APLICAȚIE INFORMATICĂ PENTRU ACTIVITATEA BIBLIOTECARĂ UTILIZÂND ORACLE SQL DEVELOPER

Student:Tolbariu Andrei

1. **Introducere**

Acest proiect își propune realizarea unei baze de date care sa faciliteze gestiunea, studenților universității, cărților și evidența împrumuturilor în cadrul bibliotecii Licuriciul din cadrul Universității de Litere Mihai Eminescu.

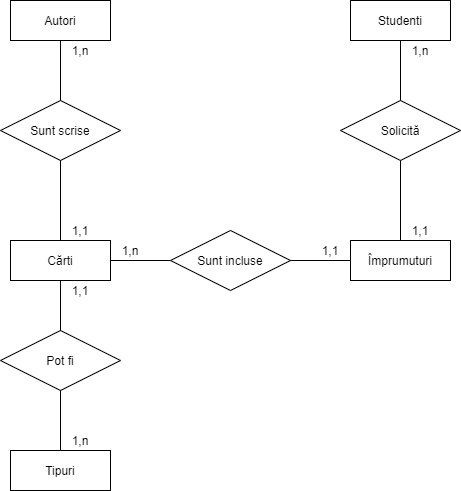
Universitatea de Litere Mihai Eminescu este o instituție de învățământ superior. Aceasta își propune să ofere cercetare și educație la cele mai înalte standarde. Biblioteca Licuriciul are sediul în interiorul U.L.M.E. și funcționează ca subsistem al universități.

Cu privire la domeniul de activitate selectat, fiind și principala sursă de informare preferată de studenții acestei universități, care oferă un volum de informații cât mai detaliate sunt referințele scrise – cărțile. Însă această bibliotecă se confruntă cu provocările impuse de era digitală și anume un sistem de gestionare rapid, care să consume cât mai puține resurse, ușor de folosit și digital.

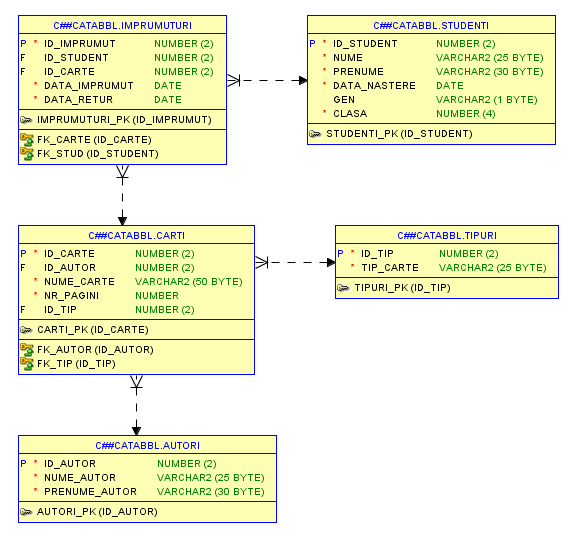
Soluția propusă constă în implementarea unei baze de date pentru gestiunea, studenților, cărților și evidența împrumuturilor.

Principalele activități ale Bibliotecii Licuriciul sunt gestiunea cărților și evidența împrumuturilor.

**3. Diagrama Entitate – Asociere**



**Schema conceptuală a bazei de date**



4.

Baza de date este formată din cele 5 tabele: Studenți, Împrumuturi, Cărți, Autori și Tipuri.

Pentru creare tabelelor am folosit comanda create table iar pentru adăugarea constrângerilor vom folosi ALTER TABLE:

Pentru tabelul STUDENTI:

Text

Description automatically generated

Pentru Tabela Imprumuturi:

Text

Description automatically generated with low confidence

Pentru Tabela Carti:

Text

Description automatically generated with low confidence

Pentru Tabela Autori:

Text

Description automatically generated

Pentru Tabela Tipuri

Text

Description automatically generated

Pentru a evidentia structura tabelelor am folosit comanda describe:

Describe STUDENTI;

------------ -------- ------------

ID\_STUDENT NOT NULL NUMBER(2)

NUME NOT NULL VARCHAR2(25)

PRENUME NOT NULL VARCHAR2(30)

DATA\_NASTERE NOT NULL DATE

GEN VARCHAR2(1)

