

Crearea și gestionarea tabelelor

1. Obiective

Scopul acestui referat este de a prezenta instrucțiunile LDD ce pot fi utilizate pentru modificarea schemei logice a unei baze de date (crearea de tabele, modificarea structurii unui tabel, redenumirea unui tabel). De asemenea, este indicat modul în care pot fi utilizate *constrângerile de integritate* pentru a preveni introducerea unor date invalide în baza de date.

După parcurgerea acestui referat, studentul va deține cunoștințele necesare:

- descrierii obiectelor din baza de date;
- pentru crearea unui tabel;
- descrierii tipurilor de date ce pot fi utilizate în momentul specificării definițiilor de coloane;
- modificării structurii unui tabel;
- ștergerii, redenumirii și trunchierii tabelelor.
- definirii constrângerilor de integritate;

O bază de date Oracle poate conține structuri de date multiple. Fiecare structură trebuie definită în faza de proiectare a bazei de date, astfel încât să poată fi creată în momentul construirii bazei de date. Tipurile de obiecte ale unei baze de date sunt date în tabelul următor.

Obiect	Descriere
Tabel	Unitatea de bază pentru stocarea datelor; compusă din linii și coloane.
Vedere	Reprezentare logică a unei submulțimi de date dintr-unul sau mai multe tabele.
Secvență	Generează valori pentru chei primare.
Index	Îmbunătățește performanțele în cazul unor interogări.
Sinonim	Furnizează alte nume obiectelor.

2. Instrucțiunea CREATE TABLE

Această instrucțiune face parte din Limbajul de Definiere a Datelor (LDD) și permite crearea unui tabel în care vor fi stocate date. În general, instrucțiunile LDD reprezintă o submulțime a instrucțiunilor SQL și sunt utilizate pentru crearea, modificarea și ștergerea obiectelor unei baze de date Oracle. Pentru ca un utilizator să poată crea un tabel, el trebuie să dețină dreptul CREATE TABLE și o zonă de stocare în care va crea obiecte de tip tabel. Dreptul de a crea un tabel este acordat de administratorul bazei de date prin intermediul instrucțiunilor LCD (Limbajul de Control al Datelor) GRANT și REVOKE.

Sintaxa instrucțiunii CREATE TABLE este următoarea:

```
CREATE TABLE [schema.]tabel  
(col tip_date [DEFAULT expr]  
[, col tip_date [DEFAULT expr] ...]);
```

unde:

schema coincide cu numele utilizatorului proprietar al tabelului;
tabel numele tabelului;
DEFAULT *expr* specifică o valoare implicită;
Col numele coloanei;
tip_date tipul de date și lungimea coloanei col.

Convenții pentru denumirea tabelor:

- numele trebuie sa înceapă cu o literă;
- numele unui tabel poate avea lungimea de 1-30 caractere;
- caracterele permise sunt numai A-Z, a-z, 0-9, _, \$ și #;
- numele unui tabel nu trebuie să coincidă cu numele altor obiecte din baza de date (obiecte ale aceluiași utilizator);
- nu trebuie sa fie un nume rezervat Oracle.

2.1 Opțiunea **DEFAULT**

Unei coloane i se poate asigna o valoare implicită utilizând opțiunea DEFAULT. Această opțiune previne introducerea unor valori NULL în tabel în cazul inserării unei linii care nu specifică o valoare pentru coloana în cauză. Valoarea implicită poate fi o valoare literală, o expresie sau o funcție SQL, cum ar fi SYSDATE sau USER, dar nu poate fi cea a unei alte coloane sau o pseudocoloană (cum ar fi NEXTVAL sau CURRVAL). Valoarea implicită trebuie să corespundă tipului de date al coloanei.

... hiredate DATE DEFAULT SYSDATE, ...

Pentru coloana hiredate s-a definit valoarea implicită returnată de funcția SYSDATE (data sistemului).

Exemplul următor creează tabelul departament, având trei coloane: DEPTNO, DNAME și LOC. Pentru a se confirma crearea tabelului dept se va executa comanda DESCRIBE.

```
CREATE TABLE departament (  
  deptno NUMBER(2),  
  dname VARCHAR2(14),  
  loc VARCHAR2(13));
```

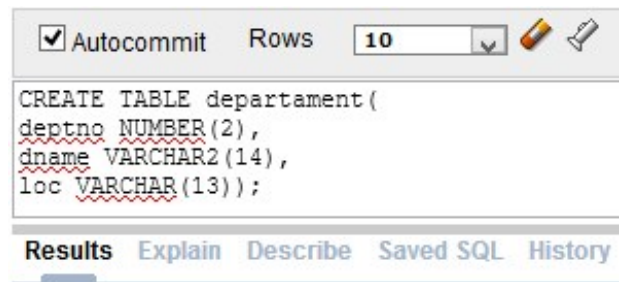


Table created.

