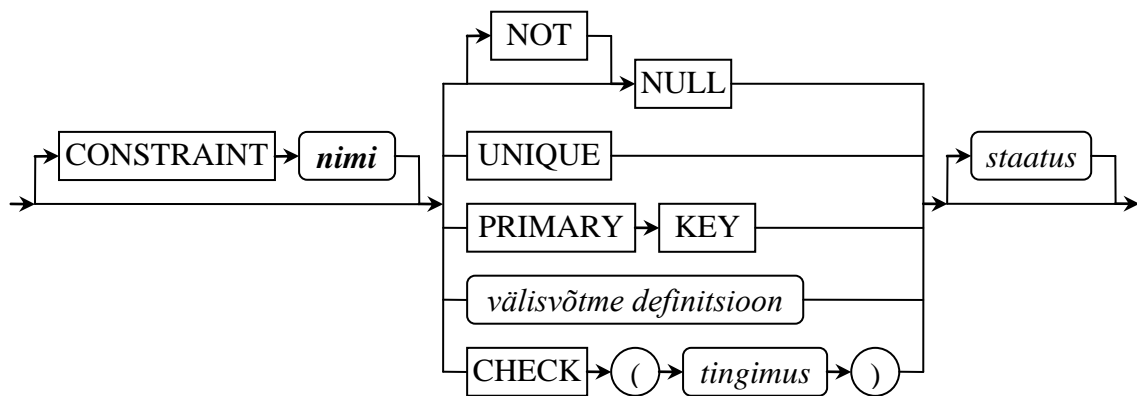




Avaldis võib sisaldada pseudoveerge: SYSDATE, USER, CURRENT\_TIMESTAMP ...

Kitsenduse lausend:

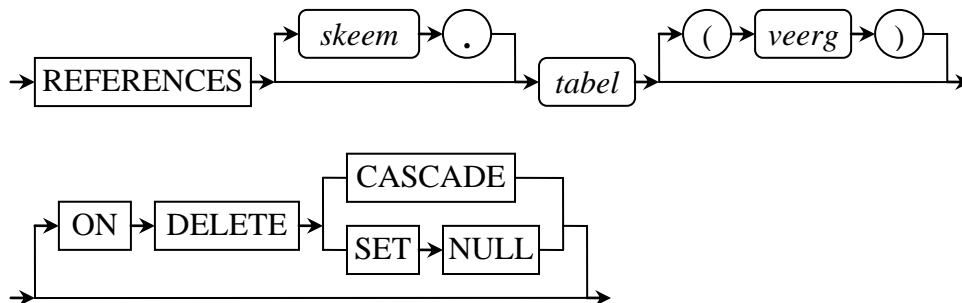


Staatus võimalikud väärtused on ENABLED ja DISABLED;

Kitsenduse CHECK puhul kehtivad järgmised reeglid:

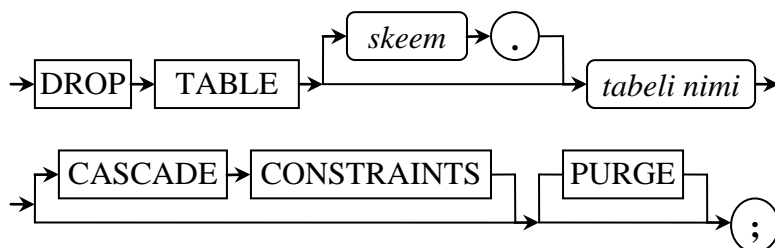
- Kitsendust CHECK ei saa rakendada vaatele;
- Kui kitsendus CHECK on defineeritud tabeli peale, siis võib opereerida ainult antud tabeli veergudega;
- Kitsendus CHECK ei või sisalduda alapäringu definitsioonis.

Välisvõtme definitsioon:



Võtmesõnade *ON DELETE* ... kasutamise korral, toimub enne ülemise tabeli rea kustutamist alumise tabeli vastavate kirjete kustutamine või nullimine.

**Tabeli kustutamine käsu süntaks:**



Kui tabelil on välisvõtmed, siis tuleb kustutamisel kasutada CASCADE CONSTRAINTS lausendit, vastasel korral antakse ORA-02449 veateade. Võttesõna PURGE kasutamise korral kustutatakse tabel täielikult andmebaasist (ei panda prügikasti e. RECYCLEBIN'i).

