;

END IF;

END LOOP;

new\_id:= idClub(new\_nama);

cek\_kode:= TO\_NUMBER(SUBSTR(new\_id, 3)) - 1;

IF cek\_kode > 0 AND cek\_kode < 10 THEN

cek\_id:= SUBSTR(new\_id, 1, 2) || '00' || cek\_kode;

ELSIF cek\_kode >= 10 AND cek\_kode < 100 THEN

cek\_id:= SUBSTR(new\_id, 1, 2) || '0' || cek\_kode;

ELSIF cek\_kode >= 100 AND cek\_kode < 1000 THEN

cek\_id:= SUBSTR(new\_id, 1, 2) || cek\_kode;

END IF;

IF UPPER(last\_id) = UPPER(cek\_id) THEN

new\_id:= UPPER(last\_id);

END IF;

new\_nama:= INITCAP(new\_nama);

new\_tahun:= TO\_DATE(TO\_CHAR(new\_tahun, 'YYYY'), 'YYYY');

IF TO\_CHAR(new\_tahun, 'YYYY') < 2000 THEN

new\_liga:= 'EN001';

ELSE

new\_liga:= 'EN004';

END IF;

UPDATE club

SET id\_club = new\_id, nama\_club = new\_nama,

tahun\_berdiri = new\_tahun, id\_liga = new\_liga

WHERE id\_club = last\_id;

newStadium(new\_id);

END;

/

SHOW ERR;

--No. 2

CREATE OR REPLACE FUNCTION idNegara (

negara VARCHAR2

) RETURN VARCHAR2

IS

ada NUMBER;

gen VARCHAR2(3);

BEGIN

gen:= UPPER(SUBSTR(negara, 1, 3));

SELECT COUNT(id\_negara) INTO ada

FROM negara

WHERE SUBSTR(id\_negara, 1, 3) = gen;

IF ada > 0 THEN

gen:= UPPER(SUBSTR(gen, 1, 2) || SUBSTR(negara, -2, 1));

END IF;

RETURN gen;

END;

/

SHOW ERR;

CREATE OR REPLACE TRIGGER newNegara

BEFORE INSERT ON negara

FOR EACH ROW

DECLARE

nama\_kembar EXCEPTION;

ada NUMBER;

BEGIN

:NEW.id\_negara:= idNegara(:NEW.nama\_negara);

:NEW.nama\_negara:= UPPER(:NEW.nama\_negara);

SELECT COUNT(nama\_negara) INTO ada

FROM negara

WHERE nama\_negara = :NEW.nama\_negara;

IF ada > 0 THEN

RAISE nama\_kembar;

END IF;

EXCEPTION

WHEN nama\_kembar THEN

RAISE\_APPLICATION\_ERROR(-20001, 'NAMA NEGARA TIDAK BOLEH KEMBAR');

END;

/

SHOW ERR;

--No. 3

CREATE OR REPLACE TRIGGER delHtrans

BEFORE DELETE ON h\_transaksi

FOR EACH ROW

DECLARE

BEGIN

DELETE FROM d\_transaksi

WHERE id\_transaksi= :OLD.id\_transaksi;

END;

/

SHOW ERR;

--No. 4

CREATE OR REPLACE TRIGGER delDtrans

BEFORE DELETE ON d\_transaksi

FOR EACH ROW

DECLARE

ht\_harga NUMBER;

dt\_harga NUMBER;

harga\_baru NUMBER;

BEGIN

SELECT biaya\_transfer INTO ht\_harga

FROM h\_transaksi

WHERE id\_transaksi = :OLD.id\_transaksi;

dt\_harga:= :OLD.subtotal;

harga\_baru:= ht\_harga - dt\_harga;

UPDATE h\_transaksi

SET biaya\_transfer = harga\_baru

WHERE id\_transaksi = :OLD.id\_transaksi;

END;

/

SHOW ERR;