DESCRIBE departament;

0.06 seconds

```
DESCRIBE departament;
```

Object Type	TABLE	Object	DEPARTMENT						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DEPARTMENT	DEPTNO	NUMBER	-	2	0	-	✓	-	-
	DNAME	VARCHAR2	14	-	-	-	✓	-	-
	LOC	VARCHAR2	13	-	-	-	✓	-	-

1 - 3

Interogarea dicționarului de date

Rezultatele execuției instrucțiunilor LDD sunt memorate în dicționarul de date. Prin interogarea dicționarului de date se pot obține diverse informații despre obiectele aparținând unui utilizator. Tabelele cele mai utilizate din dicționarul de date sunt USER_TABLES (conține informații despre tabelele create de utilizatorul curent), USER_OBJECTS (conține informații despre toate obiectele create de utilizatorul curent) și USER_CATALOG.

Afișarea tabelelor unui utilizator:

```
SELECT table_name FROM user_tables;
```

Afișarea obiectelor distincte aparținând unui utilizator:

```
SELECT DISTINCT object_name, object_type FROM user_objects;
```

Afișarea tabelelor, vederilor, sinonimelor și secvențelor unui utilizator:

```
SELECT * FROM user_catalog;
```

Deoarece tabelul USER_CATALOG are definit un sinonim, numit CAT, se poate folosi acest nume în loc de USER_CATALOG.

```
SELECT * FROM CAT;
```

Tipuri de date

Tipurile de date ce pot fi utilizate în definițiile de coloane sunt prezentate în tabelul următor:

Tipul de data	Descriere
VARCHAR2(<i>size</i>)	Dată de tip șir de caractere, de lungime variabilă (trebuie specificată dimensiunea maximă <i>size</i> ; valoarea implicită este 1, iar cea maximă 4000)
CHAR(<i>size</i>)	Dată de tip șir de caractere de lungime <i>size</i> fixă (valoarea implicită și cea minimă este 1, iar cea maximă 2000)
NUMBER(<i>p,s</i>)	Număr având <i>p</i> cifre și <i>s</i> cifre zecimale. (<i>p</i> poate lua valori între 1 și 38, <i>s</i> între -84 și 127)
DATE	Valori de tip dată calendaristică și timp cuprinse între Ianuarie 1, 4712 B.C. și Decembrie 31, 9999 A.D.
LONG	Dată de tip șir de caractere de lungime variabilă, dimensiunea maximă fiind 2 Gb.
CLOB	Dată de tip șir de caractere pe 1 octet, dimensiunea maximă fiind 4 Gb.
RAW(<i>size</i>)	Dată binară de dimensiune <i>size</i> octeți. Dimensiunea maximă este de 2000 octeți. Valoarea <i>size</i> trebuie specificată.
LONG RAW	Dată binară de dimensiune maximă 2 Gb.
BLOB	Dată binară de dimensiune maximă 4 Gb.
BFILE	Conține un pointer către un fișier binar extern bazei de date ce poate avea dimensiunea maximă de 4 Gb.

2.2 Crearea unui tabel utilizând o subinterogare

A doua metodă de creare a unui tabel constă în utilizarea clauzei AS subinterogare în instrucțiunea CREATE TABLE, care creează tabelul și inserează înregistrările furnizate de subinterogare. Sintaxa acestei instrucțiuni este:

```
CREATE TABLE tabel (coloana [, coloana ...]) AS subinterogare;
```

unde:

tabel numele tabelului;
coloana numele coloanei împreună cu valoare implicită și constrângeri de integritate (ultimele două elemente sunt opționale);
subinterogare instrucțiunea SELECT care furnizează mulțimea de înregistrări ce trebuie inserate în tabel.