## Järjendid

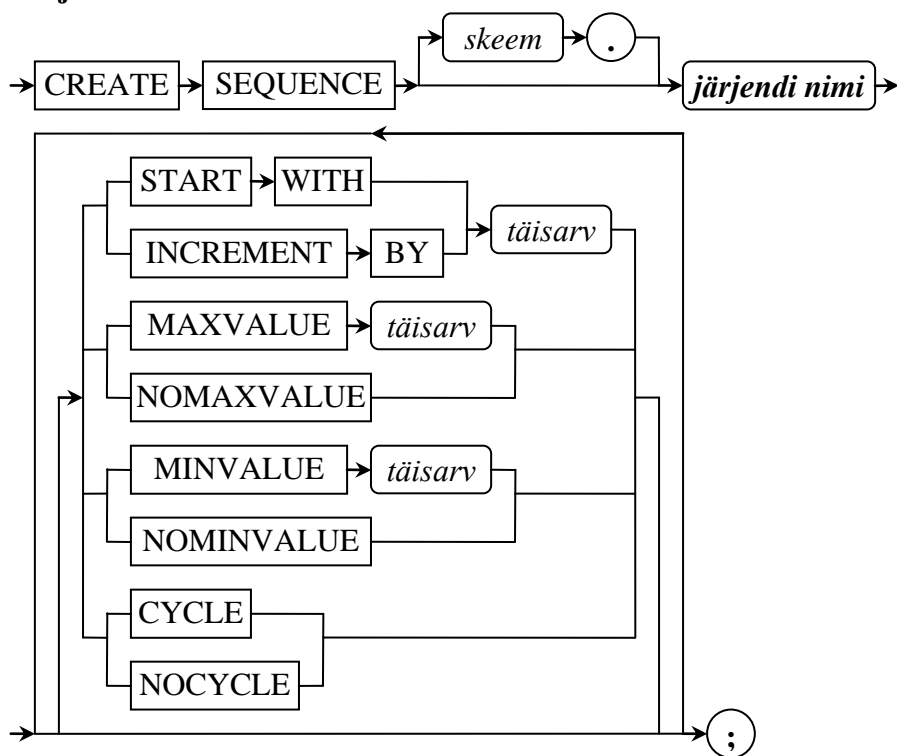
Järjend on andmebaasi objekt, kust kasutajad saavad genereerida unikaalseid täisarve. Järjendi loomiseks on kasutajal vaja CREATE SEQUENCE õigust. Igal järjendil on kaks pseudoveergu:

- CURRVAL – tagastab järjendi jooksva väärtuse;
- NEXTVAL – suurendab järjendi numbrit ja tagastab uue väärtuse.

Andmebaasi vaade, mis on seotud järjenditega:

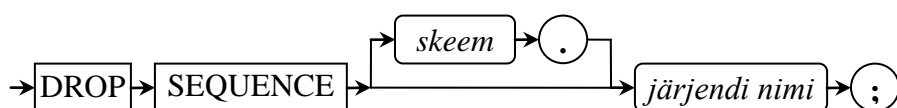
- *dba\_sequences* – kirjeldab kõiki järjendeid andmebaasis.

### Järjendi loomise käsu süntaks:



**CYCLE/NOCYCLE** – määrab, kas järjend hakkab või ei hakka korduma jõudes maksimaalse väärtuseni.

### Järjendi kustutamise käsu süntaks:



## Triger "AUTOINCREMENT"

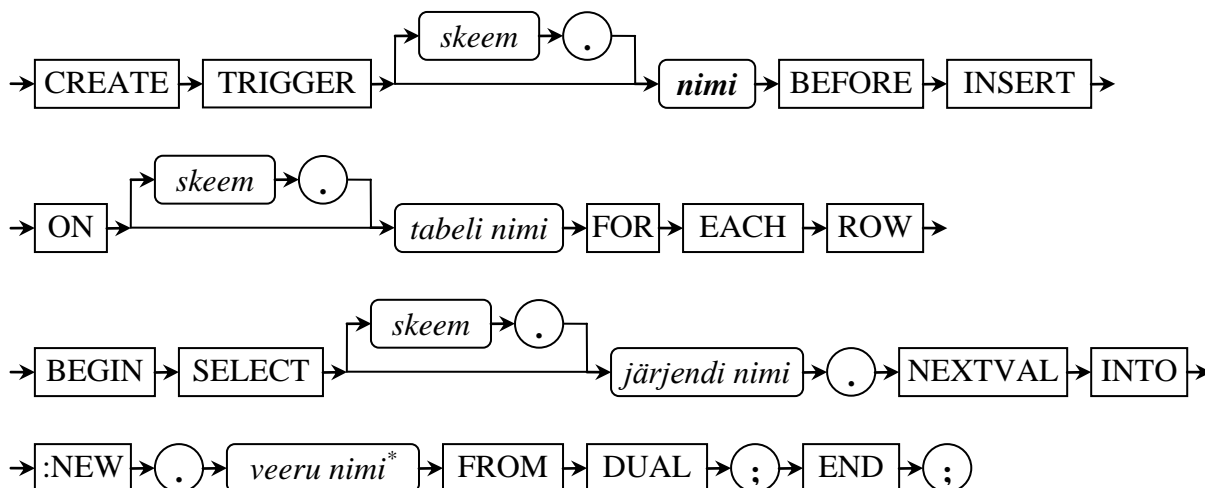
Trigerid on salvestatud PL/SQL plokid, mis on seotud tabeliga, skeemiga või andmebaasiga ja käivituvad vastava sündmuse toimumisel. Trigeri loomiseks on kasutajal vaja CREATE TRIGGER õigust. Kui triger on loodud, siis andmebaas kompileerib selle automaatselt ning lülitab ta sisse. Trigeri käsitsi sisse- ja väljalülitamiseks tuleb kasutada ALTER TRIGGER lausendi avaldises võtmesõnu ENABLE või DISABLE.

