Baza de date a unei biblioteci

[1.Descrierea modelului real](#_9s3zaad3tsxs)

[2.Regulile generale de funcționare ale bazei de date a unei biblioteci](#_p77cuwxitb2e)

[3.Entități](#_tyen6q9ei6g)

[4.Relații](#_ejz15hd4xf54)

[5.Atribute](#_sz7mhnqi9abg)

[6.Diagrama entitate-relatie](#_93yi9nqfuga1)

[7.Diagrama conceptuala](#_5qpa6qh5wkdf)

[8.Scheme relationale](#_phod8bl3qz98)

[9.Normalizari FN1, FN2, FN3](#_9lqg9mmfp2l6)

[10.Secvențe](#_iavw41e5djp1)

[11.Creare tabele](#_4l58ul4y459k)

[Carte](#_fx4hrszzwg8)

[Autori](#_q23oxwe2onbp)

[Categorie](#_hmr8oikbn8eb)

[Carte\_autor](#_lhdcux7piuq8)

[Carte\_categorie](#_srx36x1b3nly)

[Exemplar\_carte](#_yqs7uxsh3842)

[Editura](#_lng943auut84)

[Info\_imprumut](#_ldge7ajbwsg7)

[Bibliotecar](#_wx1jlze77bm7)

[Cititor](#_25rijfotuvfx)

[12.Cereri](#_cs5vfgirqjlw)

[13.Actualizare și suprimare](#_9aqwo92epsvw)

[14.View](#_28s3yz1lfrgz)

[15. Outer join , division , analiza top-n](#_2ztnyuqpi9ru)

[16. Comparare plan de executie a 2 cereri echivalente semantic](#_pu4zweegom9o)

[18.Ilustrarea a doua consistency levels](#_dcuuckpjm247)

[19. Optimizarea cererilor folosind indexare](#_cjxbuvpq780w)

# 

## **1.Descrierea modelului real**

Biblioteca este o instituție care oferă acces la o varietate de surse de informații, cum ar fi cărți, reviste, publicații periodice, filme, muzică și alte materiale media. Scopul bibliotecii este de a permite utilizatorilor sa împrumute aceste resurse pentru a-si extinde cunoștințele și pentru a se bucura de lucrurile pe care le oferă.

Bibliotecile pot varia între mărime și scop, de la biblioteci locale la biblioteci universitare sau naționale. În general, bibliotecile îsi organizeaza colecțiile în funcție de subiect, autor sau editura.

**Utilitatea bazei de date a bibliotecii**

O baza de date a unei biblioteci implica utilitati precum:

* Gestionarea, în principal, a cărților sau a altor materiale media disponibile în biblioteca, informații despre autori, edituri sau alte detalii.
* Imprumutarea și returnarea materialelor media. Baza de date trebuie sa permita gestionarea împrumuturilor și întoarcerilor de cărți și alte materiale media.
* Urmărirea utilizatorilor. Baza de date trebuie sa permita urmarirea utilizatorilor bibliotecii , anume date legate de obiectele împrumutate, și a datelor lor de contact.
* Interogari si rapoarte. Baza de date trebuie sa permita generarea de interogari si rapoarte pentru a ajuta administrarea bibliotecii.
* Gestionarea stocului. Baza de date trebuie sa permita gestionarea stocului de cărți și a altor materiale media din biblioteca.

## **2.Regulile generale de funcționare ale bazei de date a unei biblioteci**

Baza de date a bibliotecii implementate contine în total 10 tabele. Acestea permit:

* Stocarea eficientă a mai multe cărți :
  + fiecare carte are autori și categorii
  + fiecare carte are diferite exemplare diferite de la diferite edituri și din ani diferiți
* In biblioteca se pot realiza împrumuturi:
  + fiecare exemplar poate fi dat în mai multe împrumuturi la intervale de timp ce nu se suprapun
  + un împrumut contine un singur exemplar
  + toate datele despre împrumuturile efectuate de-a lungul timpului sunt stocate într-un tabel
* Biblioteca are bibliotecari prin care se realizeaza fiecare împrumut
* Biblioteca stochează date despre cititorii care o frecventează

## **3.Entități**

* **Carte**

Tabel folosit pentru descrierea informațiilor generale a unei cărți(autor,categorie).

**PK**- id\_carte

* **Autor .**

Tabel folosit pentru stocarea autorilor cărților existente din biblioteca.

**PK**- id\_autor

* **Categorie\_carte**

Tabel folosit pentru stocarea categoriilor cărților.

**PK**- id\_categorie

* **Exemplar\_carte**

Tabel ce reprezinta copile fizice ale unei carti. Acest tabel reține informațiile specifice ale unei cărți(an aparitie, editura).

**PK**- id\_exemplar

* **Editura**

Tabel folosit pentru stocarea editurilor.

**PK**- id\_editura

* **Cititor**

Tabel folosit pentru stocarea datelor celor ce împrumuta cărți.

**PK**- id\_cititor

* **Info\_imprumut**

Tabel folosit pentru stocarea datelor unui posibil împrumut.

**PK**- id\_imprumut

* **Bibliotecar**

Tabel folosit pentru stocarea datelor unui bibliotecar ce face posibil un împrumut.

**PK**- id\_bibliotecar

**Tabele asociative**

Avand prezente mai multe relații de tip many-to-many, o sa fie necesar ca in diagrama conceptuala sa apara tabelele asociative. Acestea sunt:

* **Carte\_categorie**

Tabel asociativ ce leaga cartea de categorie

**PK**- id\_carte\_categorie

* **Carte\_autori**

Tabel asociativ ce leaga cartea de autor

**PK**-id\_carte\_autori

* **Carte\_cititori**

**PK**- id\_carte\_cititori

Tabel asociativ ce lega cartea de cititor

## **4.Relații**

* + **One to many** cu exemplar\_carte.

**Cardinalitate**:

* + - O carte poate sa aiba mai multe copii fizice
    - O carte trebuie sa aiba cel putin o copie fizica.
    - Un exemplar poate sa aiba doar o carte.
    - Un exemplar trebuie sa aiba doar o carte.
  + **Many-to-many** cu categorie.

**Cardinalitate**:

* + - O carte poate sa aiba mai multe categorii.
    - O carte trebuie sa aibă măcar o categorie.
    - O categorie poate sa corespunda mai multor carti.
    - O categorie trebuie sa corespunda macar unei carti.
  + **Many-to-many** cu autor.

**Cardinalitate**:

* + - O carte poate sa aiba mai mulți autori care au contribuit la scrierea sa.
    - O carte trebuie sa aibă măcar un autor.
    - Un autor poate sa aiba mai multe cărți scrise.
    - Un autor trebuie sa aibă măcar o carte scrisă.
* **Exemplar\_carte**
  + **one-to-many** cu info\_imprumut

**Cardinalitate**:

* + - Un exemplar de carte poate sa fi trecut prin mai multe imprumuturi.
    - Un exemplar de carte nu trebuie sa treacă printr-un imprumut.
    - Un împrumut poate sa aiba doar un exemplar de carte.
    - Un împrumut trebuie sa aiba doar o carte.
  + **many-to-one** cu editura

**Cardinalitate**:

* + - Un exemplar de carte poate sa aiba doar o editura.
    - Un exemplar de carte trebuie să aibă cel puțin o editura.
    - O editură poate sa aiba mai multe exemplare de cărți.
    - O editură trebuie sa aibă măcar un exemplar de carte.
* **Cititor**
  + **one-to-many cu imprumut\_info**

**Cardinalitate:**

* + - Un cititor poate sa aiba mai multe imprumuturi.
    - Un cititor inregistrat trebuie sa aiba macar un imprumut.
    - Un imprumut poate sa aiba un cititor.
    - Un imprumut trebuie sa aiba un cititor.