Instrucțiunea următoare creează tabelul dept30, inserând informații despre angajații ce lucrează în departamentul 30. Structura tabelului dept30 este dată de lista de coloane din clauza SELECT.

```
CREATE TABLE dept30 AS SELECT empno, ename, sal*12 ANNSAL, hiredate FROM emp WHERE deptno=30;
```

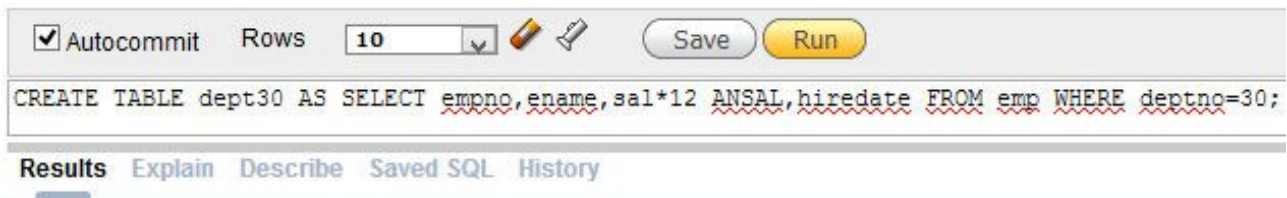


Table created.

0.10 seconds

```
DESCRIBE dept30;
```

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DEPT30	EMPNO	NUMBER	-	4	0	-	-	-	-
	ENAME	VARCHAR2	10	-	-	-	✓	-	-
	ANNSAL	NUMBER	22	-	-	-	✓	-	-
	HIREDATE	DATE	7	-	-	-	✓	-	-
									1 - 4

3. Instrucțiunea ALTER TABLE

Această instrucțiune este utilă dacă se urmărește modificarea structurii unui tabel prin:

- adăugarea unei coloane;
- ștergerea unei coloane;
- modificarea definiției unei coloane existente;
- definirea unei valori implicite pentru o coloană.

Pentru adăugarea unei coloane se va utiliza instrucțiunea ALTER TABLE împreună cu clauza ADD, iar pentru modificarea unei coloane se va utiliza clauza MODIFY. Dacă se dorește eliminarea unor coloane din definiția tabelului se va utiliza clauza DROP.

```
ALTER TABLE tabel ADD (coloana tip_date [DEFAULT expr] [, coloana tip_date ...]);
```

```
ALTER TABLE tabel MODIFY (coloana tip_date [DEFAULT expr] [, coloana tip_date ...]);
```

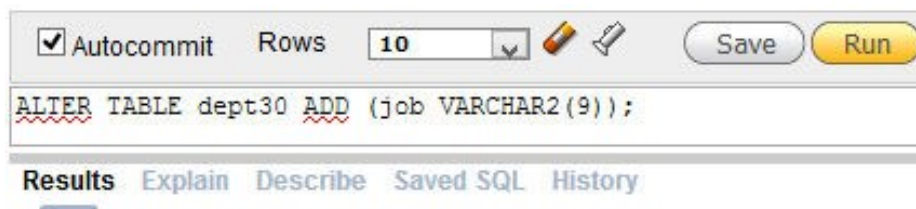
```
ALTER TABLE tabel DROP (coloana [, coloana ...]);
```

unde:

<i>tabel</i>	nume tabel;
<i>coloana</i>	nume coloană;
<i>tip_date</i>	tipul de date și dimensiunea;
DEFAULT <i>expr</i>	specifică valoarea implicită pentru coloană.

Următorul exemplu introduce o nouă coloană în tabelul emp30, numele acestei coloane fiind job, iar tipul de date asociat este varchar2(9). Pentru verificare se utilizează instrucțiunea SELECT.

```
ALTER TABLE dept30 ADD (job VARCHAR2(9));
```



```
Table altered.
```

```
0 18 seconds
```

```
SELECT * FROM dept30;
```

The screenshot shows a SQL editor interface. At the top, there is a toolbar with a checked 'Autocommit' checkbox, a 'Rows' dropdown set to '10', and 'Save' and 'Run' buttons. Below the toolbar, the SQL statement `SELECT * FROM dept30;` is entered. Underneath the statement, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History', with 'Results' being the active tab. Below the tabs, a table of results is displayed with 6 rows and 5 columns: EMPNO, ENAME, ANSAL, HIREDATE, and JOB.

