**Lucrarea de laborator N6**

**Cursoare**

**Cozma Denis**

1. ***Fie bază de date Cercetători cu următoarele tabele:***

Articole(idarticol, denarticol)

Universitate(iduniversitate, denuniversitate)

Cercetători(idcercetator, numecercetător, idarticol, iduniversitate)

***Tabelul Articole***

|  |  |
| --- | --- |
| idarticol | denarticol |
| 1 | Articol1 |
| 2 | Articol2 |
| 3 | Articol3 |
| 4 | Articol4 |
| 5 | Articol5 |

***Tabelul Autori***

|  |  |
| --- | --- |
| idarticol | idcercetator |
| 1 | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 2 |

***Tabelul Universitate***

|  |  |
| --- | --- |
| iduniversitate | denuniversitate |
| 1 | USARB |
| 2 | USM |
| 3 | ASEM |

***Tabelul Cecetatori***

|  |  |  |
| --- | --- | --- |
| idcercetator | numecercetător | iduniversitate |
| 1 | Dodu Petru | 1 |
| 2 | Lungu Vasile | 2 |
| 3 | Vrabie Maria | 1 |
| 4 | Ombun Bogdan | 3 |

*-- CREATE TABLE articole*

*-- (*

*-- idarticol INT,*

*-- denarticol VARCHAR(50)*

*-- );*

*-- INSERT INTO articole (idarticol,denarticol) VALUES*

*-- (1, 'Articol1'),*

*-- (2, 'Articol2'),*

*-- (3, 'Articol3'),*

*-- (4, 'Articol4'),*

*-- (5, 'Articol5');*

*--*

*-- CREATE TABLE autori (*

*-- IdCercetator INT(11) NOT NULL,*

*-- IdArticol INT(11) NOT NULL, PRIMARY KEY (IdCercetator, IdArticol)*

*-- );*

*--*

*-- INSERT INTO autori*

*-- (IdCercetator, IdArticol) VALUES*

*-- (1, 1),*

*-- (2, 2),*

*-- (3, 3),*

*-- (4, 4);*

*-- INSERT INTO autori*

*-- (IdCercetator, IdArticol) VALUES*

*-- (1, 1),*

*-- (2, 2),*

*-- (3, 3),*

*-- (4, 4);*

*-- CREATE TABLE universitate (*

*-- iduniversitate INT(11) NOT NULL,*

*-- denuniversitate TINYTEXT NOT NULL, PRIMARY KEY (iduniversitate)*

*-- );*

*-- INSERT INTO universitate (iduniversitate, denuniversitate) VALUES*

*-- (1, 'USARB'),*

*-- (2, 'USM'),*

*-- (3, 'ASEM');*

*-- CREATE TABLE cercetatori (*

*-- idcercetator INT(11) NOT NULL,*

*-- numecercetător TINYTEXT NOT NULL,*

*-- iduniversitate INT(11) NOT NULL, PRIMARY KEY (idcercetator)*

*-- );*

1. ***Sarcini de realizat:***
2. Creati o noua tabela, copie a tabelei emp. Stergeti inregistrarile
3. Creați o copie a tabelului Cercetatori. Ștergeți înregistrările pentru care codul universității este 1. Afișați pe ecran numărul de înregistrări eliminate.

