Gestionarea datelor unor biblioteci

În acest proiect, modelul de date va gestiona informați legate de organizarea si funcționarea unor biblioteci. Fiecare bibliotecă oferă accesul la cărțile ei, prin intermediul împrumutului, fiecărui membru al ei. Membri unei biblioteci sunt nevoiți sa dețină un permis de intrare pentru fiecare bibliotecă la care sunt înscriși.

Fiecare bibliotecă poate face o comandă, de la unul din furnizori care lucrează cu o editură. O editură poate lucra cu mai mulți furnizori.

Cărțile sunt catalogate după tipuri (ex: Cărți de poezii, Cărți pentru copii, Cărți științifice, etc), iar fiecare dintre aceste cărți se impart în subtipuri (ex: Cărți științifice despre biologie, matmatică, informatică etc).

Acest model este util deoarece ține evidența cărților împrumutate, comenzilor, membrilor și catalogarea cărților.

Modelul de date respectă anumite restricți de funcționare:

- Fiecare carte trebuie să aibă o editură.

- O carte are un singur autor.

- Un membru al biblioteci are un singur permis de intrare pentru o bibliotecă.

- Un membru nu poate împrumuta o carte fără permisul de intrare specific biblioteci din care dorește să împrumute.

- Un membru nu poate împrumuta aceeași carte de mai multe ori în aceeași zi.

- Un furnizor și o editură se află la o singură locație.

- Un angajat poate sa conducă o singură bibliotecă.

- Toate adresele se află în România.

- O carte poate avea un tip și mai multe subtipuri.

- Presupunem că nu există autori cu același tuplu de nume și prenume

**Entități**

Pentru modelul de date referitor la gestiunea unor biblioteci, structurile ANGAJATI, BIBLIOTECI, FUNCTII, SUBTIPURI, AUTORI, CARTI, ADRESE, PERMIS\_DE\_INTRARE, COMENZI, FURNIZORI, EDITURI, CITITORI reprezintă entități.

ANGAJATI = persoane care se ocupă cu funcționarea propiu-zisă a unei biblioteci. Cheia primară este *angajat\_id.*

FUNCTII = funcțiile pe care le pot avea angajații unei biblioteci. Cheia primară este *functie\_id.*

BIBLIOTECI = clădirea special amenajată pentru stocarea cărților. Aceasta poate oferi posibilitatea membrilor săi de a împrumuta cărțile stocate in ea. Cheia primară este *biblioteca\_id.*

ADRESE = date care indică localizarea pe hartă a unei persoane sau instituții. Pentru acest model vom considera că toate adresele se află în România. Cheia primară este *adresa\_id.*

AUTORI = persoană care a scris o carte. O carte poate avea un singur autor. Cheia primară este *autor\_id.*

CARTI = scriere cu un anumit subiect, tipărită și legată sau broșată în volum. Cheia primară este *carte\_id.*

CITITORI = persoană care citește o carte. Cheia primară este *cititor\_id.*

PERMIS\_DE\_INTRARE = document cu scopul de a permite accesul la cărțile aflate într-o bibliotecă. Cheia primară este *permis\_id.*

EDITURI = este o societate comercială care își joacă rolul pe piața cărților. Cheia primară este *editura\_id.*

FURNIZORI = companie care se ocupa cu distribuirea și producerea unor bunuri. Cheia primară este *furnizor\_id.*

COMENZI = acțiunea de a cumpăra produse de la un furnizor. Cheia primară este *comanda\_is.*

SUBTIPURI = subtipul unei cărți. Cheia primară este *subtip\_id.*

**Relații**

ANGAJATI\_are\_FUNCTII = relația care leagă entitățile ANGAJATI și FUNCTII reflectând legătura dintre acestea (ce funcție are un angajat). Aceasta are cardinalitatea maxima 1:n (un angajat poate avea o singură funcție și mai mulți angajați pot avea aceeași funcție) și cardinalitatea minimă este 0:1(Un angajat trebuie să aibă o functie și un angajat nu trebuie să aibă o anume funcție).

ANGAJATI\_conduce\_BIBLIOTECI = relația care leagă entitățile ANGAJAT și BIBLIOTECI refletând legătura dintre acestea (ce angajat conduce o bibliotecă). Relația maximă este 1:1 (un angajat poate să conducă o bibliotecă si o bibliotecă poate fi condusă de un angajat) și relația minimă 0:0 (un angajat nu trebuie să conducă o bibliotecă și o bibliotecă nu trebuie să fie condusă de un angajat).

ANGAJATI\_lucreaza\_BIBLIOTECI = relația care leagă entitățile ANGAJATI și BIBLIOTECI refletând legătura dintre acestea (ce angajați lucrează într-o bibliotecă). Relația maximă este 1:n (un angajat poate lucra într-o bibliotecă și într-o bibliotecă pot lucra mai mulți angajați) și relația minimă 0:1 (un angajat trebuie să lucreze într-o bibliotecă și într-o bibliotecă nu trebuie să lucreze un angajat).

ANGAJATI\_locuiesc\_ADRESE = relația care leagă entitățile ANGAJATI și ADRESE reflectând legătura dintre acestea (unde locuiesc angajați). Relația maximă este m:n (angajați pot locui la mai multe adrese și la o adresă pot locui mai mulți angajați) și relația minimă este 0:1(un angajat trebuie sa locuiască la o adresă și la o adresă nu trebuie să locuiască un angajat).

PERMIS\_DE\_INTRARE\_apartine\_BIBLIOTECI = relația care leagă entitățile BIBLIOTECI și PERMIS\_DE\_INTRARE reflectând legătura dintre acestea (de ce bibliotecă aparține permisul de intrare). Relația maximă este 1:n (un permis de intrare poate aparține de o singură bibliotecă și de o bibliotecă pot aparține mai multe permisuri de intrare) și relația minimă este 0:1 (un permis de intrare trebuie să aparțina de o bibliotecă și de o bibliotecă nu trebuie să aparțină un permis de intrare).

BIBLIOTECI\_se\_afla\_ADRESE = relația care leagă entitățile BIBLIOTECI și ADRESE reflectând legătura dintre acestea (unde se află o bibliotecă). Relația maximă este 1:n (mai multe biblioteci se pot afla la o adresă și o bibliotecă se poate afla la o adresă) și relația minimă 0:1 (o bibliotecă trebuie să se afle la o adresă și la o adresă nu trebuie să se afle o bibliotecă).

BIBLIOTECI\_face\_COMENZI = relația care leagă entitățile BIBLIOTECI și COMENZI reflectănd comenzile făcute de o bibliotecă. Relația maximă este 1:n (o bibliotecă poate face mai multe comenzi și o comandă poate să fie făcută de o bibliotecă) și relația minimă este 0:1 (o bibliotecă nu trebuie să facă o comandă și o comandă trebuie să fie făcută de o bibliotecă).

PERMIS\_DE\_INTRARE\_apartine\_CITITORI = relația de tip 3 care leagă entitățile PERMIS\_DE\_INTRARE și CITIORI reflectând permisurile de intrare ale cititorilor. Relația maximă este 1:n (un permis poate aparține unui cititor și un cititor poate avea mai multe permisuri) și relația minimă este 0:1 (un permis de intrare trebuie să aparțină unui cititor și un cititor nu trebuie să aibă un anume permis).

CITITORI\_imprumuta\_CARTI\_de\_la\_BIBLIOTECI\_cu\_un\_PERMIS\_DE\_INTRARE = relație de tip 3 care leagă entitățile PERMIS\_DE\_INTRARE, CITITORI, BIBLIOTECI și CARTI reflectând cărțile împrumutate de un cititor de la o biblioteca folosind un permis.

CITITORI\_locuiesc\_ADRESE = relație care leagă entitățile CITITORI și ADRESE reflectănd adresele cititorilor. Relația maximă este m:n (un cititor se poate afla la mai multe adrese și la o adresă se pot afla mai mulți cititori) și relația minimă 0:1 (la o adresă nu trebuie să locuiască un cititor și un cititor trebuie să locuiască la o adresă).

EDITURI\_se\_afla\_ADRESE = relație care leagă entitățile EDITURI și ADRESE reflectănd locația unde se află o editură. Relația maximă este 1:1 (o editura se poate afla la o adresă și la o adresă se poate afla o editură) și relația minimă 0:1 (o editură trebuie să se afle la o adresă și la o adresă nu trebuie să se afle o editură).

FURNIZORI\_se\_afla\_ADRESE = relație care leagă entitățile FURNIZORI și ADRESE reflectănd locația unde se află un furnizor. Relația maximă este 1:1 (un furnizor se poate afla la o adresă și la o adresă se poate afla un furnizor) și relația minimă 0:1 (un furnizor trebuie să se afle la o adresă și la o adresă nu trebuie să se afle un furnizor).

EDITURI\_lucreaza\_cu\_FURNIZORI = relația care leagă entitățile EDITURI și FURNIZORI reflectănd legătura dintre acestea (ce edituri lucreaza cu ce furnizori). Relația maximă este m:n (o editură poate lucra cu mai mulți furnizori și un furnizor poate lucra cu mai multe edituri) și relația minimă este 0:1 (o editură trebuie să lucreze cu un furnizor și un furnizor nu trebuie să lucreze cu o editură).

FURNIZORI\_se\_afla\_COMENZI = relația care leagă entitățile FURNIZORI și COMENZI reflectănd legătura dintre acestea (în ce comenzi se află un furnizor). Relația maximă este 1:n (un furnizor se poate afla în mai multe comenzi și într-o comandă se poate afla un furnizor) și relația minimă 0:1 (într-o comandă trebuie să se afle un furnizor și un furnizor nu trebuie să se afle într-o comandă).

CARTI\_se\_afla\_COMENZI = relația care leagă entitățile CARTI și FURNIZORI reflectând legătura dintre acestea (în ce comenzi se află o carte). Relația maximă este 1:n (într-o comandă se poate afla o carte și o carte se poate afla în mai multe comenzi) și relația minimă este 0:1 (într-o comandă trebuie să se afle o carte și o carte nu trebuie să se afle într-o comanda).

CARTI\_are\_EDITURI = relația care leagă entitățile CARTI și EDITURI reflectând legătura dintre acestea (ce editură are o carte). Relația maximă este 1:n (o carte poate avea o editură și o editură poate avea mai multe cărți) și relația minimă 0:1 (o carte trebuie să aibă o editură și o editură nu trebuie să aibă o carte).

AUTORI\_scris\_CARTI = relația care leagă CARTI și AUTORI reflectând legătura dintre acestea (ce autor a scris o carte). Relația maximă este 1:n (o carte poate fi scrisă de un autor și un autor poate scrie mai multe cărți) și relația minimă este 0:1 (o carte trebuie să fie scrisă de un autor și un autor nu trebuie să scrie o carte).

CARTI\_are\_SUBTIPURI = relația care leagă entitățile CARTI și SUBTIPURI reflectând legătura dintre acestea (ce subtipuri are o carte). Relația maximă este m:n (o carte poate avea mai multe subtipuri și mai multe cărți pot avea un subtip) și relația minimă 0:1 (o carte trebuie să aibă un subtip și o carte nu trebuie să aibă un anume subtip).

**Atribute**

Atributele entități ANGAJATI sunt: **angajat\_id** (variabilă de tip întreg, de lungime maximă 5 și este cheia primară a angajatului), **nume** (variabilă de tip caracter, de lungime maximă 25 și nu este nulă), **prenume** (variabilă de tip caracter, de lungime maximă 25 și nu este nulă), **cnp** (variabilă de tip caracter, de lungime maximă 13 și nu este nulă), **telefon** (variabilă de tip caracter, de lungime maximă 20 și nu este nulă), **email** (variabilă de tip caracter, de lungime maximă 320, și nu este nulă), **salariu** (variabilă de tip număr real, de lungime maximă 10 și nu este nulă), **data\_angajare** (variabilă de tip dată, care reprezintă data angajări angajatului, implicit este egală cu data curentă din sysdate), **functie\_id** (variabilă de tip întreg, de lungime maximă 5, nu este nulă, este cheia externă ce face legătura cu entitatea FUNCTII și reprezintă funcția pe care o are angajatul în bibliotecă), **biblioteca\_id** (variabilă de tip întreg, de lungime maximă 5, nu este nulă, este cheie externă ce face legătură cu entitatea BIBLIOTECI și reprezintă biblioteca în care lucrează angajatul).

Atributele entități FUNCTII sunt: **functie\_id** (variabilă de tip întreg, de lungime maximă 5 și este cheia primară a funcției), **denumire** (variabilă de tip caracter, de lungime maximă 30 și nu este nulă), **salariu\_minim** (variabilă de tip număr real, de lungime maximă 10, nu este nulă și este mai mic sau egal cu salariul maxim), **salariu\_maxim** (variabilă de tip număr real, de lungime maximă 10, nu este nulă și este mai mare sau egal cu salariul minim).

Atributele entități BIBLIOTECI sunt: **biblioteca\_id** (variabilă de tip întreg, de lungime maxima 5 și este cheia primară a biblioteci), **denumire** (variabilă de tip caracter de lungime maximă 30 și nu este nulă), **telefon** (variabilă de tip caracter, de lungime maximă 20, nu este nulă și reprezintă numărul de telefon la care poate fi contactată o bibliotecă), **manager\_id** (variabilă de tip întreg, de dimensiune maximă 5, este cheie externă ce face legătura cu entitatea ANGAJATI și reprezintă angajatul care se ocupă de condusul biblioteci), **adresa\_id** (este o variabilă de tip întreg, de lungime maximă 5, nu este nulă, este o cheie externă ce face legătura cu entitatea ADRESE și reprezintă adresa la care se află biblioteca).

Atributele entități ADRESE sunt: **adresa\_id** (variabilă de tip întreg, de lungime maximă 5 și este cheia primară a adresei), **judet** (variabilă de tip caracter, de lungime maximă 50 și nu este nulă), **localitate** (variabilă de tip caracter, de lungime maximă 50 și nu este nulă), **cod\_postal** (variabilă de tip caracter, de lungime maximă 6 și nu este nulă), **strada** (variabilă de tip caracter, de lungime maximă 50).

Atribute entități AUTORI sunt: **autor\_id** (variabilă de tip întreg, de lungime maximă 5 și este cheia primară a autorului), **nume** (variabilă de tip caracter, de lungime maximă 25 și nu este nulă), **prenume** (variabilă de tip caracter, de lungime maximă 25 și nu este nulă).

Atributele entități CARTI sunt: **carte\_id** (variabilă de tip întreg, de lungime maximă 5 și este cheia primară a cărți), **titlu** (variabilă de tip caracter, de lungime maximă 50 și nu este nulă), **editura\_id** (variabilă de tip întreg, de dimensiune maximă 5, nu este nulă, este cheie externă ce face legătura cu entitatea EDITURI și reprezintă editura cărți), **an\_aparitie** (variabilă de tip întreg, de lungime maximă 5 care reprezintă anul apariției cărți și nu este nulă), **autor\_id** (variabilă de tip întreg, de dimensiune maximă 5, nu este nulă, este cheie externă ce face legătura cu entitatea AUTORI și reprezintă autorul cărți), **tip\_carte** (variabilă de tip caracter, de lungime maximă 25 și nu este nulă).

Atributele entități CITITORI sunt: **cititor\_id** (variabilă de tip întreg, de lungime maximă 5, și este cheia primară a cititorului), **nume** (variabilă de tip caracter, de lungime maximă 25 și nu este nulă), **prenume** (variabilă de tip caracter, de lungime maximă 25 și nu este nulă), **cnp** (variabilă de tip caracter, de lungime maximă 13 și nu este nulă), **telefon** (variabilă de tip caracter, de lungime maximă 20 și nu este nulă), **email** (variabilă de tip caracter, de lungime maximă 320, și nu este nulă).

Atributele entități PERMIS\_DE\_INTRARE sunt: **permis\_id** (variabilă de tip întreg, de lungime maximă 5, și este cheia primară a permisului), **anul\_inscrieri** (variabilă de întreg, de lungime maximă 5 care reprezintă data înscrieri, implicit este egală cu data curentă din sysdate), **cititor\_id** (variabilă de tip întreg, de dimensiune maximă 5, nu este nulă, este cheie externă ce face legătura cu entitatea CITITORI și reprezintă deținătorul/deținătoarea permisului), **biblioteca\_id** (variabilă de tip întreg, de dimensiune maximă 5, nu este nulă, este cheie externă ce face legătura cu entitatea BIBLIOTECI și reprezintă biblioteca la care este valid permisul).

Atributele entități EDITURI sunt: **editura\_id** (variabilă de tip întreg, de lungime maximă 5, și este cheia primară a edituri), **denumire** (variabilă de tip caracter de lungime maximă 30 și nu este nulă), **telefon** (variabilă de tip caracter, de lungime maximă 20, nu este nulă și reprezintă numărul de telefon la care poate fi contactată o editură), **email** (variabilă de tip caracter, de lungime maximă 320, și nu este nulă), **adresa\_id** (este o variabilă de tip întreg, de lungime maximă 5, nu este nulă, este o cheie externă ce face legătura cu entitatea ADRESE și reprezintă adresa la care se află editura).

Atributele entități FURNIZORI sunt: **furnizor\_id** (variabilă de tip întreg, de lungime maximă 5, și este cheia primară a edituri), **denumire** (variabilă de tip caracter de lungime maximă 30 și nu este nulă), **telefon** (variabilă de tip caracter, de lungime maximă 20, nu este nulă și reprezintă numărul de telefon la care poate fi contactat un furnizor), **email** (variabilă de tip caracter, de lungime maximă 320, și nu este nulă), **adresa\_id** (este o variabilă de tip întreg, de lungime maximă 5, nu este nulă, este o cheie externă ce face legătura cu entitatea ADRESE și reprezintă adresa la care se află furnizorul).

Atributele entități COMENZI sunt: **comanda\_id** (variabilă de tip întreg, de lungime maximă 5, și este cheia primară a comenzi), **furnizor\_id** (variabilă de tip întreg, de dimensiune maximă 5, este cheie externă ce face legătura cu entitatea FURNIZORI și reprezintă furnizorul care se ocupă de comandă), **carte\_id** (variabilă de tip întreg, de dimensiune maximă 5, este cheie externă ce face legătura cu entitatea CARTI și reprezintă cartea care este comandată), **biblioteca\_id** (variabilă de tip întreg, de dimensiune maximă 5, este cheie externă ce face legătura cu entitatea BIBLIOTECI și reprezintă biblioteca care a făcut comanda), **nr\_exemplare** (variabilă de tip întreg, de dimensiune maximă 10, nu este nulă și este mai mare ca 0), **pret\_comanda** (variabilă de tip număr real, de lungime maximă 10 și nu este nulă).

Atributele entități SUBTIPURI sunt: **subtip\_id** (variabilă de tip întreg, de lungime maximă 5, și este cheia primară a subtipului), **denumire** (variabilă de tip caracter de lungime maximă 30 și nu este nulă).

Diagrama entitate-relație

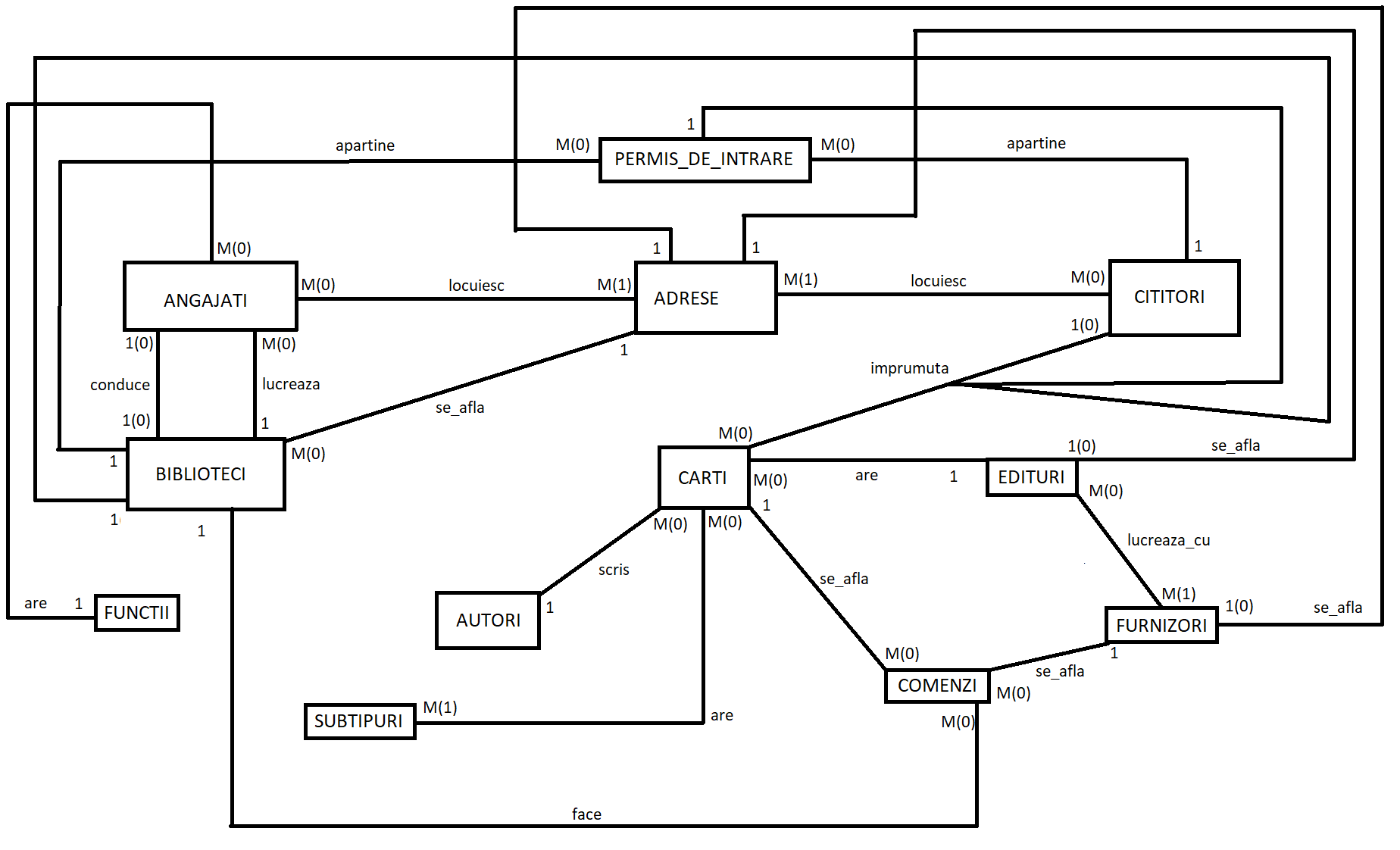
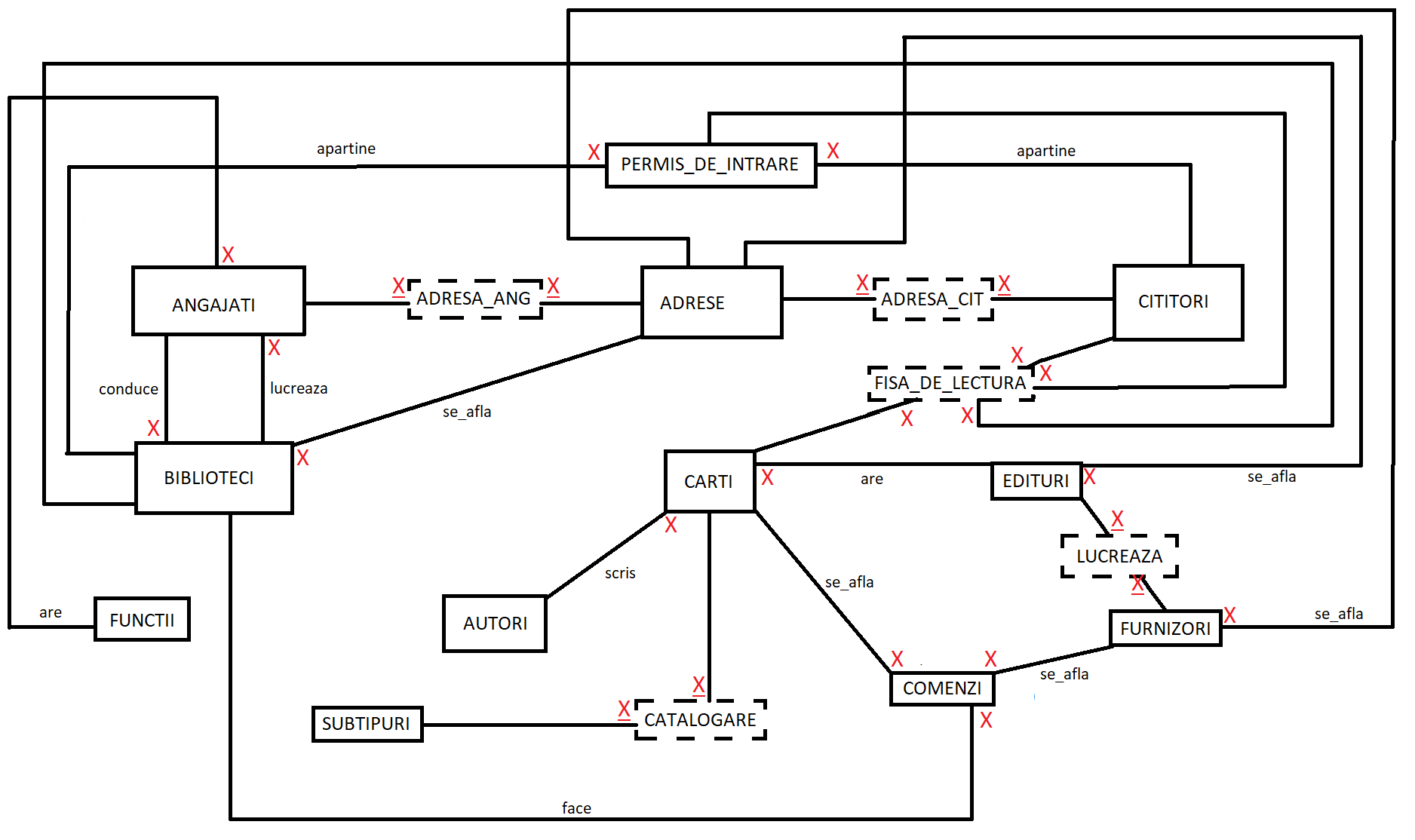
****