* **info\_imprumut**
  + **many-to-one** cu bibliotecar

**Cardinalitate**

* + - Un imprumut poate fi efectuat prin intermediul unui bibliotecar.
    - Un imprumut trebuie sa fie facut prin intermediul unui bibliotecar.
    - Un bibliotecar poate sa efectueze imprumutul a mai multi cititori.
    - Un bibliotecar trebuie sa efectueze imprumutul a cel putin un cititor.

## **5.Atribute**

**Tabele principale:**

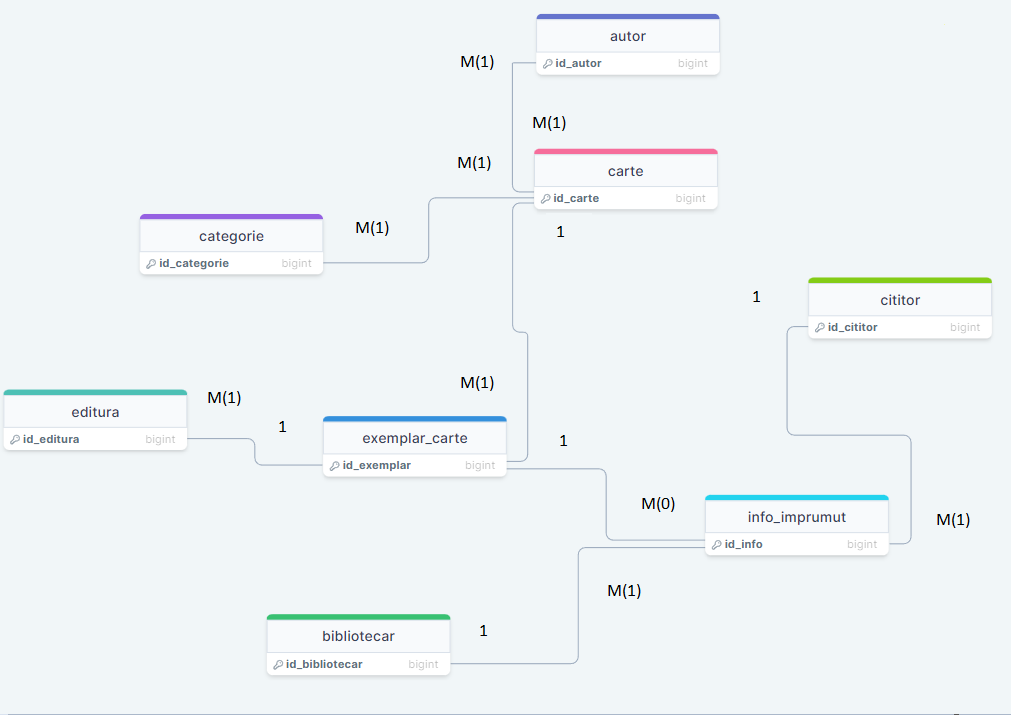
* **carte**
  + **PK id\_carte -** number(5), not null
  + **titlu\_carte -** varchar2(50)
* **autor**
  + **PK id\_autor -** number(4), not null
  + **nume\_autor -** varchar2(20) not null
  + **prenume\_autor -** varchar2(30)
    - **unique (nume\_autor, prenume\_autor)**
* **categorie**
  + **PK id\_categorie -** number(3), not null
  + **nume\_categorie -** varchar2(35)
* **exemplar\_carte**
  + **PK id\_exemplar -** number(6), not null
  + **FK id\_carte -** number(5), not null
  + **FK id\_editura -** number(3), on delete set null, on update cascade
  + **an\_publicare -** number(4)
* **editura**
  + **PK id\_editura -** number(3), not null
  + **nume\_editura -** varchar2(30)
* **info\_imprumut**
  + **PK id\_imprumut -** number(7), not null
  + **FK id\_exemplar -** number(6), not null
  + **FK id\_cititor -** number(6), not null
  + **FK id\_blibliotecar -** number(3), on delete set null
  + **data\_imprumut -** date ,not null , default sysdate
  + **data\_retur -** date , not null
* **bibliotecar**
  + **PK id\_bibliotecar -** number(3), not null
  + **nume\_bibliotecar -** varchar2(20) not null
  + **prenume\_bibliotecar -** varchar2(30)
  + **numar\_telefon\_bibliotecar -** char(10)
  + **email\_bibliotecar -** varchar2(50)
    - **unique(nume\_bibliotecar, prenume\_bibliotecar)**
* **cititor**
  + **PK id\_cititor -** number(6), not null
  + **nume\_cititor -** varchar2(20) not null
  + **prenume\_cititor -** varchar2(30)
  + **numar\_telefon\_cititor -** char(10)
  + **email\_cititor -** varchar2(50)
    - **unique(nume\_cititor, prenume\_cititor)**

**Tabele asociative:**

* **carte\_autor**
  + **PK id\_carte\_autor-** number(6), not null
  + **FK id\_carte-** number(5), not null, on delete cascade, on update cascade
  + **FK id\_autor-** number(4), not null, on delete cascade, on update cascade
    - **unique(id\_carte, id\_autor)**
* **carte\_categorie**
  + **PK id\_carte\_categorie-** number(6), not null
  + **FK id\_carte-** number(5), not null, on delete cascade, on update cascade
  + **FK id\_categorie-** number(3), not null, on delete cascade, on update cascade
    - **unique(id\_carte, id\_autor)**

## 

## **6.Diagrama entitate-relatie**



## **7.Diagrama conceptuala**

## **8.Scheme relationale**

* carte(id\_carte#, titlu\_carte, id\_exemplar)
* carte\_autor(id\_carte\_autor#,id\_carte,id\_autor)
* autor(id\_autor#, nume\_autor, prenume\_autor)
* carte\_categorie(id\_carte\_categorie#,id\_carte, id\_categorie)
* categorie(id\_categorie#, nume\_categorie)
* exemplar\_carte(id\_exemplar#,id\_carte,id\_editura, an\_publicare)
* editura(id\_editura#, nume\_editura)
* info\_imprumut(id\_imprumut#, id\_exemplar, id\_cititor, id\_bibliotecar, data\_imprumut, data\_retur, incheiere\_retur)
* bibliotecar(id\_bibliotecar#, nume\_bibliotecar, prenume\_bibliotecar, numar\_telefon, email)
* cititor(id\_cititor#, nume\_cititor, prenume\_cititor, numar\_telefon, email)

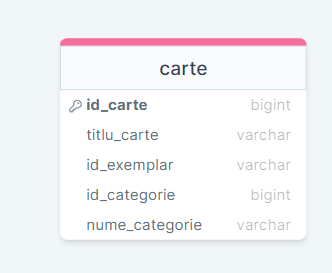
## **9.Normalizari FN1, FN2, FN3**

Diagrama propusă este deja formalizată până în FN3.

* Aceasta se regaseste in FN1 deoarece toate atributele au valori atomice. Acestea nu mai pot sa fie divizate.

