

SQL

Aplicatie: dictionar si tezaur multilingv

DDL: Trigger, Cursor

DML: Update, Insert

Proceduri si functii stocate

Trigger

```
CREATE [DEFINER = { user | CURRENT_USER }]
      TRIGGER trigger_name trigger_time trigger_event
      ON tbl_name
      FOR EACH ROW trigger_stmt
```

```
DROP TRIGGER [IF EXISTS]
[schema_name.] trigger_name
```

Cursor

```
DECLARE cursor_name CURSOR FOR select_statement
```

```
OPEN cursor_name
```

```
FETCH cursor_name INTO var_name [, var_name] ...
```

```
CLOSE cursor_name
```

SQL

Aplicatia gestioneaza un dictionar si tezaur multilingv

Entitatile cu care se opereaza sunt: limba, concept, tezaur si dictionar;

Pentru fiecare entitate se creeaza un tabel.

In tabela 'limba' sunt grupate limbile in care sunt disponibile traducerii; tabela 'concept' cuprinde toate conceptele (sau notinunile) pentru care exista traducerii in diferite limbi; in tabela 'tezaur' sunt grupate, in jurul unui concept dintr-o anume limba, acele notiuni care sunt, intr-un grad mai mare sau mai mic, sinonime cu acesta; cum aria semantica a unui termen poate diferi de la o limba la alta, sinonimele nu sunt universal valabile, ci numai pentru o anume limba.

In tabela 'dictionar' sunt furnizate traducerii pentru toate conceptele, in toate limbile disponibile.

E nevoie sa se adauge automat, la adaugarea unei limbi sau a unui nou concept, inregistrari in tabela 'dictionar'; acest proces este automatizat folosind doua triggeri (declansatoare).

```
##-----  
##  
## creare tabele limba, concept, tezaur, dictionar  
##  
##-----  
create DATABASE IF NOT EXISTS d;  
USE d;  
DROP TABLE IF EXISTS limba;  
DROP TABLE IF EXISTS concept;  
DROP TABLE IF EXISTS tezaur;  
DROP TABLE IF EXISTS dictionar;  
  
CREATE TABLE IF NOT EXISTS limba  
(id int unique auto_increment primary key,  
  nume char(20));  
  
CREATE TABLE IF NOT EXISTS concept  
(token_id char(20) unique primary key,  
  descriere char(50));  
  
CREATE TABLE IF NOT EXISTS tezaur  
(id int unique auto_increment primary key,  
  concept_token_id char(20),  
  limba_id int,  
  sinonim_token_id char(20),  
  FOREIGN KEY (concept_token_id) REFERENCES concept(token_id),  
  FOREIGN KEY (limba_id) REFERENCES limba(id),  
  FOREIGN KEY (sinonim_token_id) REFERENCES concept(token_id));  
  
CREATE TABLE IF NOT EXISTS dictionar  
(id int unique auto_increment primary key,  
  limba_id int,  
  concept_token_id char(20),  
  nume char(30),  
  FOREIGN KEY (limba_id) REFERENCES limba(id),  
  FOREIGN KEY (concept_token_id) REFERENCES concept(token_id));
```

SQL

```
## - Trigger AFTER INSERT tabela concept
## Este folosit pentru a adauga automat traduceri in dictionar, dupa fiecare inserare a unui concept
## Pentru fiecare limba existenta, se adauga cate o traducere; traducerea va contine numai un cod
## care indica faptul ca acel termen nu e inca tradus
DROP PROCEDURE IF EXISTS CURSOR_LIMBA;

DELIMITER //

CREATE PROCEDURE CURSOR_LIMBA(new_token_id char(20))

BEGIN

    DECLARE done INT DEFAULT 0;
    DECLARE nId INT;
    DECLARE cur1 CURSOR FOR SELECT id FROM limba;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;

    OPEN cur1;

    REPEAT
        FETCH cur1 INTO nId;
        IF NOT done THEN
            INSERT INTO dictionar (limba_id, concept_token_id, nume)
            VALUES(nId, new_token_id, '<enter text>');
        END IF;

    UNTIL done END REPEAT;

    CLOSE cur1;

END //
DELIMITER ;

DROP TRIGGER IF EXISTS insert_trad_dictionar;

delimiter //

CREATE TRIGGER insert_trad_dictionar AFTER INSERT ON concept

FOR EACH ROW BEGIN
    CALL CURSOR_LIMBA(NEW.token_id);
END; //

delimiter ;
```

Acest trigger se utilizeaza pentru a adauga inregistrari in tabela dictionar la adaugarea unui nou concept, cate o inregistrare pentru fiecare limba definita in tabela 'limba'.

Procedura implementeaza cursorul pentru tabela 'limba'; se parcurg toate inregistrarile tablei, si pentru fiecare inregistrare se adauga o noua inregistrare in tabela 'dictionar'. In campul 'nume' unde se adauga traducerea propriu-zisa se introduce un cod cu semnificatia 'traducere inexistentă'.

Trigger-ul apeleaza procedura CURSOR_LIMBA.

SQL

```
##-----
DROP PROCEDURE IF EXISTS CURSOR_CONCEPT;

DELIMITER //

CREATE PROCEDURE CURSOR_CONCEPT(new_limba_id INT)

BEGIN

    DECLARE done INT DEFAULT 0;
    DECLARE sId char(20);
    DECLARE cur2 CURSOR FOR SELECT token_id FROM concept;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;

    OPEN cur2;

    REPEAT
        FETCH cur2 INTO sId;
        IF NOT done THEN
            INSERT INTO dictionar (limba_id, concept_token_id, nume)
            VALUES(new_limba_id, sId, '<enter text>');

        END IF;

    UNTIL done END REPEAT;

    CLOSE cur2;

END //
DELIMITER ;

## - Trigger AFTER INSERT tabela limba
DROP TRIGGER IF EXISTS insert_limba_dictionar;

delimiter //

CREATE TRIGGER insert_limba_dictionar AFTER INSERT ON limba

FOR EACH ROW BEGIN
    CALL CURSOR_CONCEPT(NEW.id);
END; //

delimiter ;
```

Acest trigger se utilizeaza pentru a adauga inregistrari in tabela 'dictionar' la adaugarea unei noi limbi, cate o inregistrare pentru fiecare concept definit in tabela 'concept'.

Procedura implementeaza cursorul pentru tabela 'concept'; se parcurg toate inregistrarile tabelii, si pentru fiecare inregistrare se adauga o noua inregistrare in tabela 'dictionar'. In campul 'nume' unde se adauga traducerea propriu-zisa se introduce un cod cu semnificatia 'traducere inexistentă'.

Trigger-ul apeleaza procedura CURSOR_CONCEPT.

SQL

```
##-----  
##  
## inserare data in tabelele de limba, concept, tezaur, dictionar  
##  
##-----  
insert into limba  
(nume)  
VALUES  
( 'Romana' ),  
( 'Engleza' ),  
( 'Franceza' ),  
( 'Germana' );  
  
INSERT INTO concept  
(token_id,descriere)  
VALUES  
( 'OM', 'Om' ),  
( 'BARBAT', 'Barbat' ), ( 'FEMEIE', 'Femeie' ), ( 'CER', 'Cer' ), ( 'PAMANT', 'Pamant' ),  
( 'SOARE', 'Soare' ), ( 'LUNA', 'Luna' ), ( 'STEA', 'Stea' ), ( 'ASTRU', 'Astru' ),  
( 'PLANETA', 'Planeta' ), ( 'PERSOANA', 'Persoana' ), ( 'SATELIT', 'Satelit' ), ( 'COFIL', 'Copil' ),  
( 'FIINTA', 'Fiinta' ), ( 'ANIMAL', 'Animal' ), ( 'SALBATIC', 'Salbatic' ), ( 'LUNAR_ASTRAL', 'Lunar, cu referire la satelit' ),  
( 'LUNAR_CALENDARISTIC', 'Lunar, cu referire la perioada' ), ( 'SOLAR', 'Solar' ), ( 'ASTRAL', 'Astral, cu referire la astru' );  
  
INSERT INTO tezaur  
(concept_token_id,limba_id,sinonim_token_id)  
VALUES  
( 'STEA', 1, 'ASTRU' ),  
( 'ASTRU', 2, 'STEA' ),  
( 'ASTRU', 2, 'PLANETA' ),  
( 'SOLAR', 1, 'ASTRAL' ),  
( 'ASTRAL', 2, 'SOLAR' ),  
( 'LUNA', 1, 'SATELIT' ),  
( 'SATELIT', 2, 'LUNA' );
```

Inserare date in tabelele 'limba', 'concept', 'tezaur'; tabela 'dictionar' este populata prin declansarea, la adaugarea de inregistrari in tabela 'concept', trigger-ului 'insert_trad_dictionar'.

SQL

In tabela 'dictionar' s-au inserat, in urma adaugarii conceptelor in tabela 'concept', cate o inregistrare pentru fiecare limba;

Adaugand si o noua limba, se introduce, pentru fiecare concept existent in tabela 'concept', cate o inregistrare;



```
c:\wamp\bin\mysql\mysql5.1.30\bin\mysql.exe
mysql> select limba.ume As Limba, count(*) As 'Numar
-> from dictionar join limba
-> on limba.id = dictionar.limba_id
-> group by limba.ume;
+-----+-----+
| Limba  | Numar concepte |
+-----+-----+
| Engleza |          20    |
| Franceza |          20    |
| Germana |          20    |
| Romana  |          20    |
+-----+-----+
4 rows in set (0.00 sec)
mysql> _
```



```
c:\wamp\bin\mysql\mysql5.1.30\bin\mysql.exe
mysql> ## Adauga spaniola
mysql> insert into limba
-> (ume)
-> VALUES
-> ('Spaniola');
Query OK, 1 row affected (0.01 sec)

mysql>
mysql> select limba.ume As Limba, count(*) As 'Numar concepte'
-> from limba join dictionar on
-> limba.id = dictionar.limba_id
-> group by limba.ume;
+-----+-----+
| Limba  | Numar concepte |
+-----+-----+
| Engleza |          20    |
| Franceza |          20    |
| Germana |          20    |
| Romana  |          20    |
| Spaniola |          20    |
+-----+-----+
5 rows in set (0.00 sec)

mysql>
```

SQL

O vedere este o structura ne-persistenta echivalenta unei tabele temporare, rezultat fie al unei operatii de proiectie fie al unei operatii compuse de tip produs cartezian, selectie si proiectie.

Vederea 'traducere_romana' opereaza o selectie din 'dictionar' numai a inregistrarilor corespunzatoare limbii romane.

```
c:\wamp\bin\mysql\mysql5.1.30\bin\mysql.exe
mysql> CREATE VIEW traducere_romana AS
-> SELECT limba.ume As Limba, concept_token_id As Concept, dictionar.ume A
s Traducere FROM
-> limba, dictionar
-> WHERE limba.id = dictionar.limba_id
-> AND limba.ume = 'Romana';
Query OK, 0 rows affected (0.02 sec)

mysql> select * from traducere_romana;
+-----+-----+-----+
| Limba | Concept | Traducere |
+-----+-----+-----+
| Romana | OM | <enter text> |
| Romana | BARBAT | <enter text> |
| Romana | FEMEIE | <enter text> |
| Romana | CER | <enter text> |
| Romana | PAMANT | <enter text> |
| Romana | SOARE | <enter text> |
| Romana | LUNA | <enter text> |
| Romana | STEA | <enter text> |
| Romana | ASTRU | <enter text> |
| Romana | PLANETA | <enter text> |
| Romana | PERSOANA | <enter text> |
| Romana | SATELIT | <enter text> |
| Romana | COPIL | <enter text> |
| Romana | FIINTA | <enter text> |
| Romana | ANIMAL | <enter text> |
| Romana | SALBATIC | <enter text> |
| Romana | LUNAR_ASTRAL | <enter text> |
| Romana | LUNAR_CALENDARISTIC | <enter text> |
| Romana | SOLAR | <enter text> |
| Romana | ASTRAL | <enter text> |
+-----+-----+-----+
20 rows in set (0.02 sec)
```

SQL

```
## Romanian
UPDATE dictionar SET nume = 'Om' WHERE limba_id = 1 AND concept_token_id = 'OM';
UPDATE dictionar SET nume = 'Barbat' WHERE limba_id = 1 AND concept_token_id = 'BARBAT';
UPDATE dictionar SET nume = 'Femeie' WHERE limba_id = 1 AND concept_token_id = 'FEMEIE';
UPDATE dictionar SET nume = 'Cer' WHERE limba_id = 1 AND concept_token_id = 'CER';
UPDATE dictionar SET nume = 'Pamant' WHERE limba_id = 1 AND concept_token_id = 'PAMANT';
UPDATE dictionar SET nume = 'Soare' WHERE limba_id = 1 AND concept_token_id = 'SOARE';
UPDATE dictionar SET nume = 'Luna' WHERE limba_id = 1 AND concept_token_id = 'LUNA';
UPDATE dictionar SET nume = 'Stea' WHERE limba_id = 1 AND concept_token_id = 'STEA';
UPDATE dictionar SET nume = 'Astru' WHERE limba_id = 1 AND concept_token_id = 'ASTRU';
UPDATE dictionar SET nume = 'Planeta' WHERE limba_id = 1 AND concept_token_id = 'PLANETA';
UPDATE dictionar SET nume = 'Persoana' WHERE limba_id = 1 AND concept_token_id = 'PERSOANA';
UPDATE dictionar SET nume = 'Satelit' WHERE limba_id = 1 AND concept_token_id = 'SATELIT';
UPDATE dictionar SET nume = 'Copil' WHERE limba_id = 1 AND concept_token_id = 'COPIL';
UPDATE dictionar SET nume = 'Fiinta' WHERE limba_id = 1 AND concept_token_id = 'FIINTA';
UPDATE dictionar SET nume = 'Animal' WHERE limba_id = 1 AND concept_token_id = 'ANIMAL';
UPDATE dictionar SET nume = 'Salbatic' WHERE limba_id = 1 AND concept_token_id = 'SALBATIC';
UPDATE dictionar SET nume = 'Lunar' WHERE limba_id = 1 AND concept_token_id = 'LUNAR_ASTRAL';
UPDATE dictionar SET nume = 'Lunar' WHERE limba_id = 1 AND concept_token_id = 'LUNAR_CALENDARISTIC';
UPDATE dictionar SET nume = 'Solar' WHERE limba_id = 1 AND concept_token_id = 'SOLAR';
UPDATE dictionar SET nume = 'Astral' WHERE limba_id = 1 AND concept_token_id = 'ASTRAL';
```

```
## English
UPDATE dictionar SET nume = 'Human' WHERE limba_id = 2 AND concept_token_id = 'OM';
UPDATE dictionar SET nume = 'Man' WHERE limba_id = 2 AND concept_token_id = 'BARBAT';
UPDATE dictionar SET nume = 'Woman' WHERE limba_id = 2 AND concept_token_id = 'FEMEIE';
UPDATE dictionar SET nume = 'Sky' WHERE limba_id = 2 AND concept_token_id = 'CER';
UPDATE dictionar SET nume = 'Earth' WHERE limba_id = 2 AND concept_token_id = 'PAMANT';
UPDATE dictionar SET nume = 'Sun' WHERE limba_id = 2 AND concept_token_id = 'SOARE';
UPDATE dictionar SET nume = 'Moon' WHERE limba_id = 2 AND concept_token_id = 'LUNA';
UPDATE dictionar SET nume = 'Star' WHERE limba_id = 2 AND concept_token_id = 'STEA';
UPDATE dictionar SET nume = 'Star' WHERE limba_id = 2 AND concept_token_id = 'ASTRU';
UPDATE dictionar SET nume = 'Planet' WHERE limba_id = 2 AND concept_token_id = 'PLANETA';
UPDATE dictionar SET nume = 'Person' WHERE limba_id = 2 AND concept_token_id = 'PERSOANA';
UPDATE dictionar SET nume = 'Satellite' WHERE limba_id = 2 AND concept_token_id = 'SATELIT';
UPDATE dictionar SET nume = 'Child' WHERE limba_id = 2 AND concept_token_id = 'COPIL';
UPDATE dictionar SET nume = 'Being' WHERE limba_id = 2 AND concept_token_id = 'FIINTA';
UPDATE dictionar SET nume = 'Animal' WHERE limba_id = 2 AND concept_token_id = 'ANIMAL';
UPDATE dictionar SET nume = 'Wild' WHERE limba_id = 2 AND concept_token_id = 'SALBATIC';
UPDATE dictionar SET nume = 'Lunar' WHERE limba_id = 2 AND concept_token_id = 'LUNAR_ASTRAL';
UPDATE dictionar SET nume = 'Monthly' WHERE limba_id = 2 AND concept_token_id = 'LUNAR_CALENDARISTIC';
UPDATE dictionar SET nume = 'Solar' WHERE limba_id = 2 AND concept_token_id = 'SOLAR';
UPDATE dictionar SET nume = 'Astral' WHERE limba_id = 2 AND concept_token_id = 'ASTRAL';
```


SQL

```
## Francais
UPDATE dictionar SET nume = 'Homme' WHERE limba_id = 3 AND concept_token_id = 'OM';
UPDATE dictionar SET nume = 'Homme' WHERE limba_id = 3 AND concept_token_id = 'BARBAT';
UPDATE dictionar SET nume = 'Femme' WHERE limba_id = 3 AND concept_token_id = 'FEMEIE';
UPDATE dictionar SET nume = 'Ciel' WHERE limba_id = 3 AND concept_token_id = 'CER';
UPDATE dictionar SET nume = 'Terre' WHERE limba_id = 3 AND concept_token_id = 'PAMANT';
UPDATE dictionar SET nume = 'Soleil' WHERE limba_id = 3 AND concept_token_id = 'SOARE';
UPDATE dictionar SET nume = 'Lune' WHERE limba_id = 3 AND concept_token_id = 'LUNA';
UPDATE dictionar SET nume = 'Etoile' WHERE limba_id = 3 AND concept_token_id = 'STEA';
UPDATE dictionar SET nume = 'Astre' WHERE limba_id = 3 AND concept_token_id = 'ASTRU';
UPDATE dictionar SET nume = 'Planette' WHERE limba_id = 3 AND concept_token_id = 'PLANETA';
UPDATE dictionar SET nume = 'Personne' WHERE limba_id = 3 AND concept_token_id = 'PERSOANA';
UPDATE dictionar SET nume = 'Satelite' WHERE limba_id = 3 AND concept_token_id = 'SATELIT';
UPDATE dictionar SET nume = 'Enfant' WHERE limba_id = 3 AND concept_token_id = 'COPIL';
UPDATE dictionar SET nume = 'Etre' WHERE limba_id = 3 AND concept_token_id = 'FIINTA';
UPDATE dictionar SET nume = 'Animale' WHERE limba_id = 3 AND concept_token_id = 'ANIMAL';
UPDATE dictionar SET nume = 'Sauvage' WHERE limba_id = 3 AND concept_token_id = 'SALBATIC';
UPDATE dictionar SET nume = 'Lunaire' WHERE limba_id = 3 AND concept_token_id = 'LUNAR_ASTRAL';
UPDATE dictionar SET nume = 'Lunaire' WHERE limba_id = 3 AND concept_token_id = 'LUNAR_CALENDARISTIC';
UPDATE dictionar SET nume = 'Solaire' WHERE limba_id = 3 AND concept_token_id = 'SOLAR';
UPDATE dictionar SET nume = 'Astrale' WHERE limba_id = 3 AND concept_token_id = 'ASTRAL';

## Deutch
UPDATE dictionar SET nume = 'Man' WHERE limba_id = 4 AND concept_token_id = 'OM';
UPDATE dictionar SET nume = 'Man' WHERE limba_id = 4 AND concept_token_id = 'BARBAT';
UPDATE dictionar SET nume = 'Frau' WHERE limba_id = 4 AND concept_token_id = 'FEMEIE';
UPDATE dictionar SET nume = 'Sonnen' WHERE limba_id = 4 AND concept_token_id = 'SOARE';
```

SQL

Vederea 'traducere_romana' si 'traducere_engleza' dupa popularea tabelii 'dictionar' cu traduceri.

```
c:\wamp\bin\mysql\mysql5.1.30\bin\mysql.exe
mysql> select * from traducere_romana;
```

Limba	Concept	Traducere
Romana	OM	Om
Romana	BARBAT	Barbat
Romana	FEMEIE	Femeie
Romana	CER	Cer
Romana	PAMANT	Pamant
Romana	SOARE	Soare
Romana	LUNA	Luna
Romana	STEA	Stea
Romana	ASTRU	Astru
Romana	PLANETA	Planeta
Romana	PERSOANA	Persoana
Romana	SATELIT	Satelit
Romana	COFIL	Copil
Romana	FIINTA	Fiinta
Romana	ANIMAL	Animal
Romana	SALBATIC	Salbatic
Romana	LUNAR_ASTRAL	Lunar
Romana	LUNAR_CALENDARISTIC	Lunar
Romana	SOLAR	Solar
Romana	ASTRAL	Astral

```
20 rows in set (0.01 sec)
```

```
c:\wamp\bin\mysql\mysql5.1.30\bin\mysql.exe
mysql> CREATE VIEW traducere_engleza AS
  -> SELECT limba.ume As Limba, concept_token_id
s Traducere FROM
  -> limba, dictionar
  -> WHERE limba.id = dictionar.limba_id
  -> AND limba.ume = 'Engleza';
Query OK, 0 rows affected (0.00 sec)

mysql> select * from traducere_engleza;
```

Limba	Concept	Traducere
Engleza	OM	Human
Engleza	BARBAT	Man
Engleza	FEMEIE	Woman
Engleza	CER	Sky
Engleza	PAMANT	Earth
Engleza	SOARE	Sun
Engleza	LUNA	Moon
Engleza	STEA	Star
Engleza	ASTRU	Star
Engleza	PLANETA	Planet
Engleza	PERSOANA	Person
Engleza	SATELIT	Satellite
Engleza	COFIL	Child
Engleza	FIINTA	Being
Engleza	ANIMAL	Animal
Engleza	SALBATIC	Wild
Engleza	LUNAR_ASTRAL	Lunar
Engleza	LUNAR_CALENDARISTIC	Monthly
Engleza	SOLAR	Solar
Engleza	ASTRAL	Astral

```
20 rows in set (0.00 sec)

mysql>
```

SQL

```
# Verificare exista traducere din limba sursa in limba destinatie pentru termenul 'denumire'.
DROP PROCEDURE IF EXISTS EXISTA_TRADUCERE;

DELIMITER //

CREATE PROCEDURE EXISTA_TRADUCERE(traducere_din_limba char(20), traducere_in_limba char(20), denumire char(30))
BEGIN
    SET @limba_din_id = NULL, @limba_in_id = NULL, @ctid = NULL, @nume_traducere = NULL;

    SELECT @limba_din_id := id FROM limba WHERE nume = traducere_din_limba;
    SELECT @limba_in_id := id FROM limba WHERE nume = traducere_in_limba;

    SELECT @ctid := concept_token_id from dictionar
    WHERE nume = denumire AND limba_id = @limba_din_id;

    IF @ctid IS NULL THEN
        SELECT CONCAT('Nu exista cuvantul ',denumire, ' in ',traducere_din_limba, '.')
        AS Rezultat;
    ELSE
        BEGIN
            SELECT @nume_traducere := nume FROM dictionar
            WHERE concept_token_id = @ctid AND limba_id = @limba_in_id;

            IF @nume_traducere <> '<enter text>' THEN
                SELECT CONCAT('Traducerea ',denumire, ' in ',traducere_in_limba, ' este:',@nume_traducere, '.')
                AS Rezultat;
            ELSE SELECT CONCAT('Nu exista traducerea ',denumire, ' in limba ',traducere_in_limba, '.')
            AS Rezultat;
            END IF;
        END;
    END IF;

END //

DELIMITER ;
```

Procedura pentru verificarea existentei traducerii unui concept dintr-o limba in alta. Parametrii de intrare sunt limba sursa, limba destinatie si conceptul, precizate prin numele acestora.

SQL

```
# Adauga traducere din limba sursa in limba destinatie pentru termenul 'denumire'.
DROP PROCEDURE IF EXISTS ADAUGA_TRADUCERE;

DELIMITER //

CREATE PROCEDURE ADAUGA_TRADUCERE(traducere_din_limba char(20), traducere_in_limba char(20),
                                denumire_sursa char(30), denumire_destinatie char(30))
BEGIN
    SET @limba_din_id = NULL, @limba_in_id = NULL, @ctid = NULL, @nume_traducere = NULL;

    SELECT @limba_din_id := id FROM limba WHERE nume = traducere_din_limba;
    SELECT @limba_in_id := id FROM limba WHERE nume = traducere_in_limba;

    SELECT @ctid := concept_token_id from dictionar
    WHERE nume = denumire_sursa AND limba_id = @limba_din_id;

    IF @ctid IS NULL THEN
        SELECT CONCAT('Nu exista cuvantul ',denumire_sursa, ' in ',traducere_din_limba, '.')
        AS Rezultat;
    ELSE
        BEGIN
            SELECT @nume_traducere := nume FROM dictionar
            WHERE concept_token_id = @ctid AND limba_id = @limba_in_id;

            IF (@nume_traducere = '<enter text>') THEN
                UPDATE dictionar SET nume = denumire_destinatie
                WHERE concept_token_id = @ctid AND limba_id = @limba_in_id;

            ELSE SELECT CONCAT('Traducerea deja existenta. Operatie abandonata.') As Rezultat;
            END IF;

        END;

    END IF;

END //
DELIMITER ;
```

Procedura pentru adaugarea traducerii unui concept dintr-o limba in alta. Parametrii de intrare sunt limba sursa, limba destinatie si denumirea in limba sursa si traducerea in limba destinatie.

SQL

```
## Adauga un concept nou in tezaur; se introduc ca parametri traducerile si nu conceptele
DROP PROCEDURE IF EXISTS ADAUGA_SINONIM;

DELIMITER //

CREATE PROCEDURE ADAUGA_SINONIM(limba_nume char(20), denumire_concept char(30), denumire_sinonim char(30))
BEGIN
    SET @limbaid = NULL, @concept_id = NULL, @sinonim_id = NULL;

    SELECT @limbaid := id FROM limba WHERE nume = limba_nume;

    SELECT @concept_id := concept_token_id from dictionar
    WHERE nume = denumire_concept AND limba_id = @limbaid;

    SELECT @sinonim_id := concept_token_id from dictionar
    WHERE nume = denumire_sinonim AND limba_id = @limbaid;

    IF @concept_id IS NULL THEN SELECT CONCAT('Nu exista cuvantul ',denumire_concept, ' in ',limba_nume,')
        AS Rezultat;
    END IF;
    IF @sinonim_id IS NULL THEN SELECT CONCAT('Nu exista cuvantul ',denumire_sinonim, ' in ',limba_nume,')
        AS Rezultat;
    END IF;
    IF @concept_id IS NOT NULL AND @sinonim_id IS NOT NULL THEN
        BEGIN
            ## Adaugam inregistrari doar daca nu exista deja
            SET @selectat = NULL;
            SELECT @selectat := id FROM tezaur WHERE concept_token_id = @concept_id AND sinonim_token_id = @sinonim_id
            AND limba_id = @limbaid;
            IF @selectat IS NULL THEN
                BEGIN
                    INSERT INTO tezaur (concept_token_id, limba_id, sinonim_token_id)
                    VALUES (@concept_id,@limbaid, @sinonim_id);
                END;
            END IF;
        END;
    END IF;

END //
DELIMITER ;
```

Procedura pentru adaugarea unui sinonim in tezaur. Numele conceptelor se dau in traducerea din limba in care se adauga conceptul in tezaur.

SQL

```
c:\wamp\bin\mysql\mysql5.1.30\bin\mysql.exe
mysql> select * from traducere_germana
  -> where Traducere <> '<enter text>';
+-----+-----+-----+
| Limba | Concept | Traducere |
+-----+-----+-----+
| Germana | OM      | Man      |
| Germana | BARBAT  | Man      |
| Germana | FEMEIE  | Frau     |
| Germana | SOARE   | Sonnen   |
+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> CALL ADAUGA_TRADUCERE('Romana','Germana','Copil','Kinder');
+-----+-----+-----+
| @limba_din_id := id |
+-----+-----+-----+
| 1 |
+-----+-----+-----+
1 row in set (0.02 sec)

+-----+-----+-----+
| @limba_in_id := id |
+-----+-----+-----+
| 4 |
+-----+-----+-----+
1 row in set (0.03 sec)

+-----+-----+-----+
| @ctid := concept_token_id |
+-----+-----+-----+
| COPIL |
+-----+-----+-----+
1 row in set (0.05 sec)

+-----+-----+-----+
| @nume_traducere := nume |
+-----+-----+-----+
| <enter text> |
+-----+-----+-----+
1 row in set (0.06 sec)

Query OK, 1 row affected (0.06 sec)

mysql> select * from traducere_germana
  -> where Traducere <> '<enter text>';
+-----+-----+-----+
| Limba | Concept | Traducere |
+-----+-----+-----+
| Germana | OM      | Man      |
| Germana | BARBAT  | Man      |
| Germana | FEMEIE  | Frau     |
| Germana | SOARE   | Sonnen   |
| Germana | COPIL   | Kinder   |
+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> _
```