„AUTOINCREMENT” triger – seos „triger + järjend”, mida kasutatakse primaarvõtmete väärtuste genereerimiseks.

Andmebaasi vaade, mis on seotud trigeritega:

- *dba\_triggers* – kirjeldab kõiki trigereid andmebaasis.

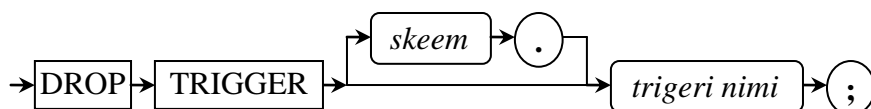
### „AUTOINCREMENT” trigeri loomise käsu süntaks:



\*Veeru nimi – veerg millele rakendatakse automaatselt unikaalsete väärtuste genereerimist (tavaliselt primaarvõtme kitsendusega).

**NB!** Antud autoincrement triger ei sisalda veatötlust ning ei pruugi sobida iga tabeli jaoks.

### „AUTOINCREMENT” trigeri kustutamise käsu süntaks:



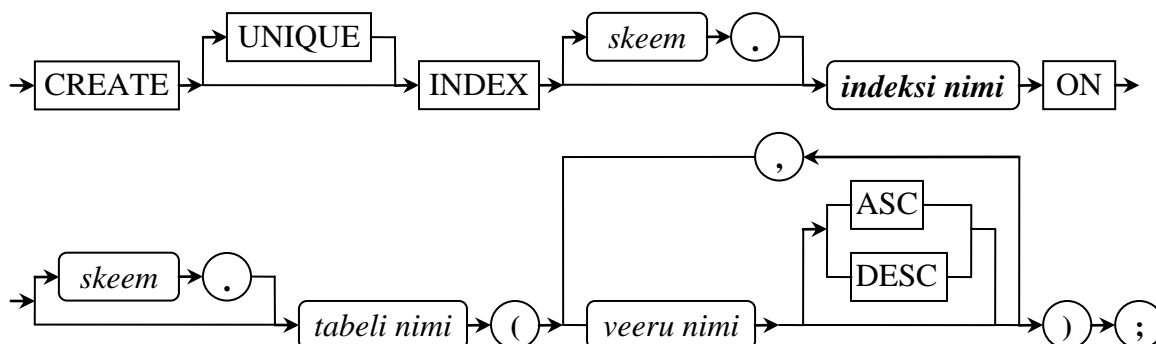
## Indeksid

Indeksid on skeemi objektid, mis sisaldavad kirjet iga indekseeritud veeru väärtuse kohta. Indeksid võimaldavad otsest ja kiiret ligipääsu ridadele. Indeksi loomiseks on kasutajal vaja CREATE INDEX õigust.

Andmebaasi vaade, mis on seotud indeksitega:

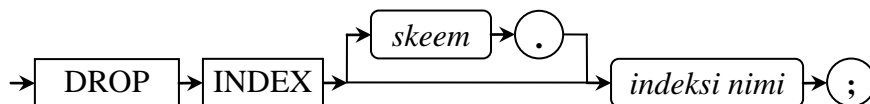
- *dba\_indexes* – kirjeldab kõiki andmebaasi indeksid.