Diagrama conceptuală



**Schemele relaționale** corespunzătoare diagramei conceptuale sunt:

ANGAJATI(angajat\_id#, nume, prenume, cnp, telefon, email, salariu, data\_angajare, functie\_id, biblioteca\_id)

FUNCTII(functie\_id#, denumire, salariu\_minim, salariu\_maxim)

BIBLIOTECI(biblioteca\_id#, denumire, telefon, manager\_id, adresa\_id)

ADRESE(adresa\_id#, judet, localitate, cod\_postal, strada)

AUTORI(autor\_id#, nume, prenume)

CARTI(carte\_id#, titlu, editura\_id, an\_aparitie, autor\_id, tip\_carte)

CITITORI(cititor\_id#, nume, prenume, cnp, telefon, email)

PERMIS\_DE\_INTRARE(permis\_id#, anul\_inscrieri, cititor\_id, biblioteca\_id)

EDITURI(editura\_id#, denumire, telefon, email, adresa\_id)

FURNIZORI(furnizor\_id#, denumire, telefon, email, adresa\_id)

COMENZI(comanda\_id#, furnizor\_id, carte\_id, biblioteca\_id, nr\_exemplare, pret\_comanda)

SUBTIPURI(subtip\_id#, denumire)

CATALOGARE(carte\_id#, subtip\_id#)

ADRESA\_ANG(angajat\_id#, adresa\_id#)

ADRESA\_CIT(cititor\_id#, adresa\_id#)

FISA\_DE\_LECTURA(fisa\_id#, cititor\_id, carte\_id, biblioteca\_id, permis\_id, data\_imprumut, data\_restituire)

LUCREAZA(editura\_id#, furnizor\_id#)

Non - FN1

|  |  |
| --- | --- |
| angajat\_id# | adrese |
| P1 | Gorj, Alba, Cluj |
| P2 | Argeș, Iași |

FN1

ANGAJATI:

|  |
| --- |
| angajat\_id# |
| P1 |
| P2 |

ADRESA\_ANG:

|  |  |
| --- | --- |
| angajat\_id# | adresa\_id# |
| P1 | A1 |
| P1 | A2 |
| P1 | A3 |
| P1 | A4 |
| P1 | A5 |

ADRESE:

|  |  |
| --- | --- |
| adresa\_id# | judet |
| A1 | Gorj |
| A2 | Alba |
| A3 | Cluj |
| A4 | Argeș |
| A5 | Iași |

Non - FN2

|  |  |  |  |
| --- | --- | --- | --- |
| carte\_id# | subtip\_id# | titlu\_carte | denumire\_subtip |
| C1 | S1 | Despre animale | animale |
| C1 | S2 | Despre animale | biologie |
| C2 | S3 | Fizică pentru începători | fizică |

FN2

CARTI:

|  |  |
| --- | --- |
| carte\_id# | titlu\_carte |
| C1 | Despre animale |
| C2 | Fizică pentru începători |

CATALOGARE:

|  |  |
| --- | --- |
| carte\_id# | subtip\_id# |
| C1 | S1 |
| C1 | S2 |
| C2 | S3 |

SUBTIPURI:

|  |  |
| --- | --- |
| subtip\_id# | denumire\_subtip |
| S1 | animale |
| S2 | biologie |
| S3 | fizică |

Non – FN3

|  |  |  |  |
| --- | --- | --- | --- |
| carte\_id# | titlu | editura | adresă\_editură |
| C1 | Cosmosul | Editura Art | Cluj |
| C2 | Poeziile lui Eminescu | Editura Aur | Iași |
| C3 | Amintiri din copilări | Editura Art | Cluj |

CARTI:

|  |  |  |
| --- | --- | --- |
| carte\_id# | titlu | editura\_id |
| C1 | Cosmosul | E1 |
| C2 | Poeziile lui Eminescu | E2 |
| C3 | Amintiri din copilări | E1 |

EDITURI:

|  |  |  |
| --- | --- | --- |
| editura\_id# | editura | adresă\_editură |
| E1 | Editura Art | Cluj |
| E2 | Editura Aur | Iași |

Non-BCNF:

FURNIZORI(furnizor\_id#, denumire, telefon#, email, adresa)

BCNF:

ADRESA\_FURNIZOR1(furnizor\_id#, denumire, email)

ADRESA\_FURNIZOR2(adresa#, telefon)

Non-FN4:

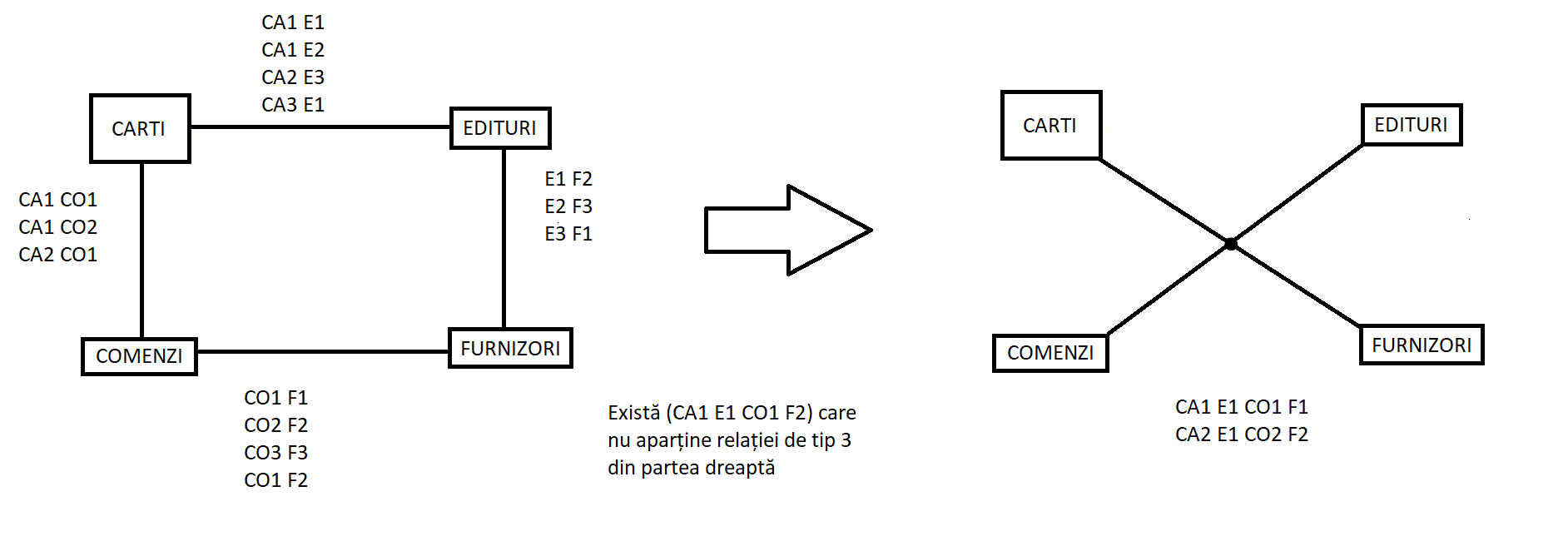
|  |  |  |
| --- | --- | --- |
| carte | tip | subtip |
| C1 | T1 | S1 |
| C1 | T2 | S2 |
| C1 | T1 | S2 |
| C1 | T2 | S1 |

FN4:

|  |  |
| --- | --- |
| carte | Tip |
| C1 | T1 |
| C1 | T2 |

|  |  |
| --- | --- |
| carte | subtip |
| C1 | S1 |
| C1 | S2 |

FN5:



DENORMALIZARE:

Denormalizarea folosită pe acest model este adaugarea editurii in entitatea CARTI, deoarece editura este un atribut foarte cautat la o carte sau numele si prenumele autorului din acelasi motiv.

Optimizarea unei cereri:

/\* Afișați codul și numele cititorilor care au citit cărțile care au apărut după 1900 sau au fost scrise de Tatham \*/

/\* Varianta neoptimizată: \*/

R1 = SELECT(CARTI, an\_aparitie > 1900)

R2 = SELECT(AUTORI, nume = ‘Tatham’)

R3 = SEMIJOIN(CARTI, R2)

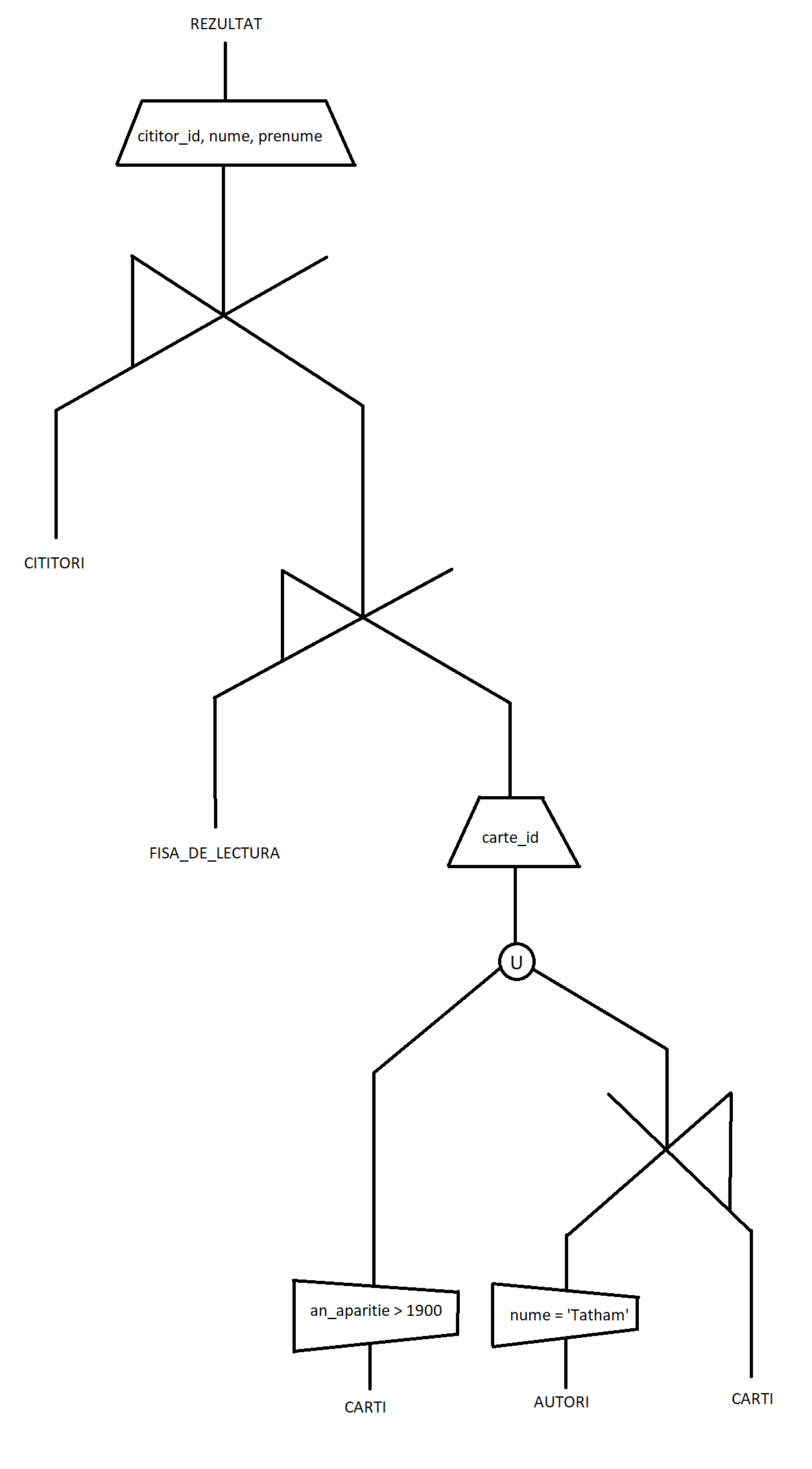
R4 = UNION(R1, R3)

R5 = PROJECT(R4, carte\_id)

R6 = SEMIJOIN(FISA\_DE\_LECTURA, R5)

R7 = SEMIJOIN(CITITORI, R6)

Rezultat = R8 = PROJECT(R7, cititor\_id, nume, prenume)



select distinct rez.cititor\_id, rez.nume, rez.prenume

from

(select cit.\*

from cititori cit, (

select fisa.\*

from fisa\_de\_lectura fisa, (select aux.carte\_id

from ( select \*

from carti

where an\_aparitie > 1900

union

select c.\*

from carti c, (select \*

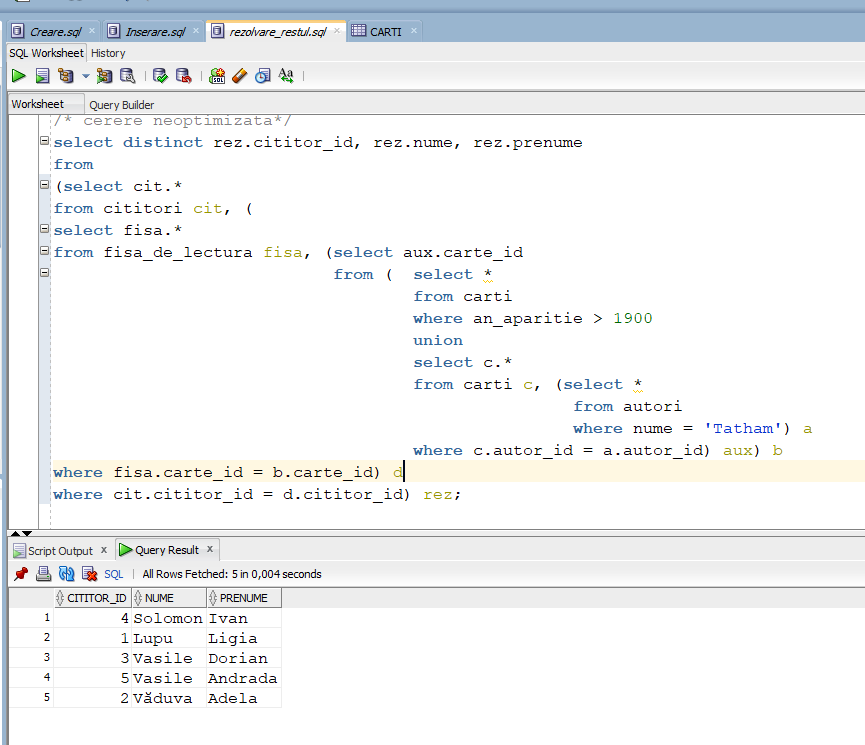
from autori

where nume = 'Tatham') a

where c.autor\_id = a.autor\_id) aux) b

where fisa.carte\_id = b.carte\_id) d

where cit.cititor\_id = d.cititor\_id) rez;



/\* Varianta optimizată: \*/

R1 = SELECT(CARTI, an\_aparitie > 1900)

R2 = PROJECT(R1, carte\_id)

R3 = SELECT(AUTORI, nume = ‘Tathan’)

R4 = PROJECT(R3, autor\_id)

R5 = PROJECT(CARTI, carte\_id, autor\_id)

R6 = SEMIJOIN(R5, R4)

R7 = PROJECT(R6, carte\_id)

R8 = UNION(R2, R7)

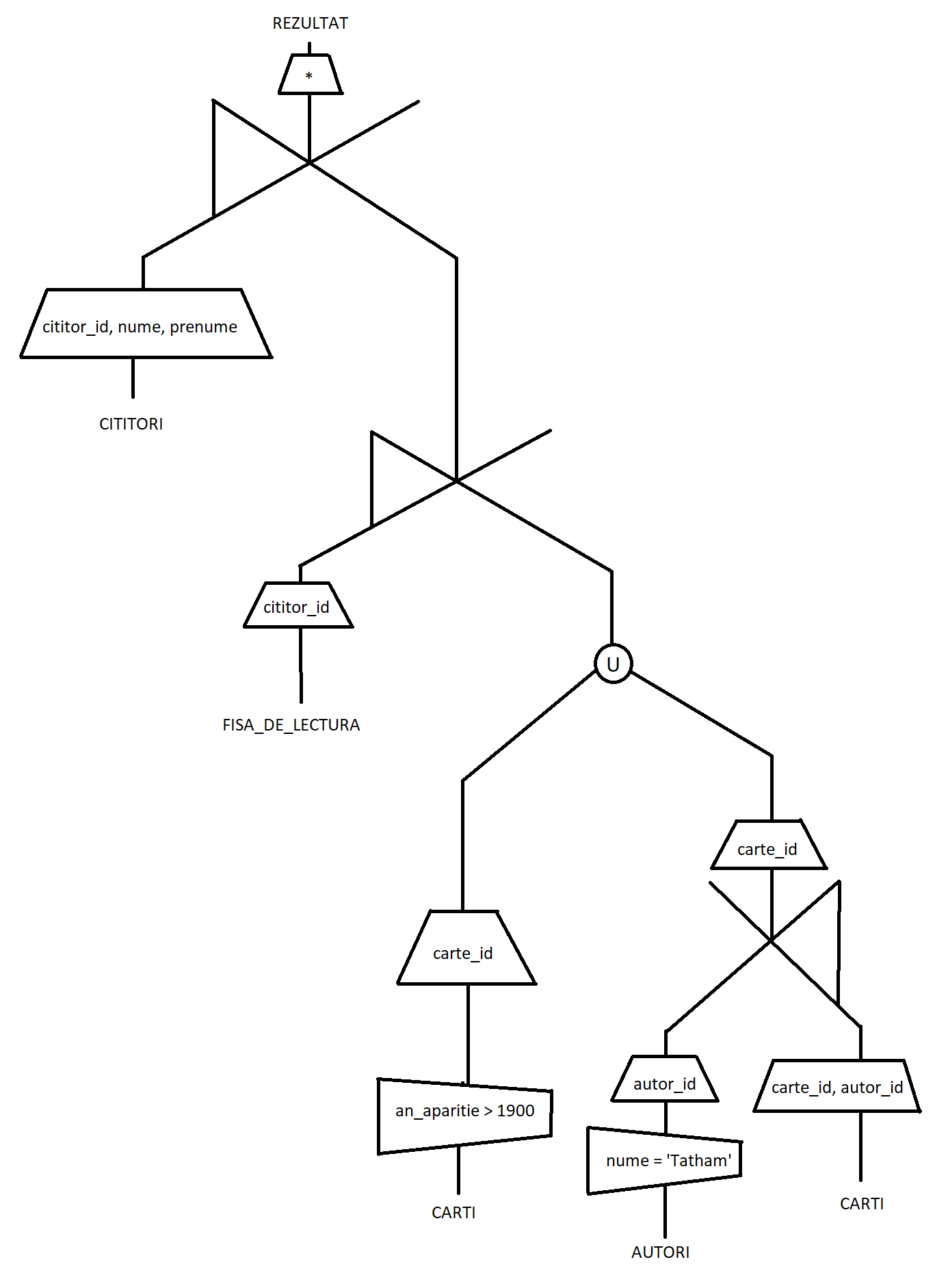
R9 = PROJECT(FISA\_DE\_LECTURA, carte\_id, cititor\_id)

R10 = SEMIJOIN(R9, R8)

R11 = PROJECT(CITITORI, cititor\_id, nume, prenume)

R12 = SEMIJOIN(R11, R10)

Rezultat = R13 = PROJECT(R12, \*)