CLASA NOT NULL NUMBER(4)

Pentru Tabelul Carti

Describe CARTI;

---------- -------- ------------

ID\_CARTE NOT NULL NUMBER(2)

ID\_AUTOR NUMBER(2)

NUME\_CARTE NOT NULL VARCHAR2(50)

NR\_PAGINI NOT NULL NUMBER

ID\_TIP NUMBER(2)

Pentru Tabelul Imprumuturi

Describe IMPRUMUTURI;

------------- -------- ---------

ID\_IMPRUMUT NOT NULL NUMBER(2)

ID\_STUDENT NUMBER(2)

ID\_CARTE NUMBER(2)

DATA\_IMPRUMUT NOT NULL DATE

DATA\_RETUR NOT NULL DATE

Pentru Tabelul Tipuri

Describe TIPURI;

--------- -------- ------------

ID\_TIP NOT NULL NUMBER(2)

TIP\_CARTE NOT NULL VARCHAR2(25)

Pentru Tabelul Autori

Describe AUTORI;

------------- -------- ------------

ID\_AUTOR NOT NULL NUMBER(2)

NUME\_AUTOR NOT NULL VARCHAR2(25)

PRENUME\_AUTOR NOT NULL VARCHAR2(30)

5. În schemă se pot observa tipurile de date ale fiecărei tabele, precum și cheile primare, cheile externe și constrângerile efectuate.

Tabela Studenți:

Table

Description automatically generated

În această tabelă se regăsesc următoarele câmpuri: ID STUDENT, NUME, PRENUME, DATA NAȘTERE, GEN, CLASA. Astfel, ID STUDENT este cheia primară a tabelei. De asemenea, câmpurile NUME, PRENUME și DATA NAȘTERE au constrângerea NOT NULL, astfel încât să nu fie posibilă introducerea unor valori nule pe câmpurile respective.

Constrangerea NOT NULL a fost exemplificata prin comanda ALTER TABLE. De mentionat, este faptul ca acest tip de constrangere poate fi adaugat si in momentul creari tabelei.

Tabela Cărți:

Table

Description automatically generated

În această tabelă se regăsesc următoarele câmpuri: ID ÎMPRUMUT, ID STUDENT, ID CARTE,

DATĂ ÎMPRUMUT și DATĂ RETUR. Câmpul ID ÎMPRUMUT este cheia primară a tabelei, iar ID STUDENT și ID CARTE sunt chei externe.

Tabela Cărți:

Table

Description automatically generated

În această tabelă se regăsesc următoarele câmpuri: ID CARTE, ID AUTOR, NUME CARTE, NR DE PAGINI ȘI ID TIP. Câmpul ID CARTE este cheia primară a tabelei, iar ID AUTOR și ID TIP sunt chei externe.

Tabela Autori:

În această tabelă se regăsesc următoarele câmpuri: ID AUTOR , NUME AUTOR și PRENUME AUTOR.

Câmpul ID AUTOR este cheia primară a tabelei, iar aceasta nu are chei externe.

Table

Description automatically generated

Tabela Tipuri:

În această tabelă se regăsesc următoarele câmpuri: ID TIP și TIP CARTE. Câmpul ID TIP este cheia primară a tabelei, iar aceasta nu are chei externe.

Graphical user interface, application, table

Description automatically generated

Ca un următor pas, s-au introdus date în toate cele 5 tabele. Astfel, s-a folosit comanda INSERT.

INSERT INTO STUDENTI VALUES(01,'Chita','Mihai',to\_date('24-03-1999','dd-mm-yyyy'),'M',1014);

INSERT INTO STUDENTI VALUES(02,'Popa','Andreea',to\_date('04-03-1998','dd-mm-yyyy'),'F',1014);

INSERT INTO STUDENTI VALUES(03,'Sima','Ioana',to\_date('18-03-1998','dd-mm-yyyy'),'F',1013);

INSERT INTO STUDENTI VALUES(04,'Zanfir','Maria',to\_date('25-02-1998','dd-mm-yyyy'),'F',1011);

INSERT INTO STUDENTI VALUES(05,'Georgesu','Adi',to\_date('21-02-1999','dd-mm-yyyy'),'M',1013);

INSERT INTO STUDENTI VALUES(06,'Ionescu','Calin',to\_date('17-06-1999','dd-mm-yyyy'),'M',1014);

