Secvențe, indecși, sinonime

- Ce este o secvență
- Cum se creează și cum se șterge o secvență
- Cum se modifică o secvență
- Ce sunt indecșii și care sunt avantajele folosirii lor
- Când este indicată folosirea indecșilor
- Cum se creează și cum se șterge un index
- Ce sunt sinonimele
- Cum se creează și cum se șterge un sinonim
- Cum se folosesc sinonimele

Secvenţe

- Imaginaţi-vă că trebuie să adăugaţi în baza de date a şcolii, datele persoanele ale noilor elevi veniţi în şcoala voastră în clasa a IX-a. Fiecărui elev trebuie să-i asociaţi un **id** unic în întreaga bază de date. Nu ştiţi însă exact care sunt id-urile elevilor deja existenţi în baza de date, pentru a şti care sunt id-urile ”libere”. Cum rezolvaţi oare această problemă?
- O variantă ar fi ca la inserarea unui nou elev să determinaţi cel mai mare id existent în baza de date, şi să-i asociaţi elevului nou inserat un id cu o unitate mai mare decât cel mai mare id. Veţi scrie o comandă de forma:

```
INSERT INTO elevi (id, nume, prenume, …)
VALUES ( SELECT max(id)+1 FROM elevi,
        ’Ionescu’, ’Ioan’, …)
```

- O astfel de soluţie poate genera probleme în cazul accesului concurent la baza de date, când este posibil ca doi utilizatori diferiţi să încerce să insereze doi elevi cu acelaşi **id**.
- Soluţia este folosirea secvenţelor. Secvenţele sunt obiecte ale bazei de date cate generează automat, în mod secvenţial, liste de numere. Acestea sunt utile când o tabelă foloseşte o cheie primară artificială, ale cărei valori dorim să le generăm automat.

Crearea şi ştergerea secvenţelor

- Sintaxa pentru crearea unei secvenţe este următoarea:

CREATE SEQUENCE *nume_secventa*

START WITH *n1*

INCREMENT BY *n2*

MAXVALUE *n3* | NOMAXVALUE

MINVALUE *n4* | NOMINVALUE

CACHE *n5* | NOCHACE

CYCLE | NOCYCLE

- Să explicăm pe rând care este rolul fiecărei opţiuni din această comandă:
  - **START WITH *n1*** – precizează de la ce valoare va începe generarea valorilor. Această opţiune este utilă atunci când câmpul pentru care dorim să generăm valori folosind această secvenţă conţine deja valori. În acest caz, vom preciza în **n1** o valoare mai mare decât toate valorile deja existente în coloana respectivă. Dacă această opţiune nu este prezentă, se va începe implicit de la valoarea **1**.
  - **INCREMENT BY n2** – precizează intervalul dintre două numere din secvenţă. Poate fi un număr întreg pozitiv sau negativ, dar nu poate fi zero. Dacă se precizează o valoare negativă, atunci valorile se vor genera în ordine descrescătoare, altfel se vor genera în ordine crescătoare. Dacă omiteţi această opţiune valoarea implicită a incrementului va fi **1**.
  - **MAXVALUE *n3*** şi respectiv **MINVALUE *n4*** – aceste clause specifică cea mai mare, respectiv cea mai mică valoare returnată de către secvenţă. **n3** şi respectiv **n4** trebuie să fie numere întrege cu maxim **9** cifre.
  - **NOMAXVALUE** – valoarea maximă generată va fi **2147483647** pentru o secvenţă cu increment pozitiv, respectiv **-1** pentru o secvenţă cu increment negativ.
  - **NOMINVALUE** – valoarea maximă generată va fi **1** pentru o secvenţă cu increment pozitiv, respectiv **-2147483647** pentru o secvenţă cu increment negativ.
  - **CACHE *n5*** – această opţiune este folosită din considerente de eficienţă. Cu această opţiune se vor genera simultan **n5** valori din secvenţă, şi numai atunci când acestea se vor epuiza se vor genera următoarele **n5** valori. În acest fel se vor face mai puţine modificări asupra bazei de date.
  - **CYCLE | NOCYCLE** – dacă specificaţi opţiunea **CYCLE** atunci când secvenţa a ajuns la valoarea maximă (respectiv minimă pentru o secvenţă cu increment negativ), secvenţa va reîncepe să genereze valori începând cu **MINVALUE** (respectiv **MAXVALUE** pentru o secvenţă cu increment negativ). Evident, dacă utilizaţi opţiunea **CYCLE** nu există nici o garanţie privind unicitatea valorilor generate.