select distinct cit.cititor\_id, cit.nume, cit.prenume

from cititori cit, (select f.cititor\_id

from fisa\_de\_lectura f, (select carte\_id

from carti c, (select autor\_id

from autori

where nume = 'Tatham') a

where c.autor\_id = a.autor\_id

union

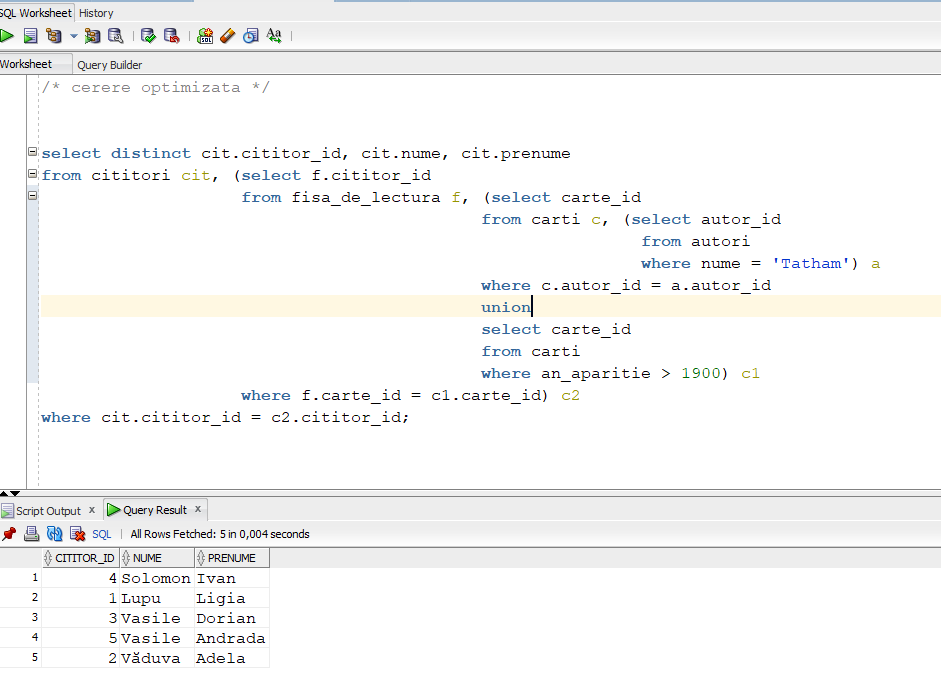
select carte\_id

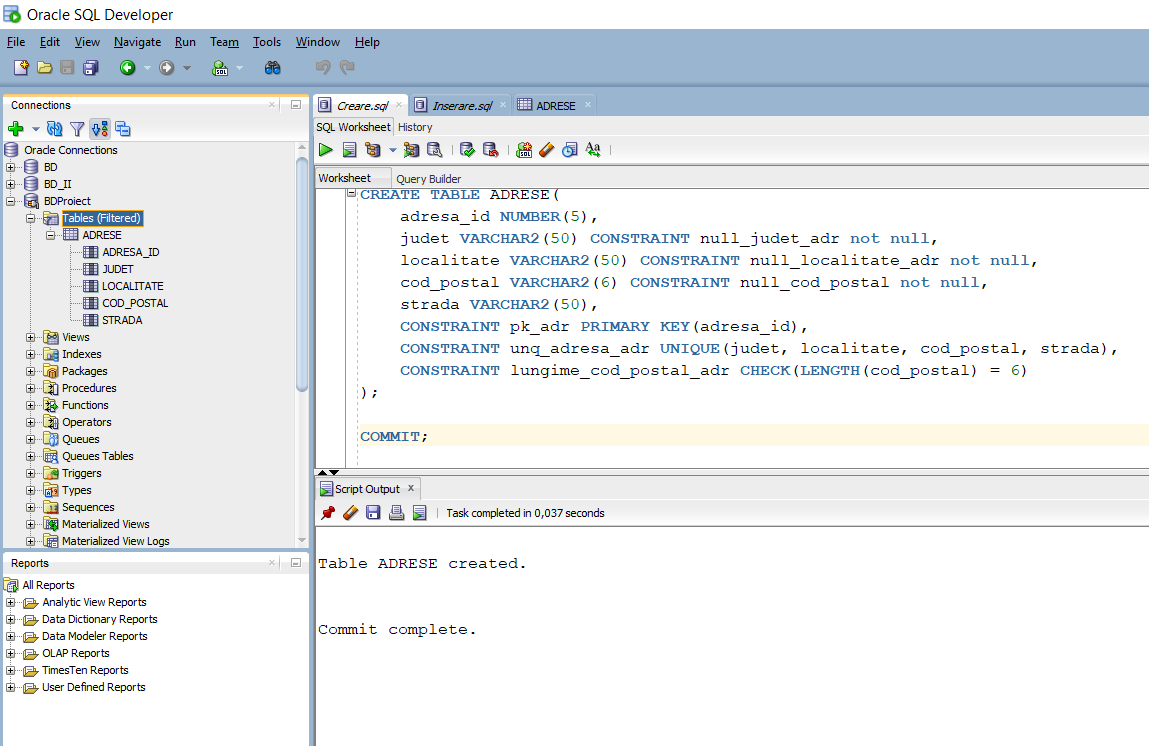
from carti

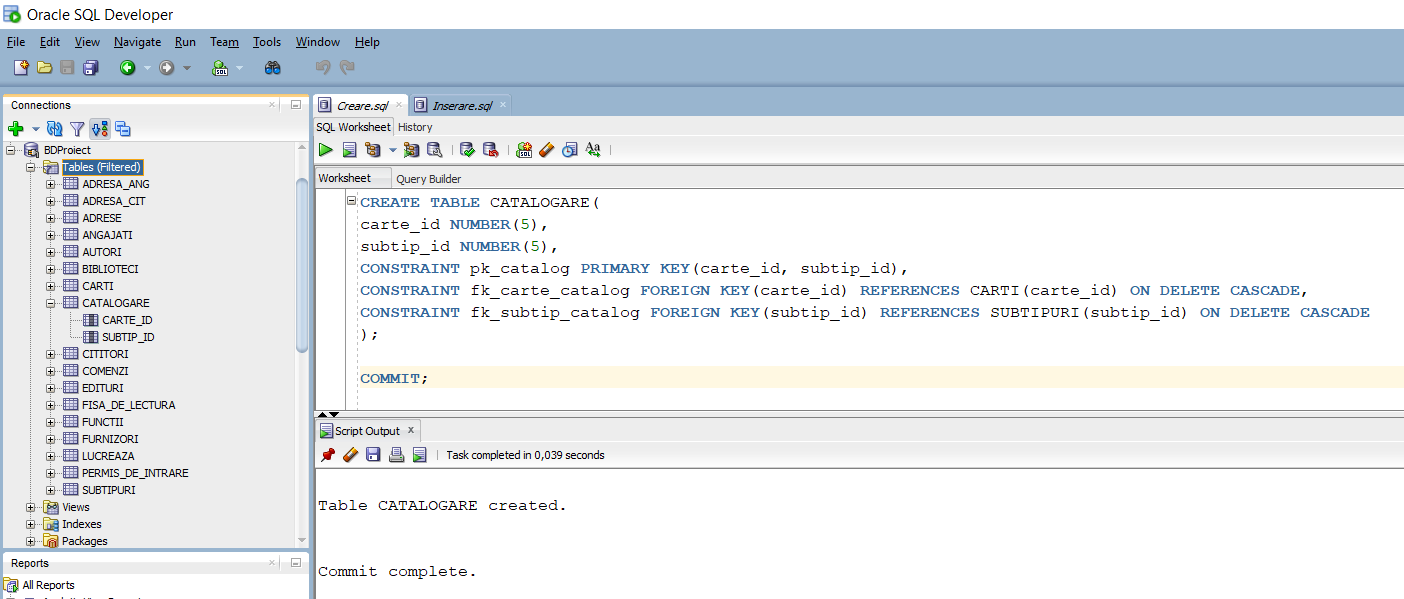
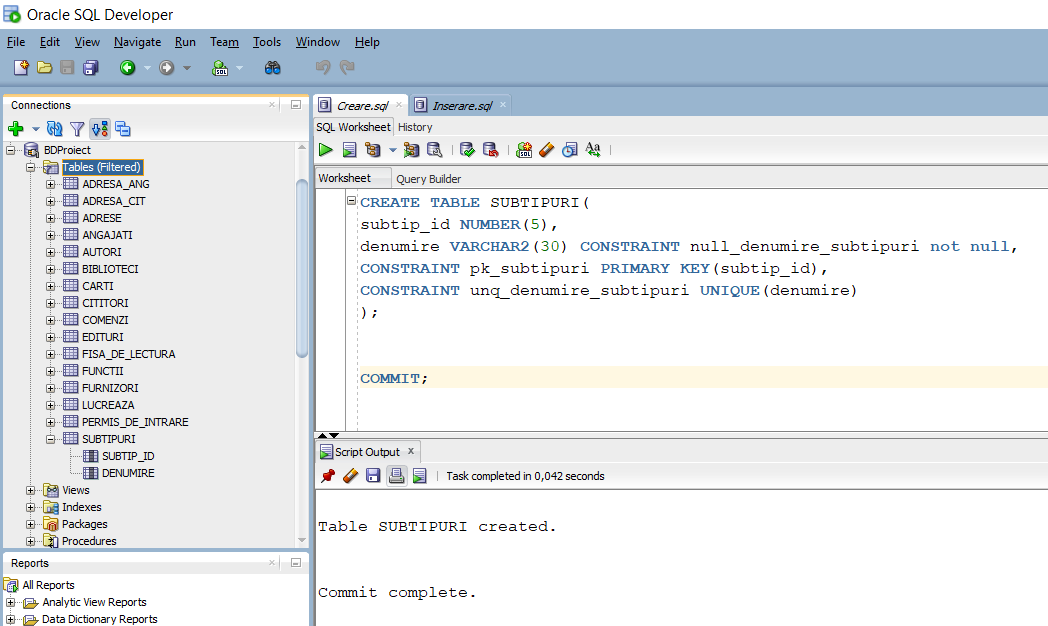
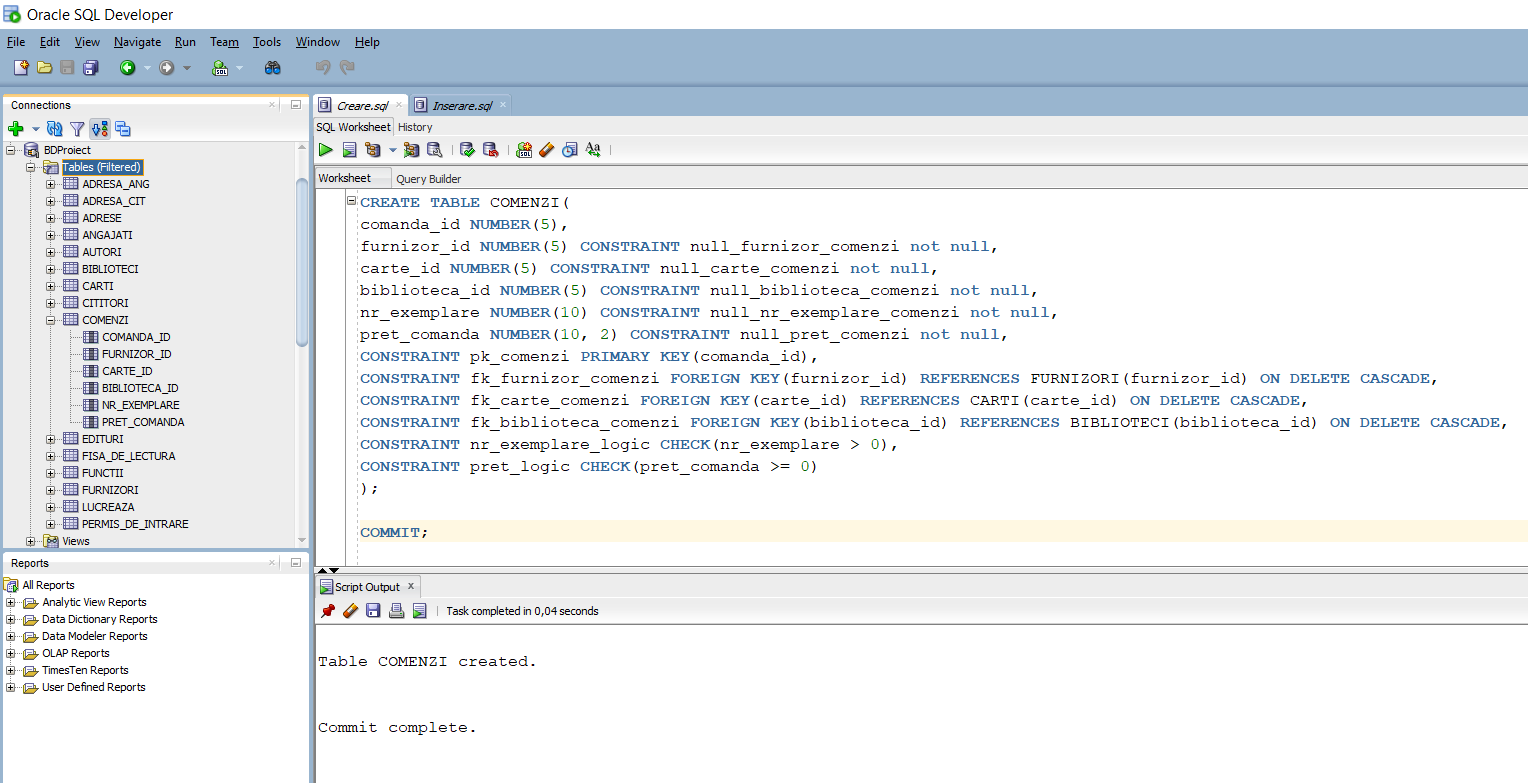
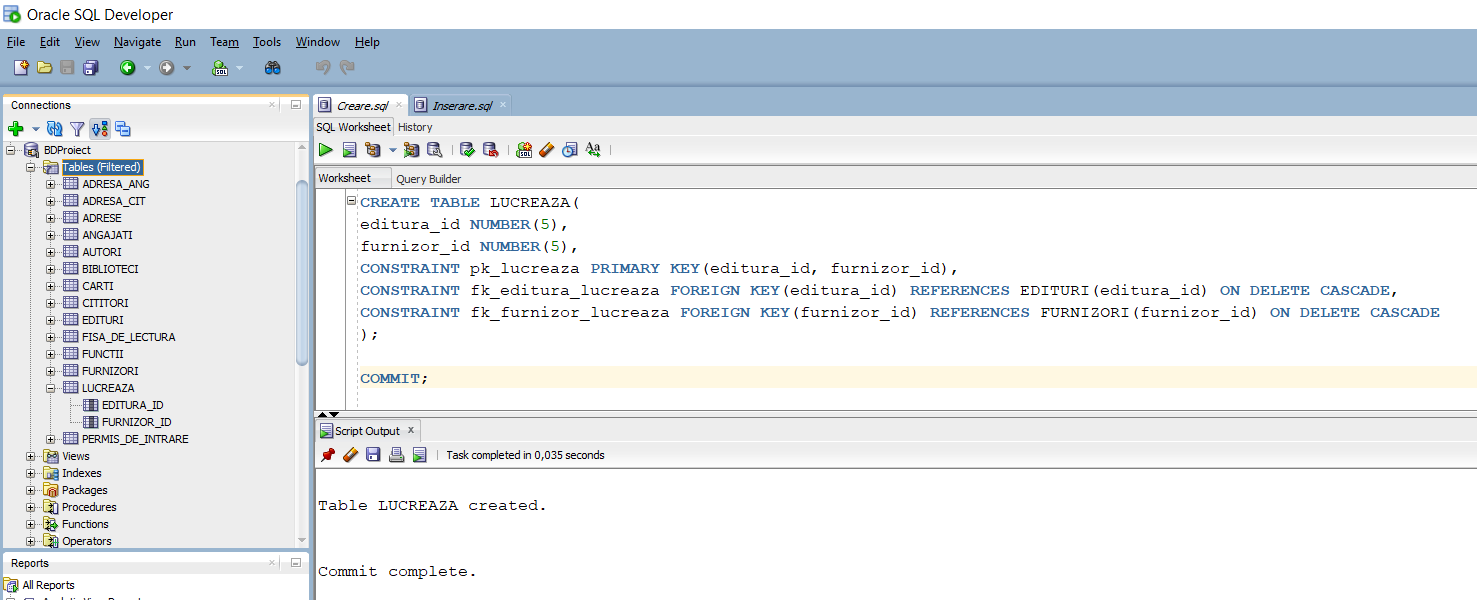