INSERT INTO STUDENTI VALUES(07,'Dumitrescu','Andrei',to\_date('14-07-1996','dd-mm-yyyy'),'M',1014);

INSERT INTO STUDENTI VALUES(08,'Grigore','Mihai',to\_date('03-08-1997','dd-mm-yyyy'),'M',1013);

INSERT INTO STUDENTI VALUES(09,'Dragnea','Ionut',to\_date('16-08-1999','dd-mm-yyyy'),'M',1013);

INSERT INTO STUDENTI VALUES(10,'Stefanescu','Denisa',to\_date('04-06-1999','dd-mm-yyyy'),'F',1012);

INSERT INTO STUDENTI VALUES(11,'Irimescu','Maria',to\_date('05-03-1998','dd-mm-yyyy'),'F',1013);

INSERT INTO STUDENTI VALUES(12,'Andreescu','Ana',to\_date('26-10-1999','dd-mm-yyyy'),'F',1011);

INSERT INTO STUDENTI VALUES(13,'Chiran','Sebastian',to\_date('28-11-1999','dd-mm-yyyy'),'M',1011);

INSERT INTO STUDENTI VALUES(14,'Dragulescu','Catalin',to\_date('16-06-1999','dd-mm-yyyy'),'M',1013);

INSERT INTO STUDENTI VALUES(15,'Ivanescu','Cristina',to\_date('18-07-1999','dd-mm-yyyy'),'F',1014);

INSERT INTO STUDENTI VALUES(16,'Barbieru','Elena',to\_date('18-07-1999','dd-mm-yyyy'),'F',1014);

INSERT INTO STUDENTI VALUES(17,'Popescu','Cristi',to\_date('12-06-1998','dd-mm-yyyy'),'M',1012);

INSERT INTO STUDENTI VALUES(18,'Marinescu','Cezar',to\_date('17-04-1999','dd-mm-yyyy'),'M',1011);

INSERT INTO STUDENTI VALUES(19,'Avram','Adrian',to\_date('30-09-1997','dd-mm-yyyy'),'M',1014);

INSERT INTO STUDENTI VALUES(20,'Radu','Bogdan',to\_date('08-03-1997','dd-mm-yyyy'),'M',1014);

INSERT INTO STUDENTI VALUES(21,'Cioiu','Maria',to\_date('12-03-1999','dd-mm-yyyy'),'F',1011);

INSERT INTO STUDENTI VALUES(22,'Popescu','Alex',to\_date('25-12-1999','dd-mm-yyyy'),'M',1014);

INSERT INTO STUDENTI VALUES(23,'Dudea','Andreea',to\_date('21-10-1998','dd-mm-yyyy'),'F',1012);

INSERT INTO STUDENTI VALUES(24,'Popa','Mihai',to\_date('22-10-1996','dd-mm-yyyy'),'M',1014);

INSERT INTO STUDENTI VALUES(25,'Petrescu','Adrian',to\_date('26-06-1996','dd-mm-yyyy'),'M',1012);

INSERT INTO STUDENTI VALUES(26,'Patrascu','Diana',to\_date('17-10-1997','dd-mm-yyyy'),'F',1012);

INSERT INTO STUDENTI VALUES(27,'Balan','Iuliana',to\_date('24-11-1996','dd-mm-yyyy'),'F',1011);

INSERT INTO STUDENTI VALUES(28,'Draghici','Elisa',to\_date('16-12-1996','dd-mm-yyyy'),'F',1013);

INSERT INTO STUDENTI VALUES(29,'Stancu','Monica',to\_date('19-08-1997','dd-mm-yyyy'),'F',1012);

INSERT INTO STUDENTI VALUES(30,'Stefan','Ioana',to\_date('20-05-1996','dd-mm-yyyy'),'F',1011);

INSERT INTO STUDENTI VALUES(31,'Branescu','Alexandra',to\_date('23-06-1996','dd-mm-yyyy'),'F',1011);

INSERT INTO STUDENTI VALUES(32,'Alexandrescu','Cristian',to\_date('23-04-1998','dd-mm-yyyy'),'M',1012);

INSERT INTO STUDENTI VALUES(33,'Zabava','Catalin',to\_date('22-04-1997','dd-mm-yyyy'),'M',1013);