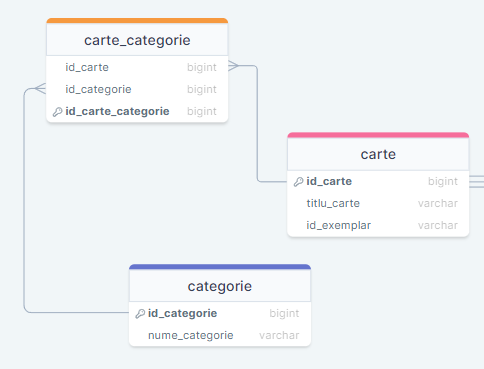
Baza de date nu ar fi respectat aceasta proprietate daca unele atribute dintr-un tabel ar fi trebuit sa aiba mai multe valori.

De exemplu, dacă tabela cărți at fi conținut și datele despre categorie:



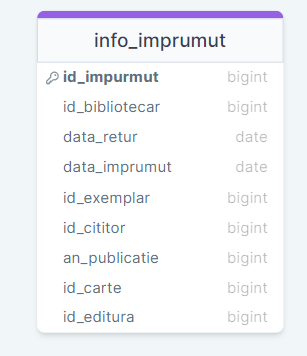
O carte poate sa aiba mai multe categorii, iar o categorie corespunde mai multor cărți. Așadar, exista o relație many-to-many, ceea ce duce la nerespectarea proprietății FN1.

După normalizare:



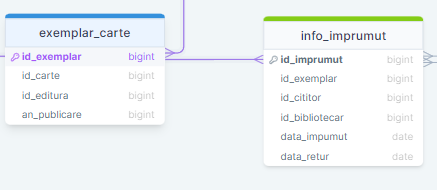
* Aceasta se regaseste in FN2 deoarece se regaseste in FN1 și fiecare atribut depinde functional de cheia primara.

De exemplu, dacă info\_imprumut ar fi conținut datele din carte\_exemplar:



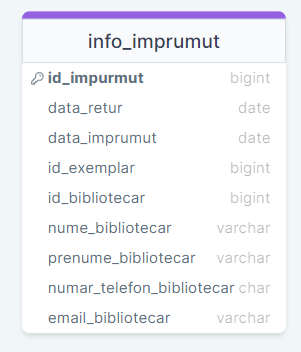
Atributele an\_publicatie, id\_carte, id\_editura si id\_exemplar nu depind functional de id\_imprumut, deci astfel nu s-ar respecta FN2.

După normalizare:



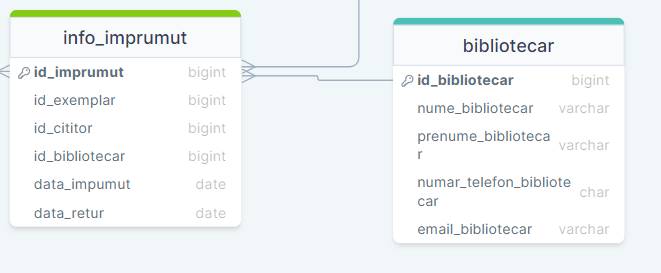
* Aceasta se regaseste in FN3 deoarece se regaseste in FN2 și nu exista atribute care sa depinda indirect una fata de cealalta.

De exemplu, dacă info\_imprumut ar fi conținut datele din bibliotecar:

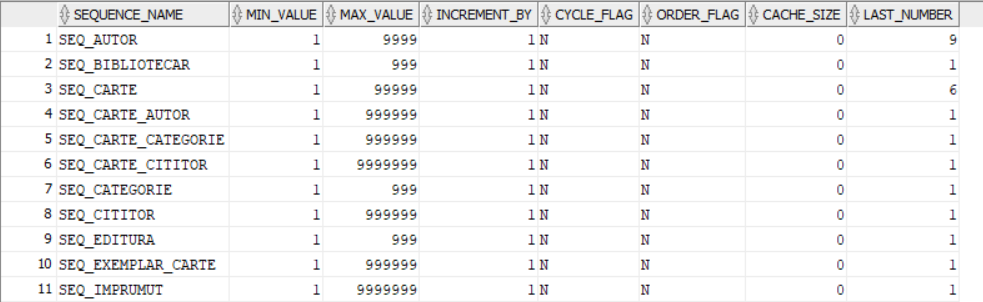


Cum atributele id\_bibliotecar, nume\_bibliotecar, prenume\_bibliotecar, numar\_telefon\_bibliotecar, email\_bibliotecar depind indirect una fata de cealalta, acest tabel nu respecta FN3.

După normalizare:

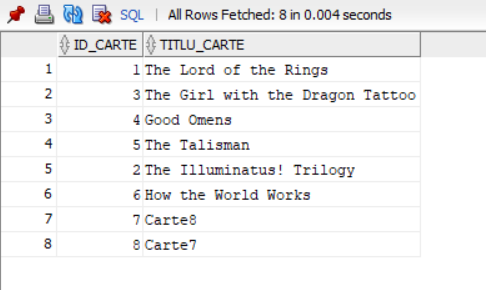


## **10.Secvențe**

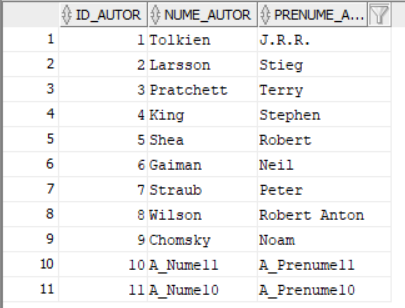


## **11.Creare tabele**

### **Carte**



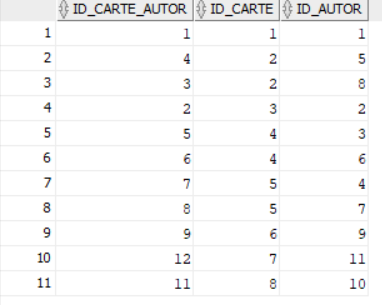
### **Autori**



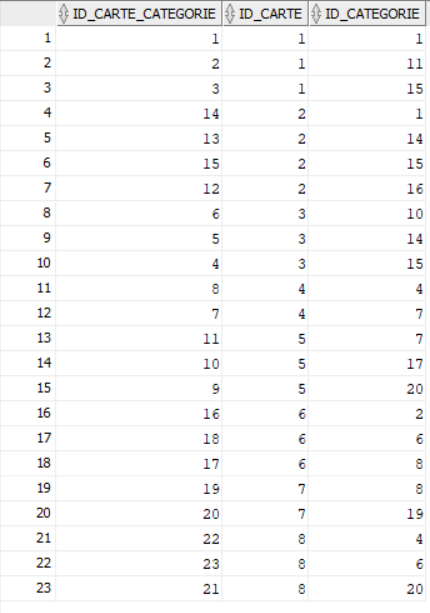
### **Categorie**



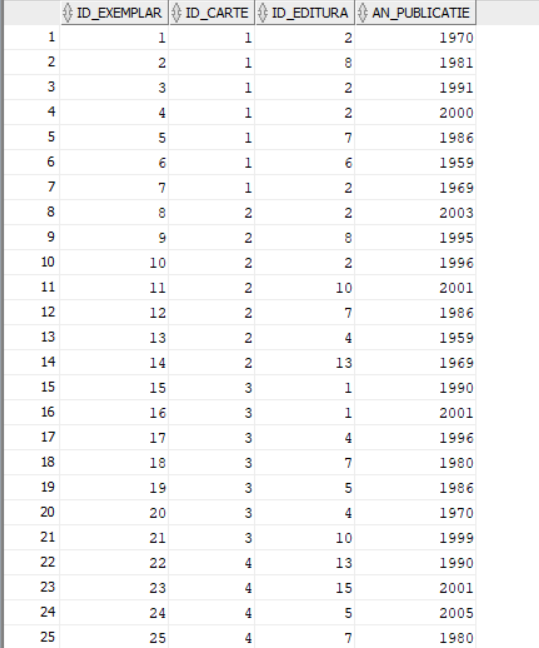
### **Carte\_autor**

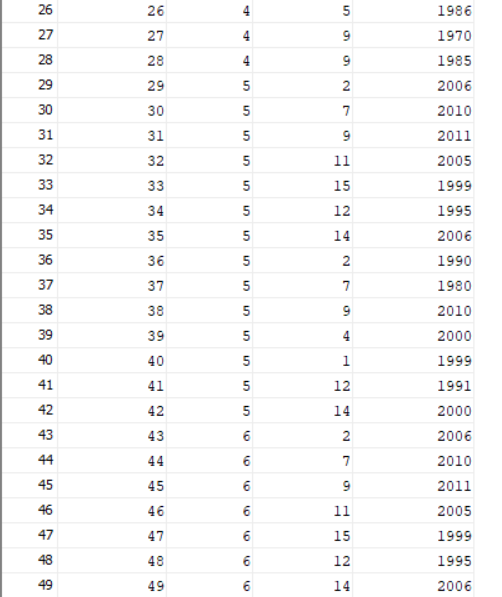


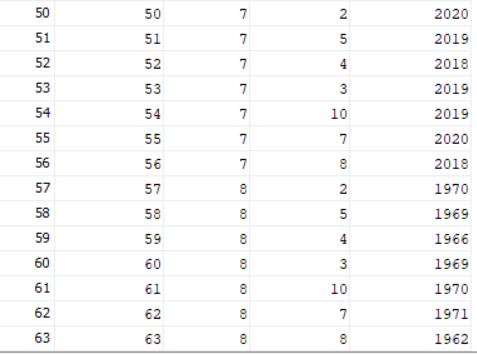
### **Carte\_categorie**



### **Exemplar\_carte**



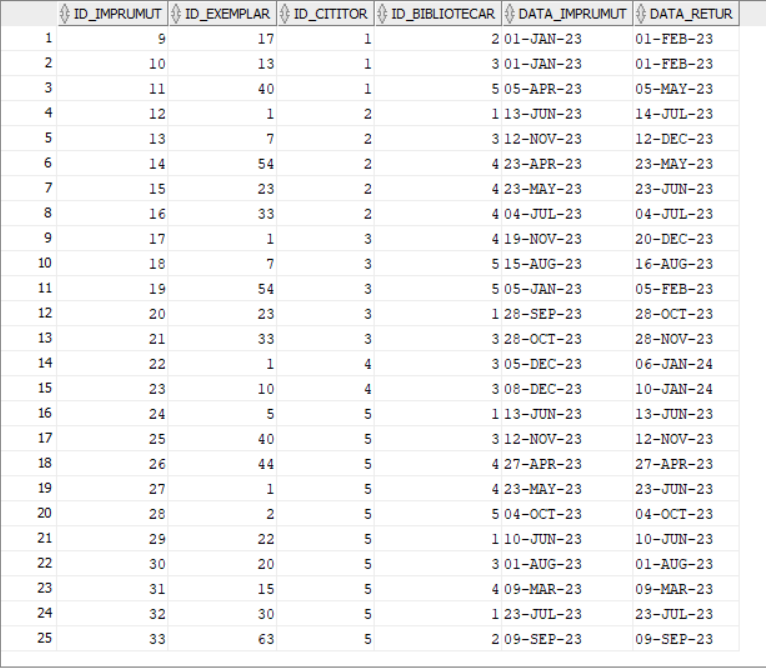




### **Editura**



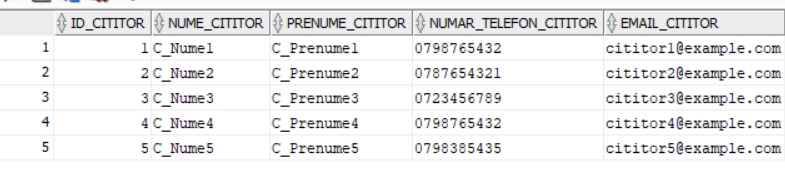
### **Info\_imprumut**



### **Bibliotecar**



### **Cititor**

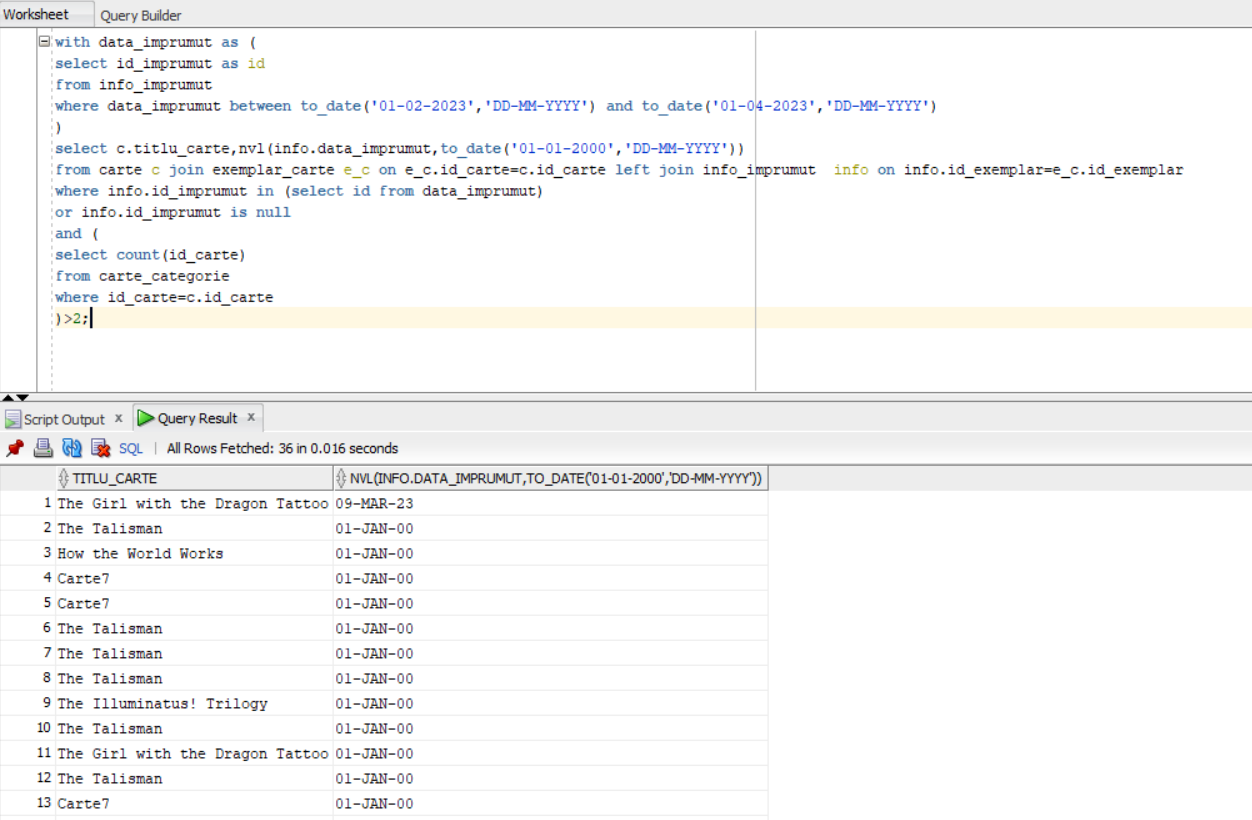


## **12.Cereri**

1. Sa se afiseze titlul tuturor exemplarelor de cărți care au mai mult de 2 categorii și care au fost împrumutate între 01-02-2023 și 01-04-2023. Dacă cartile nu au fost împrumutate, se va afișa data 01-01-2000