creez o copie a tabelului cercetatori

*-- CREATE TABLE laborator3.cercetatori\_copie LIKE laborator3.cercetatori;*

*-- copii datele din tabelul cercetatori in cercetatori\_copie*

*-- INSERT INTO lab\_3.cercetatori\_copie SELECT \* FROM lab\_3.cercetatori;*

*-- Fac o procedura in care voi face un cursor. Fara el cursoru nu va lucra*

*-- /\* variabila hadler - a\*/*

*-- Declare done integer default 0;*

*--*

*/\*declararea cursorului\*/*

**Declare** BankCursor **Cursor** **for** **select** \* **from** cercetatori\_copie **where** iduniversitate = 1;

*/\*HANDLER programare, pe care o vom explica mai jos\*/*

**DECLARE** **CONTINUE** **HANDLER** **FOR** **SQLSTATE** '02000' **SET** done=1;

**Set** counter = 0;

*/\* deschiderea cursorului \*/*

**Open** BankCursor;

*/\*scoaterea datelor \*/*

**WHILE** done = 0 **DO**

**FETCH** BankCursor **INTO** id\_cercet,cercet\_name,cercet\_univ,cercet\_calif;

**delete** **from** cercetatori\_copie **where** idcercetator = id\_cercet;

**Set** counter = counter +1;

*-- select id\_cercet;*

**END** **WHILE**;

**select** counter **as** 'оnregistrările eliminate';

*/\*inchiderea cursorului \*/*

**Close** BankCursor;

**END**;

**END**

1. Pentru tabelul Cercetatori, folosind un cursor să se afișeze numele cercetătorului și denumirea universității care au cel puțin două articole. Cu ajutorului unui alt cursor să se afișeze lista cercetătorilor care au un articol. Să se definească pentru fiecare listă un antet. Între rezultatele cursoarelor să se tragă o linie.

DELIMITER $$

*--*

*-- CREATE PROCEDURE proc\_cursors2*

*-- BEGIN*

*-- Begin*

*-- /\* variabila unde noi scoatem datele \*/*

*-- Declare id\_cercet int;*

*-- Declare cercet\_name VARCHAR(30);*

*-- Declare cercet\_univ VARCHAR(30);*

*-- Declare cercet\_art int;*

*--*

*--*

*-- /\* var hadler - a\*/*

*-- DECLARE more\_rows BOOLEAN DEFAULT TRUE;*

*--*

*-- /\*Initializare cursorului 1\*/*

*-- Declare BankCursor Cursor for*

*-- select count(autori.IdArticol) as Number, cercetatori\_copie.numecercetător, universitate.denuniversitate*

*-- from cercetatori\_copie*

*-- inner join autori on autori.IdCercetator = cercetatori\_copie.idcercetator*

*-- inner join universitate on universitate.iduniversitate = cercetatori\_copie.iduniversitate*

*-- group by autori.IdCercetator*

*-- having count(autori.IdArticol) >=2;*

*--*

*-- /\*declararea cursorului 2\*/*

*-- Declare BankCursor2 Cursor for*

*-- select count(autori.IdArticol) as Number, cercetatori\_copie.numecercetător, universitate.denuniversitate*

*-- from cercetatori\_copie*

*-- inner join autori on autori.IdCercetator = cercetatori\_copie.idcercetator*

*-- inner join universitate on universitate.iduniversitate = cercetatori\_copie.iduniversitate*

*-- group by autori.IdCercetator*

*-- having count(autori.IdArticol) = 1;*

*--*

*--*

*-- /\*HANDLER 1\*/*

*-- DECLARE CONTINUE HANDLER FOR NOT FOUND SET more\_rows = FALSE;*

*--*

*-- Set counter = 0;*

*-- /\* deschiderea cursorului 1\*/*

*-- Open BankCursor;*

*-- /\*scoatem datele \*/*

*-- DROP TABLE IF EXISTS date\_cercetator;*

*-- CREATE TEMPORARY TABLE date\_cercetator(*

*-- cercet\_art int,*

*-- cercet\_name VARCHAR(30),*

*-- cercet\_univ VARCHAR(30)*

*-- );*

*-- insert into date\_cercetator (cercet\_art,cercet\_name ,cercet\_univ) values (null, 'Cercetatorile', 'cu cel putin 2 articole');*

*--*

*-- while more\_rows do*

*-- FETCH BankCursor INTO cercet\_art,cercet\_name,cercet\_univ;*

*-- IF more\_rows THEN*

*-- insert into date\_cercetator (cercet\_art,cercet\_name ,cercet\_univ) values (cercet\_art, cercet\_name, cercet\_univ);*

*-- END IF;*

*-- End while;*

*-- /\*inchiderea cursorul 1\*/*

*-- Close BankCursor;*

*--*

*-- -- ---------------------------*

*-- begin*

*--*

*-- Declare more\_rows2 BOOLEAN DEFAULT TRUE;*

*-- /\*HANDLER 1\*/*

*-- DECLARE CONTINUE HANDLER FOR NOT FOUND SET more\_rows2 = FALSE;*

*--*

*-- /\* deschiderea cursor 2\*/*

*-- Open BankCursor2;*

*--*

*-- insert into date\_cercetator (cercet\_art,cercet\_name ,cercet\_univ) values (null, '--------', '--------'), (null, 'Cercetatorile', 'cu un articol');*

*-- while more\_rows2 do*

*-- FETCH BankCursor2 INTO cercet\_art,cercet\_name,cercet\_univ;*

*-- IF more\_rows2 THEN*

*-- insert into date\_cercetator (cercet\_art,cercet\_name ,cercet\_univ) values (cercet\_art, cercet\_name, cercet\_univ);*

*-- END IF;*

*-- End while;*

*-- select \* from date\_cercetator;*

*-- DROP TABLE IF EXISTS date\_cercetator;*

*-- /\*inchiderea cursor2 \*/*

*-- Close BankCursor2;*

*-- end;*

*--*

*-- END;*

*--*

*-- END;*

1. Folosind un cursor care va afișa în trei variabile diferite pentru a afișa un mesaj (primul nume =, al doilea nume =, al treilea nume = ) însoțit de numele primelor 3 cercetători ordonați în ordine alfabetică a numelui.

DELIMITER $$

*-- CREATE PROCEDURE proc\_cursors3 BEGIN BEGIN*

*-- /\* variabila unde scoatem datele \*/ DECLARE cercet\_name1 VARCHAR(30); DECLARE cercet\_name2 VARCHAR(30); DECLARE cercet\_name3 VARCHAR(30); DECLARE counter INT; DECLARE MESSAGE\_TEXT VARCHAR(200);*

*--*

*-- /\* variabila hadler - a\*/ DECLARE more\_rows BOOLEAN DEFAULT TRUE;*

*--*

*-- /\*declarrea cursor 1\*/ DECLARE BankCursor CURSOR FOR*

*-- SELECT cercetatori\_copie.numecercetător*

*-- FROM cercetatori\_copie*

*-- ORDER BY cercetatori\_copie.numecercetător*

*-- LIMIT 3;*

*--*

*-- /\*HANDLER \*/ DECLARE CONTINUE HANDLER FOR NOT FOUND SET more\_rows = FALSE;*

*--*

*-- /\* deschidem cursor\*/ OPEN BankCursor;*

*-- /\*scoatem datele \*/ SET counter = 0; WHILE more\_rows DO*

*-- IF COUNTER = 0 THEN FETCH BankCursor INTO cercet\_name1; END IF;*

*-- IF COUNTER = 0 THEN FETCH BankCursor INTO cercet\_name2; END IF;*

*-- IF COUNTER = 0 THEN FETCH BankCursor INTO cercet\_name3; END IF;*

*--*

*-- -- IF more\_rows THEN*

*--*

*-- -- END IF;*

*-- END WHILE;*

*-- SELECT CONCAT('primul nume =',cercet\_name1, ', al doilea nume =', cercet\_name2, ', al treilea nume =', cercet\_name3) AS message;*

*--*

*-- -- Acest mesaj va aparea ca eroare*

*-- SELECT CONCAT('primul nume =',cercet\_name1, ', al doilea nume =', cercet\_name2, ', al treilea nume =', cercet\_name3) INTO MESSAGE\_TEXT; SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT =MESSAGE\_TEXT;*

*-- /\*inchiderea cursor \*/ CLOSE BankCursor; END; END;*

1. Pentru tabela emp, fără a modifica conţinutul tabelei, folositi un cursor pentru a afisa numele angajatului,
2. Folosind un cursor să se afișeze pentru fiecare cercetător numărul de articole scrise în formatul:

Nume – Dodu, prenume – Petru, articole – 1

*-- DELIMITER $$*

*-- CREATE PROCEDURE proc\_cursors4 BEGIN BEGIN*

*-- /\* variabila unde scoatem datele \*/ DECLARE cercet\_name VARCHAR(30); DECLARE cercet\_surnname VARCHAR(30); DECLARE counter\_art INT; DECLARE MESSAGE\_TEXT VARCHAR(200);*