### Indeksi loomise käsu süntaks:



ASC/DESC – kasutatakse sorteerimisindeksite puhul.

### Indeksi kustutamise käsu süntaks:



## Prügikast

Igal kasutajal on olemas isiklik prügikast – koht kuhu pannakse kustutatud tabelid.

Andmebaasi vaade, mis on seotud prügikastiga:

- *dba\_recyclebin* – väljastab informatsiooni andmebaasis olevate prügikastide kohta.

Samuti võib prügikasti sisu vaatamiseks kasutada SQL\*Plus käsku SHOW RECYCLEBIN.

## Päringud, mis on seotud skeemi objektide haldamisega

Tabeli loomise päringu näidis:

```
SQL> CREATE TABLE näidis.näidis_tabel(  
 1 veerg_1 NUMBER NOT NULL PRIMARY KEY,  
 2 veerg_2 VARCHAR2(32) NOT NULL,  
 3 veerg_3 DATE DEFAULT SYSDATE,  
 4 veerg_4 NUMBER CHECK(veerg_4 IN (1,2,3))  
 5 ) TABLESPACE näidis_tabeliruum;
```

Tabeli kustutamise päringu näidis:

```
SQL> DROP TABLE näidis.näidis_tabel;
```

Päring kustutatud objekti(tabeli) taastamiseks:

```
SQL> FLASHBACK TABLE näidis_tabel  
 1 TO BEFORE DROP;
```

Järjendi loomise näidis:

```
SQL> CREATE SEQUENCE näidis.näidis_järjend  
 1 START WITH 1  
 2 INCREMENT BY 1  
 3 NOMAXVALUE  
 4 MINVALUE 1  
 5 NOCYCLE;
```

AUTOINCREMENT trigeri loomise näidis:

```
SQL> CREATE TRIGGER ti_näidis_trigger  
 1 BEFORE INSERT  
 2 ON näidis.näidis_tabel  
 3 FOR EACH ROW  
 4 BEGIN  
 5 IF :new.veerg_1 is null then  
 6 SELECT näidis.näidis_järjend.NEXTVAL into :new.veerg_1 FROM DUAL;  
 7 END IF;  
 8 END;  
 9 /
```

Päring prügikasti tühjendamiseks:

```
SQL> PURGE RECYCLEBIN;
```

Sorteerimisindeksi loomise näidis:

```
SQL> CREATE INDEX näidis.näidis_indeks ON  
 1 näidis.näidis_tabel(veerg_2 ASC, veerg_3 DESC);
```

## Ülesanded

1. Logige sisse kasutajana *sybase*.
2. Looge järgmised tabelid:

Tabeli nimi	Veeru nimi	Andmetüüp	Kitsendused	Autoincrement
Klubi	Id	Number	Primaarvõti, NOT NULL	Jah
	Nimi	Varchar2(100)	NOT NULL	Ei
	Aadress	Varchar2(70)	NOT NULL, Vaikeväärtus <i>Tartu</i>	Ei
Isik	Id	Number	Primaarvõti, NOT NULL	Jah
	Eesnimi	Varchar2(50)	NOT NULL	Ei
	Perenimi	Varchar2(50)	NOT NULL	Ei
	Isikukood	Number		Ei
	Klubi	Number	Välisvõti tabelisse Klubi veerule Id	Ei
Turniir	Id	Number	Primaarvõti, NOT NULL	Jah
	Nimetus	Varchar2(100)	NOT NULL	Ei
	Toimumiskoht	Varchar2(100)		Ei
	Alguskuupaev	Date	NOT NULL	Ei
	Loppkuupaev	Date		Ei
Partii	Id	Number	Primaarvõti, NOT NULL	Jah
	Turniir	Number	NOT NULL, Välisvõti tabelisse Turniir veerule Id	Ei
	Algushetk	Timestamp	NOT NULL	Ei
	Lopphetk	Timestamp		Ei
	Valge	Number	NOT NULL, Välisvõti tabelisse Isik veerule Id	Ei
	Must	Number	NOT NULL, Välisvõti tabelisse Isik veerule Id	Ei
	Valge_tulemus	Number	Lubatud väärtused: 0,1,2	Ei
	Musta_tulemus	Number	Lubatud väärtused: 0,1,2	Ei
	Kokkuvote	Varchar2(400)		Ei

3. Looge järgmised sortimisindeksid:
  - a. Turniiri algusaegade peale (nimeks *ix\_turniirialgus*);
  - b. Partiide kahanevalt valge ja musta tulemuse peale (nimeks *ix\_partii\_tulemused\_desc*);