Se folosesc:

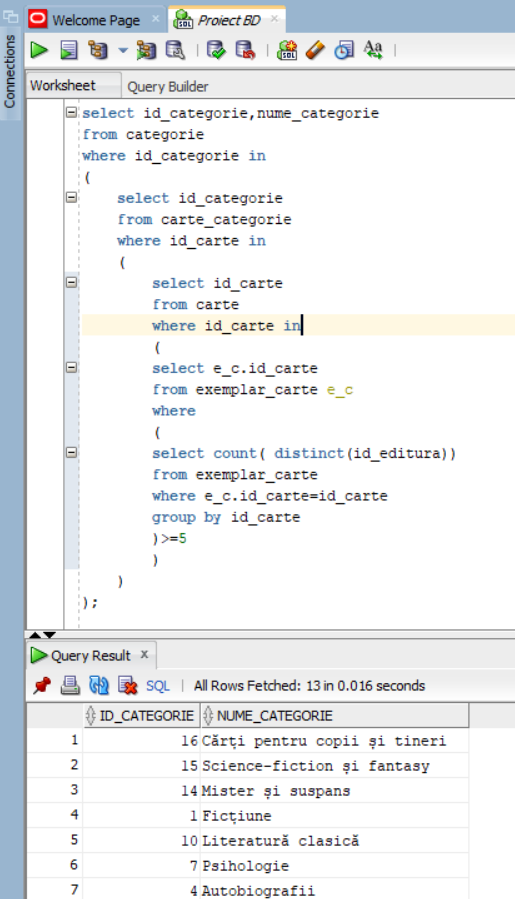
* 1 bloc de cerere (clauza with)
* 3 functii pe date calendaristice (to\_date)



2. Sa se afiseze categoriile care corespund unor cărți care au mai mult de 5 exemplare în biblioteca.

Se folosesc:

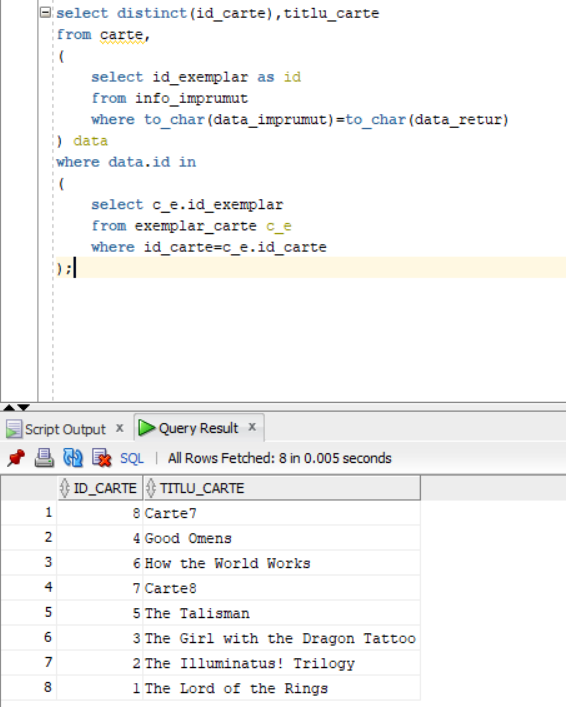
* grupări de date cu subcereri nesincronizate în care intervin cel puțin 3 tabele, funcții grup, filtrare la nivel de grupuri



3. Un cititor are data de împrumut egala cu cea de retur dacă și numai dacă acesta studiaza la biblioteca. Sa se afiseze id-ul și titlul cărților folosite în acest scop.

Se folosesc:

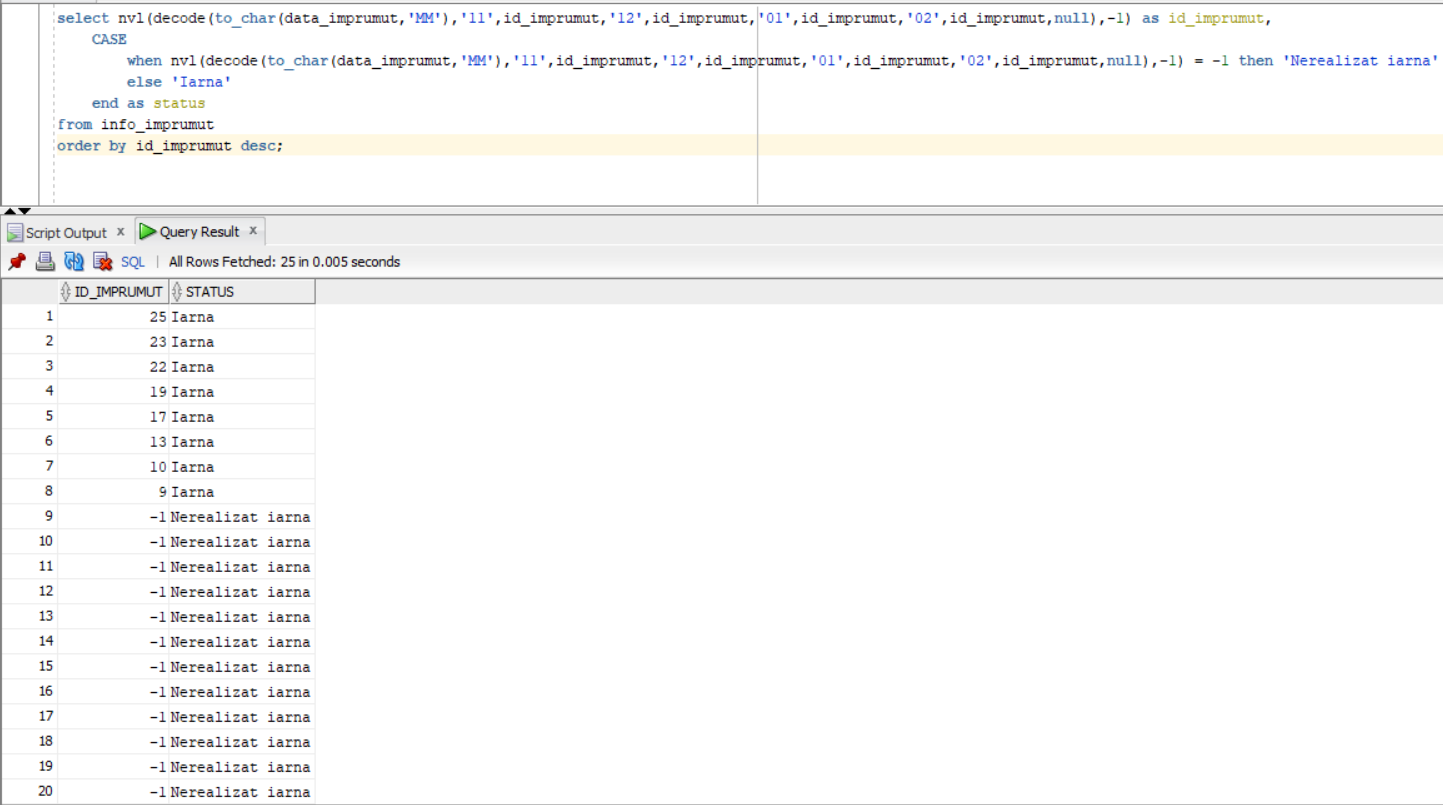
* subcereri nesincronizate în clauza FROM
* 2 funcții pe șiruri de caractere (to\_char)



4. Sa se afiseze id-ul tuturor împrumuturilor realizate in noiembrie, decembrie, ianuarie si februarie sau -1 dacă împrumutul respectiv este realizat in alta luna. Pentru fiecare sa se afiseze mesajul 'iarna' dacă îndeplineste conditia sau 'Nerealizat iarna' în caz contrar. Ordonați descrescător după id-ul afișat.