- De exemplu, comanda:

CREATE SEQUENCE sec1

START WITH 1 INCREMENT BY 1

- creează o secvenţă care va genera valori din **1** în **1**, începând cu **1**, adică va genera în ordine valorile **1**, **2**, **3**, etc.
- Comanda

CREATE SEQUENCE sec2

START WITH 120 INCREMENT BY -3

- creează o secvenţă care va genera valori descrescătoare din **3** în **3**, începând cu **120**, adică va genera în ordine valorile **120**, **117**, **114**, etc.
- Ştergerea unei secvenţe se face simplu cu comanda **DROP SEQUENCE**.

UTILIZAREA SECVENŢELOR

- Să vedem acum cum generăm efectiv valorile din secvenţă. Vom folosi două pseudocoloane speciale numite **NEXTVAL** şi respectiv **CURRVAL**. **NEXTVAL** generează următoarea valoare din secvenţă, în timp ce **CURVAL** este folosită pentru a afla care a fost valoarea care tocmai a fost generată.
- Pentru exemplificare, creăm secvenţa

CREATE SEQUENCE sec3

START WITH 5 INCREMENT BY 3

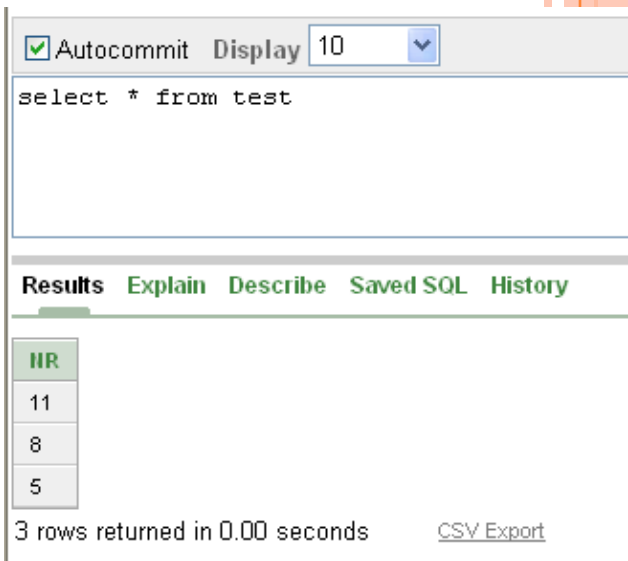
- şi tabela

CREATE TABLE test(nr number(3))

- şi rulăm de **3** ori comanda:

INSERT INTO test values(sec3.NEXTVAL)

- În acest fel conţinutul tabelei este **5**, **8**, **11**



- Dacă rulăm acum comanda

SELECT sec3.currval FROM dual

- se va afişa valaoarea **11**, adică exact ultima valoare generată de către secvenţă.
- **Atenţie!** Pseudocoloanele **NEXTVAL** şi **CURRVAL** nu pot fi folosite în următoarele contexte:
  - în clauza **SELECT** a unei vederi
  - într-o comandă **SELECT** care foloseşte opţiunea **DISTINCT**
  - într-o comandă **SELECT** care foloseşte clauzele **GROUP BY**, **HAVING**, sau **ORDER BY**.
  - într-o subinterogare din cadrul unei comenzi **SELECT**, **DELETE** sau **UPDATE**.
  - Într-o opţiune **DEFAULT** a comenzii **CREATE TABLE** sau **ALTER TABLE**.