INSERT INTO STUDENTI VALUES(34,'Radu','Alexandra',to\_date('14-09-1999','dd-mm-yyyy'),'F',1013);

INSERT INTO STUDENTI VALUES(35,'Damascu','Radu',to\_date('17-12-1999','dd-mm-yyyy'),'M',1012);

INSERT INTO STUDENTI VALUES(36,'Zamfir','Cristiana',to\_date('19-08-1998','dd-mm-yyyy'),'F',1011);

commit;

INSERT INTO AUTORI VALUES(01,'F.Scott','Fitzgerald');

INSERT INTO AUTORI VALUES(02,'Dickens','Charles');

INSERT INTO AUTORI VALUES(03,'Austen','Jane');

INSERT INTO AUTORI VALUES(04,'Woolf','Virginia');

INSERT INTO AUTORI VALUES(05,'Orwell','George');

INSERT INTO AUTORI VALUES(06,'Kafka','Franz');

INSERT INTO AUTORI VALUES(07,'Dahl','Roald');

INSERT INTO AUTORI VALUES(08,'Hugo','Victor');

INSERT INTO AUTORI VALUES(09,'Carroll','Lewis');

INSERT INTO AUTORI VALUES(10,'Bronte','Emily');

commit;

INSERT INTO TIPURI VALUES(01,'FICTIUNE');

INSERT INTO TIPURI VALUES(02,'NON-FICTIUNE');

INSERT INTO TIPURI VALUES(03,'SF');

INSERT INTO TIPURI VALUES(04,'FICTIUNE ISTORICA');

INSERT INTO TIPURI VALUES(05,'POEZIE');

INSERT INTO TIPURI VALUES(06,'THRILLER');

INSERT INTO TIPURI VALUES(07,'FABULA');

INSERT INTO TIPURI VALUES(08,'COMEDIE');

INSERT INTO TIPURI VALUES(09,'DRAMA');

INSERT INTO TIPURI VALUES(10,'FILOZOFIE');

commit;

INSERT INTO CARTI VALUES (01,02,'Great Expectations',505,01);

INSERT INTO CARTI VALUES (02,04,'Mrs Dalloway',194,01);

INSERT INTO CARTI VALUES (03,10,'Wuthering Heights',464,04);

INSERT INTO CARTI VALUES (04,06,'The Metamorphosis',201,03);

INSERT INTO CARTI VALUES (05,09,'Through the Looking-Glass',208,03);

INSERT INTO CARTI VALUES (06,08,'Les MisÃ©rables',1463,04);

INSERT INTO CARTI VALUES (07,07,'Matilda',240,03);

INSERT INTO CARTI VALUES (08,04,'To the Lighthouse',204,01);

INSERT INTO CARTI VALUES (09,10,'Poems by Currer, Ellis and Acton Bell',218,05);

INSERT INTO CARTI VALUES (10,07,'James and the Giant Peach',146,03);

INSERT INTO CARTI VALUES (11,02,'Oliver Twist',608,01);

INSERT INTO CARTI VALUES (12,04,'The Waves',297,01);

INSERT INTO CARTI VALUES (13,03,'Pride and Prejudice',279,09);

INSERT INTO CARTI VALUES (14,01,'The Great Gatsby',200,04);

INSERT INTO CARTI VALUES (15,05,'1984',237,03);

commit;

INSERT INTO IMPRUMUTURI VALUES (01,15,03,TO\_DATE('01-05-2020','dd-mm-yyyy'),TO\_DATE('01-07-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (02,02,04,TO\_DATE('01-04-2020','dd-mm-yyyy'),TO\_DATE('01-06-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (03,14,05,TO\_DATE('01-07-2020','dd-mm-yyyy'),TO\_DATE('01-09-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (04,31,07,TO\_DATE('01-06-2020','dd-mm-yyyy'),TO\_DATE('01-08-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (05,27,02,TO\_DATE('01-07-2020','dd-mm-yyyy'),TO\_DATE('01-09-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (06,09,03,TO\_DATE('01-02-2020','dd-mm-yyyy'),TO\_DATE('01-04-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (07,17,12,TO\_DATE('01-05-2020','dd-mm-yyyy'),TO\_DATE('01-07-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (08,18,14,TO\_DATE('01-03-2020','dd-mm-yyyy'),TO\_DATE('01-05-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (09,10,09,TO\_DATE('01-01-2020','dd-mm-yyyy'),TO\_DATE('01-03-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (10,29,08,TO\_DATE('01-07-2020','dd-mm-yyyy'),TO\_DATE('01-09-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (11,02,01,TO\_DATE('01-04-2020','dd-mm-yyyy'),TO\_DATE('01-06-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (12,16,06,TO\_DATE('01-05-2020','dd-mm-yyyy'),TO\_DATE('01-07-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (13,18,11,TO\_DATE('01-04-2020','dd-mm-yyyy'),TO\_DATE('01-06-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (14,18,15,TO\_DATE('01-02-2020','dd-mm-yyyy'),TO\_DATE('01-04-2020','dd-mm-yyyy'));