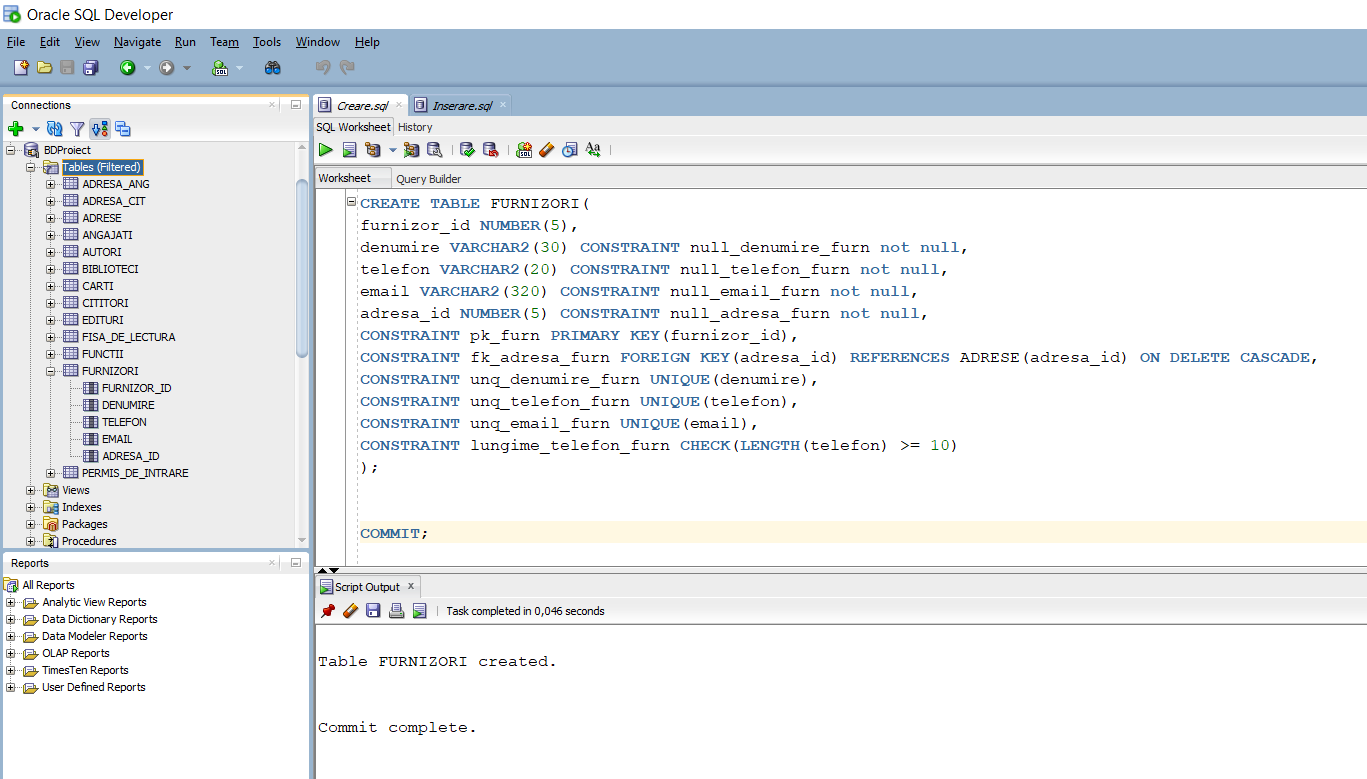
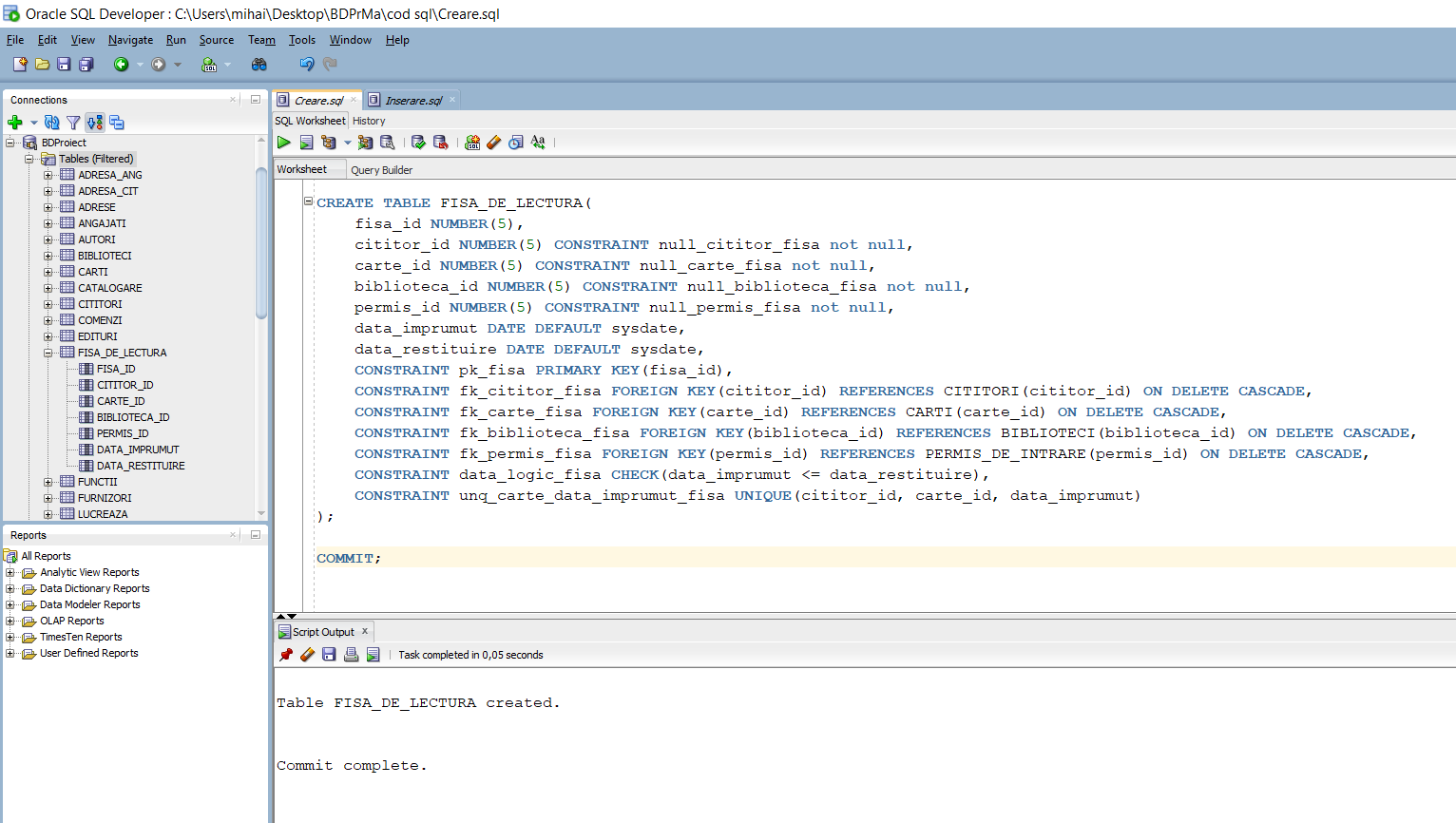
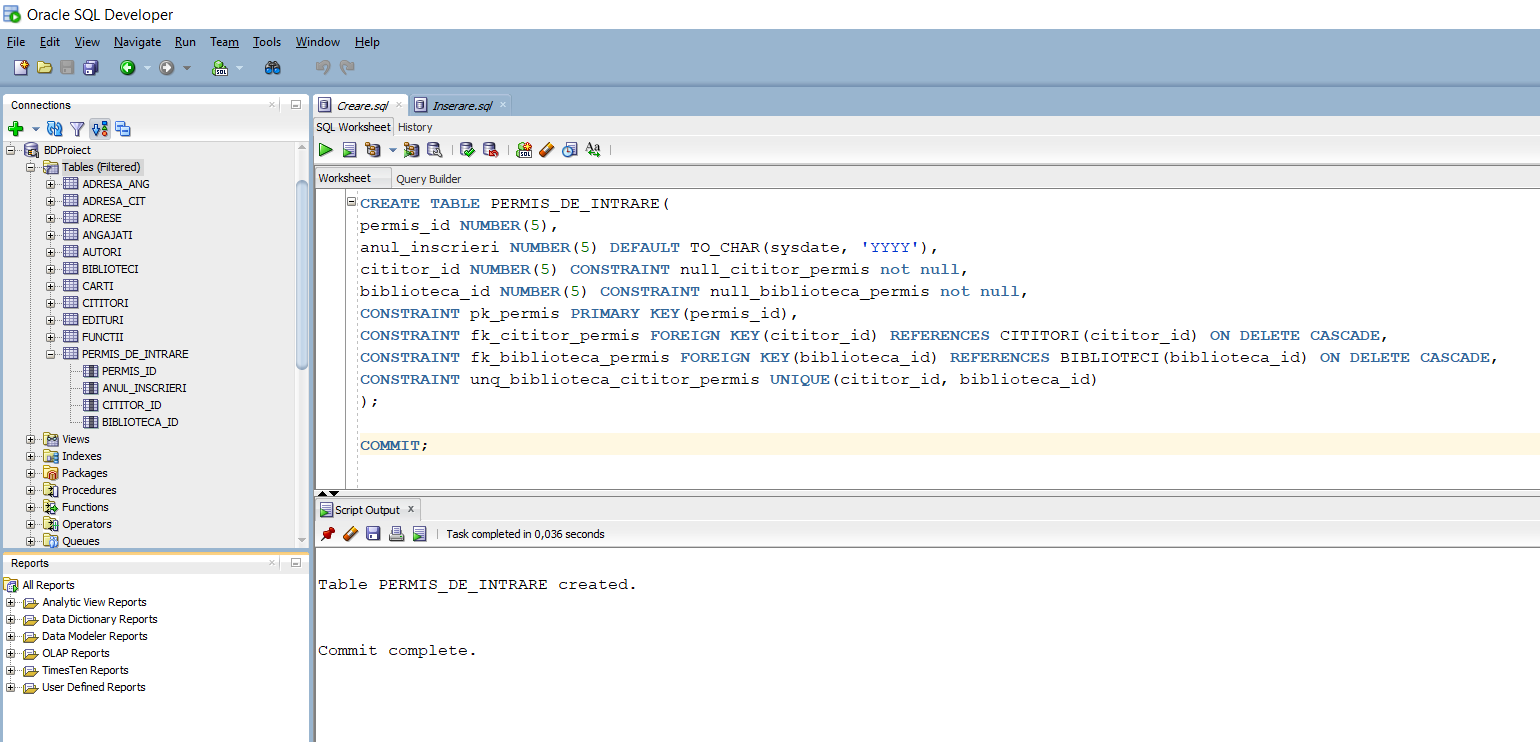
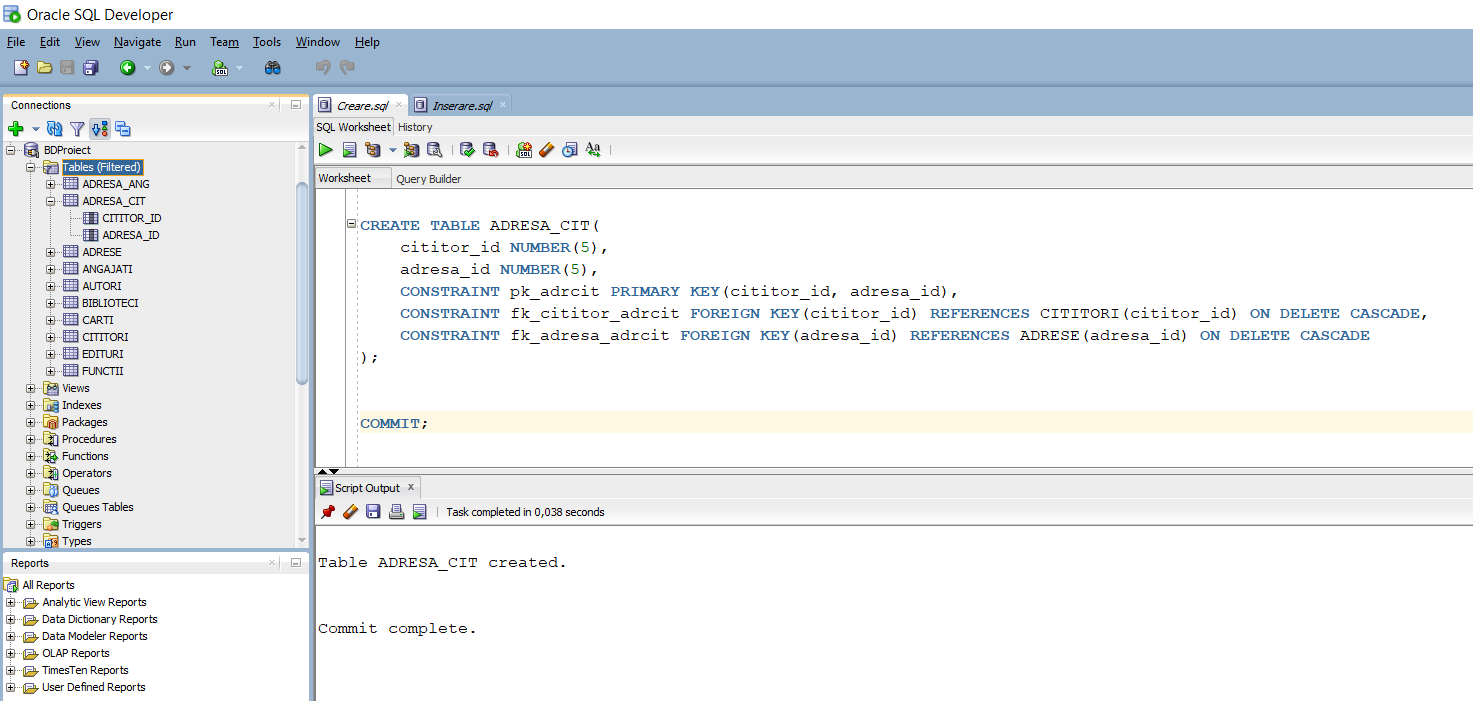
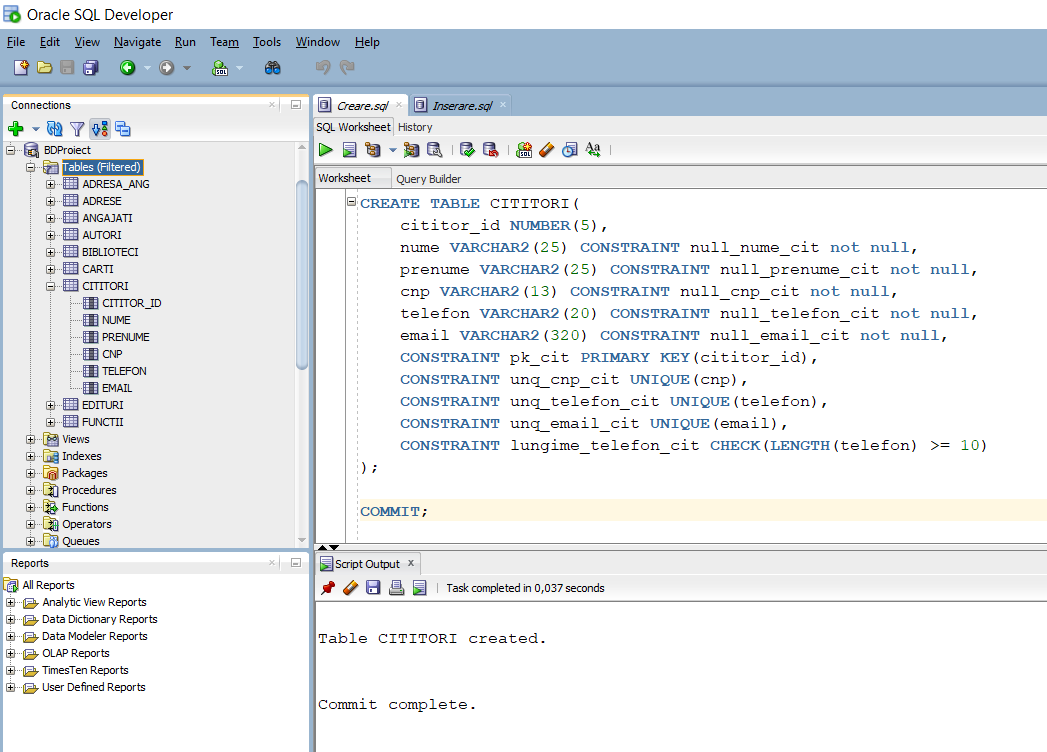
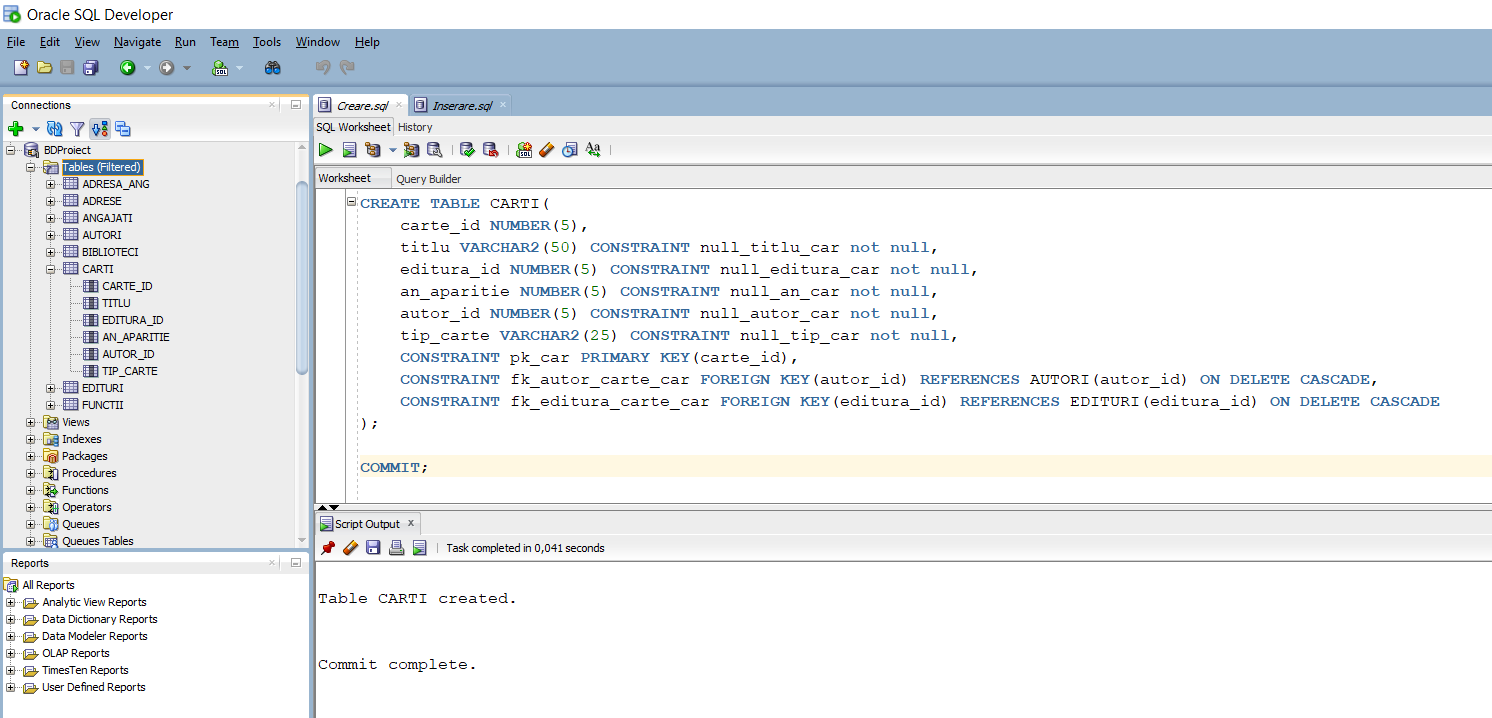
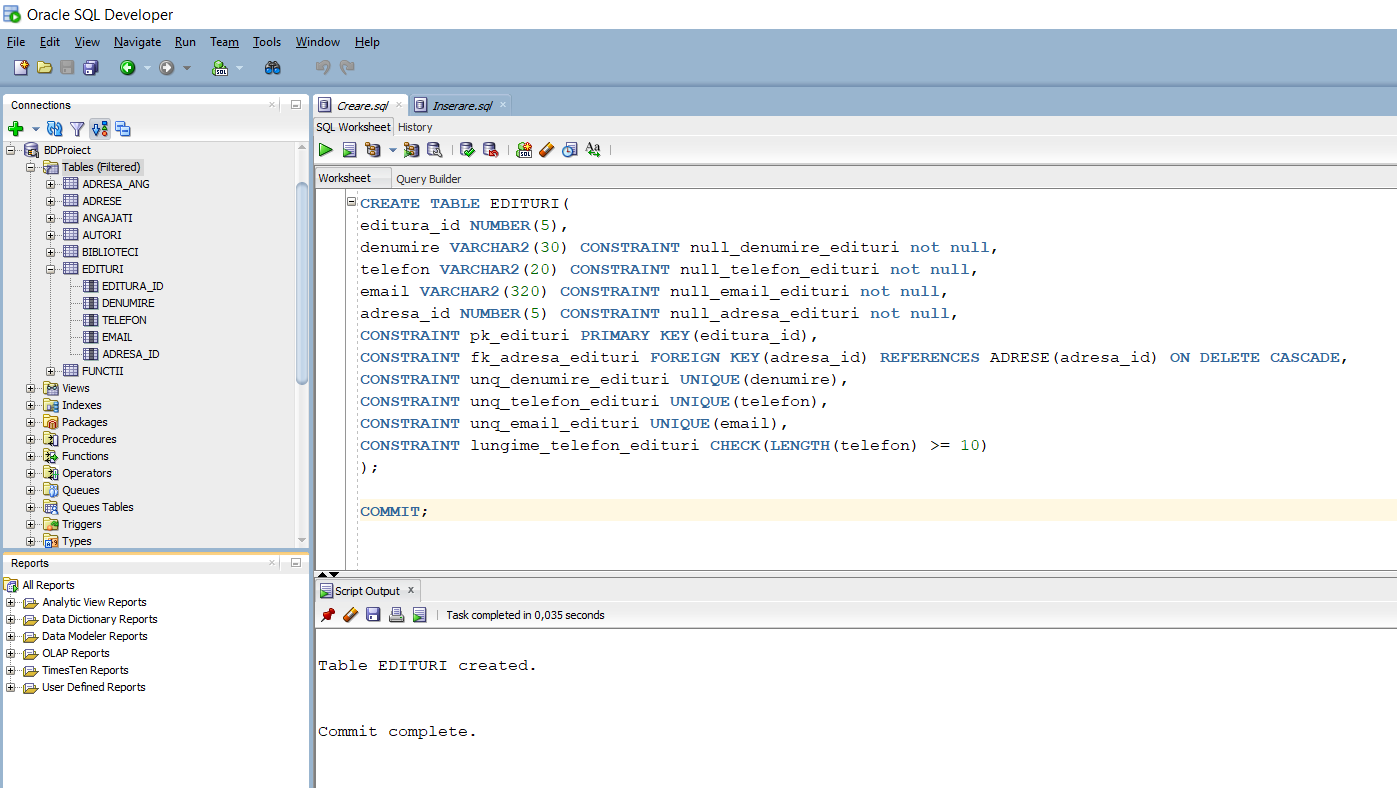
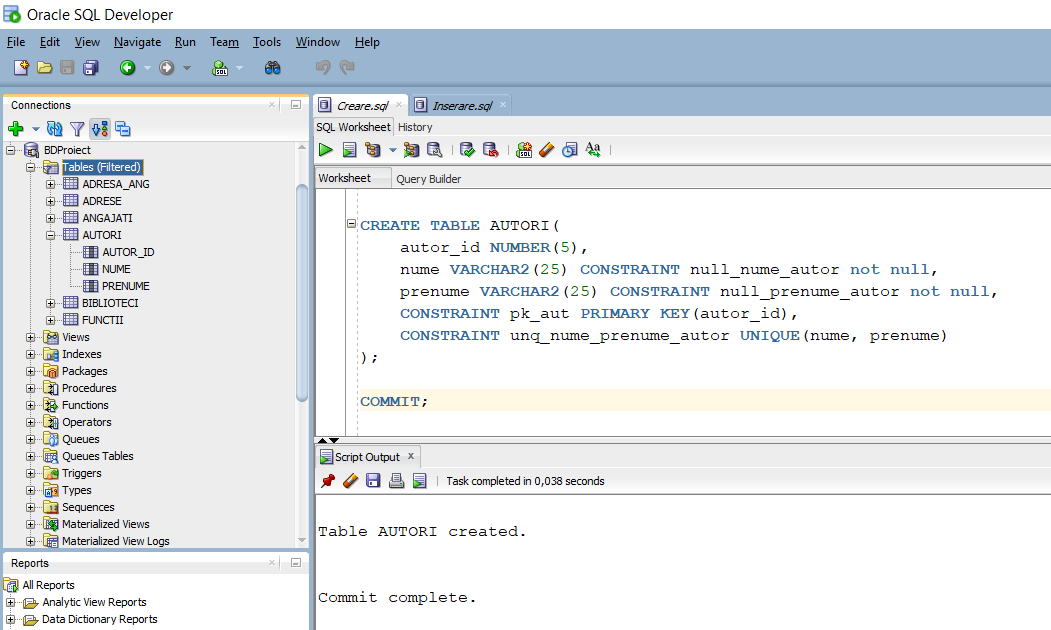
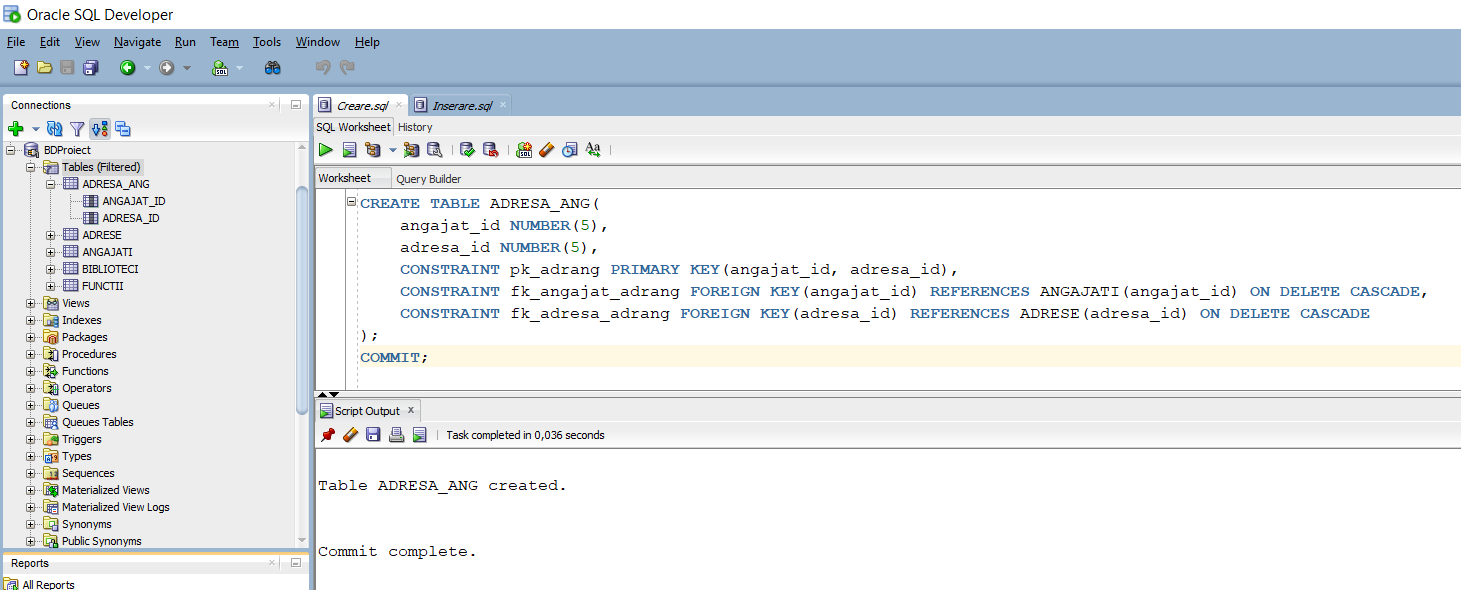
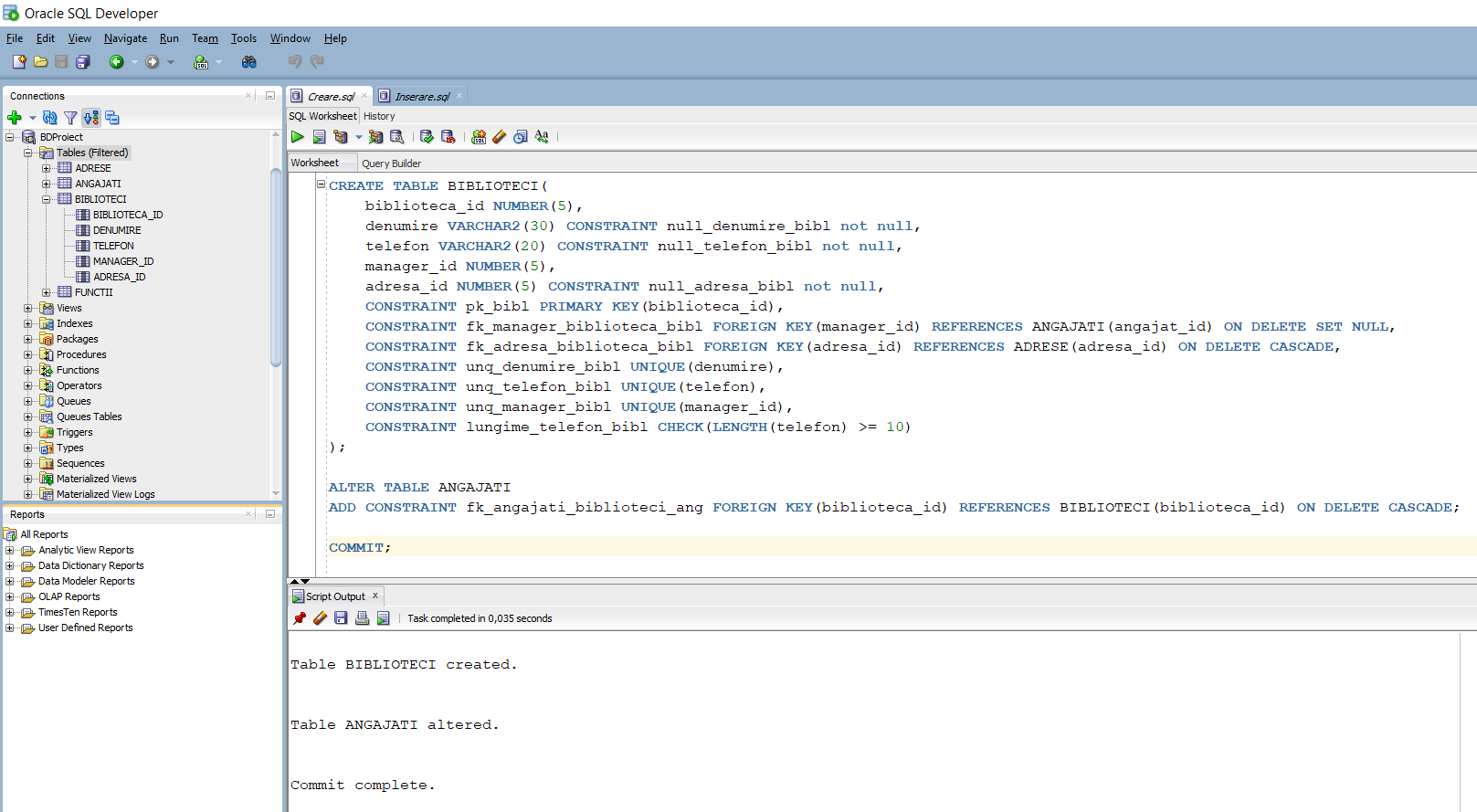
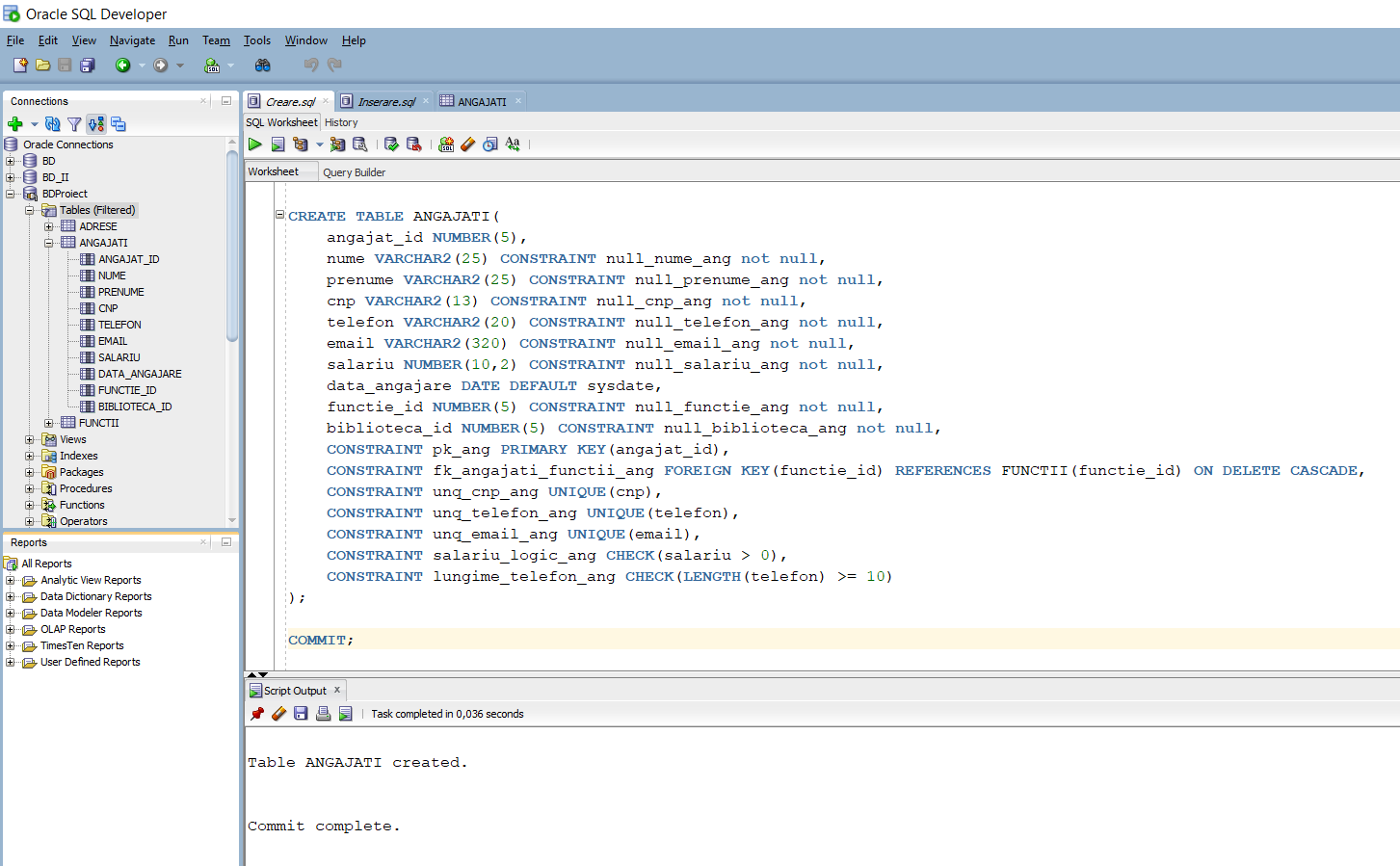
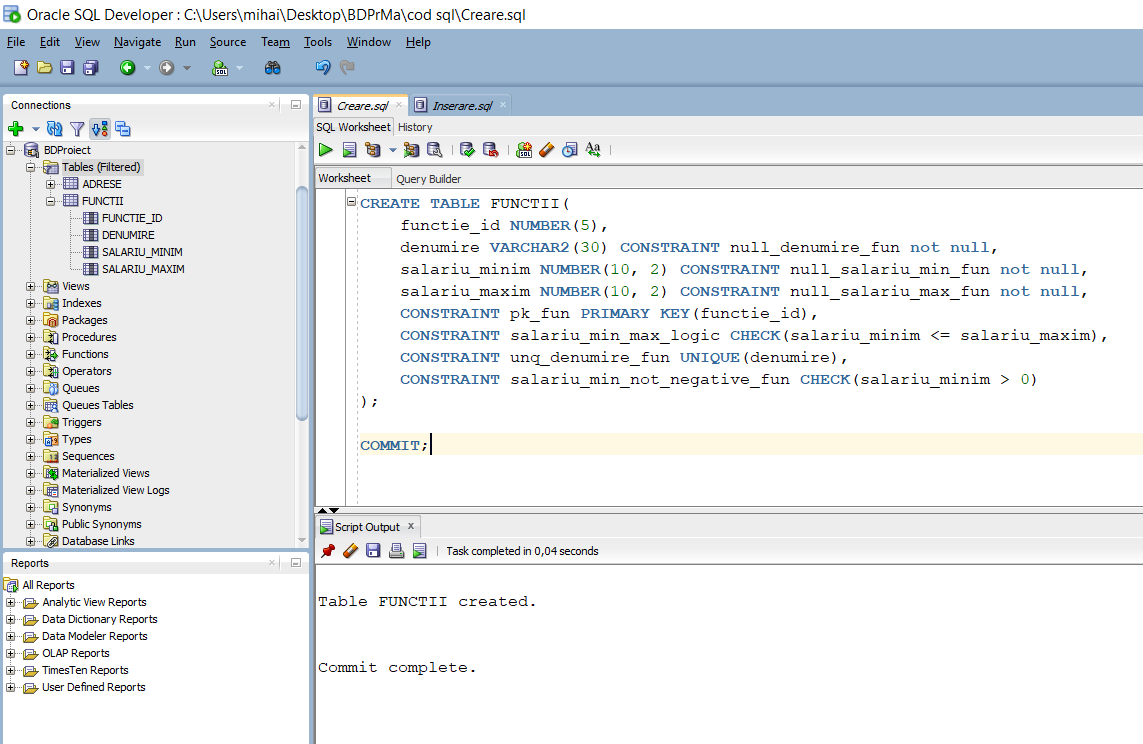
where an\_aparitie > 1900) c1