Modificarea secvenţelor

- Comanda **ALTER SEQUENCE** care permite modificarea unei secvenţe are sintaxa similară cu cea a comenzii **CREATE SEQUENCE**:
- **CREATE SEQUENCE *nume_secventa***
- **INCREMENT BY *n2***
- **MAXVALUE *n3* | NOMAXVALUE**
- **MINVALUE *n4* | NOMINVALUE**
- **CACHE *n5* | NOCHACE**
- **CYCLE | NOCYCLE**
- Modificarea unei secvenţe va afecta doar valorile ce se vor genera ulterior. La modificarea unei secvenţe trebuie să se ţină cont de câteva restricţii. De exemplu nu se poate stabili o valoare în clauza **MAXVALUE** care să fie mai mică decât ultima valoare care a fost deja generată de către secvenţă.

- Să experimentăm puţin opţiunea de modificare a unei secvenţe. Să rulăm, pe rând, următoarele comenzi:

```
CREATE SEQUENCE sec4;
CREATE TABLE test1 (n NUMBER(2), v NUMBER(2));
INSERT INTO test1 values(1, sec4.NEXTVAL);
INSERT INTO test1 values(2, sec4.NEXTVAL);
INSERT INTO test1 values(3, sec4.NEXTVAL);
INSERT INTO test1 values(4, sec4.CURRVAL);
ALTER SEQUENCE sec4 INCREMENT BY -5
   MINVALUE -200;
INSERT INTO test1 values(5, sec4.NEXTVAL);
INSERT INTO test1 values(6, sec4.NEXTVAL);
INSERT INTO test1 values(7, sec4.NEXTVAL);
```

- După aceste comenzi, conţinutul tabelei **test** va fi cel din tabelul alăturat:

N	V
1	1
2	2
3	3
4	3
5	-2
6	-7
7	-12

INDECŞI

- Să presupunem că am creat o tabelă cu comanda:

CREATE TABLE test (id integer, content varchar)

- şi am inserat o mulţime de linii în această tabelă. La un moment dat avem nevoie să rulăm o interogare de forma:

SELECT content FROM test WHERE id = 5;

- Serverul bazei de date va trebui să parcurgă întreaga tabelă test, linie de linie, pentru a căuta toate liniile pentru care **id**-ul este 5. Dacă tabela conţine foarte multe linii şi doar puţine linii (poate chiar nici una) for fi returnate de către interogarea anterioară, această metodă este clar ineficientă.
- Pentru a un acces direct şi rapid la liniile unei tabele, se vor folosi indecşii.
- Indecşii unei tabele funcţionează similar cu indexul unei cărţi de specialitate. Într-un astfel de index, aflat de obicei la sfârşitul unei cărţi se găsesc principalii termeni şi concepte întâlnite în cartea respectivă, sortaţi alfabetic, indicându-se în dreptul fiecărui termen pagina sau paginile la care poate fi întâlnit termenul respectiv în carte. O persoană interesată de un anumit termen, nu va citi întreaga carte, ci va căuta în index pagina sau paginile corespunzătoare.

- Există două tipuri de indecşi:
  - ***indecşi unici*** – sunt generaţi automat pentru coloanele ce fac parte din cheia primară sau asupra cărora s-a definit o constrângere **UNIQUE**.
  - ***indecşi non-unici*** – care sunt definiţi de către utilizator.
- Crearea unui index se realizează cu comanda:

  CREATE INDEX *nume_index*

  ON *nume_tabela*(*coloana1*, *coloana2*, ... , *coloanan*)

- De exemplu, dacă dorim să creştem viteza operaţiilor de căutare după coloana **nume** din tabela **elevi** vom crea următorul index:

  CREATE INDEX elevi_idx1

  ON carti(nume)
- Într-un index putem include mai multe coloane ale unei tabele, ca în următorul exemplu:

  CREATE INDEX elevi_idx2

  ON carti(nume, prenume)
- De asemenea pot fi incluse în index expresii, nu doar coloane ale unei tabele:

  CREATE INDEX elevi_idx3

  ON carti(UPPER(nume), UPPER(prenume))