Se folosesc:

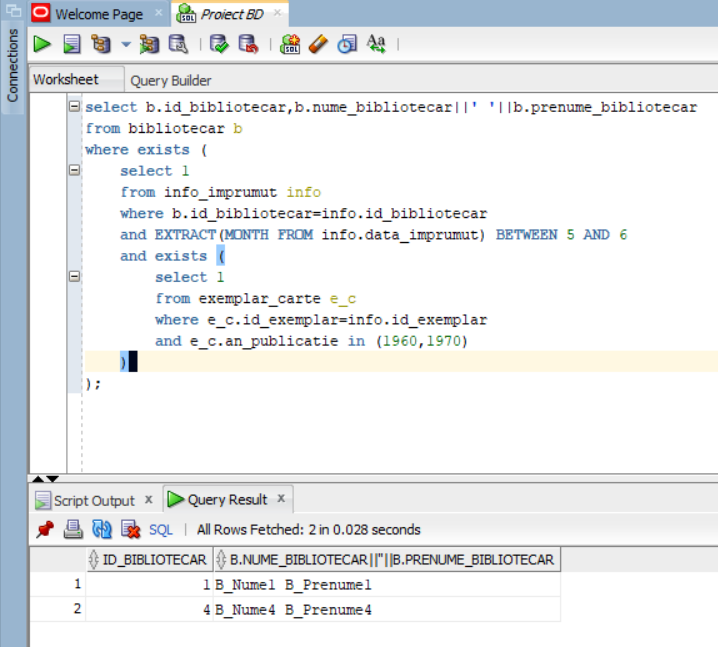
* ordonări și utilizarea funcțiilor NVL și DECODE
* 2 funcții pe șiruri de caractere (to\_char)
* o expresie CASE



5. Sa se afiseze date despre bibliotecarii ce au dat în împrumut între iunie și iulie cărți publicate între anii 1960 1970.

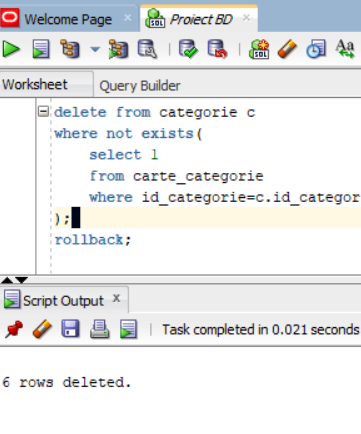
Se folosesc:

* subcereri sincronizate în care intervin cel puțin 3 tabele

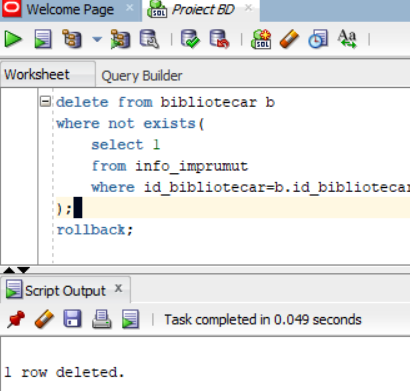


## **13.Actualizare și suprimare**

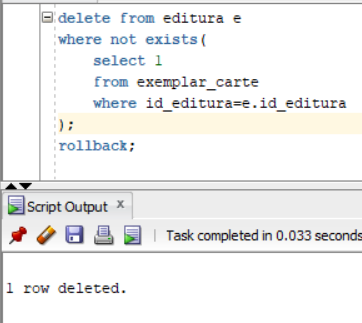
1. Sa se șteargă categoriile care nu corespund nici unei cărți



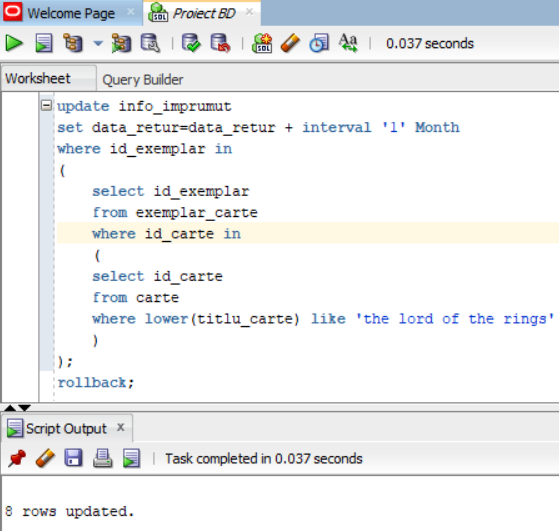
1. Sa se stearga bibliotecarii care nu au efectuat niciun imprumut



1. Sa se stearga editurile care nu corespund niciunui exemplar



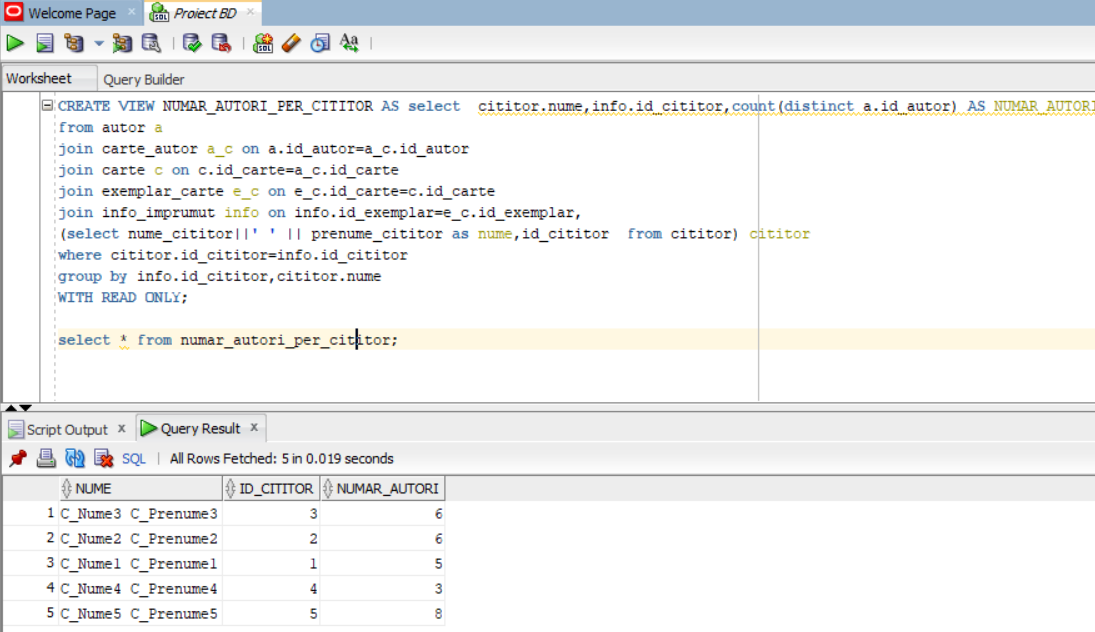
1. Sa se adauge o luna la data de retur tuturor împrumuturilor cărții 'Lord of the Rings'