EMPNO	ENAME	ANSAL	HIREDATE	JOB
7698	BLAKE	34200	05/01/1981	-
7499	ALLEN	19200	02/20/1981	-
7521	WARD	15000	02/22/1981	-
7654	MARTIN	15000	09/28/1981	-
7844	TURNER	18000	09/08/1981	-
7900	JAMES	11400	12/03/1981	-

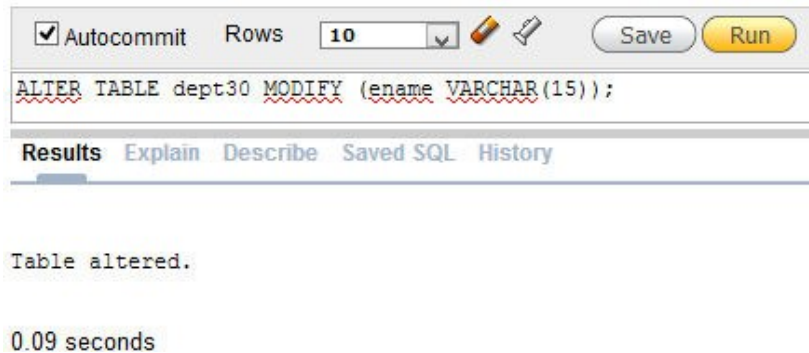
6 rows returned in 0.03 seconds [Download](#)

Observații:

Cu ajutorul instrucțiunii ALTER TABLE se pot adăuga, modifica și șterge coloane. În cazul inserării unei coloane nu se poate specifica locul de apariție al acesteia în schema tabelului. Noua coloană devine automat ultima coloană. Dacă tabelul conține înregistrări în momentul adăugării unei coloane noi, atunci noua coloană va fi inițializată cu valori NULL pentru toate înregistrările.

Se pot modifica specificațiile coloanelor utilizând instrucțiunea ALTER TABLE cu clauza MODIFY. Modificările permise sunt schimbarea tipului de date, a dimensiunii, a valorii inițiale și impunerea unei constrângeri NOT NULL.

```
ALTER TABLE dept30 MODIFY (ename VARCHAR(15));
```



Observații:

- se poate mări precizia sau scala unei coloane numerice;
- se poate micșora lățimea unei coloane dacă aceasta conține numai valori NULL sau dacă tabelul nu are înregistrări;
- se poate schimba tipul de date dacă în coloana respectivă există numai valori NULL;
- se poate converti o coloană de tip CHAR la tip VARCHAR2 sau invers dacă aceasta;
- conține valori NULL sau dacă nu se modifică lățimea;
- schimbarea valorii implicite pentru o coloană afectează numai inserările ulterioare în tabel.

4. Ștergerea unui tabel

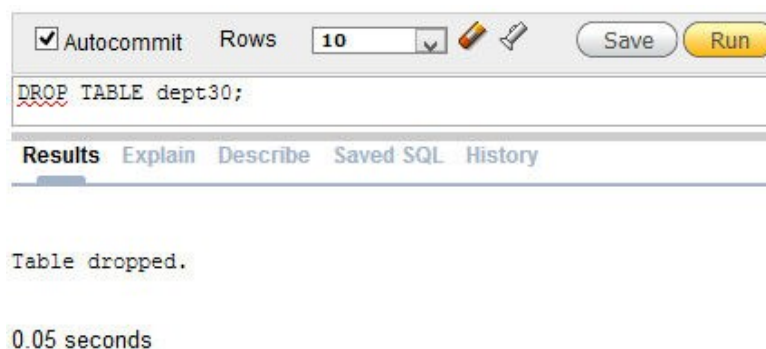
Comanda DROP TABLE șterge definiția unui tabel. Atunci când se șterge un tabel, baza de date pierde toate înregistrările din tabel, împreună cu indecșii asociați acestuia.

Sintaxa instrucțiunii DROP TABLE este:

```
DROP TABLE nume_tabel;
```

Exemplu:

```
DROP TABLE dept30;
```



Observații:

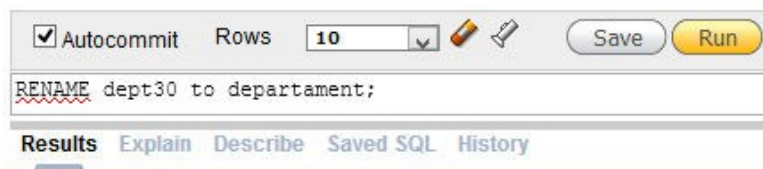
- toate datele sunt șterse;
- orice obiect dependent de tabel (vedere sau sinonim) va continua să existe, dar va fi invalid;
- orice tranzacție în curs va fi finalizată;
- numai utilizatorul care a creat tabelul sau cel care are privilegiul DROP ANY TABLE poate șterge un tabel.