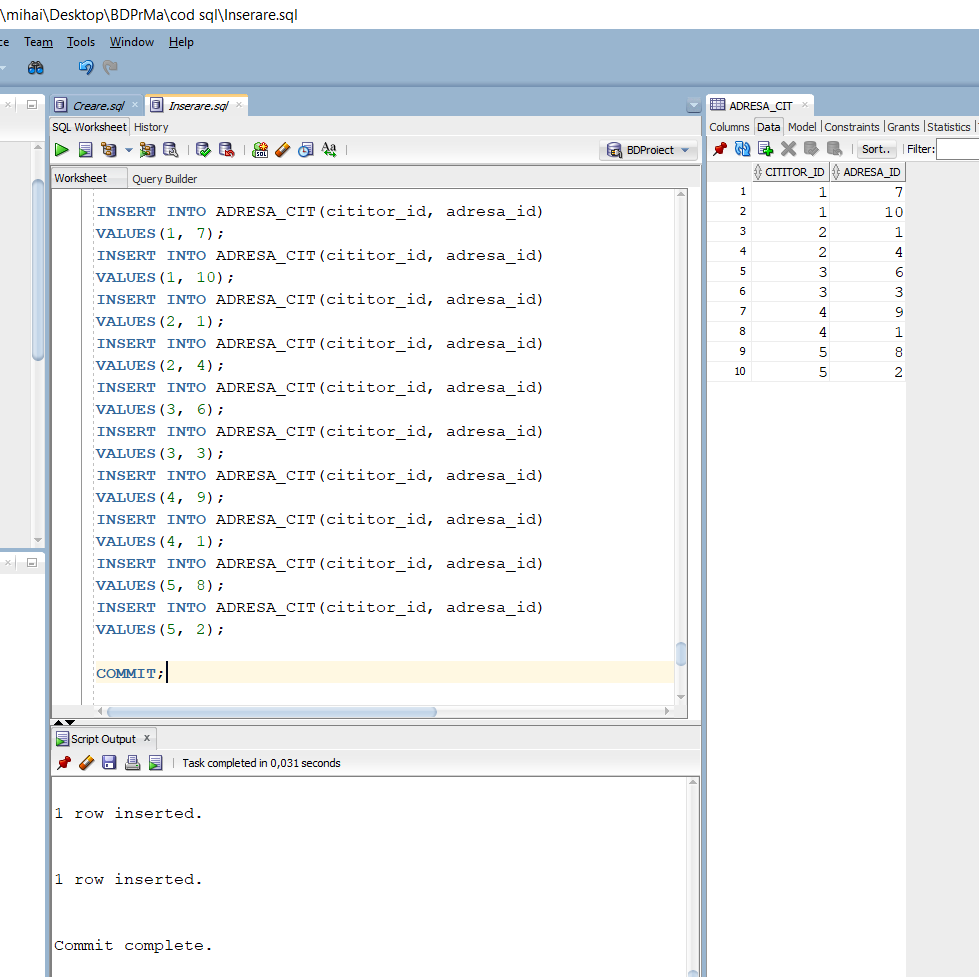
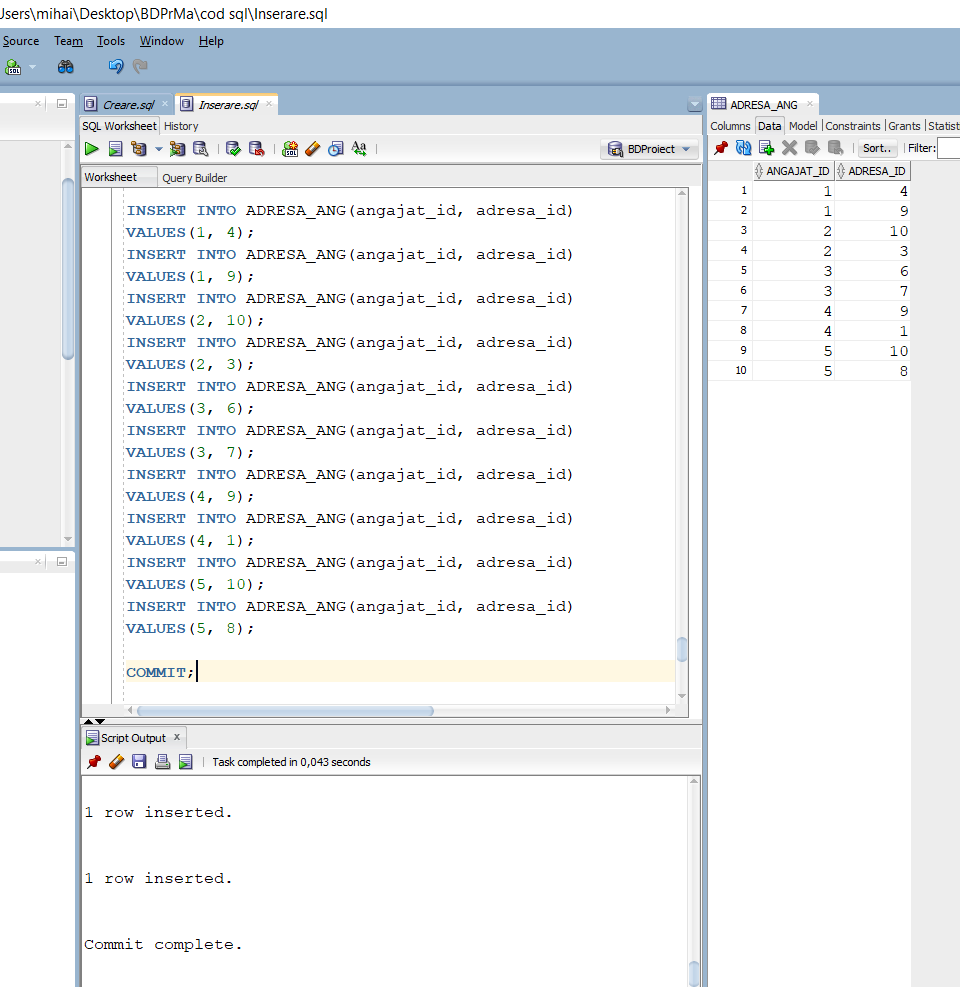
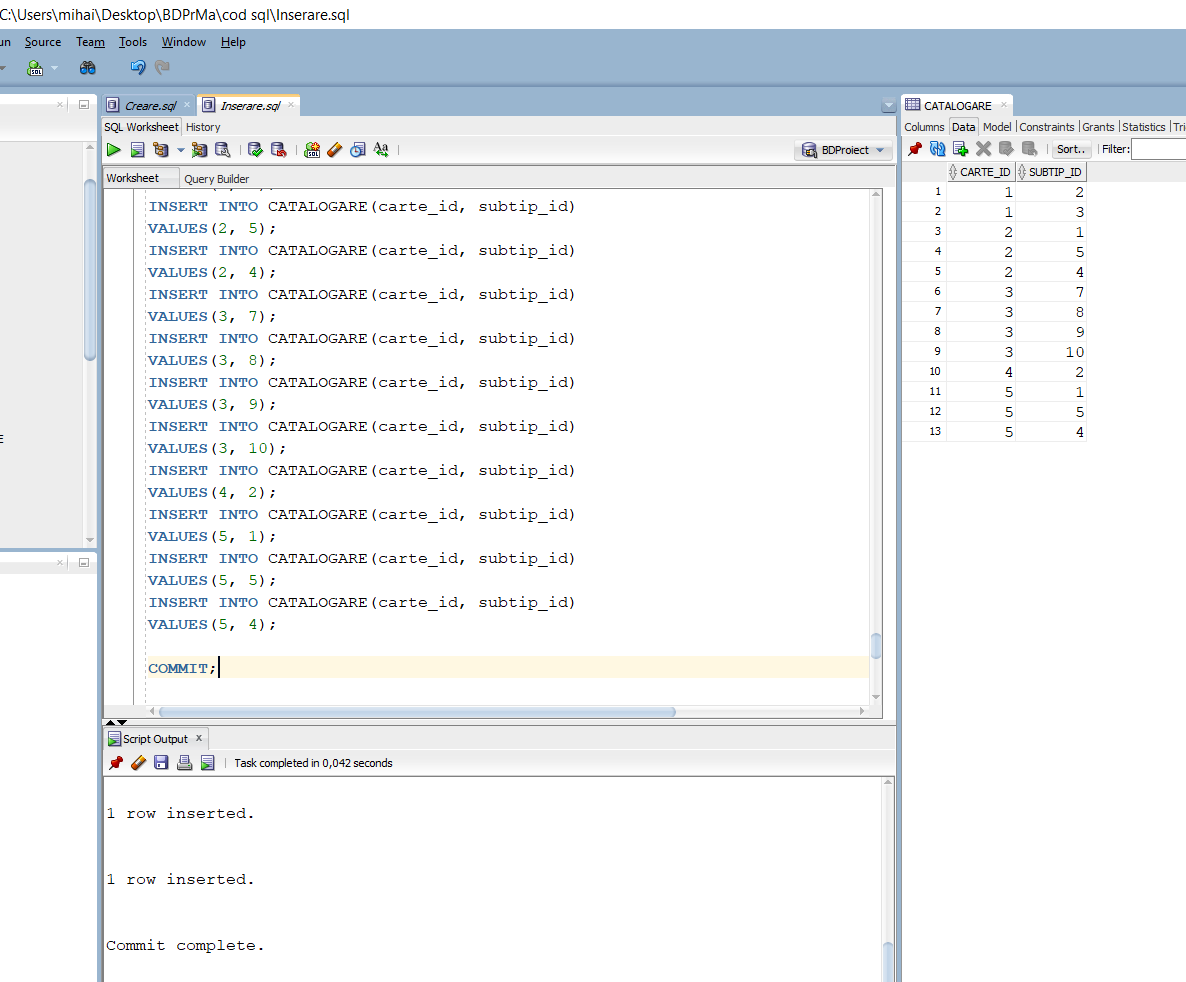
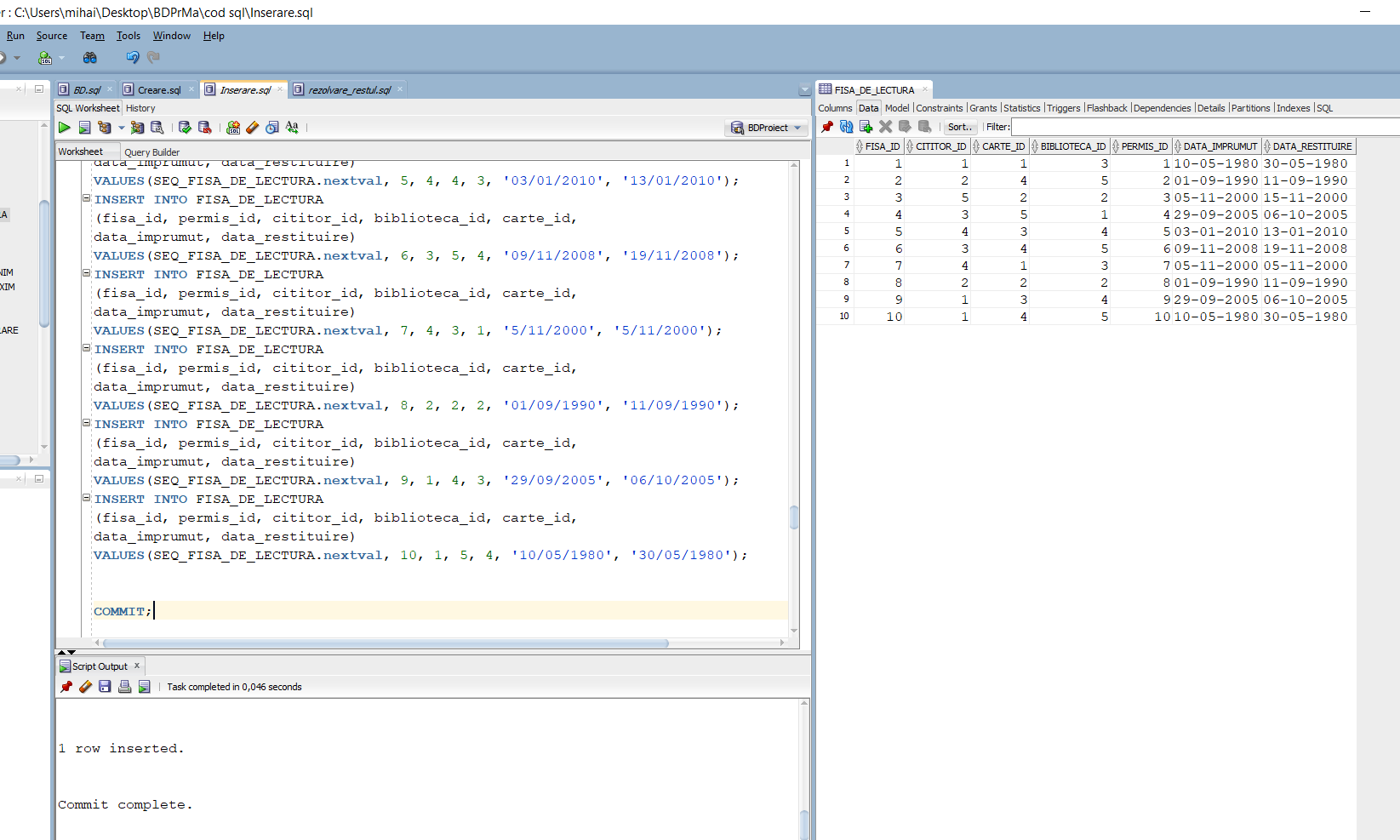
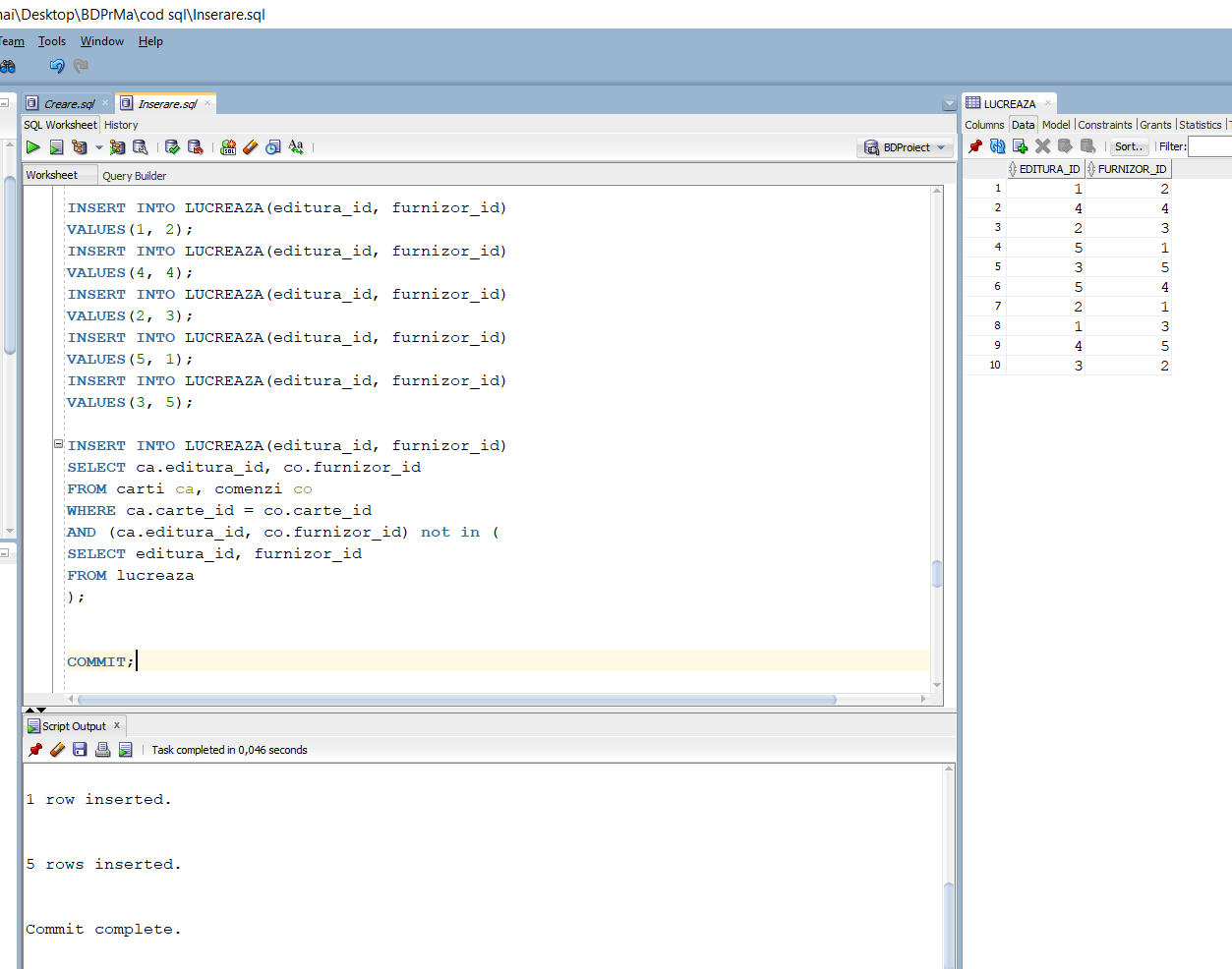
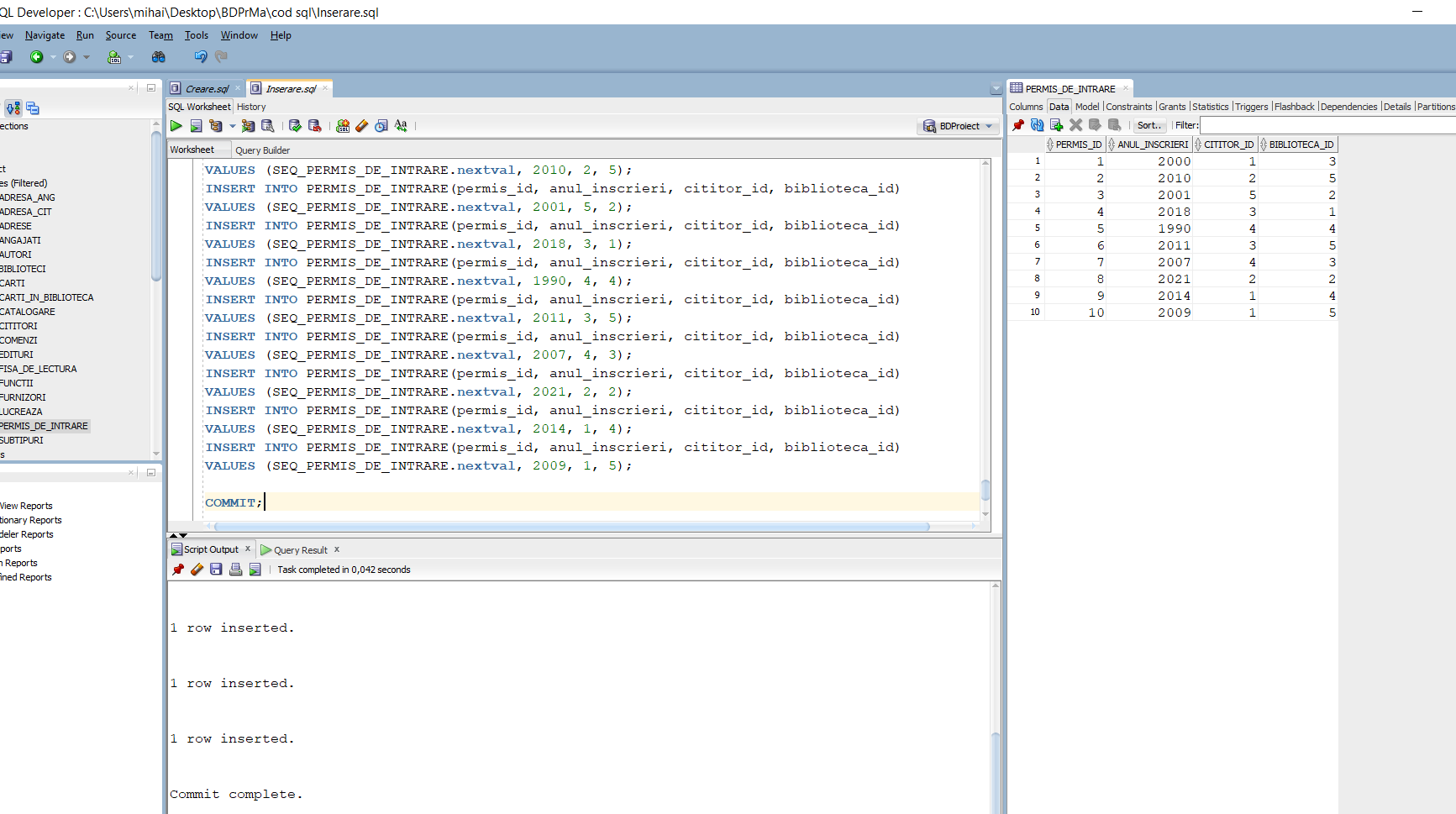
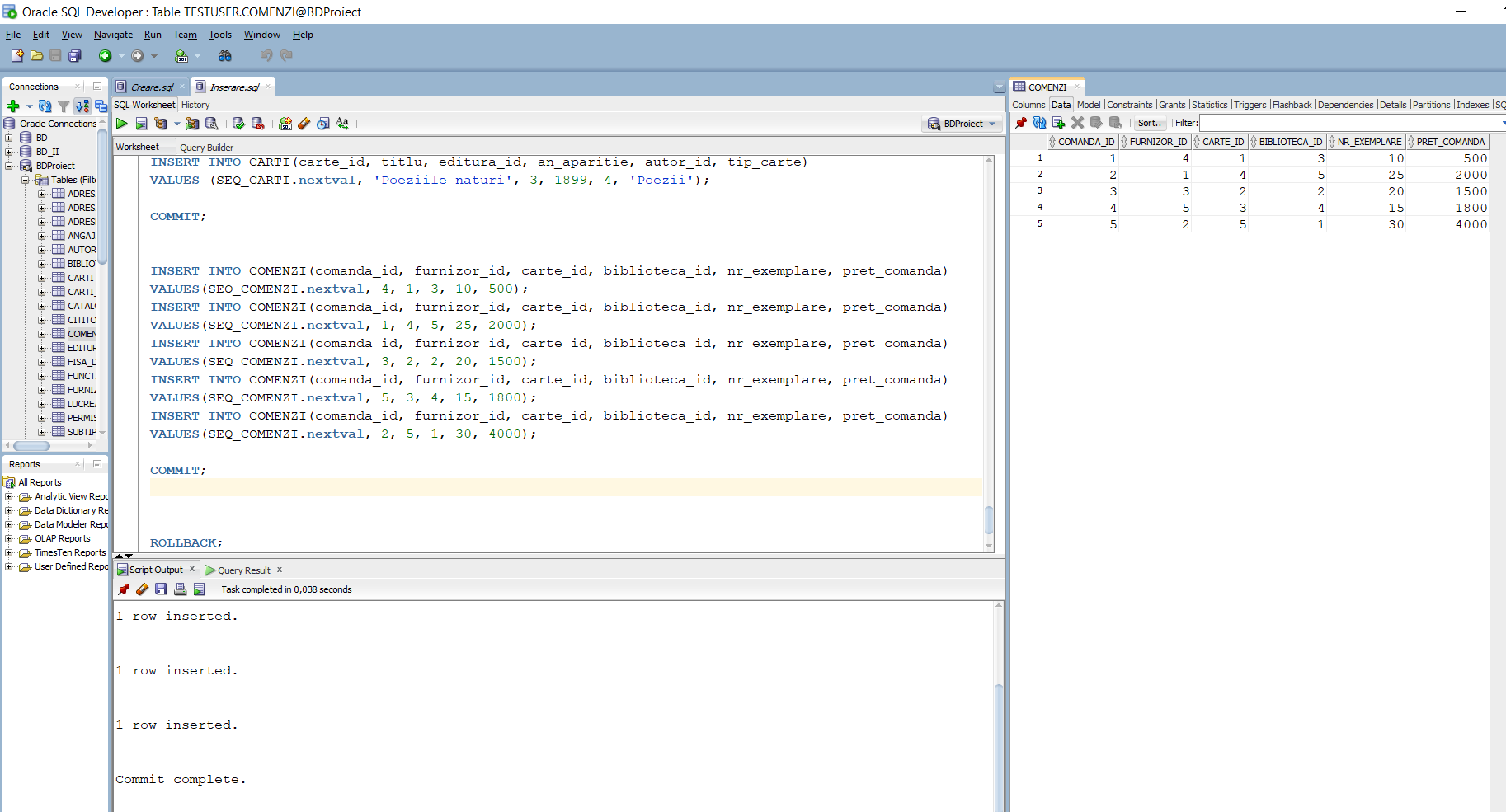
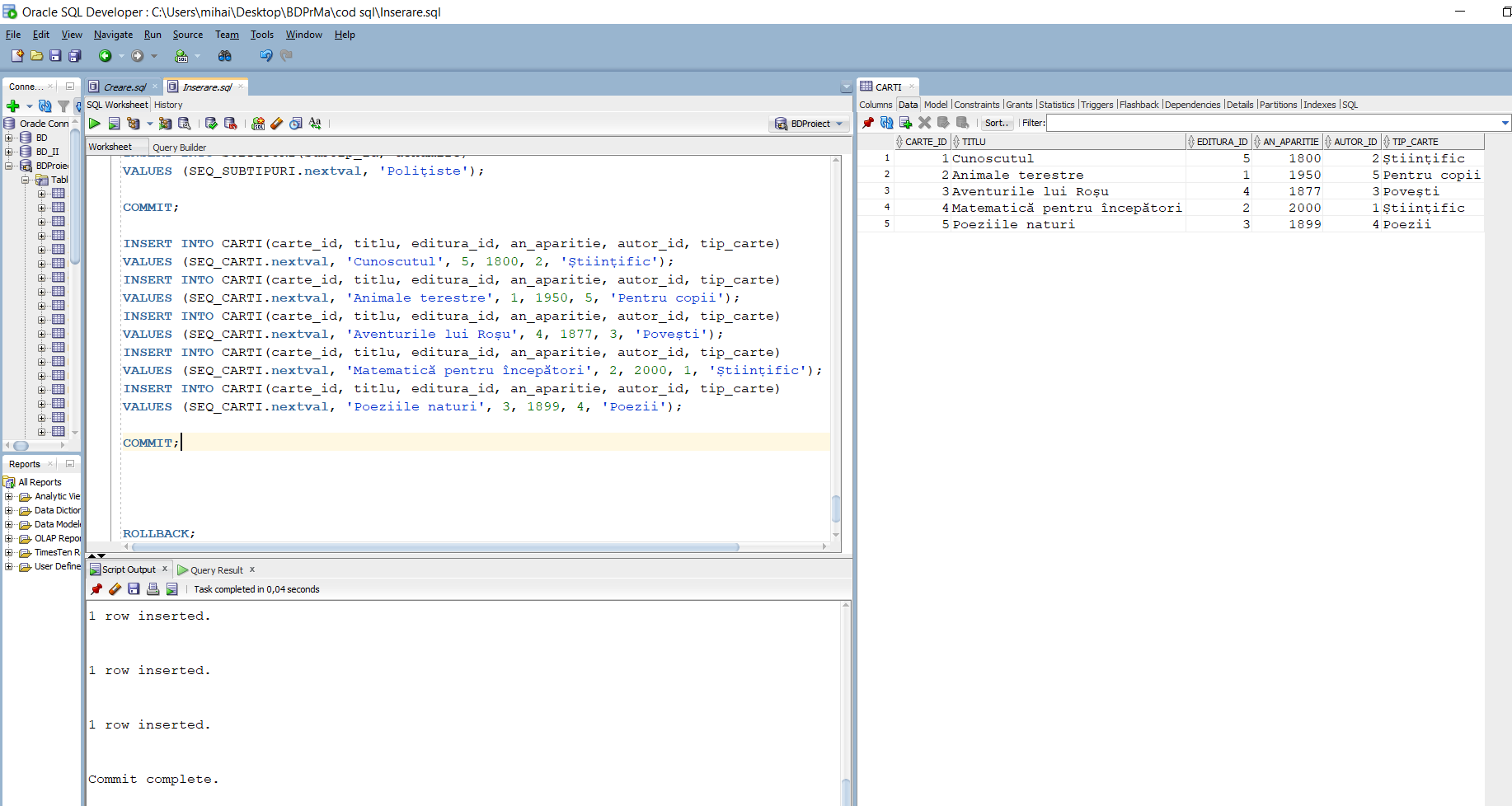
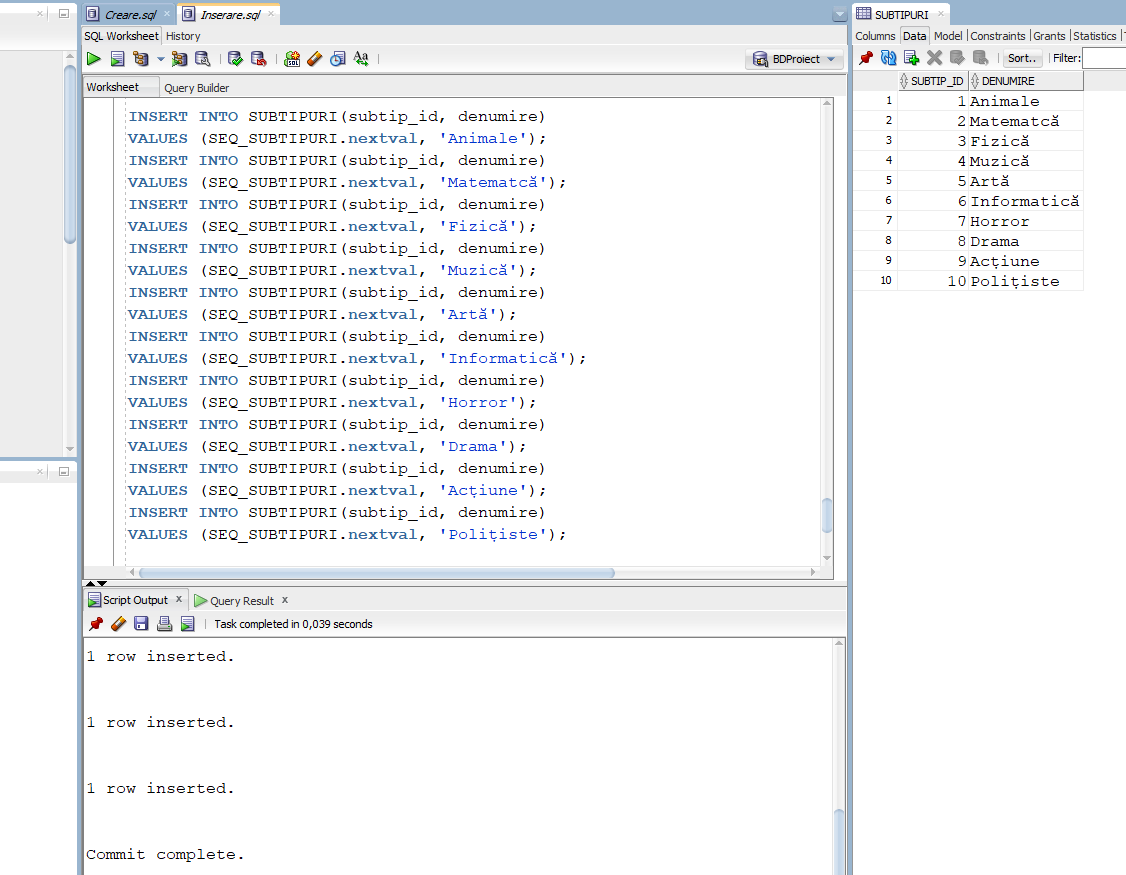
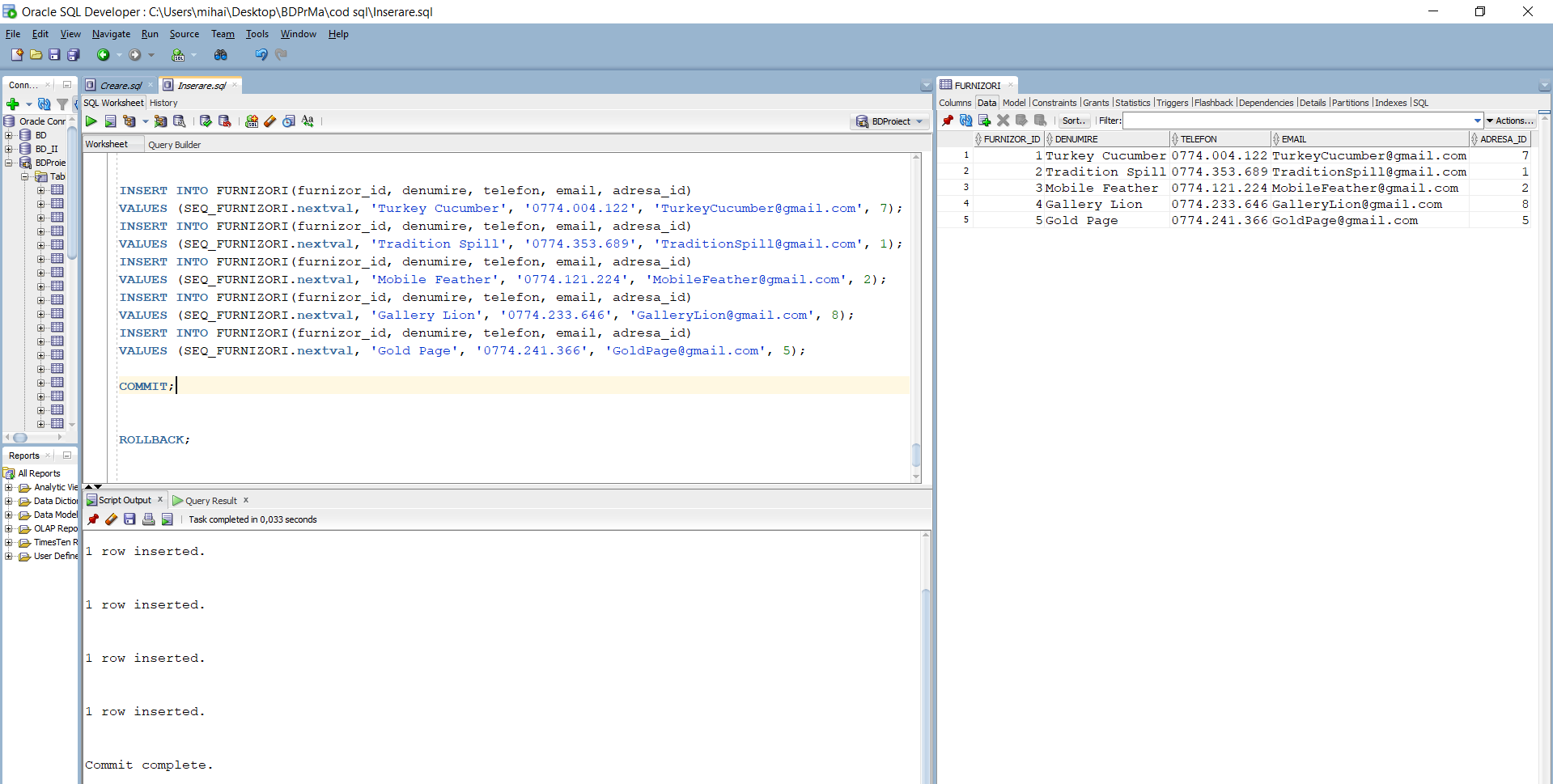