## **14.View**

Numarul de autori pentru cartile imprumutate de-a lungul timpului pentru toti cititorii

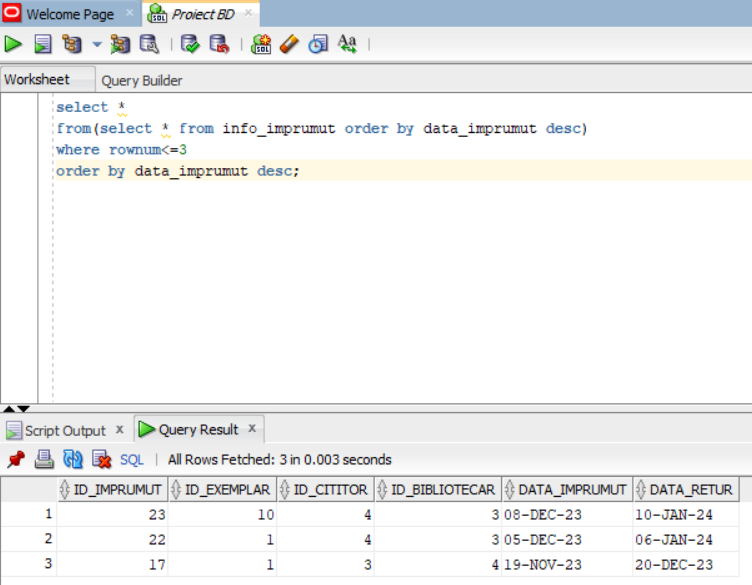
* Singura operatie LMD care se poate folosi este select
* restul de delete,update,insert sunt nepermise



## **15. Outer join , division , analiza top-n**

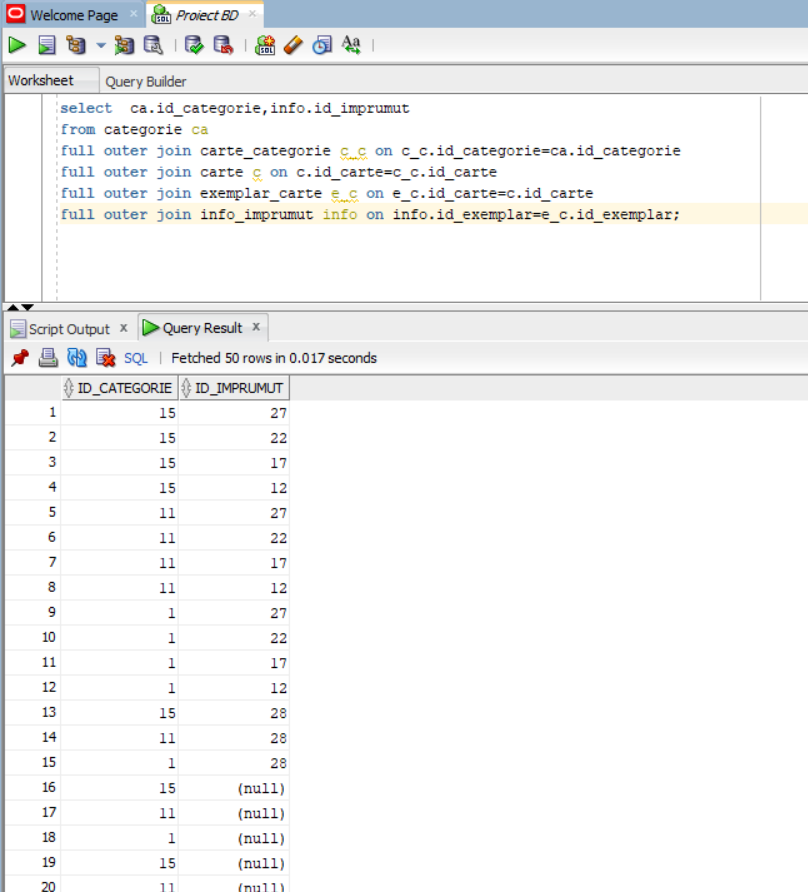
**Analiza top-n**

Sa se afiseze primele 3 cele mai recente imprumuturi



**Outer join**

Sa se afiseze pentru fiecare exemplar de carte, categoria sa si imprumuturile din care a facut parte. Daca nu a fost imprumutata o sa se afiseze null



**Division(varianta count)**

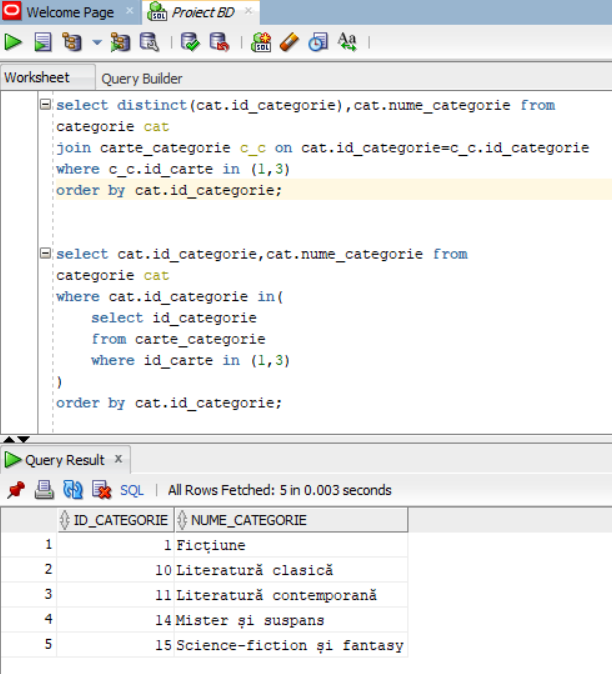
Sa se obtina codurile cartilor ce au numai categoriile ce incep cu 'c'.