*--*

*-- /\* variabila hadler - a\*/ DECLARE more\_rows BOOLEAN DEFAULT TRUE;*

*--*

*-- /\*DEclararea cursorului 1\*/ DECLARE BankCursor CURSOR FOR*

*-- SELECT SUBSTRING\_INDEX(SUBSTRING\_INDEX(cercetatori\_copie.numecercetător, ' ', 1), ' ', -1) AS first\_name, SUBSTRING\_INDEX(SUBSTRING\_INDEX(cercetatori\_copie.numecercetător, ' ', 2), ' ', -1) AS last\_name, COUNT(autori.IdCercetator) AS articole*

*-- FROM cercetatori\_copie*

*-- INNER JOIN autori ON autori.IdCercetator = cercetatori\_copie.idcercetator*

*-- GROUP BY cercetatori\_copie.idcercetator;*

*--*

*-- /\*HANDLER \*/ DECLARE CONTINUE HANDLER FOR NOT FOUND SET more\_rows = FALSE;*

*-- DROP TABLE IF EXISTS date\_cercetator;*

*-- CREATE TEMPORARY TABLE date\_cercetator(*

*-- cercet\_name VARCHAR(30),*

*-- cercet\_surnname VARCHAR(30),*

*-- counter\_art VARCHAR(30)*

*-- );*

*-- /\* deschiderea cursorului\*/ OPEN BankCursor;*

*-- /\*citirea datelor \*/ SET MESSAGE\_TEXT = ''; WHILE more\_rows DO*

*-- IF more\_rows THEN FETCH BankCursor INTO cercet\_name, cercet\_surnname, counter\_art;*

*-- INSERT INTO date\_cercetator(cercet\_name,cercet\_surnname,counter\_art) VALUES (CONCAT('Nume – ', cercet\_surnname), CONCAT('prenume – ', cercet\_name), CONCAT('articole – ', counter\_art)); SET MESSAGE\_TEXT = CONCAT(MESSAGE\_TEXT, '| Nume – ', cercet\_surnname, ', prenume – ', cercet\_name, ', articole – ', counter\_art); END IF; END WHILE;*

*-- SELECT \**

*-- FROM date\_cercetator;*

*-- DROP TABLE IF EXISTS date\_cercetator;*

*--*

*-- /\*inchiderea cursor \*/ CLOSE BankCursor; SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT =MESSAGE\_TEXT; END; END;*

1. Folosind un cursor să se afișeze pentru fiecare articol autorii acestui în formatul:

Denumirea articolului – Articol1, autori - Dodu Petru.

DELIMITER $$

*-- CREATE PROCEDURE proc\_cursors5 BEGIN BEGIN*

*-- /\* variabila unde scoatem datele \*/ DECLARE result VARCHAR(150); DECLARE MESSAGE\_TEXT VARCHAR(200);*

*--*

*-- /\* variabila hadler - a\*/ DECLARE more\_rows BOOLEAN DEFAULT TRUE;*

*--*

*-- /\*declararea cursor 1\*/ DECLARE BankCursor CURSOR FOR*

*-- SELECT CONCAT ('Denumirea articolului – ', cercetatori\_copie.numecercetător, ', autori - ',*

*-- articole.denarticol) AS result*

*-- FROM cercetatori\_copie*

*-- INNER JOIN autori ON autori.IdCercetator = cercetatori\_copie.idcercetator*

*-- INNER JOIN articole ON articole.idarticol = autori.IdArticol;*

*-- /\*HANDLER \*/*

*-- DECLARE CONTINUE HANDLER FOR NOT FOUND SET more\_rows = FALSE;*

*--*

*-- /\* deschiderea cursor\*/*

*-- Open BankCursor;*

*-- /\*citirea datelor\*/*

*-- set MESSAGE\_TEXT = '';*

*-- while more\_rows do*

*-- FETCH BankCursor INTO result;*

*-- IF more\_rows THEN*

*-- select result;*

*-- set MESSAGE\_TEXT = Concat(MESSAGE\_TEXT, result, ' | ');*

*-- END IF;*

*-- End while;*

*--*

*--*

*-- /\*inchiderea cursor \*/*

*-- Close BankCursor;*

*-- SIGNAL SQLSTATE '45000' set MESSAGE\_TEXT =MESSAGE\_TEXT;*

*-- END;*

*--*

*-- END;*