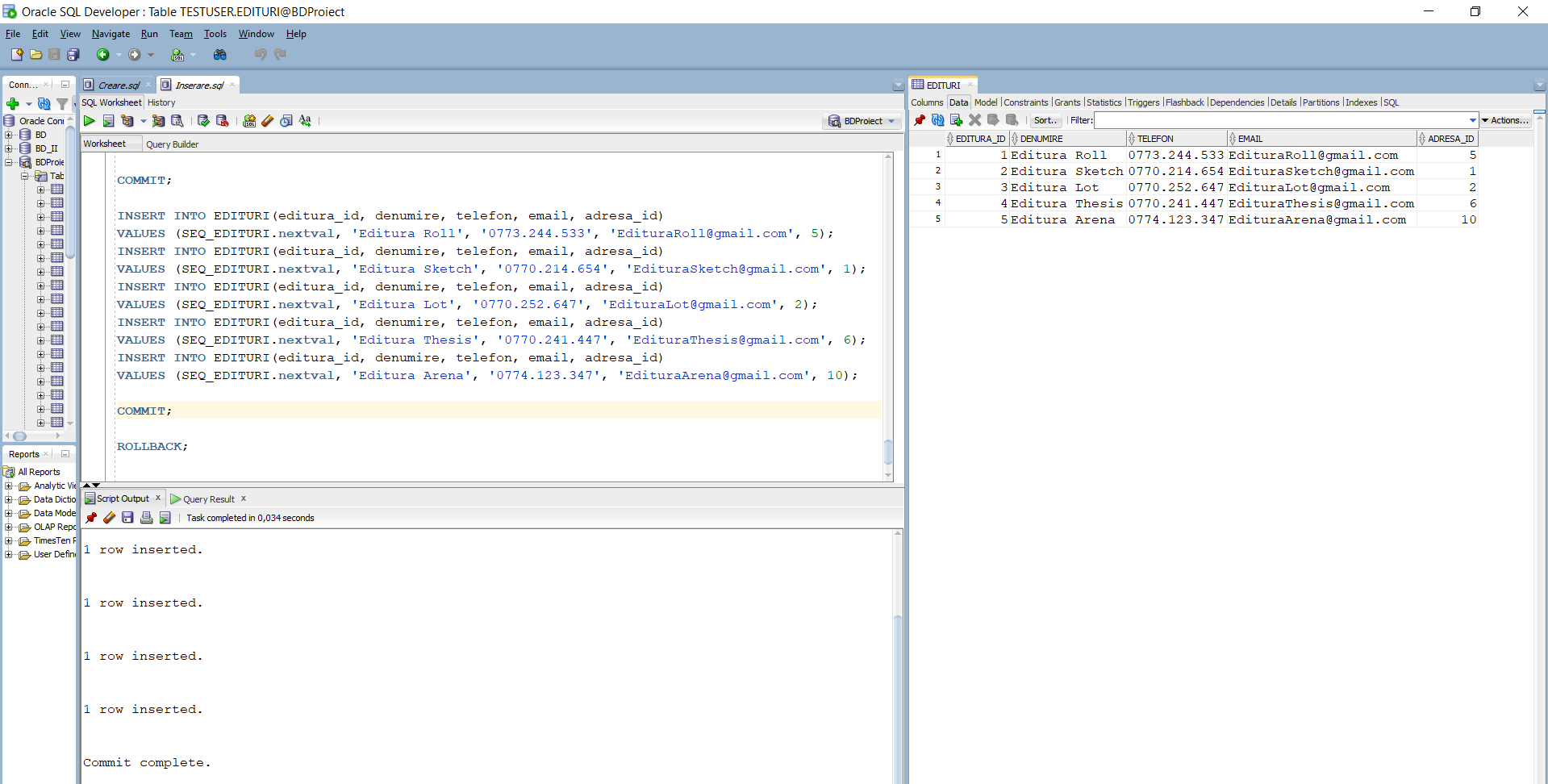
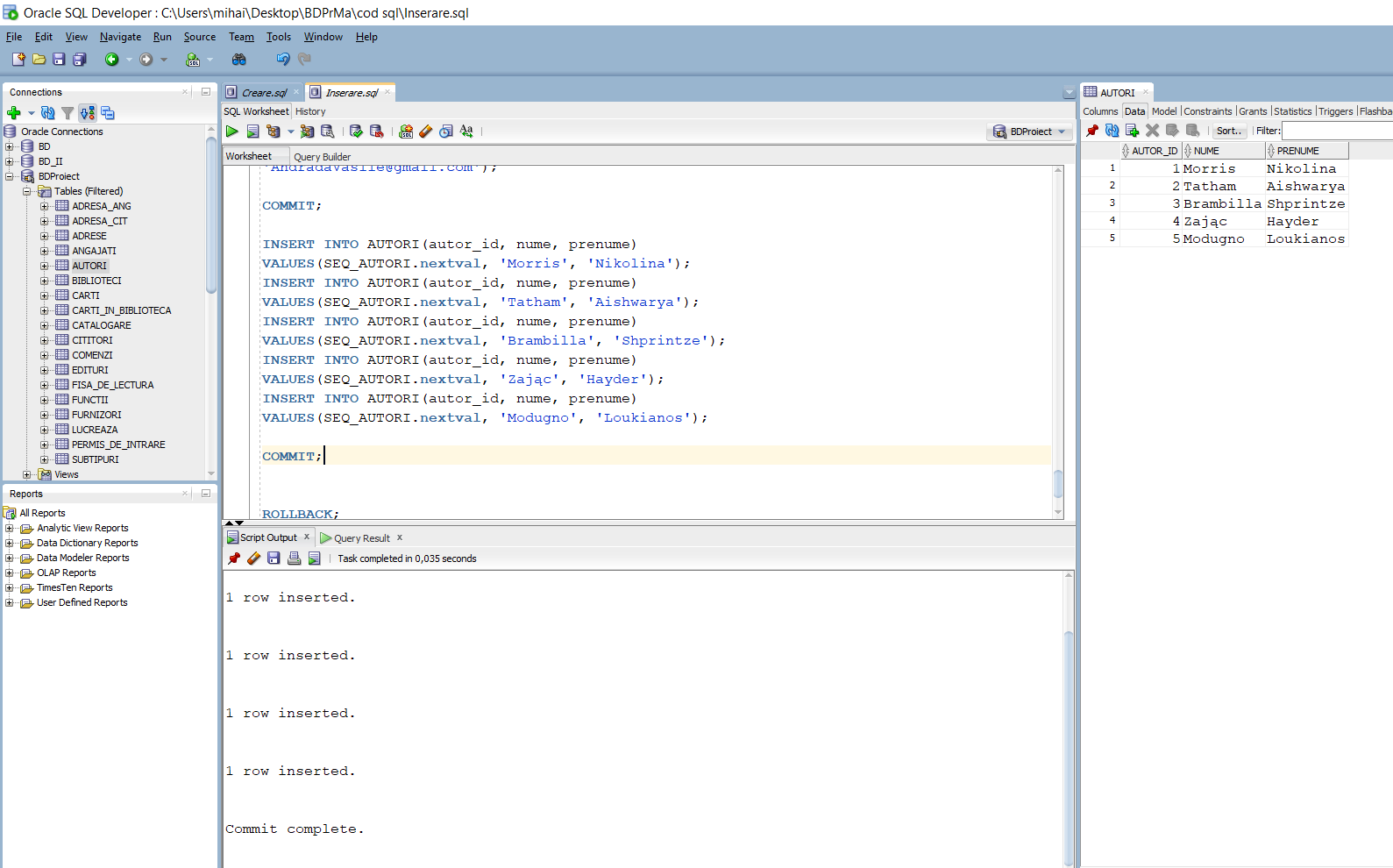
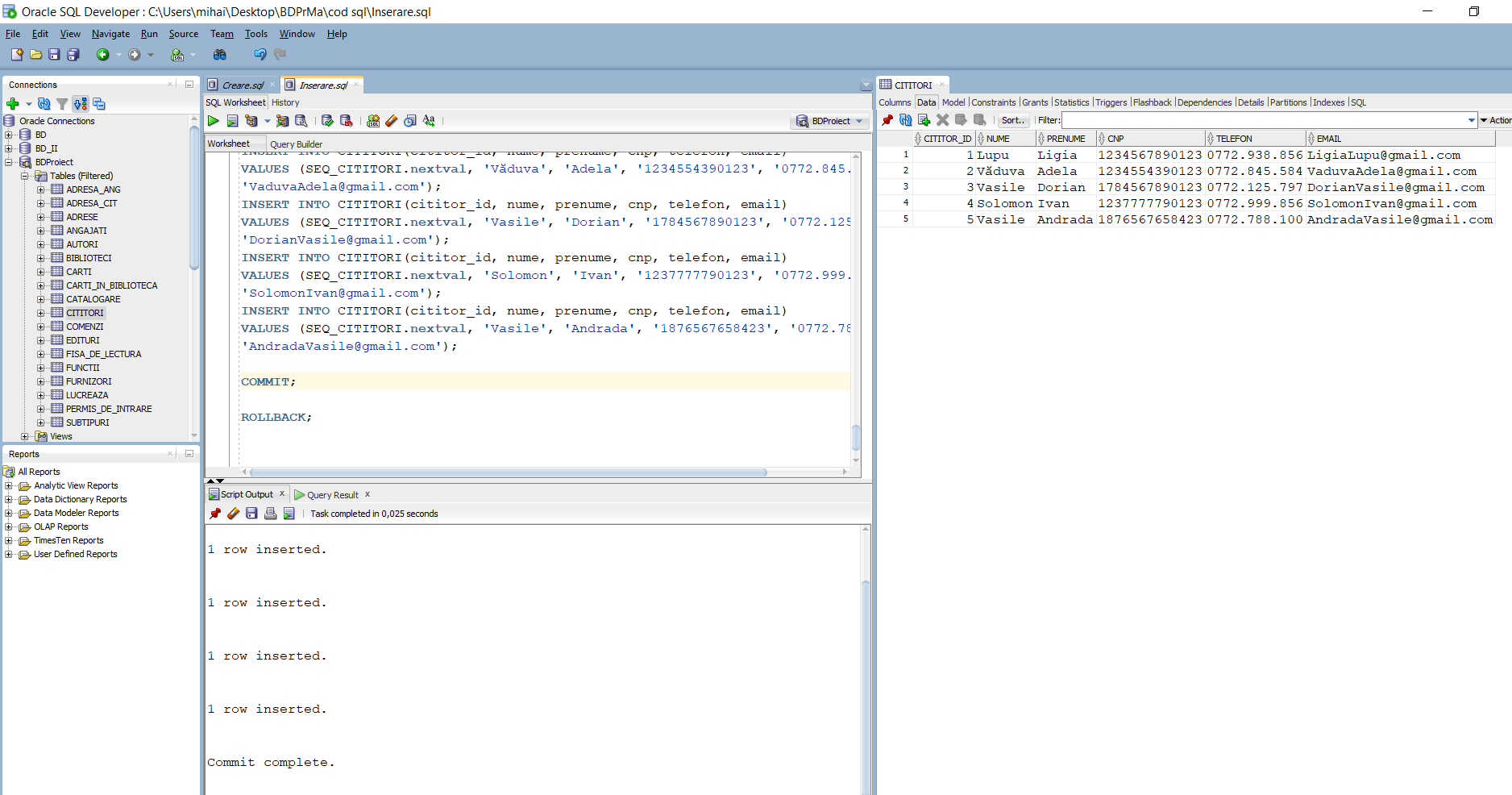
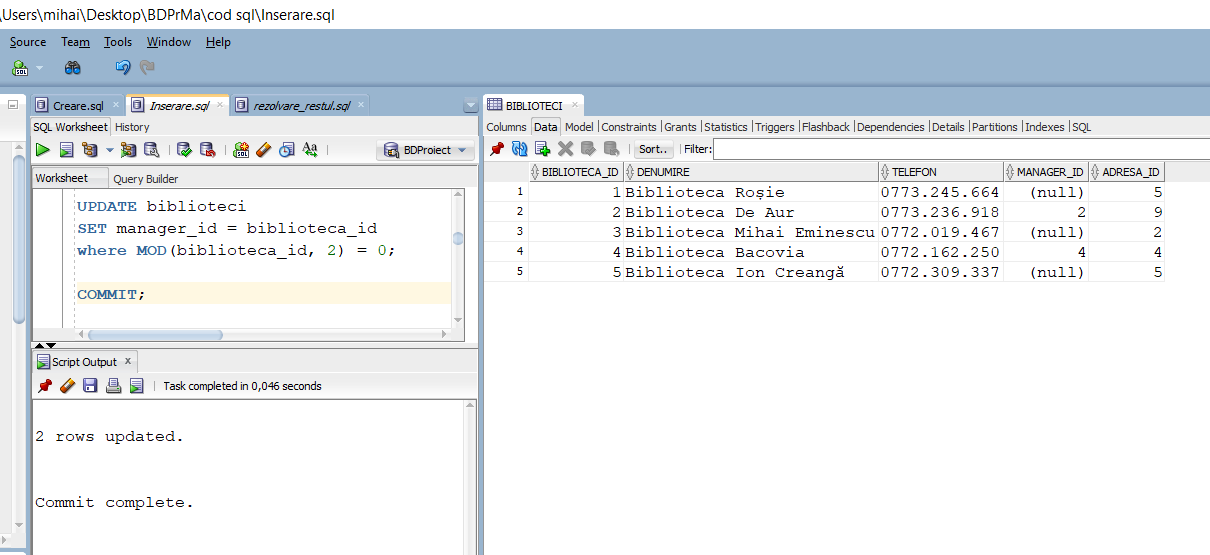
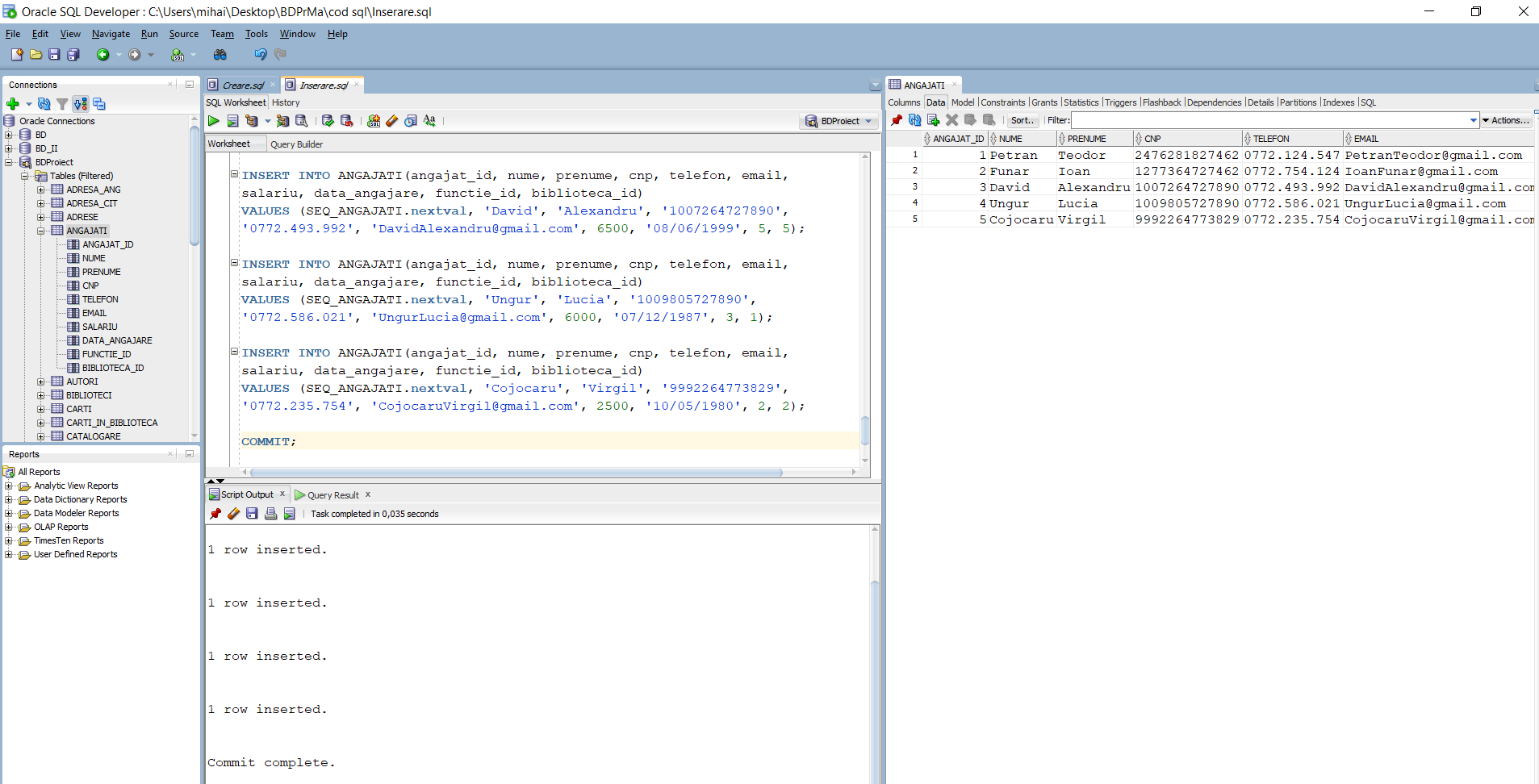
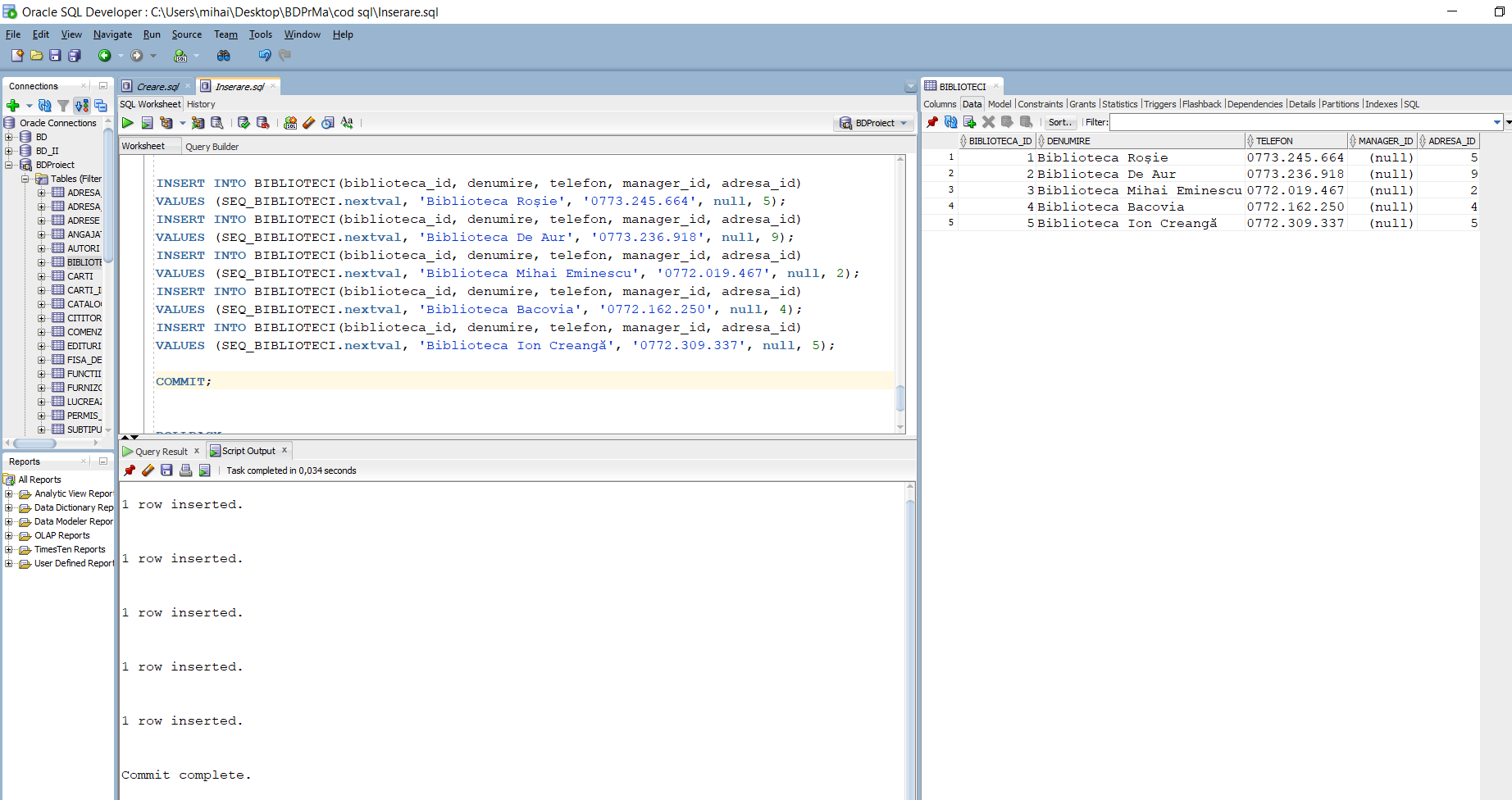
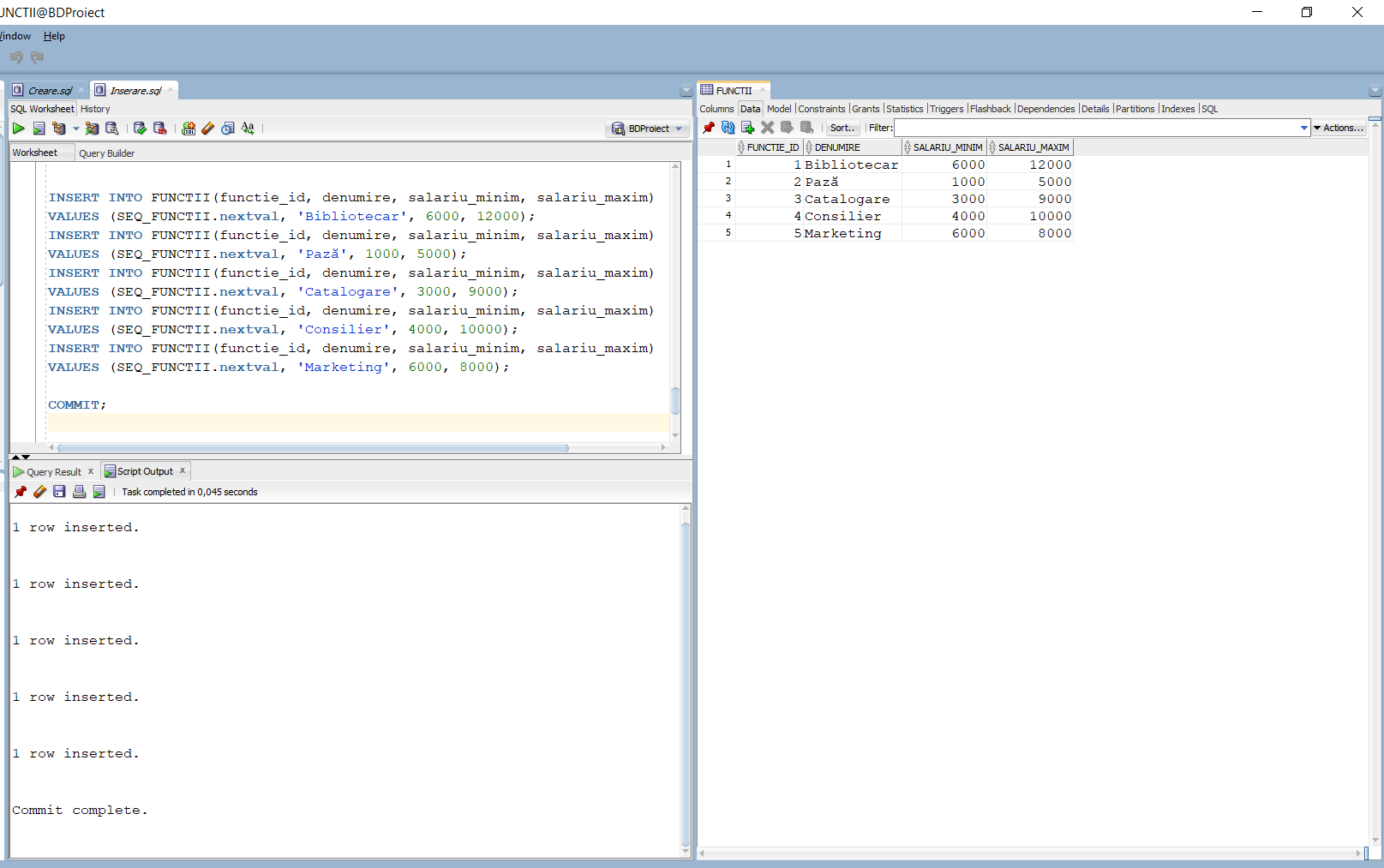
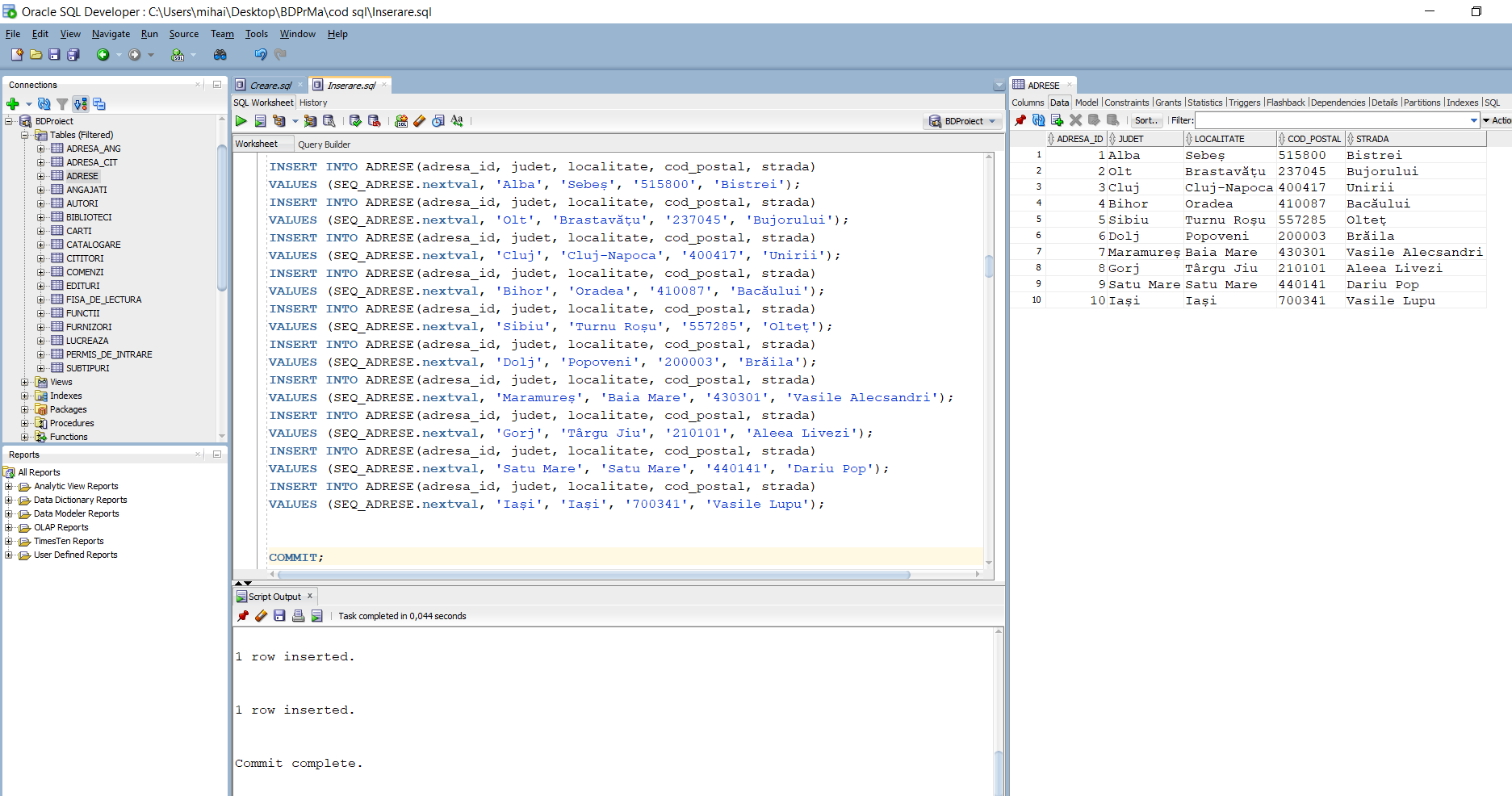
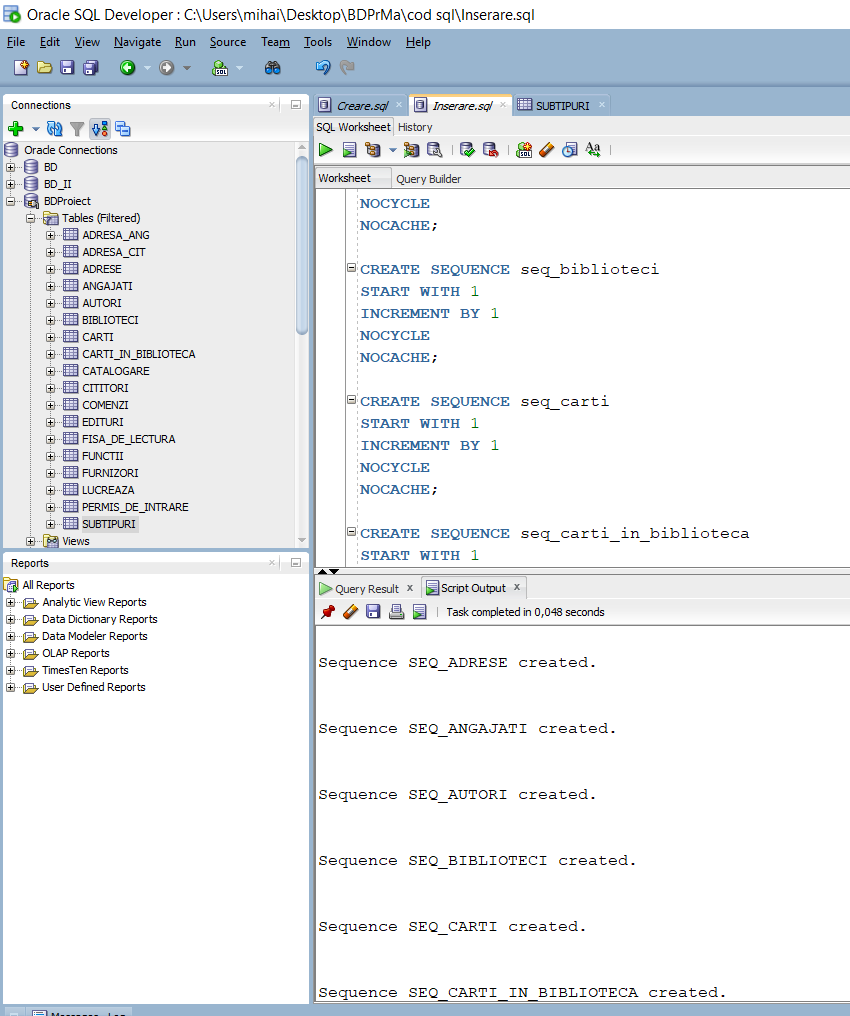
where f.carte\_id = c1.carte\_id) c2

where cit.cititor\_id = c2.cititor\_id;



Creare tabele: 



Inserare date:

Rezolvari:

/\* Afișați datele despre cărțile care au fost citite de un

cititor care are litera 'u' în numele lui, au un permis făcut după anul 2010,

titlul cărți conține litera 'p' și afișați dacă cartea a fost scrisă de Modugno\*/

select c.titlu, c.tip\_carte, su.denumire subtip,

decode(a.nume, 'Modugno', 'Este scrisa de Modugno', 'Nu este scrisa de Modugno') as scrisa\_de\_moudgno

from carti c, autori a, catalogare ca, subtipuri su

where c.carte\_id = ca.carte\_id

and ca.subtip\_id = su.subtip\_id

and c.autor\_id = a.autor\_id

and c.carte\_id in (

select fisa.carte\_id

from fisa\_de\_lectura fisa, cititori cit, permis\_de\_intrare perm

where fisa.cititor\_id = cit.cititor\_id

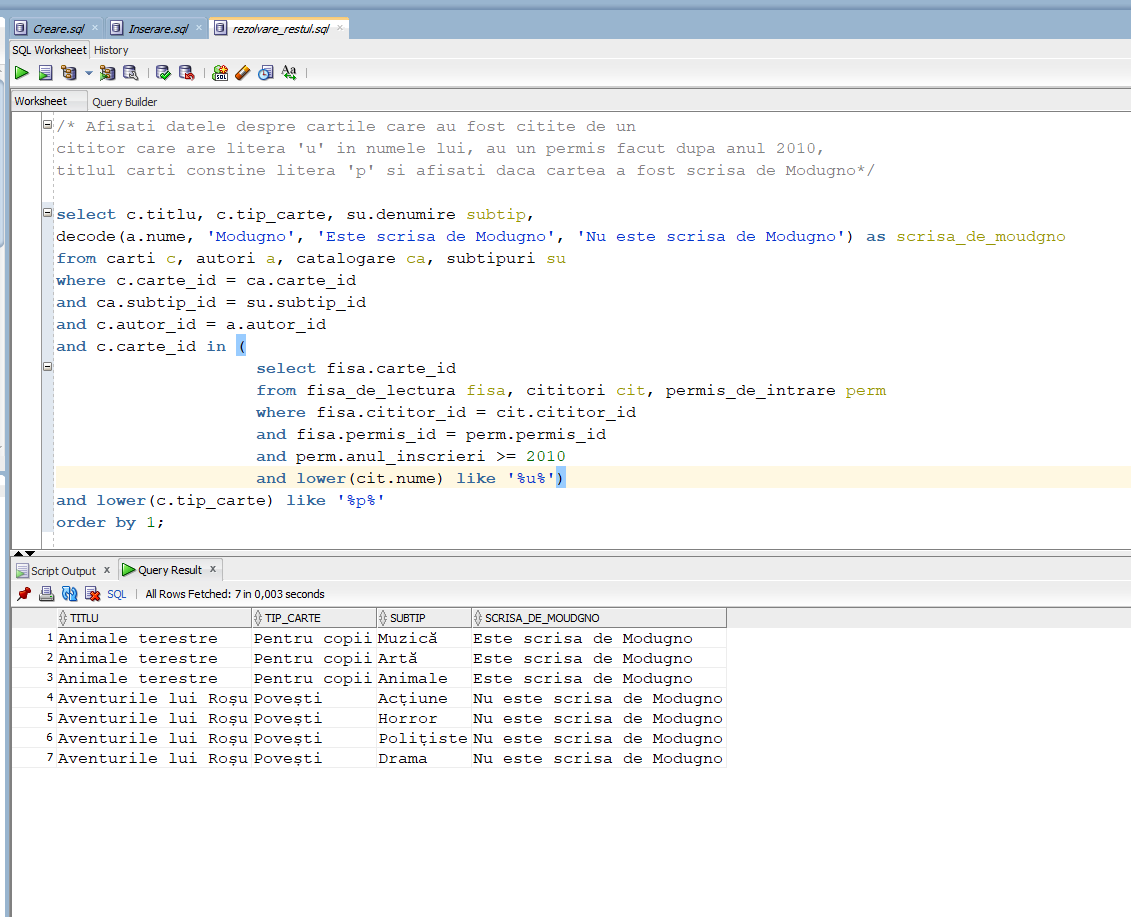
and fisa.permis\_id = perm.permis\_id

and perm.anul\_inscrieri >= 2010

and lower(cit.nume) like '%u%')

and lower(c.tip\_carte) like '%p%'

order by 1;



/\* Afișați numele, prenume și titlul cărți

pentru cititori care au citit o carte de la editura Sketch

scrisă de autorul cu numele Morris\*/

select distinct cit.nume, cit.prenume, ca.titlu

from cititori cit, fisa\_de\_lectura fisa, carti ca

where cit.cititor\_id = fisa.cititor\_id

and ca.carte\_id = fisa.carte\_id

and exists (

select 1

from carti c, autori a, edituri e

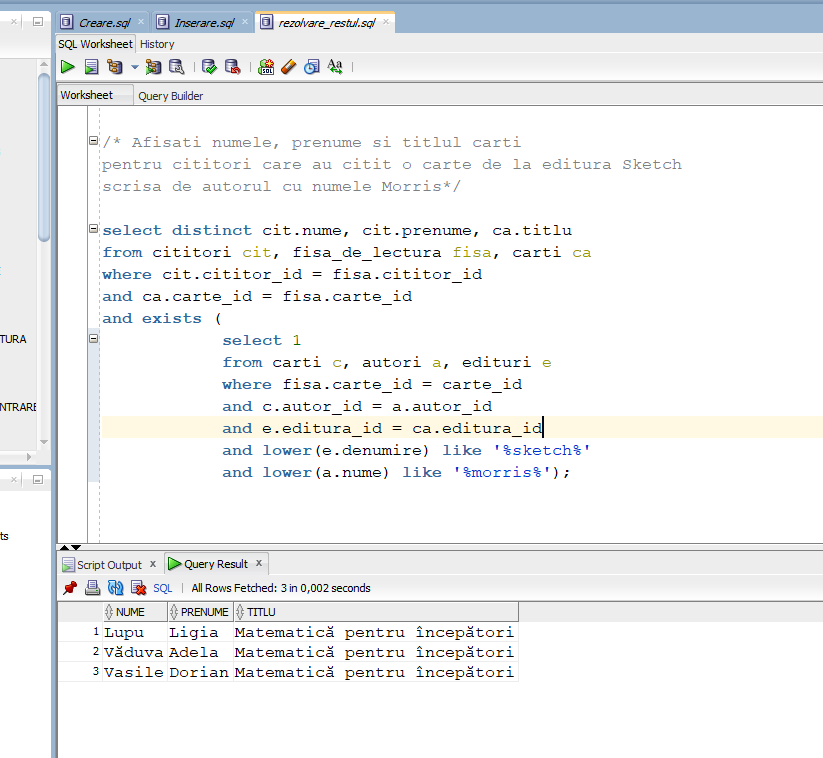
where fisa.carte\_id = carte\_id

and c.autor\_id = a.autor\_id

and e.editura\_id = ca.editura\_id

and lower(e.denumire) like '%sketch%'

and lower(a.nume) like '%morris%');



/\* Să se afișeze cărțile citite de, cititorul care a citit cel mai mult

și acum câți ani și câte luni au fost împrumutate ele \*/

select c.nume, c.prenume, ca.titlu, concat('Imprumutata de el acum ',

to\_char(sysdate, 'YYYY') - to\_char(fisa.data\_imprumut, 'YYYY')) || ' de ani.' ani,

'Imprumutat de el acum '||FLOOR(MONTHS\_BETWEEN(SYSDATE,fisa.data\_imprumut))||' luni.' luni

from fisa\_de\_lectura fisa, cititori c, carti ca,(select fisa.cititor\_id, count(\*)

from fisa\_de\_lectura fisa

group by fisa.cititor\_id

having count(\*) = (select max(count(\*))

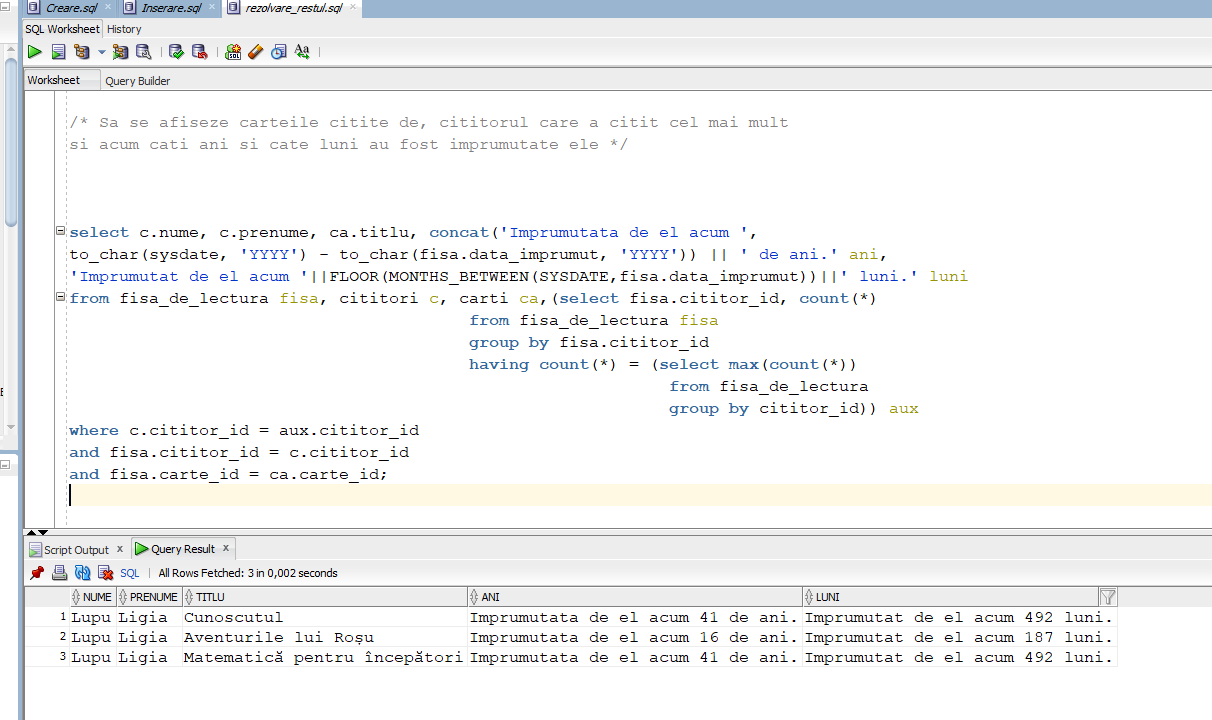
from fisa\_de\_lectura

group by cititor\_id)) aux

where c.cititor\_id = aux.cititor\_id

and fisa.cititor\_id = c.cititor\_id

and fisa.carte\_id = ca.carte\_id;

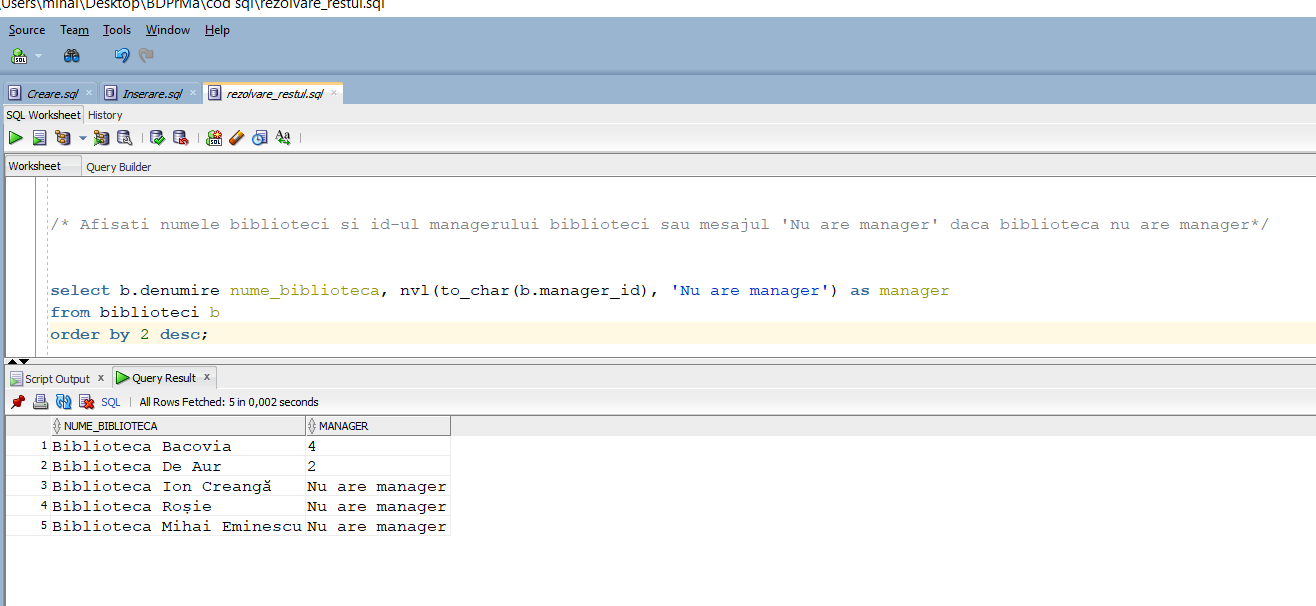


/\* Afișați numele biblioteci și id-ul managerului biblioteci sau mesajul 'Nu are manager' dacă biblioteca nu are manager\*/

select b.denumire nume\_biblioteca, nvl(to\_char(b.manager\_id), 'Nu are manager') as manager

from biblioteci b

order by 2 desc;



/\* Afișați numele, mesajul 'Angajatul are salariul > 6000' dacă angajatul are salariul mai mare ca 6000 sau

mesajul 'Angajatul nu are salariul > 6000' în caz contrar și numărul zilelor rămase până la

sfârșitul luni relativ cu data angajări salariatului\*/

with salariu6000 as (

select angajat\_id

from angajati

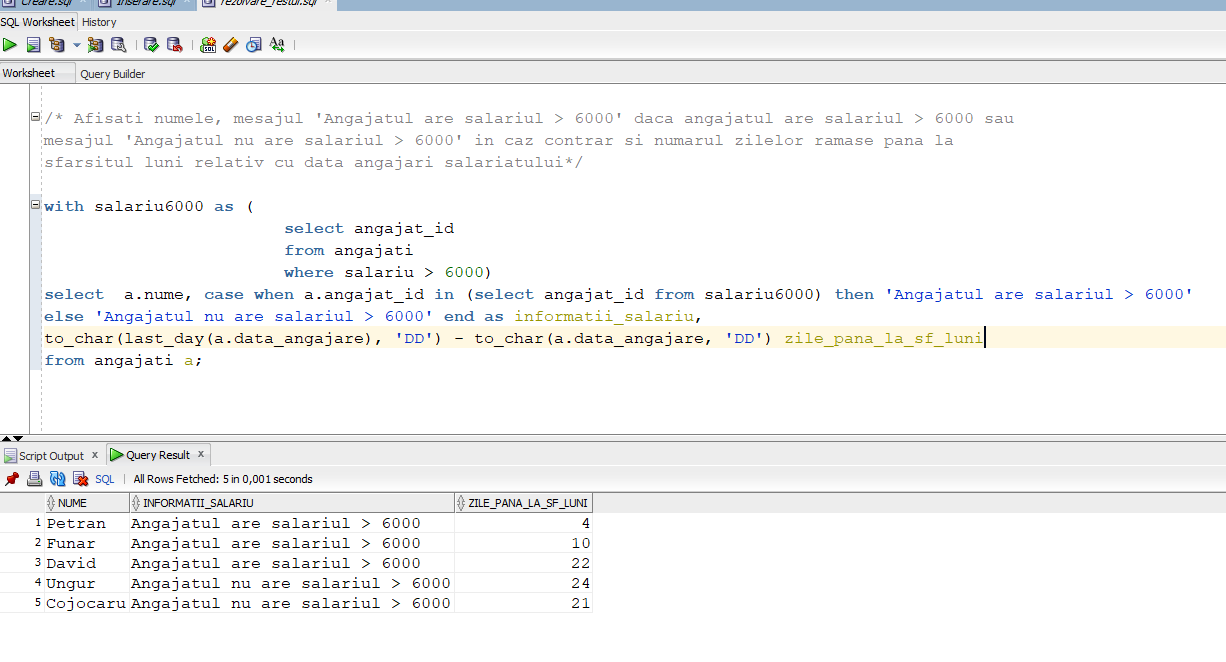
where salariu > 6000)

select a.nume, case when a.angajat\_id in (select angajat\_id from salariu6000) then 'Angajatul are salariul > 6000'

else 'Angajatul nu are salariul > 6000' end as informatii\_salariu,

to\_char(last\_day(a.data\_angajare), 'DD') - to\_char(a.data\_angajare, 'DD') zile\_pana\_la\_sf\_luni

from angajati a;



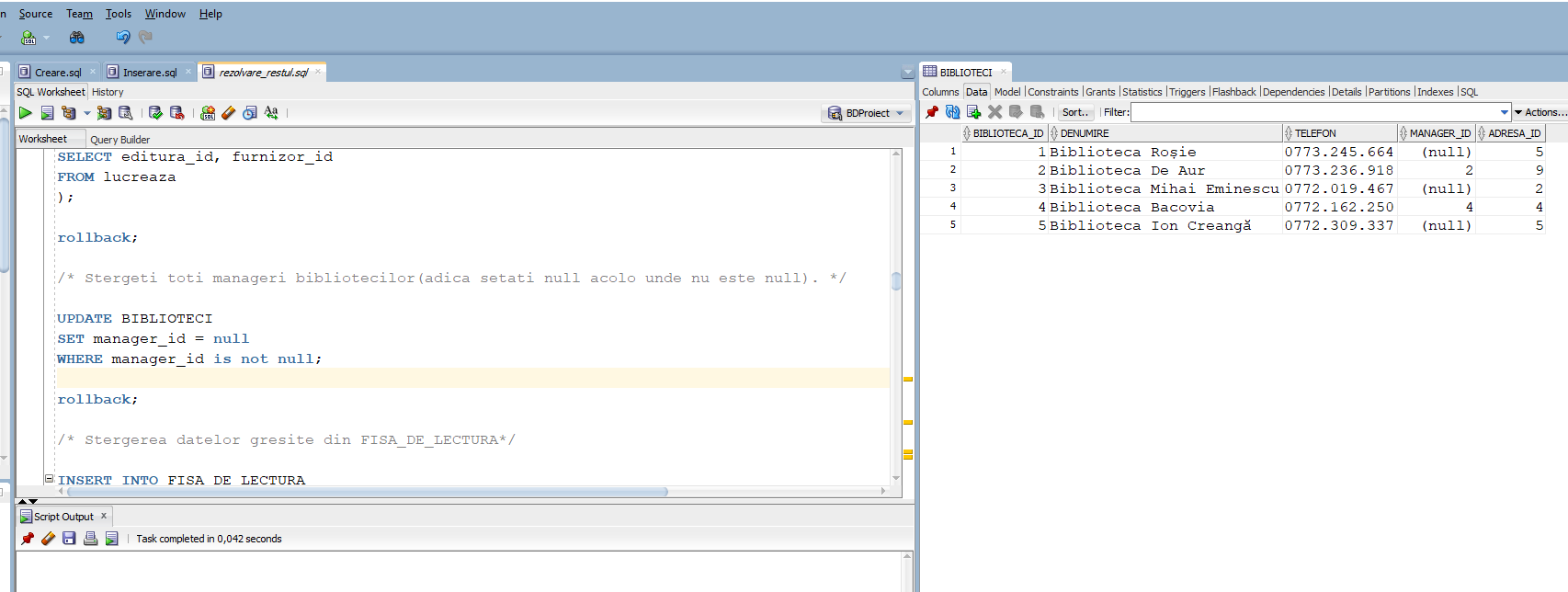
Operati de actualizare sau suprimare:

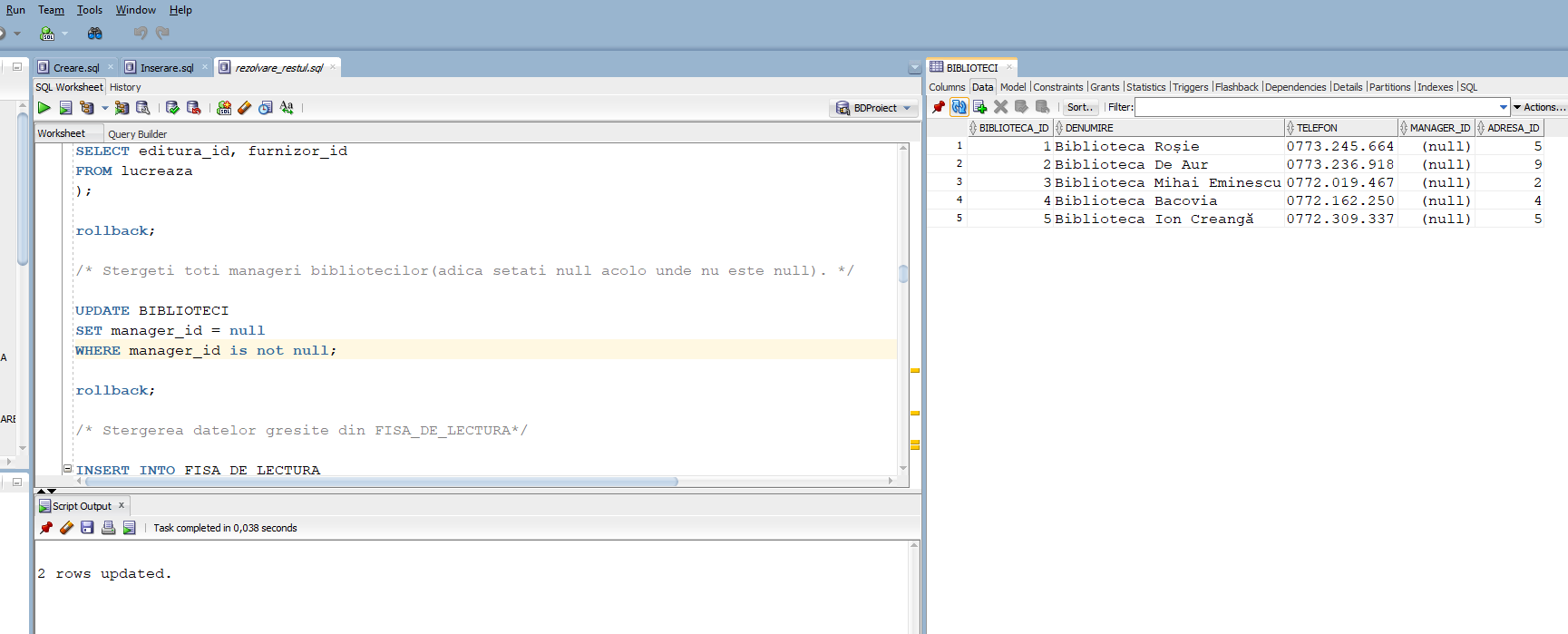
/\* Ștergeți toți manageri bibliotecilor(adică setați null acolo unde nu este null). \*/

UPDATE BIBLIOTECI

SET manager\_id = null

WHERE manager\_id is not null;





/\* Ștergerea datelor greșite din FISA\_DE\_LECTURA\*/

INSERT INTO FISA\_DE\_LECTURA

(fisa\_id, permis\_id, cititor\_id, biblioteca\_id, carte\_id,

data\_imprumut, data\_restituire)

VALUES(11, 10, 1, 5, 1, '11/05/1980', '30/05/1980');

DELETE FROM FISA\_DE\_LECTURA f

WHERE f.carte\_id not in (

select carte\_id

from comenzi

where biblioteca\_id = f.biblioteca\_id)

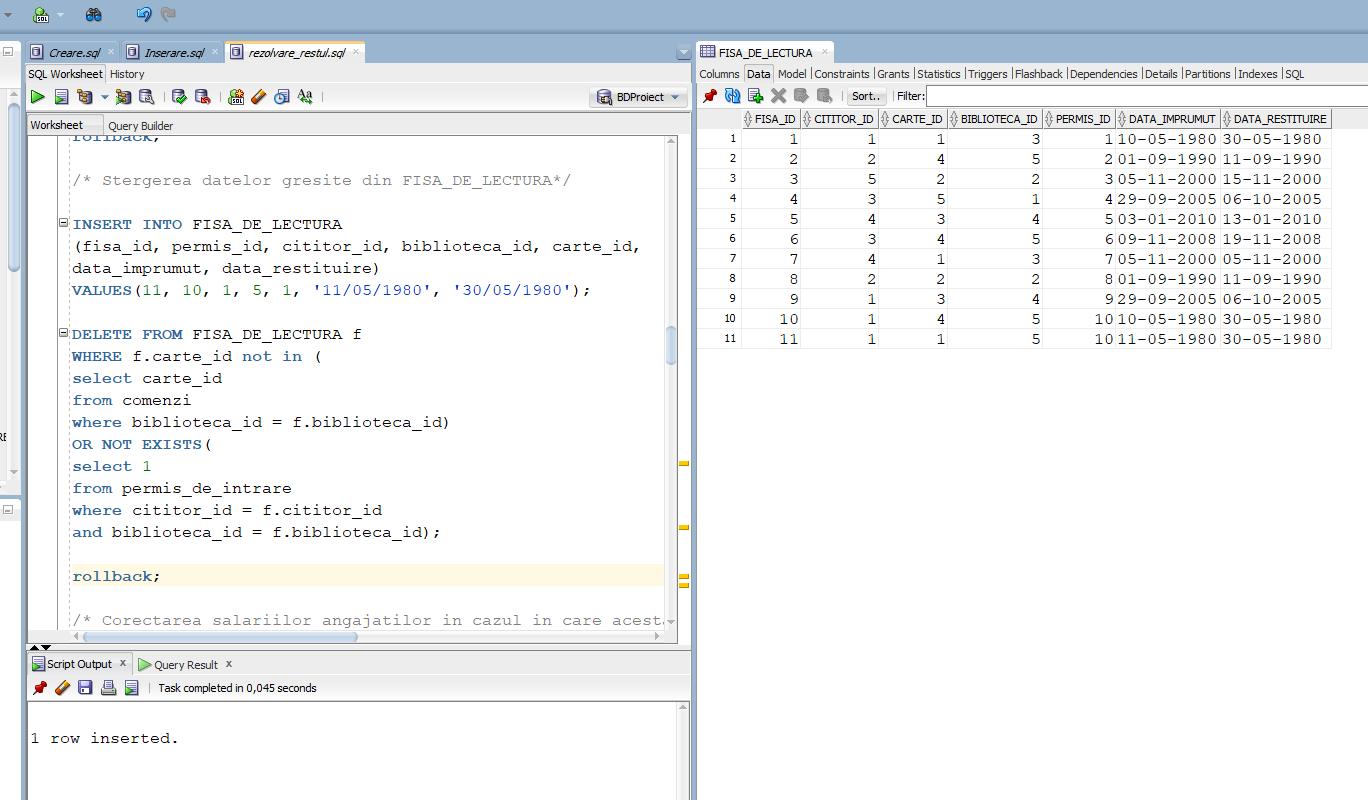
OR NOT EXISTS(

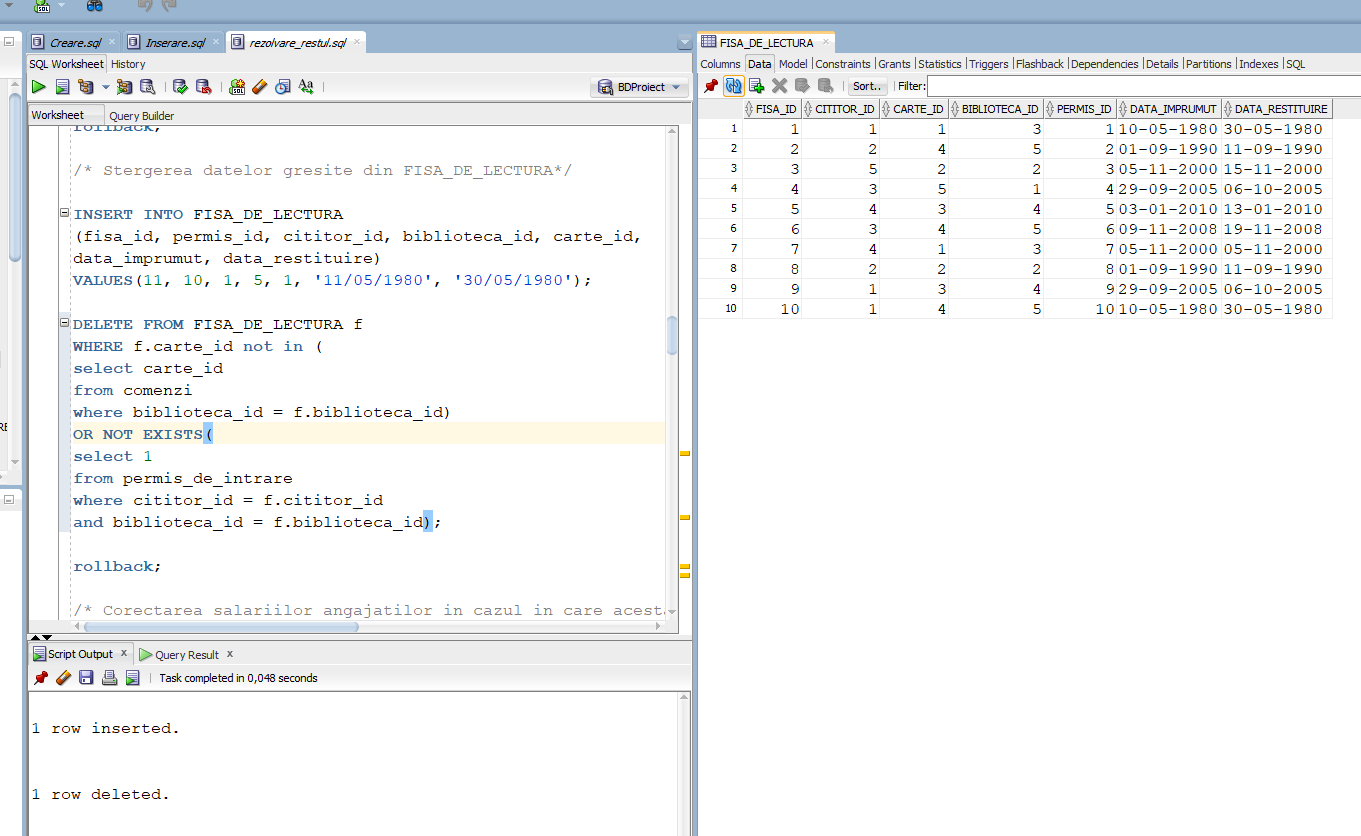
select 1

from permis\_de\_intrare

where cititor\_id = f.cititor\_id

and biblioteca\_id = f.biblioteca\_id);





/\* Corectarea salariilor angajaților în cazul în care acesta este greșit \*/

INSERT INTO ANGAJATI(angajat\_id, nume, prenume, cnp, telefon, email,

salariu, data\_angajare, functie\_id, biblioteca\_id)

VALUES (6, 'Adam', 'Nicu?or', '9882264773829',

'0772.002.778', 'AdamNicu?or@gmail.com', 5000, '10/05/1980', 1, 1);

UPDATE ANGAJATI a

SET a.salariu = (

select salariu\_minim

from functii

where functie\_id = a.functie\_id)

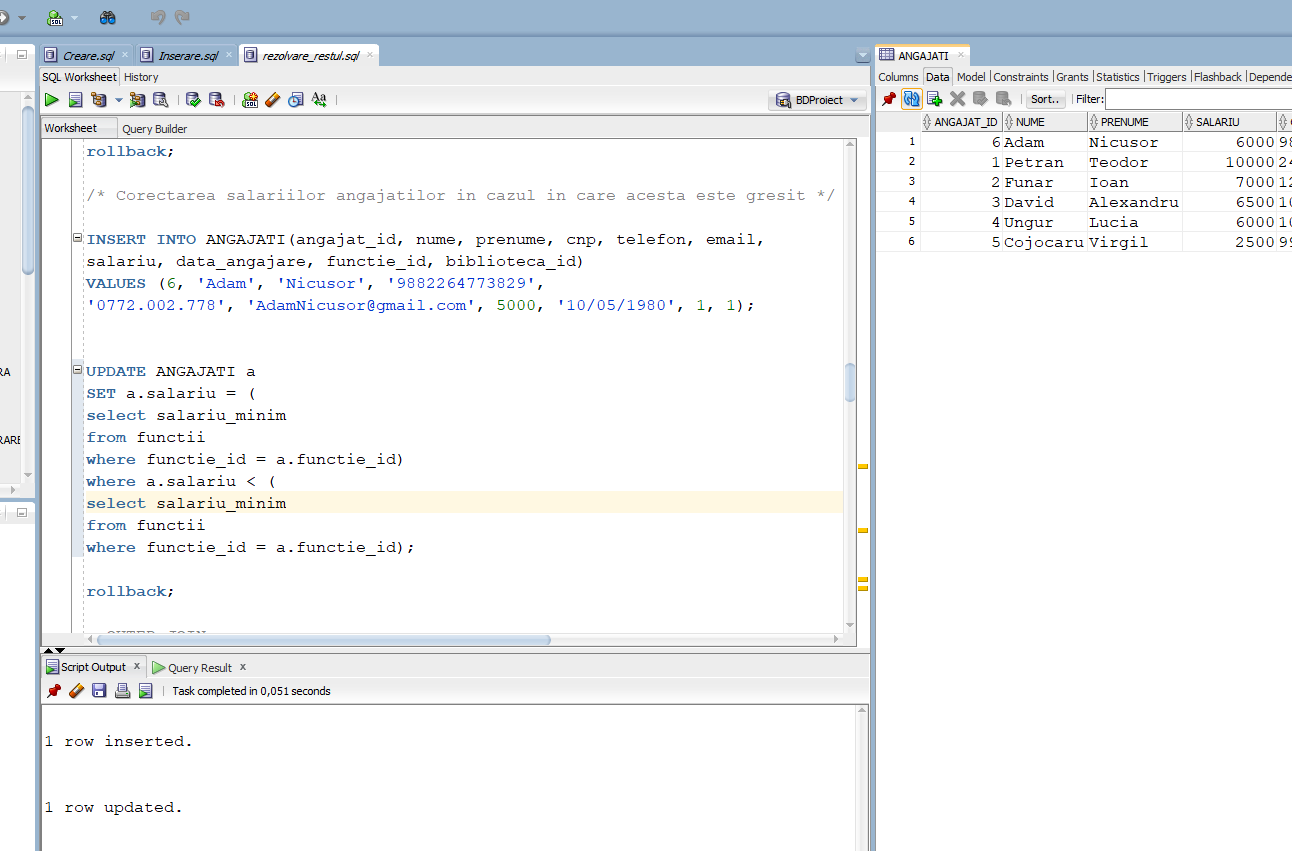
where a.salariu < (

select salariu\_minim

from functii

where functie\_id = a.functie\_id);





Cereri outer-join și division:

/\* Să se afișeze informați despre manageri bibliotecilor \*/

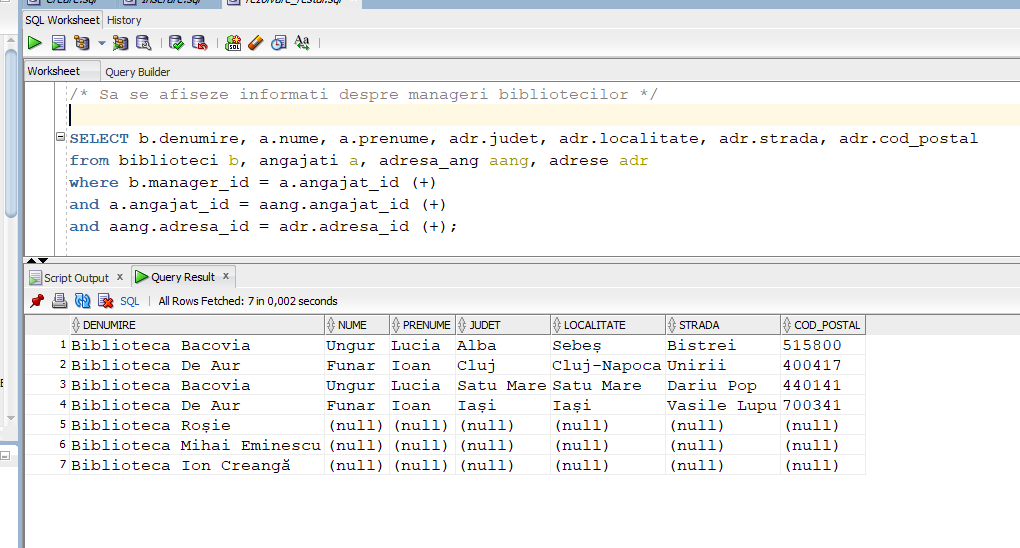
SELECT b.denumire, a.nume, a.prenume, adr.judet, adr.localitate, adr.strada, adr.cod\_postal

from biblioteci b, angajati a, adresa\_ang aang, adrese adr

where b.manager\_id = a.angajat\_id (+)

and a.angajat\_id = aang.angajat\_id (+)

and aang.adresa\_id = adr.adresa\_id (+);



/\* Să se afișeze toți angajați care locuiesc la toate județele care au cel putin o literă 'o' în ele \*/

INSERT INTO ANGAJATI(angajat\_id, nume, prenume, cnp, telefon, email,

salariu, data\_angajare, functie\_id, biblioteca\_id)

VALUES (6, 'Dumitru', 'Carol', '9966664773829',

'0772.205.288', 'DumitruCarol@gmail.com', 2500, '10/05/1980', 2, 2);

INSERT INTO ADRESA\_ANG(angajat\_id, adresa\_id)

VALUES(6, 2);

INSERT INTO ADRESA\_ANG(angajat\_id, adresa\_id)

VALUES(6, 4);

INSERT INTO ADRESA\_ANG(angajat\_id, adresa\_id)

VALUES(6, 6);

INSERT INTO ADRESA\_ANG(angajat\_id, adresa\_id)

VALUES(6, 8);

/\* Codul de mai sus este adăugat cu scopul de a afișa un răspuns deoarece in baza de date nu există un angajat care să satisfacă cererea \*/

select \*

from angajati a

where not exists (

select 1

from adrese adr

where lower(judet) like '%o%'

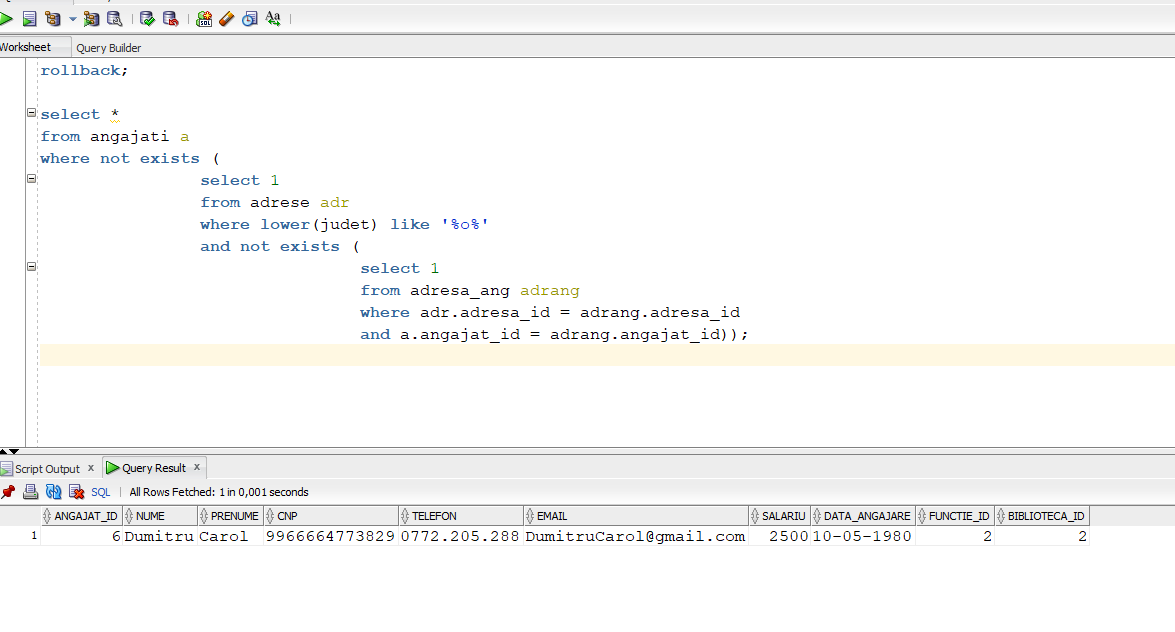
and not exists (

select 1

from adresa\_ang adrang

where adr.adresa\_id = adrang.adresa\_id

and a.angajat\_id = adrang.angajat\_id));



/\* Să se afișeze toți furnizori care lucrează cu toate editurile \*/

INSERT INTO FURNIZORI(furnizor\_id, denumire, telefon, email, adresa\_id)

VALUES (6, 'Auxiliar', '0774.789.316', 'Auxiliar@gmail.com', 5);

INSERT INTO LUCREAZA(editura\_id, furnizor\_id)

VALUES(1, 6);

INSERT INTO LUCREAZA(editura\_id, furnizor\_id)

VALUES(2, 6);

INSERT INTO LUCREAZA(editura\_id, furnizor\_id)

VALUES(3, 6);

INSERT INTO LUCREAZA(editura\_id, furnizor\_id)

VALUES(4, 6);

INSERT INTO LUCREAZA(editura\_id, furnizor\_id)

VALUES(5, 6);

/\* Codul de mai sus este adăugat cu scopul de a afișa un răspuns deoarece in baza de date nu există un furnizor care să satisfacă cererea \*/

select \*

from furnizori a

where not exists (

select 1

from edituri b

where not exists (

select 1

from lucreaza c

where b.editura\_id = c.editura\_id

and a.furnizor\_id = c.furnizor\_id));