- Pentru a şterge un index folosiţi comanda **DROP INDEX**. Indecşii pot fi adăugaţi şi şterşi în orice moment fără a afecta tabela pe care o indexează în nici un fel, ei fiind fizic şi logic independenţi de tabela pe care o indexează. Totuşi, atunci când veţi şterge o tabelă, se vor şterge automat toţi indecşii definiţi pe tabela respectivă.
- Odată creat un index, nu mai este necesară nici o intervenţie, acesta fiind actualizat automat după fiecare modificare efectuată asupra tabelei. De asemenea indexul va fi folosit automat în interogări care pot câştiga de pe urma folosirii sale.
- Un index definit pe o coloană care face parte dintr-o condiţie de join, poate duce la creşterea semnificativă a vitezei de executare a join-ului respectiv.

- Aşadar, este indicată crearea unui index atunci când:
- coloana care se indexează conţine o plajă mare de valori
- coloana care se indexează conţine multe valori nule (valorile nule nu sunt incluse în index)
- una sau mai multe coloane sunt frecvent folosite împreună în clauza **WHERE** sau în condiţiile de join.
- Tabela este mare şi majoritatea interogărilor returnează un număr mic de linii din această tabelă (**~5%** din numărul total de înregistrări)

- Când NU este indicat să creaţi un index? Atunci când:
- tabela este mică, în acest caz căutarea secvenţială este acceptabilă
- Coloanele nu sunt foarte des folosite în clauza **WHERE** a interogărilor
- majoritatea interogărilor returnează un număr mare de înregistrări (mai mult de **5%** din numărul total de înregistrări)
- se efectuează multe operaţii de inserare, ştergere sau modificare asupra tabelei. După fiecare astfel de operaţie sistemul trebuie să actualizeze indexul, operaţie consumatoare de timp
- Coloanele indexate sunt referite cel mai ades ca parte a unor expresii.

Sinonime

- După cum ştiţi sinonimul este un cuvânt cu exact acelaşi înţeles cu un alt cuvânt, adică un cuvânt care poate fi folosit în locul altui cuvânt
- Similar în dialectul bazelor de date, administratorul unei baze de date poate defini nume echivalente pentru un obiect al bazei de date.
- În principal vom defini un sinonim pentru un obiect al bazei de date pentru a simplifica referirea la acel obiect.
- De exemplu pentru a interoga **tabela1** din schema unui alt utilizator, fie acesta **user1**, atunci vom referi această tabelă prin prefixarea numelui tabelei cu numele utilizatorului în a cărui schemă se găseşte tabela, adică vom scrie **user1.tabela1**. Dacă numele utilizatorului este însă **RO_L2_SQL01_S12** iar tabela se numeşte **d_track_listings**, va trebui să scriem **RO_L2_SQL01_S12.d_track_listings** pentru a ne referi la acea tabelă, ceea ce este destul de neplăcut. Pentru aceasta vom defini un sinonim mai scurt pentru tabela respectivă

- Sintaxa comenzii de creare a unui sinonim este:
  CREATE [PUBLIC] SYNONYM *nume_sinonim*
  FOR *obiect*
- De exemplu:
  CREATE SYNONYM ana_track
  FOR RO_L2_SQL01_S12.d_track_listings
- În continuare, vom putea folosi acest sinonim în locul numelui complet al tabelei.
- Se pot defini sinonime pentru tabele, vederi, secvenţe, proceduri sau alte obiecte ale bazei de date.
- Opţiunea **PUBLIC** este folosită de către administratorul bazei de date pentru a crea un sinonim accesibil tuturor utilizatorilor bazei de date. În mod implicit un sinonim este privat.
- Ştergerea unui sinonim se face cu comanda **DROP SYNONYM**.

- În această lecție am învățat:
- Ce este o secvență
- Cum se creează și cum se șterge o secvență
- Cum se modifică o secvență
- Ce sunt indecșii și care sunt avantajele folosirii lor
- Când este indicată folosirea indecșilor
- Cum se creează și cum se șterge un index
- Ce sunt sinonimele
- Cum se creează și cum se șterge un sinonim
- Cum se folosesc sinonimele