5. Redenumirea unui obiect din baza de date

Pentru a modifica numele unui tabel, a unei vederi, secvențe sau sinonim se utilizează instrucțiunea RENAME.

```
RENAME nume_vechi TO nume_nou;
```

```
RENAME dept30 TO department;
```



```
Statement processed.
```

```
0.05 seconds
```

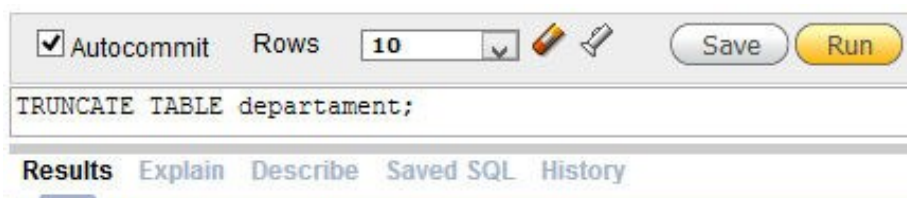
Notă: Numai proprietarul poate modifica numele unui obiect.

6. Trunchierea unui tabel

Instrucțiunea TRUNCATE TABLE face parte din LDD și șterge toate înregistrările din tabelul specificat, eliberând spațiul folosit de tabel. Operațiunea este ireversibilă, în sensul că instrucțiunea ROLLBACK nu va restaura conținutul tabelului.

```
TRUNCATE TABLE nume_tabel;
```

```
TRUNCATE TABLE department;
```



```
Table truncated.
```

```
0.15 seconds
```

Pentru ștergerea înregistrărilor dintr-un tabel se poate utiliza instrucțiunea DELETE, dar aceasta instrucțiune nu eliberează spațiul de stocare.

7. Adăugarea comentariilor de tabel și de coloană

Se pot adăuga comentarii având maxim 2000 octeți unei coloane, unui tabel sau unei vederi utilizând instrucțiunea COMMENT.

```
COMMENT ON <TABLE tabel | COLUMN tabel.col> IS text;
```

unde:

tabel numele tabelului;
col numele coloanei din tabel;
text textul comentariului, încadrat de apostrofuri (' ').

Comentariile sunt stocate în dicționarul de date și pot fi vizualizate în coloana COMMENTS a uneia din următoarele vederi:

ALL_COL_COMMENTS; - conține toate comentariile de coloane

USER_COL_COMMENTS; - conține comentariile asociate coloanelor utilizatorului curent

ALL_TAB_COMMENTS; - conține toate comentariile de tabele

USER_TAB_COMMENTS. - conține toate comentariile asociate tabelelor utilizatorului curent

```
SQL> COMMENT ON TABLE emp IS 'Employee Information';  
Comment created.
```

Se poate renunța la un comentariu, setându-l ca fiind șirul vid ('').

```
COMMENT ON TABLE emp IS '';
```

Probleme:

1. Creați tabelul DEPARTMENT având structura din tabelul următor și verificați crearea tabelului.

Column Name	Id	Name
Nulls/Unique		
Datatype	Number	Varchar2
Length	7	25

2. Introduceți înregistrări în tabelul DEPARTMENT folosind înregistrările din tabelul DEPT. Includeți doar coloanele de care aveți nevoie.

3. Creați tabelul EMPLOYEE având structura din tabelul următor.

Column Name	ID	LAST_NAME	FIRST_NAME	DEPT_ID
Nulls/Unique				
Datatype	Number	Varchar2	Varchar2	Number
Length	7	10	25	7

4. Modificați tabelul EMPLOYEE pentru a permite stocarea unor nume mai lungi (de exemplu de lungime 25 de caractere) în coloana LAST_NAME.

5. Verificați că cele două tabele create sunt stocate în dicționarul de date (*Indicație*: afișați datele de interes din vederea USER_TABLES).

6. Creați tabelul EMPLOYEE2 având aceeași structură ca și tabelul EMP, incluzând doar coloanele EMPNO, ENAME și DEPTNO. Redenumiți coloanele în noul tabel astfel: ID, LAST_NAME, DEPT_ID. Se vor prelua înregistrările din tabelul emp.
7. Ștergeți tabelul EMPLOYEE.
8. Redenumiți tabelul EMPLOYEE2 în EMPLOYEE.
9. Adăugați comentarii la definițiile celor 2 tabele, descriind semnificația tabelului respectiv.
10. Vizualizați modificările din dicționarul de date.