INSERT INTO IMPRUMUTURI VALUES (15,21,01,TO\_DATE('01-08-2020','dd-mm-yyyy'),TO\_DATE('01-10-2020','dd-mm-yyyy'));

commit;

Table

Description automatically generated

În imaginea precedentă sunt reprezentate datele din tabela STUNDETI.

Table

Description automatically generated

În imaginea precedentă sunt reprezentate datele din tabela IMPRUMUTURI.

Graphical user interface, text, application, table

Description automatically generated

În imaginea precedentă sunt reprezentate datele din tabela CĂRȚI.

Table, Excel

Description automatically generated

În imaginea precedentă sunt reprezentate datele din tabela AUTORI.

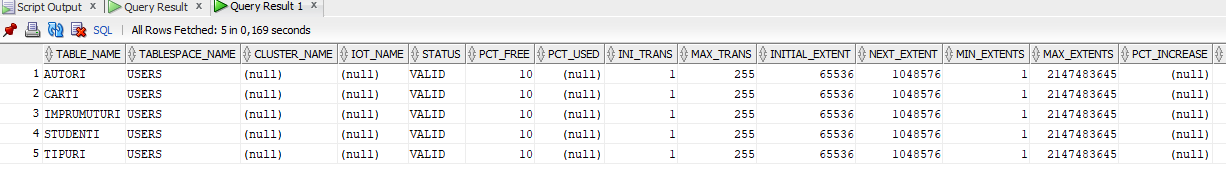
Graphical user interface, table

Description automatically generated with medium confidence

În imaginea precedentă sunt reprezentate datele din tabela TIPURI.

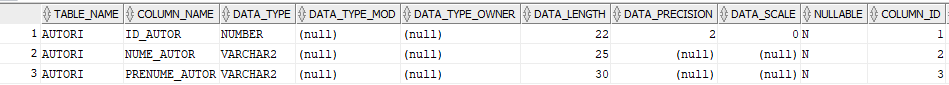
In urmatorul output, se prezinta tabelele existente in „user\_tables”.

*SELECT \* FROM USER\_TABLES;*

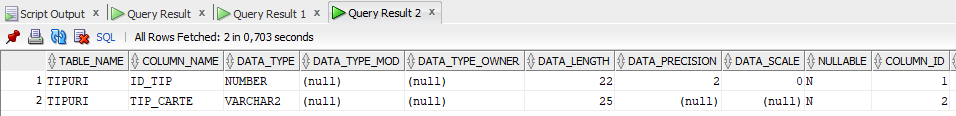


Putem vizualiza structura tabelelor folosind clauza select pe „USER\_TAB\_COLS”.

*select \* from user\_tab\_cols where table\_name = 'AUTORI';*

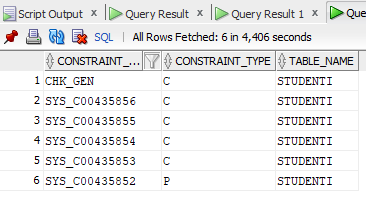


*select \* from user\_tab\_cols where table\_name = 'TIPURI';*

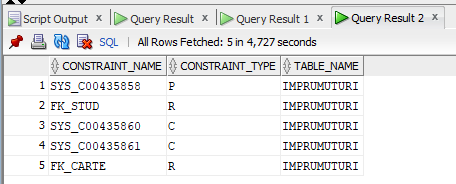


Se poate folosi tabela „USER\_CONSTRAINTS” pentru a arata constrangerile unei tabele:

*SELECT CONSTRAINT\_NAME, CONSTRAINT\_TYPE,TABLE\_NAME FROM USER\_CONSTRAINTS WHERE TABLE\_NAME LIKE 'STUDENTI' ;*



*SELECT CONSTRAINT\_NAME, CONSTRAINT\_TYPE,TABLE\_NAME FROM USER\_CONSTRAINTS WHERE TABLE\_NAME LIKE 'IMPRUMUTURI' ;*



6. In continuare, vor fi exemplificate cateva comenzi de prelucrare pe tabele.

Afișarea studenților care au returnat cărțile împrumutate mai devreme de luna August.

*SELECT NUME, PRENUME, CLASA*

*FROM STUDENTI*

*WHERE ID\_STUDENT ÎN*

*(SELECT ID\_STUDENT FROM IMPRUMUTURI*

*WHERE EXTRACT (MONTH FROM DATA\_RETUR) <8)*

*ORDER BY NUME;*

Table

Description automatically generated

Afișarea studenților care au împrumutat cărți intre luna aprilie și iunie.

*SELECT ID\_STUDENT, NUME, PRENUME, CLASA*

*FROM STUDENTI*

*WHERE ID\_STUDENT ÎN*

*(SELECT ID\_STUDENT FROM IMPRUMUTURI*

*WHERE EXTRACT (MONTH FROM DATA\_IMPRUMUT) BETWEEN 4 AND 6)*

*ORDER BY NUME;*

Graphical user interface, application, table

Description automatically generated

Afișarea tuturor studenților care au împrumutat cărți.

*SELECT S.ID\_STUDENT, S.NUME, S.PRENUME*

*FROM STUDENTI S*

*WHERE S.ID\_STUDENT*

*ÎN (SELECT I.ID\_STUDENT FROM IMPRUMUTURI I*

*WHERE I.ID\_STUDENT = S.ID\_STUDENT) ORDER BY ID\_STUDENT ASC;*

Table

Description automatically generated

Afișarea studenților care nu au împrumutat cărți.

*SELECT S.ID\_STUDENT, S.NUME, S.PRENUME*

*FROM STUDENTI S*

*WHERE S.ID\_STUDENT*

*NOT ÎN (SELECT I.ID\_STUDENT FROM IMPRUMUTURI I*

*WHERE I.ID\_STUDENT = S.ID\_STUDENT) ORDER BY ID\_STUDENT ASC;*



Afișarea autorilor, cărților pe care le-au scris, cat și tipul acestora.

*SELECT A.NUME\_AUTOR, A.PRENUME\_AUTOR, C.NUME\_CARTE, T.TIP\_CARTE*

*FROM AUTORI A, CARTI C, TIPURI T*

*WHERE A.ID\_AUTOR=C.ID\_AUTOR AND C.ID\_TIP=T.ID\_TIP*

*ORDER BY 1;*

Table

Description automatically generated

Afișarea studenților, numele cărților, data împrumutului pentru cărțile de tipul ficțiune istorica.

*SELECT S.NUME, S.PRENUME, C.NUME\_CARTE, T.TIP\_CARTE, I.DATA\_IMPRUMUT*

*FROM*

*STUDENTI S, IMPRUMUTURI I, TIPURI T, CARTI C*

*WHERE*

*S.ID\_STUDENT = I.ID\_STUDENT AND I.ID\_CARTE=C.ID\_CARTE AND C.ID\_TIP=T.ID\_TIP*

*AND TIP\_CARTE= UPPER ('FICTIUNE ISTORICA') AND*

*EXTRACT (MONTH FROM DATA\_IMPRUMUT)>=3*

*ORDER BY NUME;*

Graphical user interface, text, table

Description automatically generated with medium confidence

Afișarea numărului mediu de pagini a operelor în funcție de autor.

*SELECT A.NUME\_AUTOR, AVG (C.NR\_PAGINI) AS MEDIE\_NR\_PAG FROM AUTORI A, CARTI C*

*WHERE A.ID\_AUTOR=C.ID\_AUTOR*

*GROUP BY A.NUME\_AUTOR;*

Graphical user interface, application, table

Description automatically generated

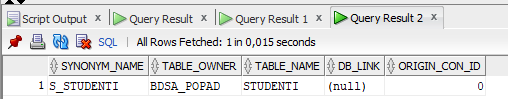
De asemenea, se pot crea si sinonime peste tabele.

*CREATE OR REPLACE SYNONYM S\_STUDENTI FOR STUDENTI ;*

Astfel, acest sinonim va avea toate datele din tabela Studenti.

*SELECT \* FROM USER\_SYNONYMS;*

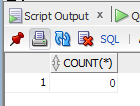
Se poate vizualiza definitia lui in tabela USER\_SYNONYMS.



De asemenea, putem sa ii dam Drop pentru a-l sterge si sa interogam aceeasi tabela pentru a vedea ca nu mai exista:

*DROP SYNONYM S\_STUDENTI;*

*SELECT COUNT(\*) FROM USER\_SYNONYMS;*



Count-ul este 0, deoarece nu mai aveam alte sinonime definite.

7.

S-a creat o secvență prin care se incrementează id-ul studenților în vederea inserării unor noi studenți.

*CREATE SEQUENCE SEQUENCE2*

*START WITH 36*

*INCREMENT BY 1*

*MINVALUE 0*

*MAXVALUE 100*

*CYCLE;*



În continuare, se vor realiza câteva actualizări pe tabele.

În primul rând, s-a efectuat o modificare a clasei studentului cu id-ul 37.

*UPDATE STUDENTI*

*SET CLASA = 1012*

*WHERE ID\_STUDENT = 37;*



De asemenea, s-a modificat data împrumutului s-a modificat pentru studentul cu id-ul 15, deoarece fusese introdusă eronat.

*UPDATE IMPRUMUTURI*

*SET DATA\_IMPRUMUT = '01-10-2020'*

*WHERE ID\_IMPRUMUT = 15;*



Se poate folosi si comanda DELETE pentru a sterge anumite inregistrari din tabele.

*DELETE FROM STUDENTI WHERE ID\_STUDENT = 37; COMMIT;*

8.

Se creeaza si o vizualizare noua, join intre tabela AUTORI si CARTI avand conditia ca numarul de pagini al cartilor sa fie mai mare de 250 de pagini.

*CREATE OR REPLACE VIEW AUTORI\_CARTI(ID\_AUTOR,NUME\_CARTE,NR\_PAGINI,ID\_TIP)*

*AS SELECT A.ID\_AUTOR,C.NUME\_CARTE,C.NR\_PAGINI,C.ID\_TIP FROM AUTORI A*

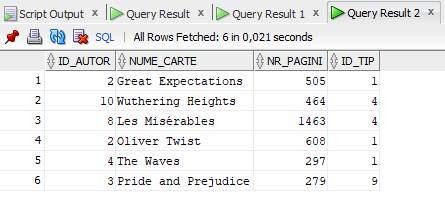
*INNER JOIN CARTI C*

*ON C.ID\_AUTOR = A.ID\_AUTOR*

*WHERE C.NR\_PAGINI > 250;*

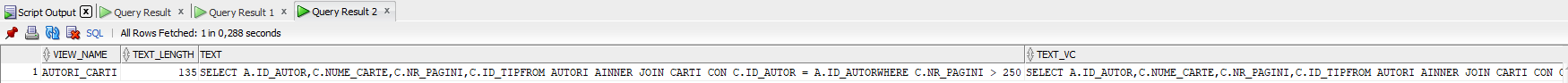
Se selecteaza continutul tabelei:

*SELECT \* FROM AUTORI\_CARTI;*



Se poate observa view-ul interogand tabela USER\_VIEWS.

*SELECT \* FROM USER\_VIEWS;*



Se poate observa ca in coloana TEXT\_VC avem definitia vizualizarii.

Alte Exemple de Viewuri:

Un View pentru vizualizarea Carților de ficțiune

Graphical user interface, text, application

Description automatically generated

Un View pentru vizualizarea carților împrumutate:

Table

Description automatically generated with medium confidence

Un View pentru vizualizarea elevilor ce au împrumutat carțile intre lunile Aprilie și Iunie

Graphical user interface, text, application

Description automatically generated

Un View pentru a vizualiza carțile împrumutate

Graphical user interface

Description automatically generated with low confidence