## 

## **16. Comparare plan de executie a 2 cereri echivalente semantic**

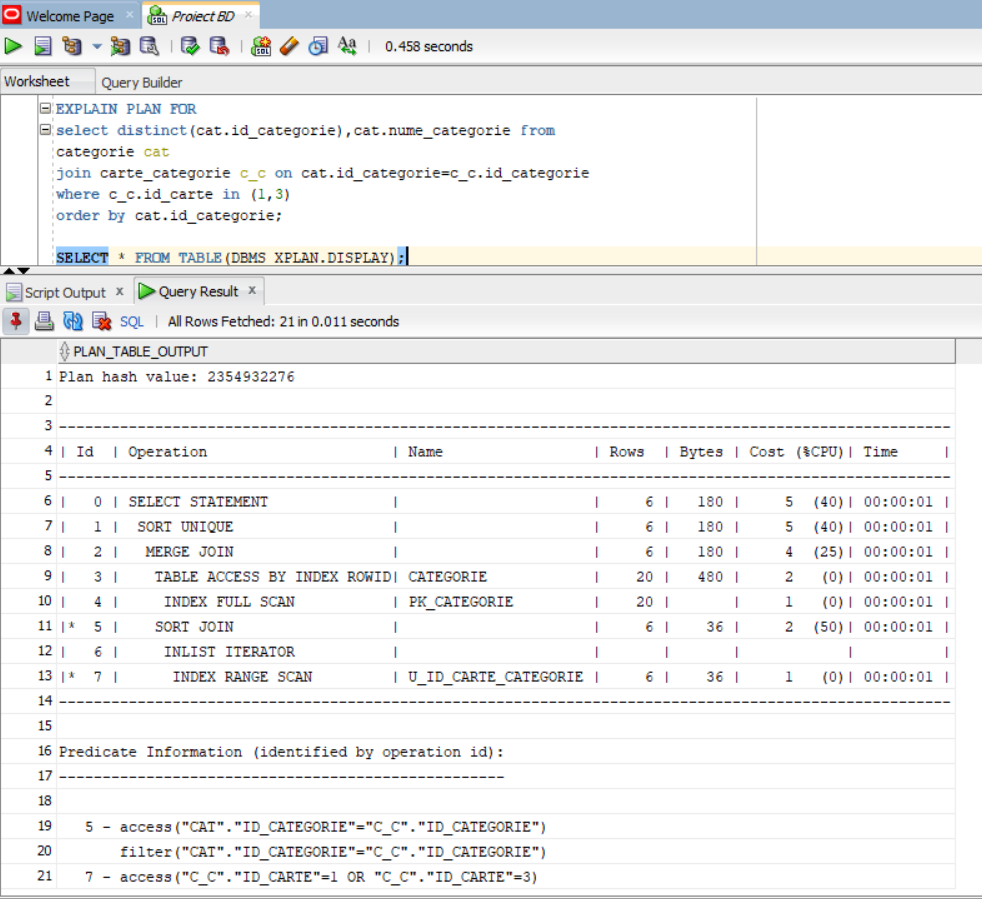
**Cererile propriu-zise**

Sa se afișeze toate datele despre categoriile pentru cărțile cu id-ul egal cu 1 sau 3.



Cele 2 cereri sunt echivalente semantic, rezultatele acestora sunt mereu aceeleasi.

**Planurile de execuție ale cererilor**



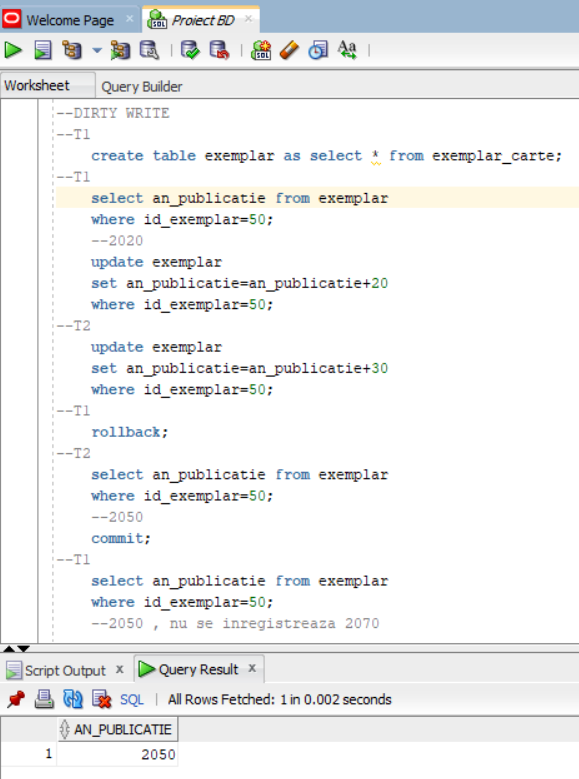
## 

După cum se observa planul de execuție al cererilor diferă, desi cele doua sunt echivalente semantic. Ambele planuri implica accesul la tabelul “categorie”, apoi se face join cu tabelul “carte\_categorie” folosindu-se de id\_carte\_categorie. Diferenta principala dintre cele doua cereri este modul în care procesează înregistrările din tabelul “categorie”.

Se observa diferenta intre join-uri: prima cerere folosește “merge join” , iar a doua folosește “merge join semi”. De asemenea prima cerere este mai ineficienta deoarece acesta trebuie sa selecteze elementele unice, ceea ce implica în folosirea mai multor randuri și a mai mult spatiu din memorie.

## **18.Ilustrarea a doua consistency levels**

**Dirty Write**



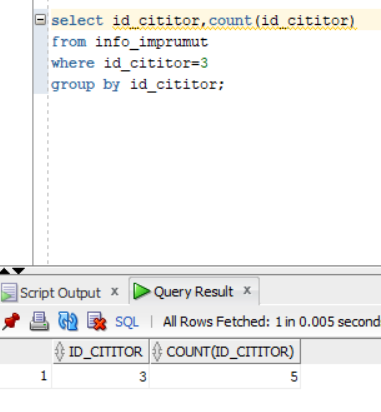
**Lost Update**



## **19. Optimizarea cererilor folosind indexare**

1. Sa se obtina cate împrumuturi s-au realizat pentru cititorul cu id-ul egal cu 3.

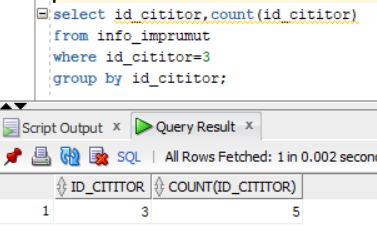
Înainte de a crea indexul:



Se creeaza un index pentru tabelul info\_imprumut ce marchează id\_cititor:

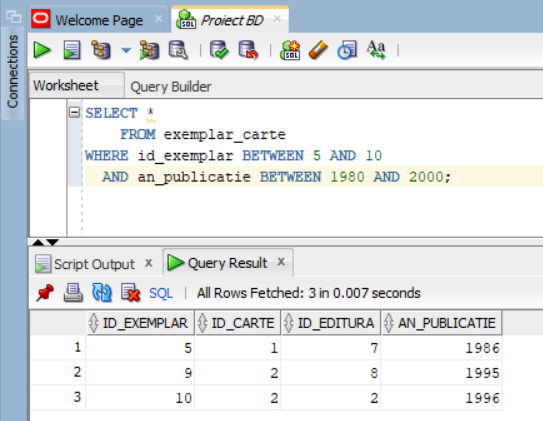


Cererea după crearea indexului:

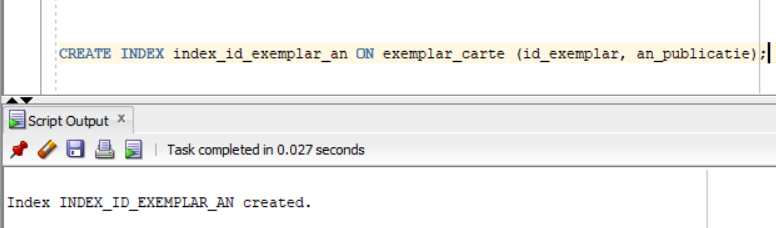
****

1. Sa se afiseze datele exemplarelor de cărți cu id-ul între 5 și 10 publicate între anii 1980 și 2000

Înainte de a crea indexul:

****

Se creeaza un index pentru tabelul info\_imprumut ce marchează id\_cititor:

****

Cererea după crearea indexului:

