

Proiect Baze de Date

**Student : Butuza Alexandru
Grupa 1066**

CERINTE PROIECT ORACLE

1. Definirea schemei bazei de date – tabele, restrictii de integritate. Se utilizează comenzile CREATE, ALTER, DROP.
2. Exemple cu operatiile de actualizare a datelor (comenzile DML – INSERT, UPDATE, DELETE, MERGE (optional) pentru inregistrari).
3. Exemple de interogari variate (SELECT):
 - Utilizarea operatorilor de comparatie
 - Join-uri
 - Utilizarea functiilor de grup si conditii asupra acestora
 - Utilizarea functiilor numerice, de tip caracter, pentru data si timp
 - Construirea de expresii cu DECODE si CASE
 - Utilizarea operatorilor UNION, MINUS, INTERSECT
 - Subcereri (cereri imbricate)
4. Gestiunea altor obiecte ale bazei de date: vederi, indecsi, sinonime, secvente etc.

1. Descrierea Bazei de Date:

Obiectivul proiectului este să se creeze o bază de date pentru gestionarea unei biblioteci. Aceasta îi va permite bibliotecii să urmărească toate împrumuturile făcute. Baza de date va conține următoarele tabele:

Tabela Carti:

Atribute: ID_Carte (cheie primară), Titlu, Autor, An_Publicare, ID_Gen (cheie externă către tabelul Genuri)

Tabela Genuri:

Atribute: ID_Gen (cheie primară), Nume_Gen

Tabela Imprumuturi:

Atribute: ID_Imprumut (cheie primară), ID_Carte (cheie externă către tabelul Carti), Data_Imprumut, Data_Restituire, ID_Utilizator (cheie externă către tabelul Utilizatori)

Tabela Utilizatori:

Atribute: ID_Utilizator (cheie primară), Nume_Utilizator, Email

Tabela Carti:

ID_Carte	PRIMARY KEY
Titlu	
Autor	
, An_Publicare	
ID_Gen	FOREIGN KEY catre Genuri

Tabela Genuri:

ID_Gen	PRIMARY KEY
Nume_Gen	

Tabela Imprumuturi:

ID_Imprumut	PRIMARY KEY	
ID_Carte	FOREIGN KEY catre Carti	
Data_Imprumut		
Data_Restituire		
ID_Utilizator	FOREIGN KEY catre Utilizatori	

Tabela Utilizatori:

ID_Utilizator	PRIMARY KEY
Nume_Utilizator	

Email	
-------	--

Restricții și tipuri de legături:

Fiecare tabel are o cheie primara unica pentru a putea identifica in mod unic fiecare inregistrare.

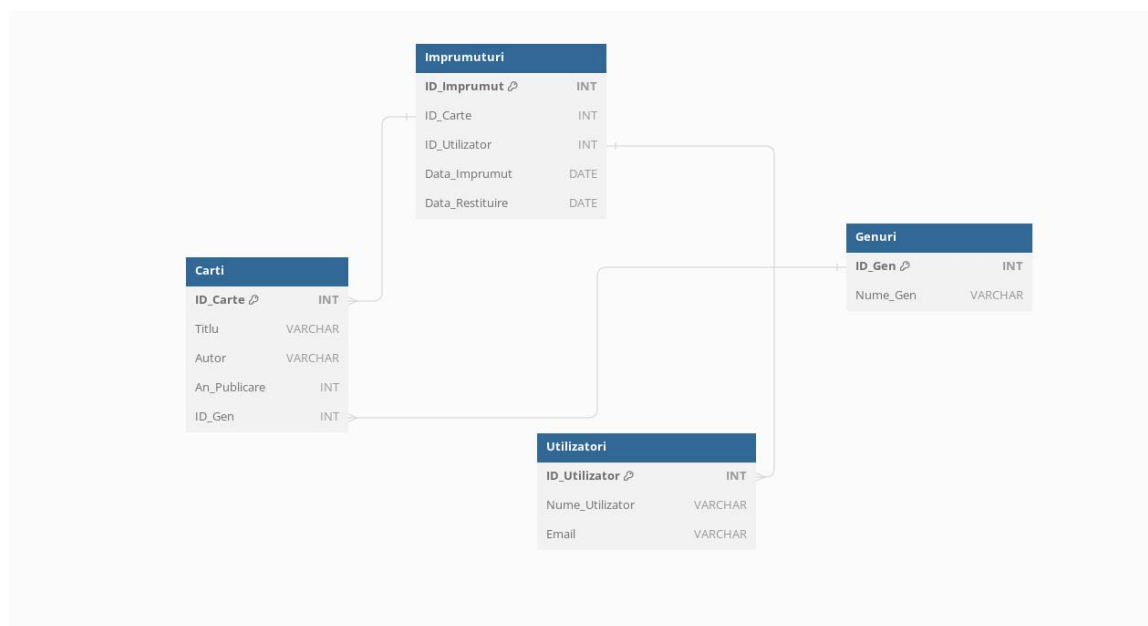
Relatiile straine sunt folosite pentru a conecta inregistrarile dintre tabele.

Legătura dintre tabela Carti și Genuri este realizată prin cheia externă ID_Gen.

Legătura dintre tabela Imprumuturi și Carti este realizată prin cheia externă ID_Carte.

Legătura dintre tabela Imprumuturi și Utilizatori este realizată prin cheia externă ID_Utilizator.

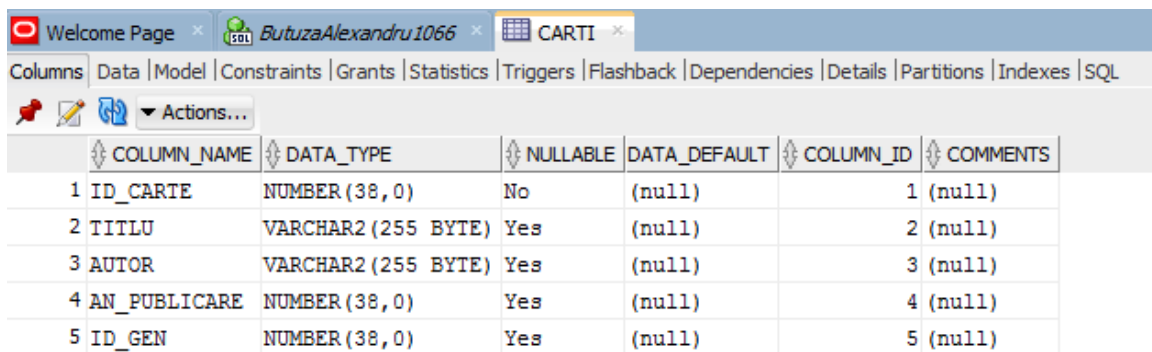
2.Diagrama bazei de date:



3.Crearea tabelelor:

Tabela cartii:

```
CREATE TABLE Carti (  
    ID_Carte INT PRIMARY KEY,  
    Titlu VARCHAR(255),  
    Autor VARCHAR(255),  
    An_Publicare INT,  
    ID_Gen INT,  
    FOREIGN KEY (ID_Gen) REFERENCES Genuri(ID_Gen)  
);
```

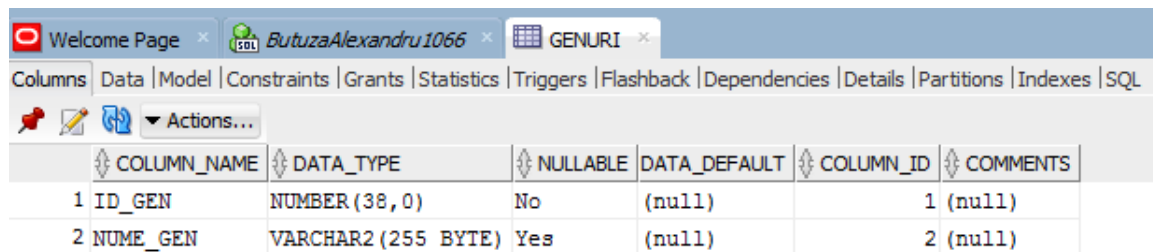


The screenshot shows a database management interface with a tab labeled 'CARTI'. Below the tab, there is a navigation bar with options: Columns, Data, Model, Constraints, Grants, Statistics, Triggers, Flashback, Dependencies, Details, Partitions, Indexes, and SQL. The 'Columns' tab is selected, displaying a table with the following structure:

	↕ COLUMN_NAME	↕ DATA_TYPE	↕ NULLABLE	DATA_DEFAULT	↕ COLUMN_ID	↕ COMMENTS
1	ID_CARTE	NUMBER(38,0)	No	(null)	1	(null)
2	TITLU	VARCHAR2(255 BYTE)	Yes	(null)	2	(null)
3	AUTOR	VARCHAR2(255 BYTE)	Yes	(null)	3	(null)
4	AN_PUBLICARE	NUMBER(38,0)	Yes	(null)	4	(null)
5	ID_GEN	NUMBER(38,0)	Yes	(null)	5	(null)

Tabela Genuri:

```
CREATE TABLE Genuri (  
    ID_Gen INT PRIMARY KEY,  
    Nume_Gen VARCHAR(255)  
);
```

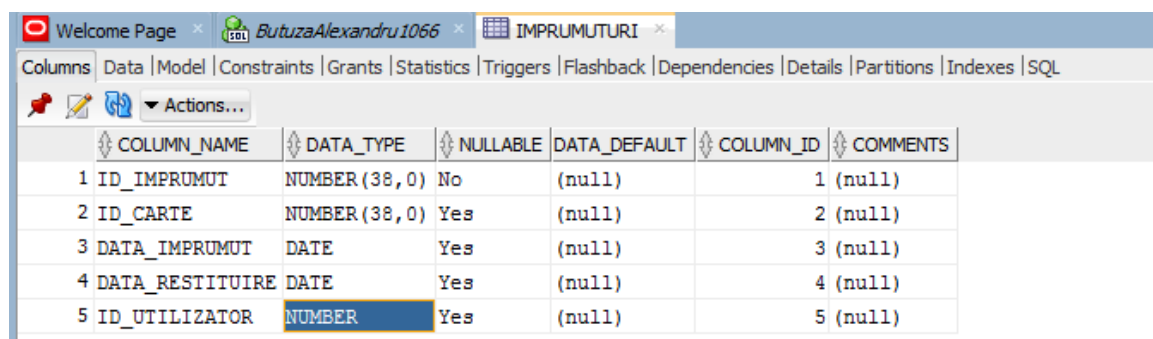


The screenshot shows the SQL Developer interface with the 'GENURI' table selected. The 'Columns' tab is active, displaying the following table structure:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_GEN	NUMBER (38, 0)	No	(null)	1	(null)
2	NUME_GEN	VARCHAR2 (255 BYTE)	Yes	(null)	2	(null)

Tabela imprumuturi:

```
CREATE TABLE Imprumuturi (  
    ID_Imprumut INT PRIMARY KEY,  
    ID_Carte INT,  
    ID_Utilizator INT,  
    Data_Imprumut DATE,  
    Data_Restituire DATE,  
    FOREIGN KEY (ID_Carte) REFERENCES Carti(ID_Carte),  
    FOREIGN KEY (ID_Utilizator) REFERENCES Utilizatori(ID_Utilizator)  
);
```



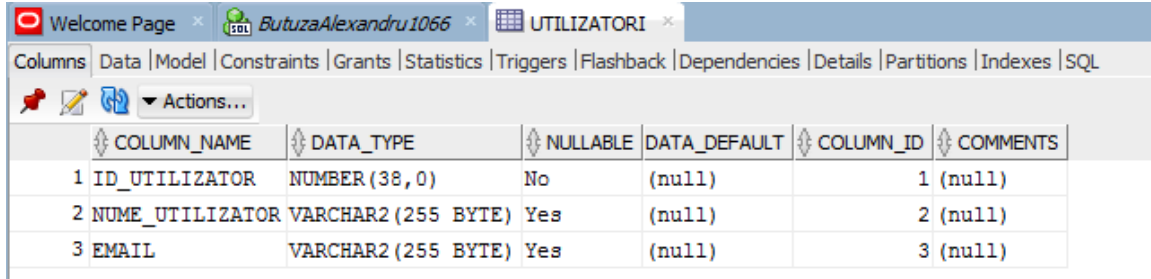
The screenshot shows the SQL Developer interface with the 'IMPRUMUTURI' table selected. The 'Columns' tab is active, displaying the following table structure:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_IMPRUMUT	NUMBER (38, 0)	No	(null)	1	(null)
2	ID_CARTE	NUMBER (38, 0)	Yes	(null)	2	(null)
3	DATA_IMPRUMUT	DATE	Yes	(null)	3	(null)
4	DATA_RESTITUIRE	DATE	Yes	(null)	4	(null)
5	ID_UTILIZATOR	NUMBER	Yes	(null)	5	(null)

Tabela Utilizatori:

```
CREATE TABLE Utilizatori (  
    ID_Utilizator INT PRIMARY KEY,  
    Nume_Utilizator VARCHAR(255),  
    Email VARCHAR(255)
```

);



The screenshot shows the SQL Developer interface with the 'UTILIZATORI' table selected. The 'Columns' tab is active, displaying the following table structure:

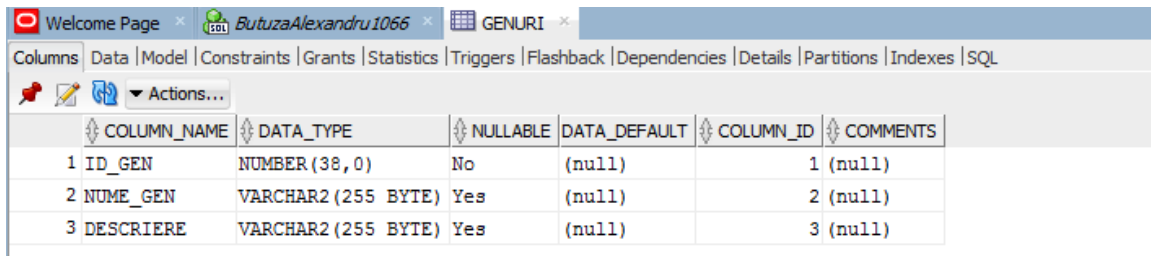
	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_UTILIZATOR	NUMBER(38,0)	No	(null)	1	(null)
2	NUME_UTILIZATOR	VARCHAR2(255 BYTE)	Yes	(null)	2	(null)
3	EMAIL	VARCHAR2(255 BYTE)	Yes	(null)	3	(null)

4. Actualizarea Structurii Tabelor și Modificarea Restrictiilor de Integritate:

4.1 Adăugarea unei coloane pentru "Genuri":

```
ALTER TABLE Genuri
```

```
ADD COLUMN Descriere VARCHAR(255);
```



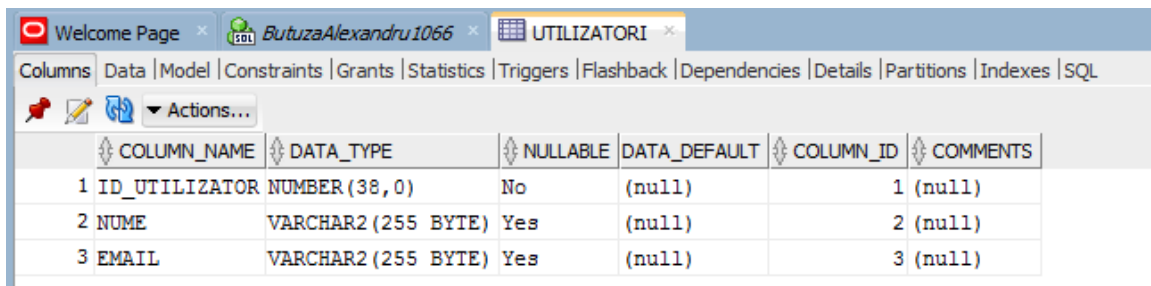
The screenshot shows the SQL Developer interface with the 'GENURI' table selected. The 'Columns' tab is active, displaying the following table structure:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_GEN	NUMBER(38,0)	No	(null)	1	(null)
2	NUME_GEN	VARCHAR2(255 BYTE)	Yes	(null)	2	(null)
3	DESCRIERE	VARCHAR2(255 BYTE)	Yes	(null)	3	(null)

4.2 Schimbarea numelui unei coloane pentru "Utilizatori":

```
ALTER TABLE Utilizatori
```

```
RENAME COLUMN Nume_Utilizator TO Nume;
```



The screenshot shows the SQL Developer interface with the 'UTILIZATORI' table selected. The 'Columns' tab is active, displaying the following table structure:

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_UTILIZATOR	NUMBER(38,0)	No	(null)	1	(null)
2	NUME	VARCHAR2(255 BYTE)	Yes	(null)	2	(null)
3	EMAIL	VARCHAR2(255 BYTE)	Yes	(null)	3	(null)

5. Adăugarea (min 10, max 20) de înregistrări în fiecare tabelă

Tabela Genuri:

```
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(3, 'Fictiune', 'Gen literar fictiv în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(4, 'Științifico-Fantastic', 'Gen literar SF în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(5, 'Mister', 'Gen literar de mister în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(6, 'Romantic', 'Gen literar romantic în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(7, 'Istoric', 'Gen literar istoric în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(8, 'Poezie', 'Gen literar de poezie în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(9, 'Teatru', 'Gen literar de teatru în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(10, 'Adventure', 'Gen literar de aventuri în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES(11, 'Biografie', 'Gen literar de biografie în limba română'),
INSERT INTO Genuri(ID_Gen,Nume_Gen,Descriere)
VALUES (12, 'Comedie', 'Gen literar de comedie în limba română');
```


Query Builder

```
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (3, 'Fictiune', 'Gen literar fictiv în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (4, 'Științifico-Fantastic', 'Gen literar SF în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (5, 'Mister', 'Gen literar de mister în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (6, 'Romantic', 'Gen literar romantic în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (7, 'Istoric', 'Gen literar istoric în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (8, 'Poezie', 'Gen literar de poezie în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (9, 'Teatru', 'Gen literar de teatru în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (10, 'Adventure', 'Gen literar de aventuri în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (11, 'Biografie', 'Gen literar de biografie în limba română');
INSERT INTO Genuri (ID_Gen, Nume_Gen, Descriere)
VALUES (12, 'Comedie', 'Gen literar de comedie în limba română');
```

Script Output

Task completed in 0.35 seconds

1 row inserted.

1 row inserted.

	ID_GEN	NUME_GEN	DESCRIERE
1	3	Fictiune	Gen literar fictiv în limba română
2	4	Științifico-Fantastic	Gen literar SF în limba română
3	5	Mister	Gen literar de mister în limba română
4	6	Romantic	Gen literar romantic în limba română
5	7	Istoric	Gen literar istoric în limba română
6	8	Poezie	Gen literar de poezie în limba română
7	9	Teatru	Gen literar de teatru în limba română
8	10	Adventure	Gen literar de aventuri în limba română
9	11	Biografie	Gen literar de biografie în limba română
10	12	Comedie	Gen literar de comedie în limba română

Tabela Carti:

```
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen)
VALUES (1, 'Harry Potter și Piatra Filozofala', 'J.K. Rowling', 1997, 3);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(2, 'Dune',
'Frank Herbert', 1965, 4);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(3,
'Sherlock Holmes: Studioul în Rosu', 'Arthur Conan Doyle', 1887, 5);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(4, 'Romeo
și Julieta', 'William Shakespeare', 1597, 6);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(5,
'Cronicile din Narnia: Leul, Vrăjitoarea și Dulapul', 'C.S. Lewis', 1950, 7);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(6, 'Poezii
Selectate', 'Mihai Eminescu', 1884, 8);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(7, 'Hamlet',
'William Shakespeare', 1603, 9);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(8,
'Robinson Crusoe', 'Daniel Defoe', 1719, 10);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(9, 'Steve
Jobs: Biografia', 'Walter Isaacson', 2011, 11);
INSERT INTO Carti(ID_Carte, Titlu, Autor, An_Publicare, ID_Gen) VALUES(10, 'O
noapte furtunoasă', 'I.L. Caragiale', 1885, 12);
```

Worksheet Query Builder

```

INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen)
VALUES (1, 'Harry Potter și Piatra Filozofala', 'J.K. Rowling', 1997, 3);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (2, 'Dune', 'Frank Herbert', 1965, 4);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (3, 'Sherlock Holmes: Studioul în Rosu', 'Arthur Conan Doyle', 1887, 5);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (4, 'Romeo și Julieta', 'William Shakespeare', 1597, 6);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (5, 'Cronicile din Narnia: Leul, Vrăjitoarea și Dulapul', 'C.S. Lewis', 1950, 7);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (6, 'Poezii Selectate', 'Mihai Eminescu', 1884, 8);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (7, 'Hamlet', 'William Shakespeare', 1603, 9);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (8, 'Robinson Crusoe', 'Daniel Defoe', 1719, 10);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (9, 'Steve Jobs: Biografia', 'Walter Isaacson', 2011, 11);
INSERT INTO Carti(ID_Carte,Titlu,Autor,An_Publicare,ID_Gen) VALUES (10, 'O noapte furtunoasă', 'I.L. Caragiale', 1885, 12);

```

Script Output

Task completed in 0.273 seconds

1 row inserted.

1 row inserted.

ID_CARTE	TITLU	AUTOR	AN_PUBLICARE	ID_GEN
1	Harry Potter și Piatra Filozofala	J.K. Rowling	1997	3
2	Dune	Frank Herbert	1965	4
3	Sherlock Holmes: Studioul în Rosu	Arthur Conan Doyle	1887	5
4	Romeo și Julieta	William Shakespeare	1597	6
5	Cronicile din Narnia: Leul, Vrăjitoarea și Dulapul	C.S. Lewis	1950	7
6	Poezii Selectate	Mihai Eminescu	1884	8
7	Hamlet	William Shakespeare	1603	9
8	Robinson Crusoe	Daniel Defoe	1719	10
9	Steve Jobs: Biografia	Walter Isaacson	2011	11
10	O noapte furtunoasă	I.L. Caragiale	1885	12

Tabela Utilizatori:

```

INSERT INTO
Imprumuturi(ID_Impumut,ID_Carte,ID_Utilizator>Data_Impumut>Data_Restituire)
VALUES(1, 1, 1, TO_DATE('2022-01-05', 'YYYY-MM-DD'), TO_DATE('2022-01-15',
'YYYY-MM-DD'));
INSERT INTO
Imprumuturi(ID_Impumut,ID_Carte,ID_Utilizator>Data_Impumut>Data_Restituire)
VALUES(2, 2, 2, TO_DATE('2022-01-08', 'YYYY-MM-DD'), TO_DATE('2022-01-18',
'YYYY-MM-DD'));

```

```

INSERT                                                    INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(3, 3, 3, TO_DATE('2022-01-10', 'YYYY-MM-DD'), TO_DATE('2022-01-20',
'YYYY-MM-DD'));
INSERT                                                    INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(4, 4, 4, TO_DATE('2022-01-12', 'YYYY-MM-DD'), TO_DATE('2022-01-22',
'YYYY-MM-DD'));
INSERT                                                    INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(5, 5, 5, TO_DATE('2022-01-15', 'YYYY-MM-DD'), TO_DATE('2022-01-25',
'YYYY-MM-DD'));
INSERT                                                    INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(6, 6, 6, TO_DATE('2022-01-18', 'YYYY-MM-DD'), TO_DATE('2022-01-28',
'YYYY-MM-DD'));
INSERT                                                    INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(7, 7, 7, TO_DATE('2022-01-20', 'YYYY-MM-DD'), TO_DATE('2022-01-30',
'YYYY-MM-DD'));
INSERT                                                    INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(8, 8, 8, TO_DATE('2022-01-22', 'YYYY-MM-DD'), TO_DATE('2022-02-01',
'YYYY-MM-DD'));
INSERT                                                    INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(9, 9, 9, TO_DATE('2022-01-25', 'YYYY-MM-DD'), TO_DATE('2022-02-05',
'YYYY-MM-DD'));
INSERT                                                    INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(10, 10, 10, TO_DATE('2022-01-28', 'YYYY-MM-DD'), TO_DATE('2022-02-
08', 'YYYY-MM-DD'));

```

Worksheet Query Builder

```
INSERT INTO Utilizatori(ID_Utilizator, Nume, Email)
VALUES(1, 'Ana Popescu', 'ana.popescu@example.com');
INSERT INTO Utilizatori VALUES(2, 'Mihai Ionescu', 'mihai.ionescu@example.com');
INSERT INTO Utilizatori VALUES(3, 'Elena Vasile', 'elena.vasile@example.com');
INSERT INTO Utilizatori VALUES(4, 'Andrei Stoica', 'andrei.stoica@example.com');
INSERT INTO Utilizatori VALUES(5, 'Maria Radu', 'maria.radu@example.com');

INSERT INTO Utilizatori VALUES(6, 'Gabriel Dumitrescu', 'gabriel.dumitrescu@example.com');
INSERT INTO Utilizatori VALUES(7, 'Cristina Gheorghiu', 'cristina.gheorghiu@example.com');
INSERT INTO Utilizatori VALUES(8, 'Alexandru Barbu', 'alexandru.barbu@example.com');
INSERT INTO Utilizatori VALUES(9, 'Laura Dinu', 'laura.dinu@example.com');
INSERT INTO Utilizatori VALUES(10, 'Ionela Marin', 'ionela.marin@example.com');
```

Script Output x

Task completed in 0.278 seconds

1 row inserted.

1 row inserted.

UTILIZATORI			
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partition			
Sort.. Filter:			
	ID_UTILIZATOR	NUME	EMAIL
1	1	Ana Popescu	ana.popescu@example.com
2	2	Mihai Ionescu	mihai.ionescu@example.com
3	3	Elena Vasile	elena.vasile@example.com
4	4	Andrei Stoica	andrei.stoica@example.com
5	5	Maria Radu	maria.radu@example.com
6	6	Gabriel Dumitrescu	gabriel.dumitrescu@example.com
7	7	Cristina Gheorghiu	cristina.gheorghiu@example.com
8	8	Alexandru Barbu	alexandru.barbu@example.com
9	9	Laura Dinu	laura.dinu@example.com
10	10	Ionela Marin	ionela.marin@example.com

Tabela Imprumuturi :

```
INSERT INTO
Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(1, 1, 1, TO_DATE('2022-01-05', 'YYYY-MM-DD'), TO_DATE('2022-01-15',
'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(2, 2, 2, TO_DATE('2022-01-08', 'YYYY-MM-
DD'), TO_DATE('2022-01-18', 'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(3, 3, 3, TO_DATE('2022-01-10', 'YYYY-MM-
DD'), TO_DATE('2022-01-20', 'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(4, 4, 4, TO_DATE('2022-01-12', 'YYYY-MM-
DD'), TO_DATE('2022-01-22', 'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(5, 5, 5, TO_DATE('2022-01-15', 'YYYY-MM-
DD'), TO_DATE('2022-01-25', 'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(6, 6, 6, TO_DATE('2022-01-18', 'YYYY-MM-
DD'), TO_DATE('2022-01-28', 'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(7, 7, 7, TO_DATE('2022-01-20', 'YYYY-MM-
DD'), TO_DATE('2022-01-30', 'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(8, 8, 8, TO_DATE('2022-01-22', 'YYYY-MM-
DD'), TO_DATE('2022-02-01', 'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(9, 9, 9, TO_DATE('2022-01-25', 'YYYY-MM-
DD'), TO_DATE('2022-02-05', 'YYYY-MM-DD')),
INSERT INTO Imprumuturi VALUES(10, 10, 10, TO_DATE('2022-01-28', 'YYYY-
MM-DD'), TO_DATE('2022-02-08', 'YYYY-MM-DD'));
```

The screenshot shows a SQL query execution interface. The main window displays the following SQL script:

```
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire)
VALUES(1, 1, 1, TO_DATE('2022-01-05', 'YYYY-MM-DD'), TO_DATE('2022-01-15', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(2, 2, 2, TO_DATE('2022-01-08', 'YYYY-MM-DD'), TO_DATE('2022-01-18', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(3, 3, 3, TO_DATE('2022-01-10', 'YYYY-MM-DD'), TO_DATE('2022-01-20', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(4, 4, 4, TO_DATE('2022-01-12', 'YYYY-MM-DD'), TO_DATE('2022-01-22', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(5, 5, 5, TO_DATE('2022-01-15', 'YYYY-MM-DD'), TO_DATE('2022-01-25', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(6, 6, 6, TO_DATE('2022-01-18', 'YYYY-MM-DD'), TO_DATE('2022-01-28', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(7, 7, 7, TO_DATE('2022-01-20', 'YYYY-MM-DD'), TO_DATE('2022-01-30', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(8, 8, 8, TO_DATE('2022-01-22', 'YYYY-MM-DD'), TO_DATE('2022-02-01', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(9, 9, 9, TO_DATE('2022-01-25', 'YYYY-MM-DD'), TO_DATE('2022-02-05', 'YYYY-MM-DD'));
INSERT INTO Imprumuturi(ID_Impmut,ID_Carte,ID_Utilizator,Data_Impmut,Data_Restituire) VALUES(10, 10, 10, TO_DATE('2022-01-28', 'YYYY-MM-DD'), TO_DATE('2022-02-08', 'YYYY-MM-DD'));
```

The 'Script Output' window at the bottom shows the result of the execution:

```
1 row inserted.
```

The status bar at the bottom indicates 'Task completed in 0.265 seconds'.

Welcome Page × ButuzaAlexandru1066 × IMPRUMUTURI ×						
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL						
Sort.. Filter:						
	ID_IMPRUMUT	ID_CARTE	DATA_IMPRUMUT	DATA_RESTITUIRE	ID_UTILIZATOR	
1	1	1	05-JAN-22	15-JAN-22	1	
2	2	2	08-JAN-22	18-JAN-22	2	
3	3	3	10-JAN-22	20-JAN-22	3	
4	4	4	12-JAN-22	22-JAN-22	4	
5	5	5	15-JAN-22	25-JAN-22	5	
6	6	6	18-JAN-22	28-JAN-22	6	
7	7	7	20-JAN-22	30-JAN-22	7	
8	8	8	22-JAN-22	01-FEB-22	8	
9	9	9	25-JAN-22	05-FEB-22	9	
10	10	10	28-JAN-22	08-FEB-22	10	

6.Actualizarea inregistrarilor

6.1 UPDATE Carti

```
SET An_Publicare = 2005
WHERE ID_Carte = 1;
```

6.2 UPDATE Genuri

```
SET Descriere = 'Gen literar fictiv în limba română, actualizat'
WHERE ID_Gen = 3;
```

6.3 UPDATE Imprumuturi

```
SET Data_Restituire = TO_DATE('2022-01-20', 'YYYY-MM-DD')
WHERE ID_Imprumut = 1;
```

6.4 UPDATE Utilizatori

```
SET Email = 'ana.popescu@gmail.com'
WHERE ID_Utilizator = 1;
```

Worksheet Query Builder

```
-- Actualizare inregistrare in tabelul "Carti"
UPDATE Carti
SET An_Publicare = 2005
WHERE ID_Carte = 1;

-- Actualizare inregistrare in tabelul "Genuri"
UPDATE Genuri
SET Descriere = 'Gen literar fictiv in limba română, actualizat'
WHERE ID_Gen = 3;

-- Actualizare inregistrare in tabelul "Imprumuturi"
UPDATE Imprumuturi
SET Data_Restituire = TO_DATE('2022-01-20', 'YYYY-MM-DD')
WHERE ID_Imprumut = 1;

-- Actualizare inregistrare in tabelul "Utilizatori"
UPDATE Utilizatori
SET Email = 'ana.popescu@gmail.com'
WHERE ID_Utilizator = 1;
```

Script Output x

Task completed in 0.21 seconds

1 row updated.

1 row updated.

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL			
Sort.. Filter:			
ID_GEN	NUME_GEN	DESCRIERE	
1	3 Fictiune	Gen literar fictiv în limba română, actualizat	
2	4 Științifico-Fantastic	Gen literar SF în limba română	
3	5 Mister	Gen literar de mister în limba română	
4	6 Romantic	Gen literar romantic în limba română	
5	7 Istoric	Gen literar istoric în limba română	
6	8 Poezie	Gen literar de poezie în limba română	
7	9 Teatru	Gen literar de teatru în limba română	
8	10 Adventure	Gen literar de aventuri în limba română	
9	11 Biografie	Gen literar de biografie în limba română	
10	12 Comedie	Gen literar de comedie în limba română	

Welcome Page ButuzaAlexandru1066 CARTI				
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL				
Sort.. Filter:				
ID_CARTE	TITLU	AUTOR	AN_PUBLICARE	ID_GEN
1	1 Harry Potter și Piatra Filozofala	J.K. Rowling	2005	3
2	2 Dune	Frank Herbert	1965	4
3	3 Sherlock Holmes: Studioul în Rosu	Arthur Conan Doyle	1887	5
4	4 Romeo și Julieta	William Shakespeare	1597	6
5	5 Cronicile din Narnia: Leul, Vrăjitoarea și Dulapul	C.S. Lewis	1950	7
6	6 Poezii Selectate	Mihai Eminescu	1884	8
7	7 Hamlet	William Shakespeare	1603	9
8	8 Robinson Crusoe	Daniel Defoe	1719	10
9	9 Steve Jobs: Biografia	Walter Isaacson	2011	11
10	10 O noapte furtunoasă	I.L. Caragiale	1885	12

Welcome Page ButuzaAlexandru1066 UTILIZATORI		
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions		
Sort.. Filter:		
ID_UTILIZATOR	NUME	EMAIL
1	1 Ana Popescu	ana.popescu@gmail.com
2	2 Mihai Ionescu	mihai.ionescu@example.com
3	3 Elena Vasile	elena.vasile@example.com
4	4 Andrei Stoica	andrei.stoica@example.com
5	5 Maria Radu	maria.radu@example.com
6	6 Gabriel Dumitrescu	gabriel.dumitrescu@example.com
7	7 Cristina Gheorghiu	cristina.gheorghiu@example.com
8	8 Alexandru Barbu	alexandru.barbu@example.com
9	9 Laura Dinu	laura.dinu@example.com
10	10 Ionela Marin	ionela.marin@example.com

Welcome Page x ButuzaAlexandru1066 x IMPRUMUTURI x						
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Inde						
Sort.. Filter:						
	ID_IMPRUMUT	ID_CARTE	DATA_IMPRUMUT	DATA_RESTITUIRE	ID_UTILIZATOR	
1	1	1	05-JAN-22	20-JAN-22	1	
2	2	2	08-JAN-22	18-JAN-22	2	
3	3	3	10-JAN-22	20-JAN-22	3	
4	4	4	12-JAN-22	22-JAN-22	4	
5	5	5	15-JAN-22	25-JAN-22	5	
6	6	6	18-JAN-22	28-JAN-22	6	
7	7	7	20-JAN-22	30-JAN-22	7	
8	8	8	22-JAN-22	01-FEB-22	8	
9	9	9	25-JAN-22	05-FEB-22	9	
10	10	10	28-JAN-22	08-FEB-22	10	

7.Stergerea si recuperarea unei tabele

7.1 Stergerea tabelului:

drop table Utilizatori
cascade constraints

7.2 Recuperarea tabelului:

flashback table Utilizatori
to before drop

Welcome Page x ButuzaAlexandru1066 x IMPRUMUTURI x

0.55199999 seconds

Worksheet Query Builder

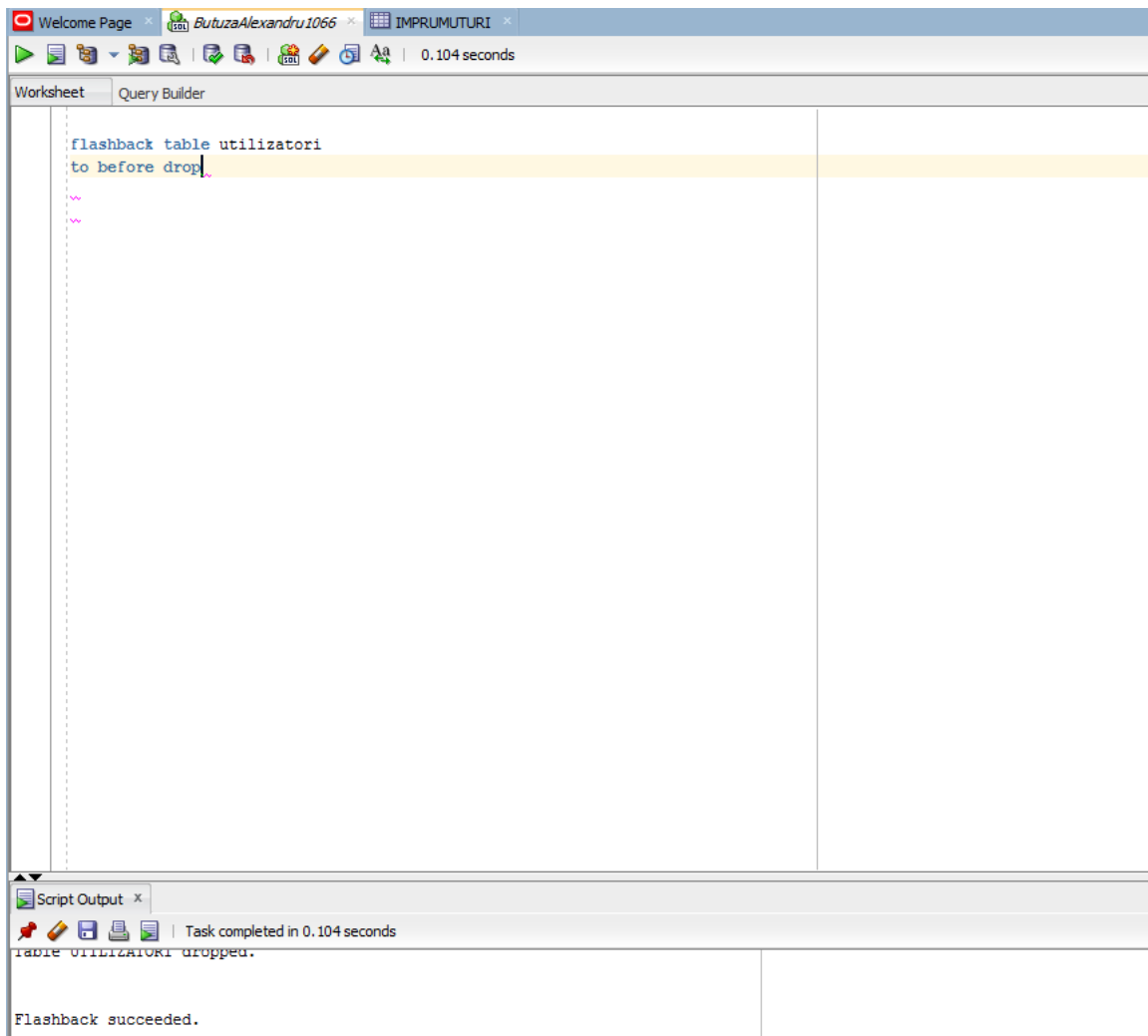
```
drop table Utilizatori  
cascade constraints
```

Script Output x

Task completed in 0.552 seconds

1 row updated.

Table UTILIZATORI dropped.



8.Exemple de interogari variate

8.1 Afișează toate cărțile distincte, inclusiv cele împrumutate și cele disponibile

```
SELECT Titlu  
FROM (  
    SELECT Titlu FROM Carti  
    UNION  
    SELECT (SELECT Titlu FROM Carti WHERE ID_Carte = Imprumuturi.ID_Carte)  
AS Titlu  
FROM Imprumuturi  
);
```

Worksheet Query Builder

-- Exemplul 1 | Afișează toate cărțile distincte, inclusiv cele împrumutate și cele disponibile

```
SELECT Titlu
FROM (
  SELECT Titlu FROM Carti
  UNION
  SELECT (SELECT Titlu FROM Carti WHERE ID_Carte = Imprumuturi.ID_Carte) AS Titlu
  FROM Imprumuturi
);
```

Script Output x

Task completed in 0.152 seconds

TITLU
Harry Potter și Piatra Filozofala
Dune

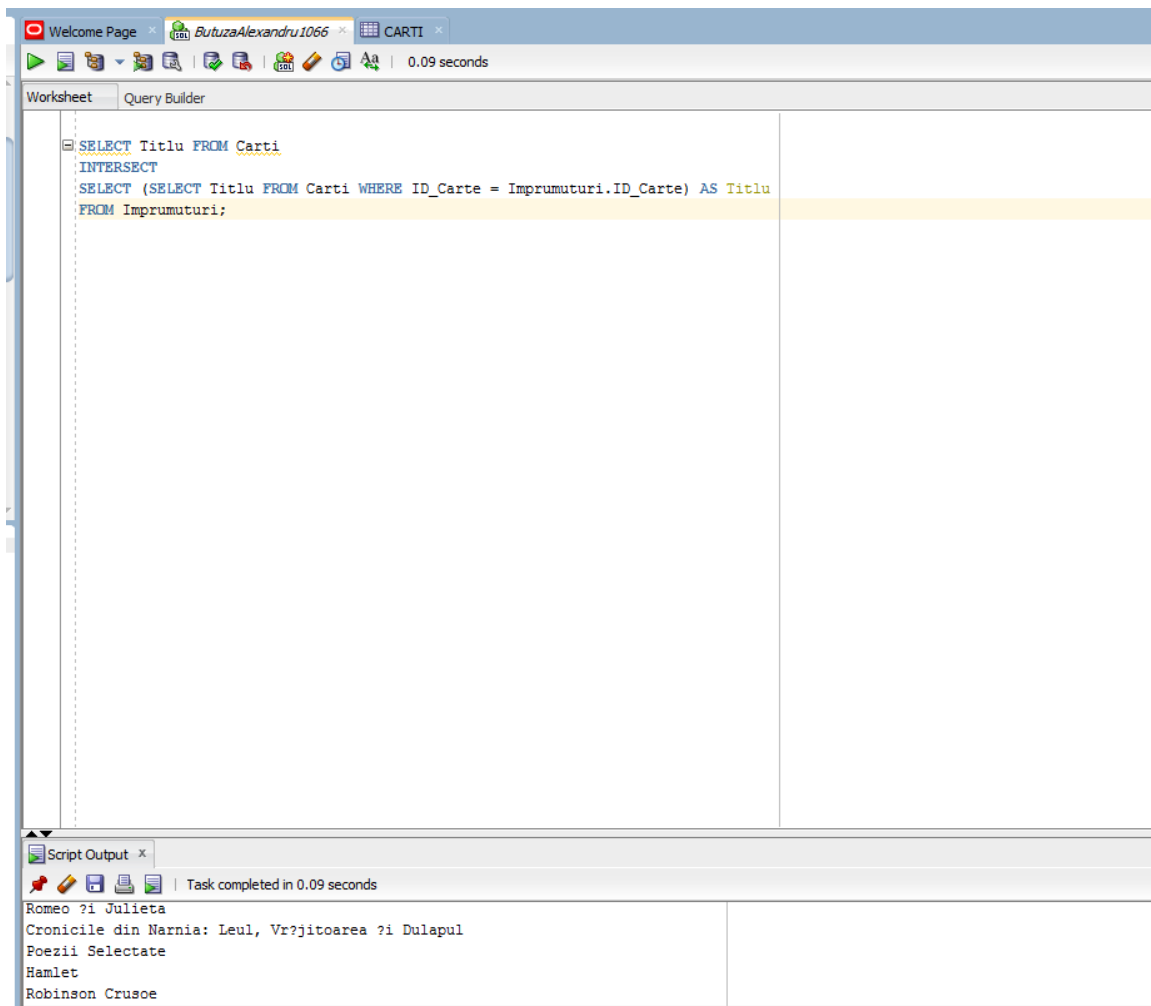
8.2 Afișează cărțile împrumutate care sunt și disponibile:

```
SELECT Titlu FROM Carti
```

```
INTERSECT
```

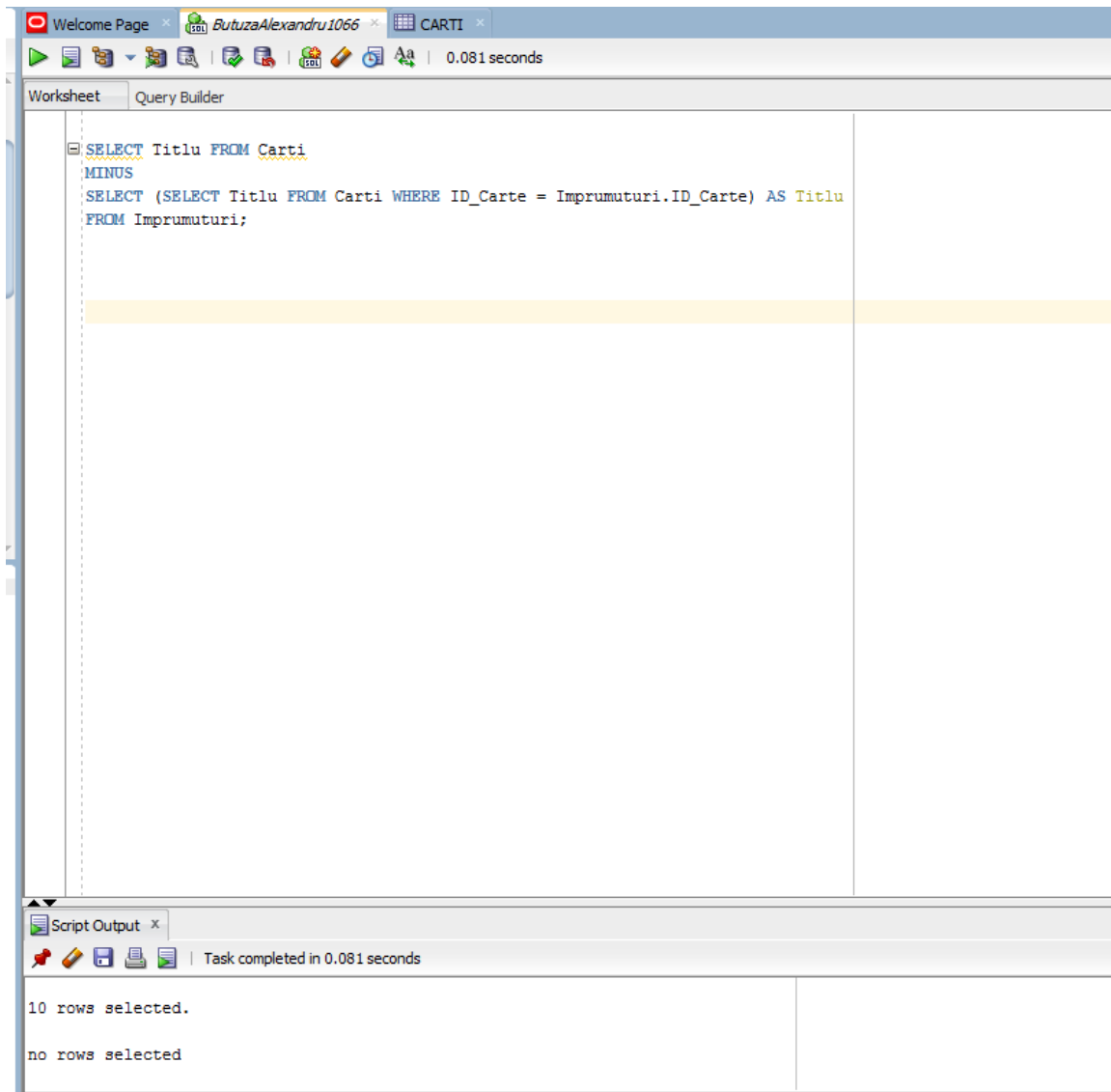
```
SELECT (SELECT Titlu FROM Carti WHERE ID_Carte = Imprumuturi.ID_Carte) AS  
Titlu
```

```
FROM Imprumuturi;
```



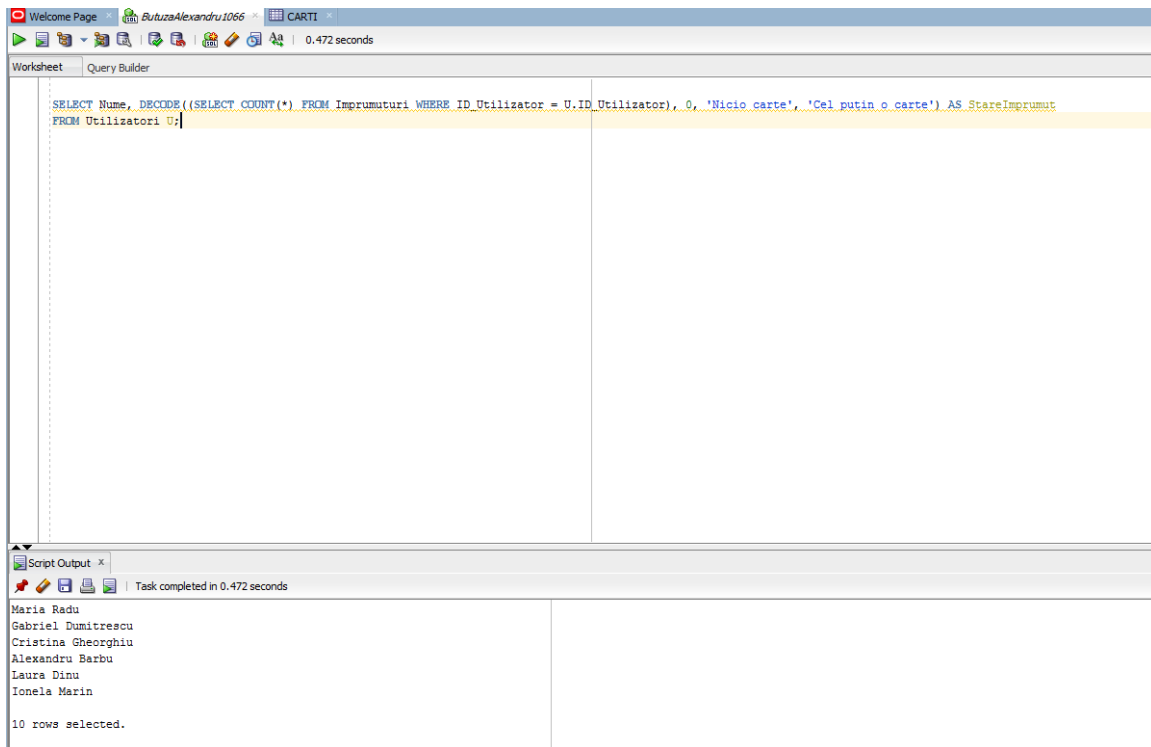
8.3 Afişează cărţile disponibile care nu au fost împrumutate :

```
SELECT Titlu FROM Carti  
MINUS  
SELECT (SELECT Titlu FROM Carti WHERE ID_Carte = Imprumuturi.ID_Carte) AS  
Titlu  
FROM Imprumuturi;
```



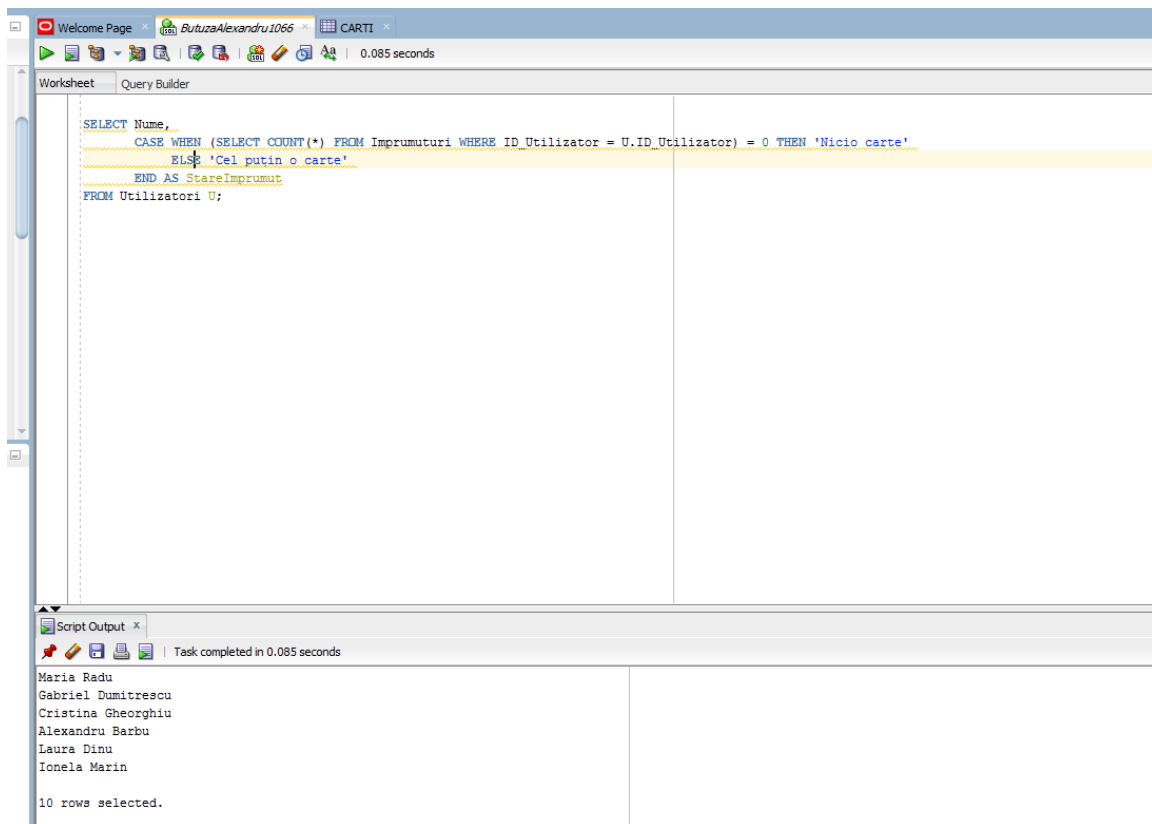
8.4 Afișează numele utilizatorilor și numărul de cărți împrumutate folosind DECODE :

```
SELECT Nume, DECODE((SELECT COUNT(*)
FROM Imprumuturi
WHERE ID_Utilizator = U.ID_Utilizator), 0, 'Nicio carte', 'Cel puțin o carte')
AS StareImprumut
FROM Utilizatori U;
```



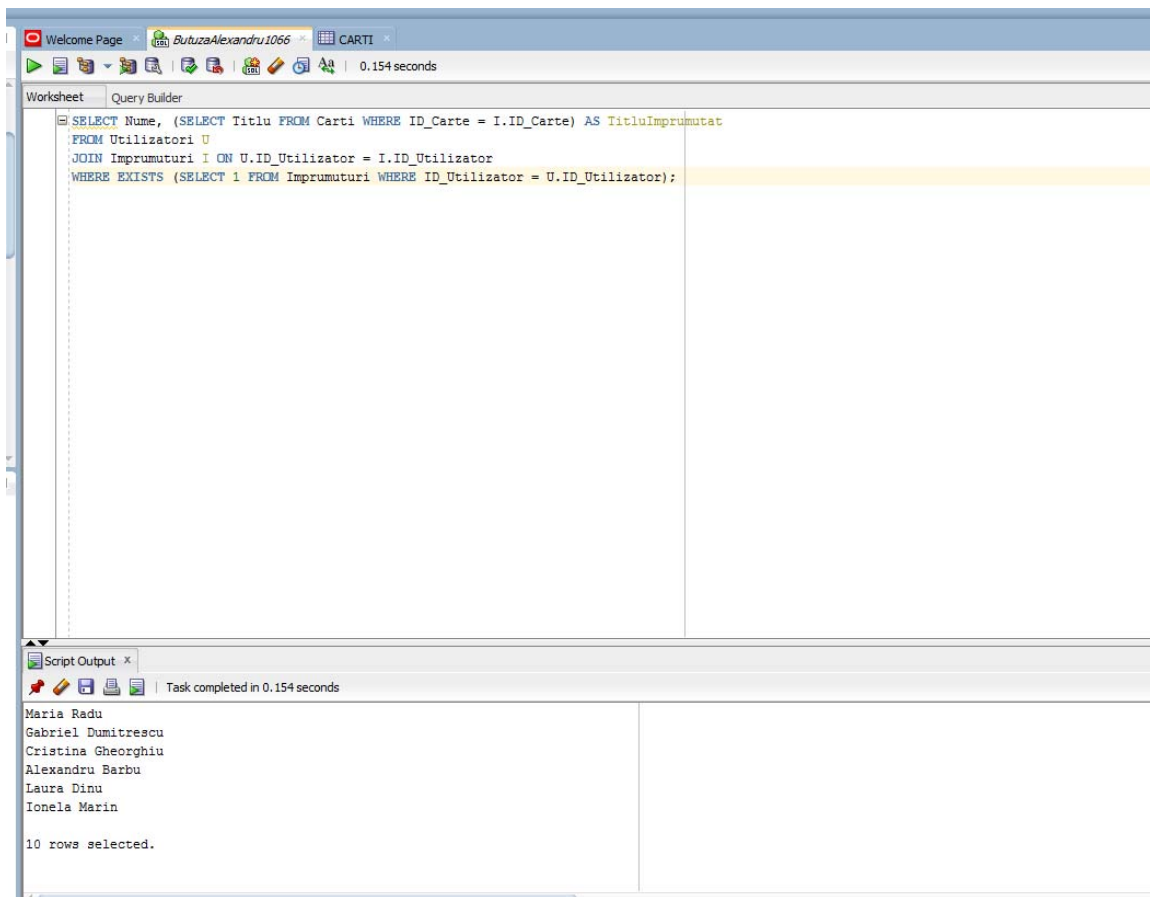
8.5 Afășează numele utilizatorilor și numărul de cărți împrumutate folosind CASE.

```
SELECT Nume,
       CASE WHEN (SELECT COUNT(*) FROM Imprumuturi
WHERE ID_Utilizator = U.ID_Utilizator) = 0 THEN 'Nicio carte'
       ELSE 'Cel puțin o carte'
       END AS StareImprumut
FROM Utilizatori U;
```

8.6 Afișează utilizatorii care au împrumutat cel puțin o carte și detaliile acestora.

```
SELECT Nume, (SELECT Titlu FROM Carti WHERE ID_Carte = I.ID_Carte)  
AS TitluImprumutat  
FROM Utilizatori U  
JOIN Imprumuturi I ON U.ID_Utilizator = I.ID_Utilizator  
WHERE EXISTS (SELECT 1 FROM Imprumuturi  
WHERE ID_Utilizator = U.ID_Utilizator);
```



8.7 Afişează data actuală

SELECT SYSDATE FROM dual;

The screenshot shows a SQL query editor window with the following tabs: 'Welcome Page', 'ButuzaAlexandru1066', and 'CARTI'. The 'Query Builder' tab is active, displaying the query: `SELECT SYSDATE FROM dual;`. The execution time is 0.085 seconds. Below the query editor is a 'Script Output' window showing the results of the query. The output indicates that 10 rows were selected, and the date is 05-JAN-24.

Worksheet Query Builder

```
SELECT SYSDATE FROM dual;
```

Script Output x

Task completed in 0.085 seconds

Ionela Marin

10 rows selected.

SYSDATE

05-JAN-24

8.8 Afișează numărul total de cărți în fiecare gen

```
SELECT Nume_Gen, COUNT(*) AS NumarCarti
FROM Carti
JOIN Genuri ON Carti.ID_Gen = Genuri.ID_Gen
GROUP BY Nume_Gen;
```

Screenshot of a SQL query execution interface showing the Query Builder and Script Output panels.

Query Builder:

```
SELECT Nume_Gen, COUNT(*) AS NumarCarti
FROM Carti
JOIN Genuri ON Carti.ID_Gen = Genuri.ID_Gen
GROUP BY Nume_Gen;
```

Script Output:

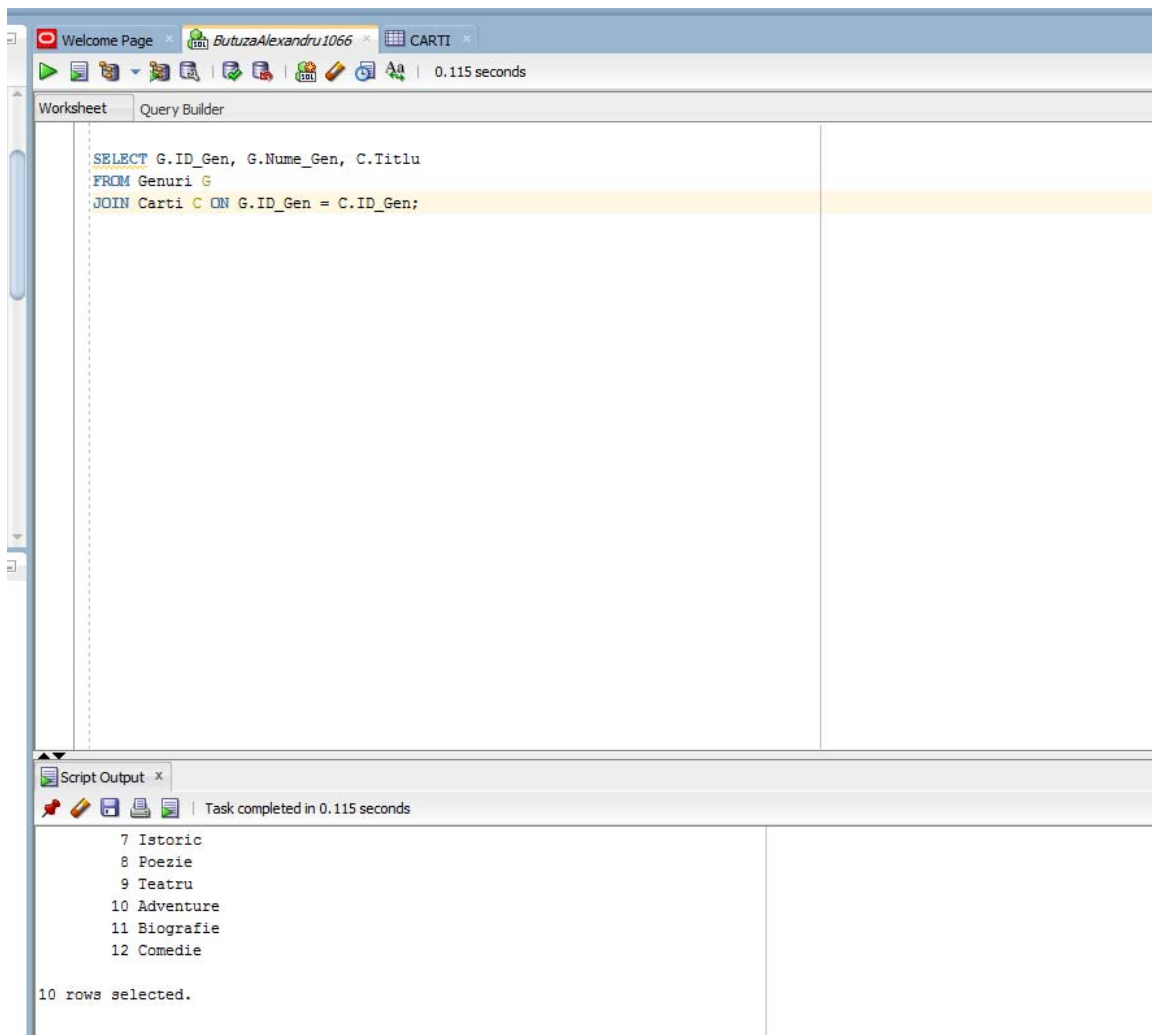
Task completed in 0.112 seconds

Istoric
Poezie
Teatru
Adventure
Biografie
Comedie

10 rows selected.

8.9 Afișează structura ierarhică a genurilor de cărți

```
SELECT G.ID_Gen, G.Nume_Gen, C.Titlu
FROM Genuri G
JOIN Carti C ON G.ID_Gen = C.ID_Gen;
```



8.10 Afișează informații complete despre împrumuturile făcute, inclusiv detalii despre cărți și utilizatori

```
SELECT Imprumuturi.ID_Impmut, Utilizatori.Nume AS NumeUtilizator, Carti.Titlu,
Imprumuturi.Data_Impmut, Imprumuturi.Data_Restituire
FROM Imprumuturi
JOIN Utilizatori ON Imprumuturi.ID_Utilizator = Utilizatori.ID_Utilizator
JOIN Carti ON Imprumuturi.ID_Carte = Carti.ID_Carte;
```

The screenshot shows a SQL query builder interface. The top bar includes a 'Welcome Page' button, a user profile 'ButuzaAlexandru1066', and a 'CARTI' button. Below this is a toolbar with various icons and a timer showing '0.098 seconds'. The main area is divided into 'Worksheet' and 'Query Builder' tabs. The 'Query Builder' tab displays the following SQL query:

```
SELECT Imprumuturi.ID_Imprumut, Utilizatori.Nume AS NumeUtilizator, Carti.Titlu, Imprumuturi.Data_Imprumut, Imprumuturi.Data_Restituire
FROM Imprumuturi
JOIN Utilizatori ON Imprumuturi.ID_Utilizator = Utilizatori.ID_Utilizator
JOIN Carti ON Imprumuturi.ID_Carte = Carti.ID_Carte;
```

Below the query, the 'Script Output' tab shows the results of the query. It indicates 'Task completed in 0.098 seconds' and lists 10 rows selected:

5	Maria Radu
6	Gabriel Dumitrescu
7	Cristina Gheorghiu
8	Alexandru Barbu
9	Laura Dinu
10	Ionela Marin

10 rows selected.

8.11 Afişează toţi utilizatorii care au împrumutat cărţi

SELECT DISTINCT U.Nume

FROM Utilizatori U

JOIN Imprumuturi I ON U.ID_Utilizator = I.ID_Utilizator;

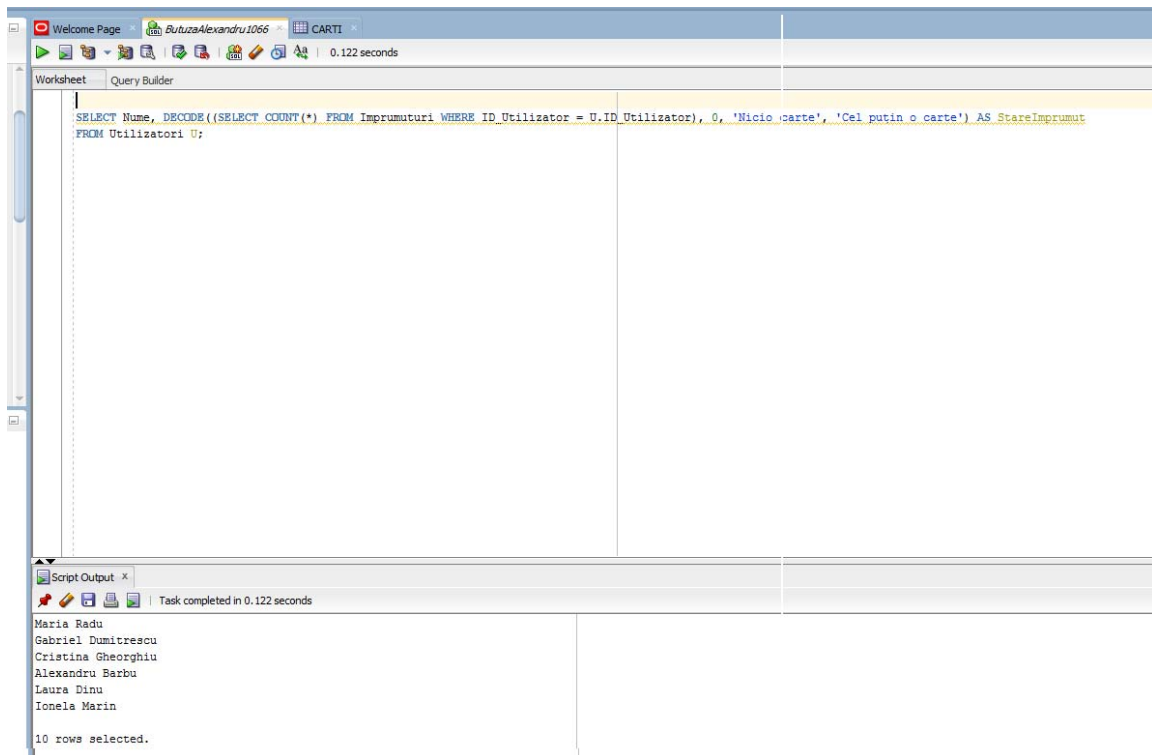
The screenshot shows a SQL query editor window with the following SQL query:

```
SELECT DISTINCT U.Nume  
FROM Utilizatori U  
JOIN Imprumuturi I ON U.ID_Utilizator = I.ID_Utilizator;
```

The query is executed, and the results are displayed in the Script Output window. The output shows the names of the users who have borrowed books:

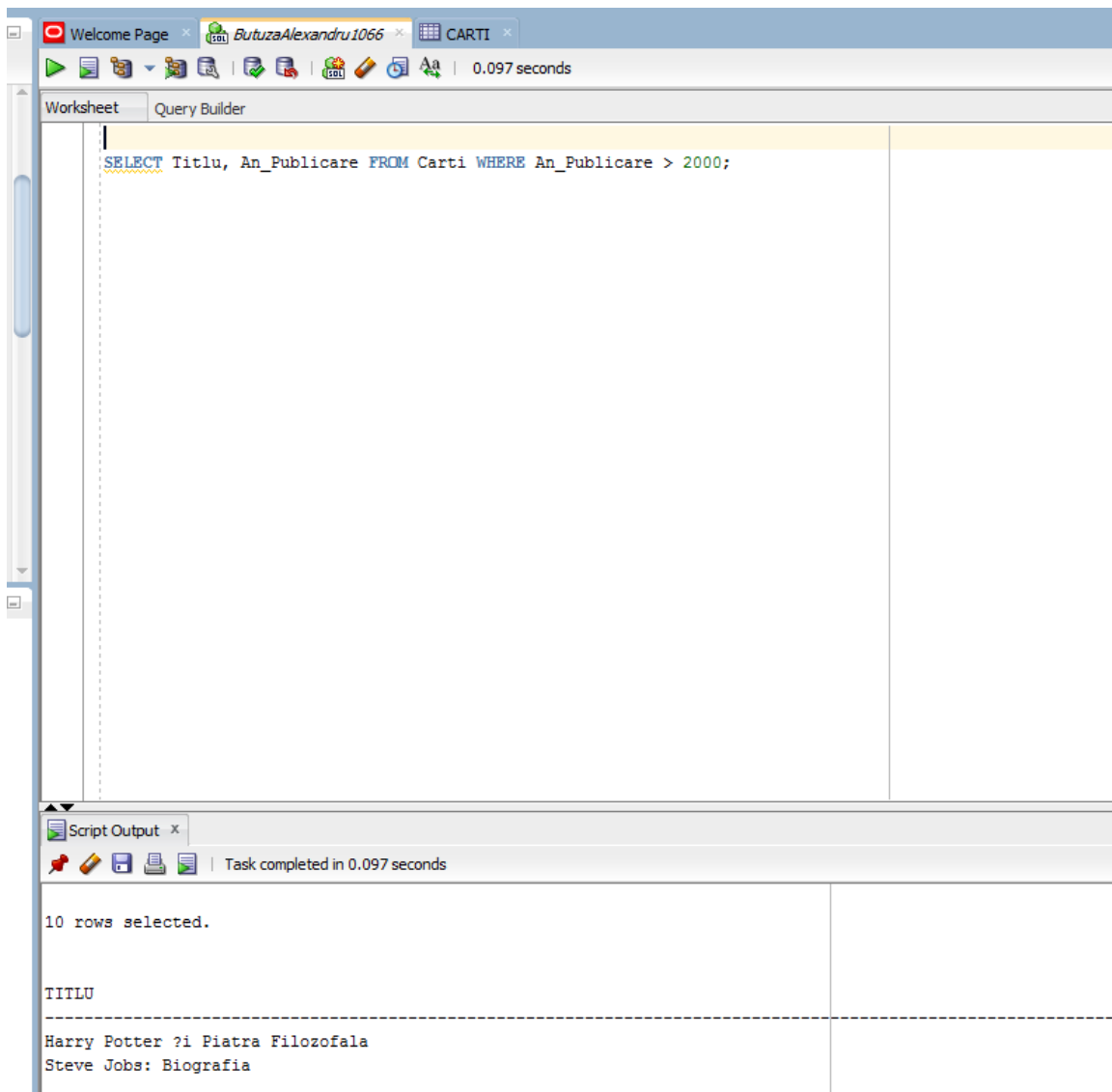
```
Maria Radu  
Gabriel Dumitrescu  
Cristina Gheorghiu  
Alexandru Barbu  
Laura Dinu  
Ionela Marin  
  
10 rows selected.
```

8.12 Afişează numele utilizatorilor şi numărul de cărţi împrumutate folosind DECODE.
SELECT Nume, DECODE((SELECT COUNT(*) FROM Imprumuturi WHERE
ID_Utilizator = U.ID_Utilizator), 0, 'Nicio carte', 'Cel puţin o carte') AS StareImprumut
FROM Utilizatori U;



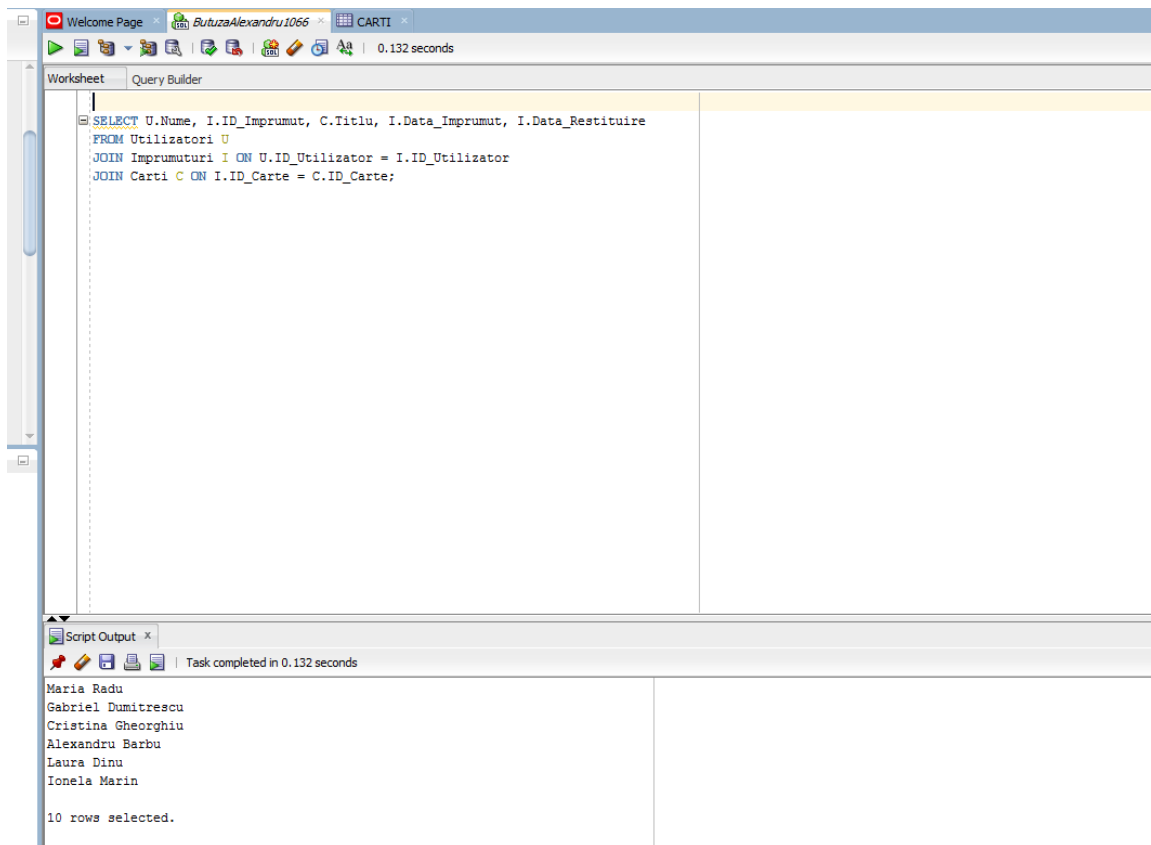
8.13 Afیșează cărțile publicate după anul 2000:

`SELECT Titlu, An_Publicare FROM Carti WHERE An_Publicare > 2000;`



8.14 Afișează toate împrumuturile cu detalii complete despre utilizator și carte:

```
SELECT U.Nume, I.ID_Impmut, C.Titlu, I.Data_Impmut, I.Data_Restituire
FROM Utilizatori U
JOIN Impmuturi I ON U.ID_Utilizator = I.ID_Utilizator
JOIN Carti C ON I.ID_Carte = C.ID_Carte;
```



8.15 Afişează numărul total de cărți împrumutate de fiecare utilizator:

```
SELECT U.Nume, COUNT(I.ID_Imprumut) AS NumarImprumuturi
FROM Utilizatori U
LEFT JOIN Imprumuturi I ON U.ID_Utilizator = I.ID_Utilizator
GROUP BY U.Nume;
```

Worksheet Query Builder

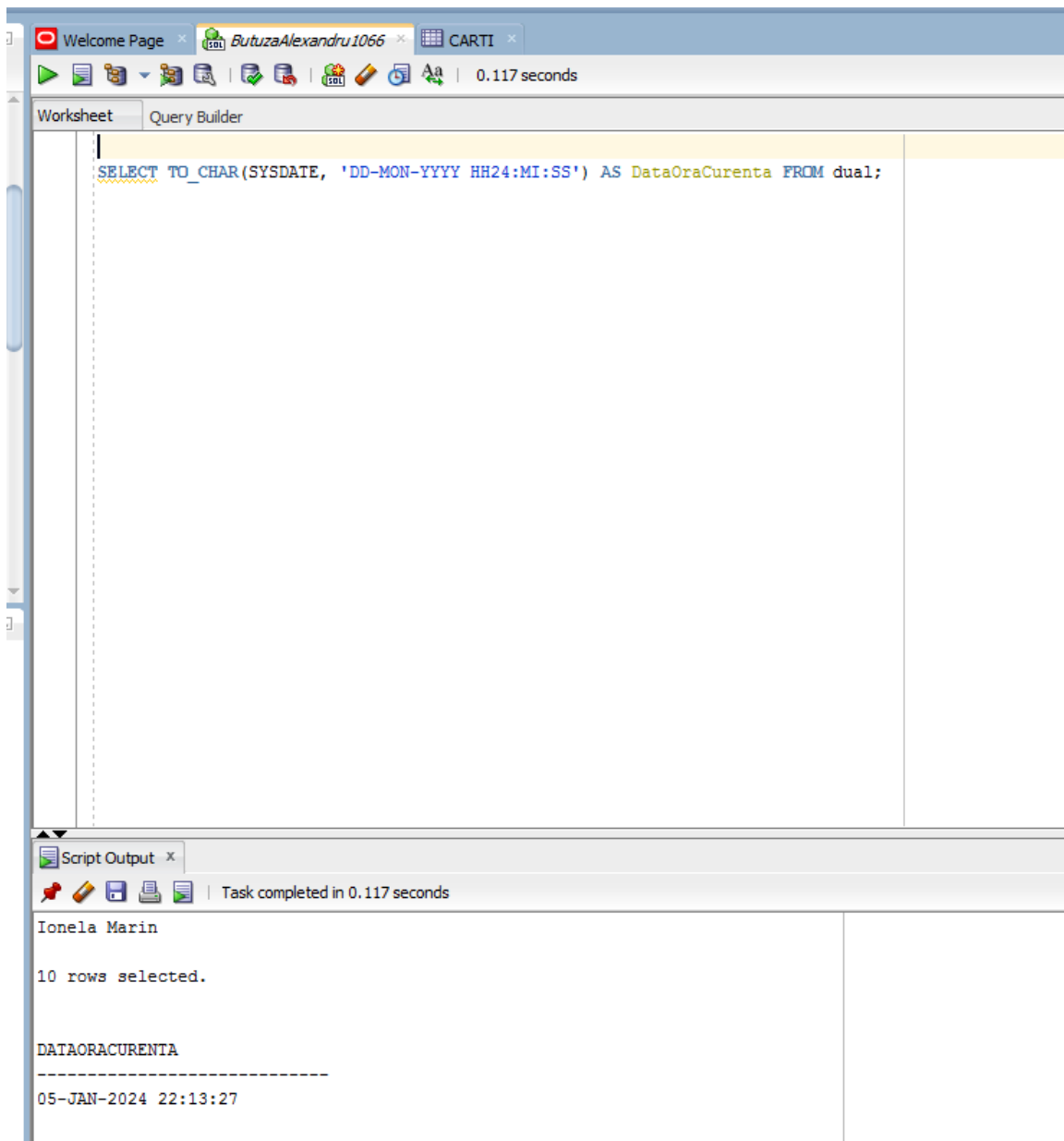
```
SELECT U.Nume, COUNT(I.ID_Impmut) AS NumarImpmuturi
FROM Utilizatori U
LEFT JOIN Impmuturi I ON U.ID_Utilizator = I.ID_Utilizator
GROUP BY U.Nume;
```

Script Output x

Task completed in 0.087 seconds

NUME
Ana Popescu
Mihai Ionescu
Elena Vasile
Andrei Stoica
Maria Radu
Gabriel Dumitrescu
Cristina Gheorghiu
Alexandru Barbu

8.16 Afișează data actuală și ora curentă într-un format personalizat:
SELECT TO_CHAR(SYSDATE, 'DD-MON-YYYY HH24:MI:SS') AS DataOraCurenta
FROM dual;



8.17

```
ALTER TABLE Carti
ADD Disponibila VARCHAR2(1);
UPDATE Carti
SET Disponibila = CASE WHEN MOD(ID_Carte, 2) = 0 THEN 'Y' ELSE 'N' END;
SELECT Titlu, DECODE(Disponibila, 'Y', 'Disponibilă', 'Indisponibilă') AS
StareDisponibilitate
FROM Carti;
```

Worksheet Query Builder

```

ALTER TABLE Carti
ADD Disponibila VARCHAR2(1);
UPDATE Carti
SET Disponibila = CASE WHEN MOD(ID_Carte, 2) = 0 THEN 'Y' ELSE 'N' END;
SELECT Titlu, DECODE(Disponibila, 'Y', 'Disponibilă', 'Indisponibilă') AS StareDisponibilitate
FROM Carti;

```

Script Output x

Task completed in 0.205 seconds

Cronicile din Narnia: Leul, Vrăjitoarea și Dulapul
Poezii Selectate
Hamlet
Robinson Crusoe
Steve Jobs: Biografia
O noapte furtunoasă
10 rows selected.

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL

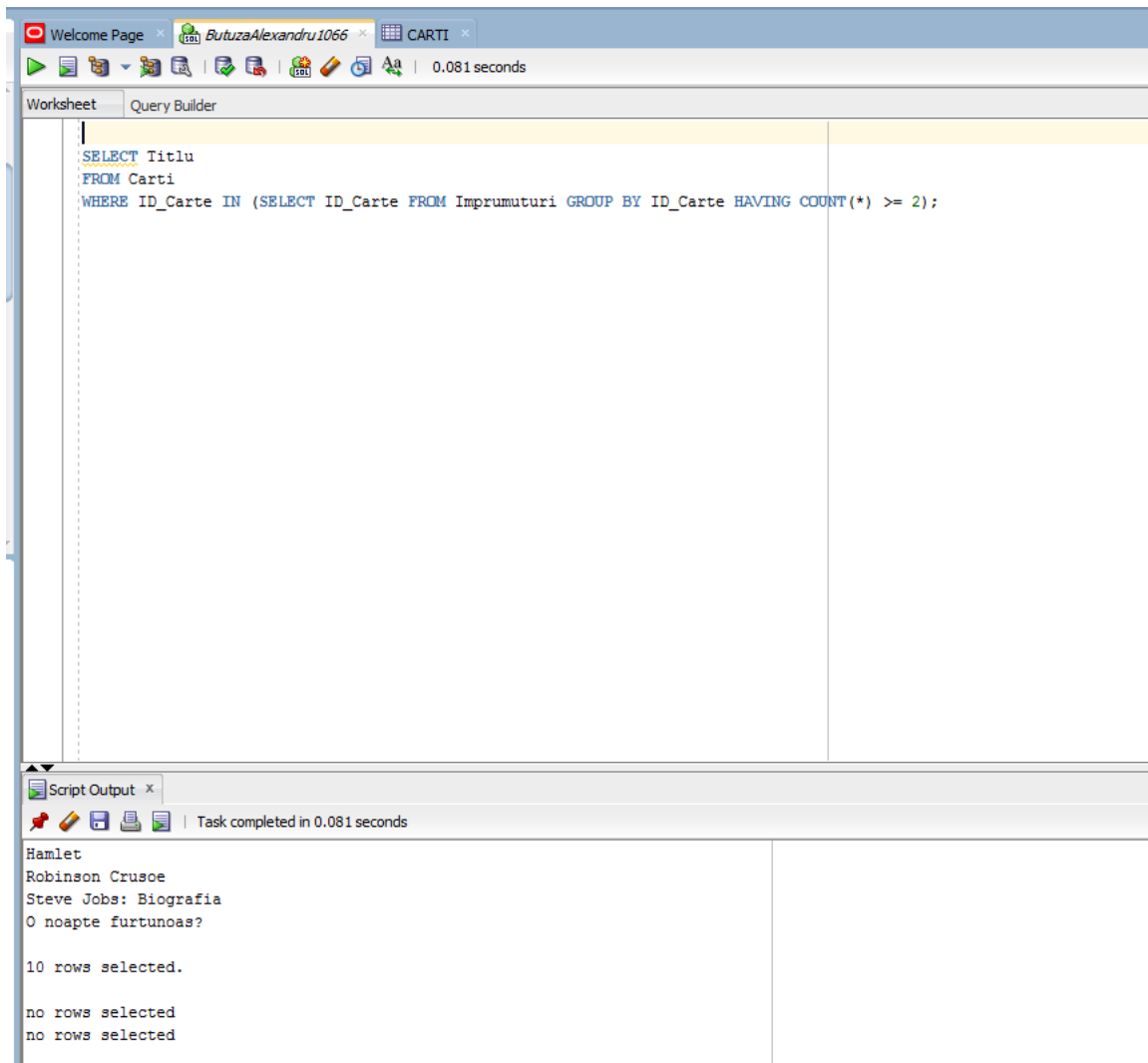
ID_CARTE	TITLU	AUTOR	AN_PUBLCARE	ID_GEN	DISPONIBILA
1	1 Harry Potter și Piatra Filozofala	J.K. Rowling	2005	3	N
2	2 Dune	Frank Herbert	1965	4	Y
3	3 Sherlock Holmes: Studioul în Rosu	Arthur Conan Doyle	1887	5	N
4	4 Romeo și Julieta	William Shakespeare	1597	6	Y
5	5 Cronicile din Narnia: Leul, Vrăjitoarea și Dulapul	C.S. Lewis	1950	7	N
6	6 Poezii Selectate	Mihai Eminescu	1884	8	Y
7	7 Hamlet	William Shakespeare	1603	9	N
8	8 Robinson Crusoe	Daniel Defoe	1719	10	Y
9	9 Steve Jobs: Biografia	Walter Isaacson	2011	11	N
10	10 O noapte furtunoasă	I.L. Caragiale	1885	12	Y

8.18 Afیșează cărțile care au fost împrumutate de cel puțin două ori:

```

SELECT Titlu
FROM Carti
WHERE ID_Carte IN (SELECT ID_Carte FROM Imprumuturi GROUP BY ID_Carte
HAVING COUNT(*) >= 2);

```



8.19 Afișează titlurile și un mesaj în funcție de data limită de restituire a cărților.

```
SELECT C.Titlu,
CASE
    WHEN I.Data_Restituire < SYSDATE THEN 'Întârziat'
    WHEN I.Data_Restituire = SYSDATE THEN 'Azi este termenul limită'
    ELSE 'La timp'
END AS StareRestituire
FROM Carti C
JOIN Imprumuturi I ON C.ID_Carte = I.ID_Carte;
```

The screenshot shows a SQL query editor with the following query:

```

SELECT C.Titlu,
CASE
  WHEN I.Data_Restituire < SYSDATE THEN 'întârziat'
  WHEN I.Data_Restituire = SYSDATE THEN 'Azi este termenul limită'
  ELSE 'La timp'
END AS StareRestituire
FROM Carti C
JOIN Imprumuturi I ON C.ID_Carte = I.ID_Carte;

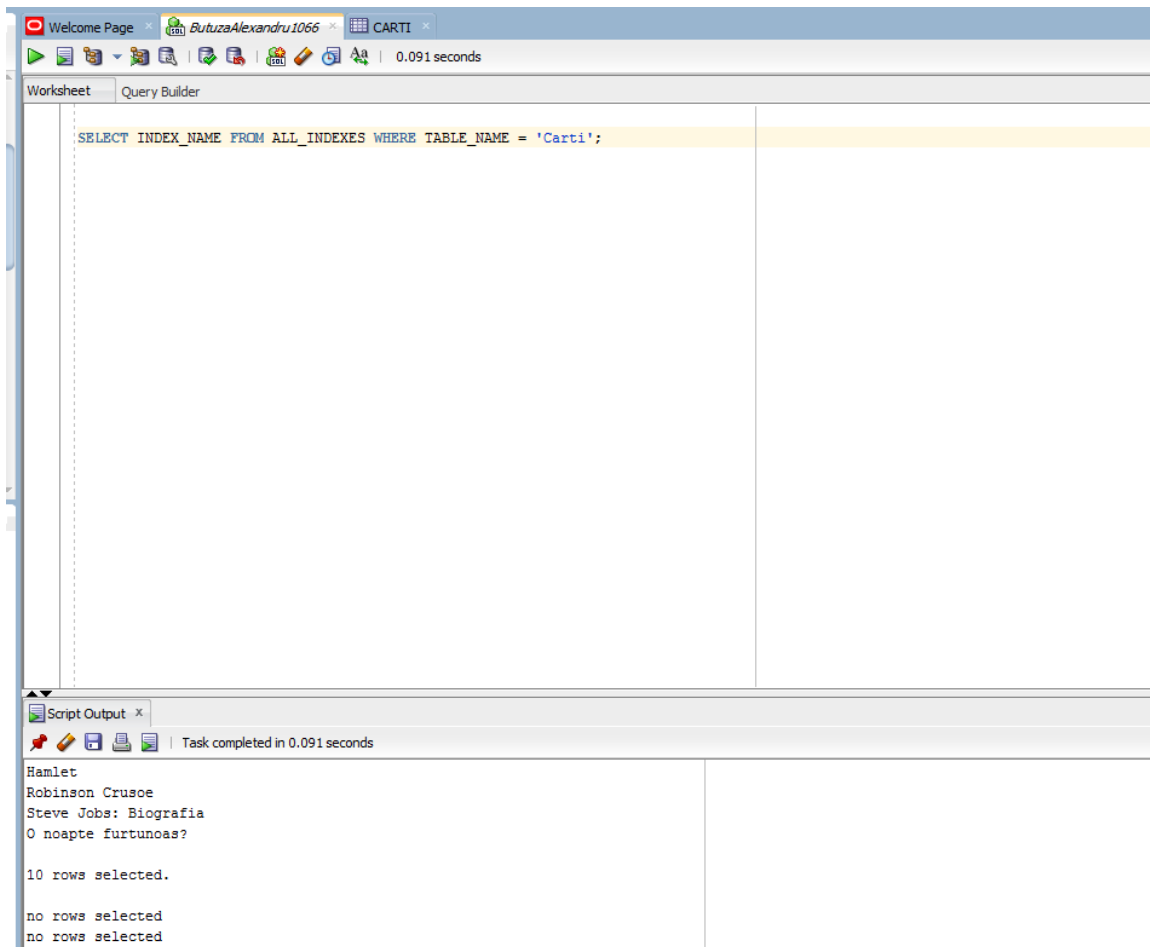
```

The script output window shows the results of the query, with the column header 'TITLU' and the following rows:

TITLU
Harry Potter ?i Piatra Filozofala
Dune
Sherlock Holmes: Studioul în Rosu
Romeo ?i Julieta
Cronicile din Narnia: Leul, Vr?jitoarea ?i Dulapul
Poezii Selectate
Hamlet
Robinson Crusoe

8.20 Afîșează indecșii care există pe tabela "Carti":

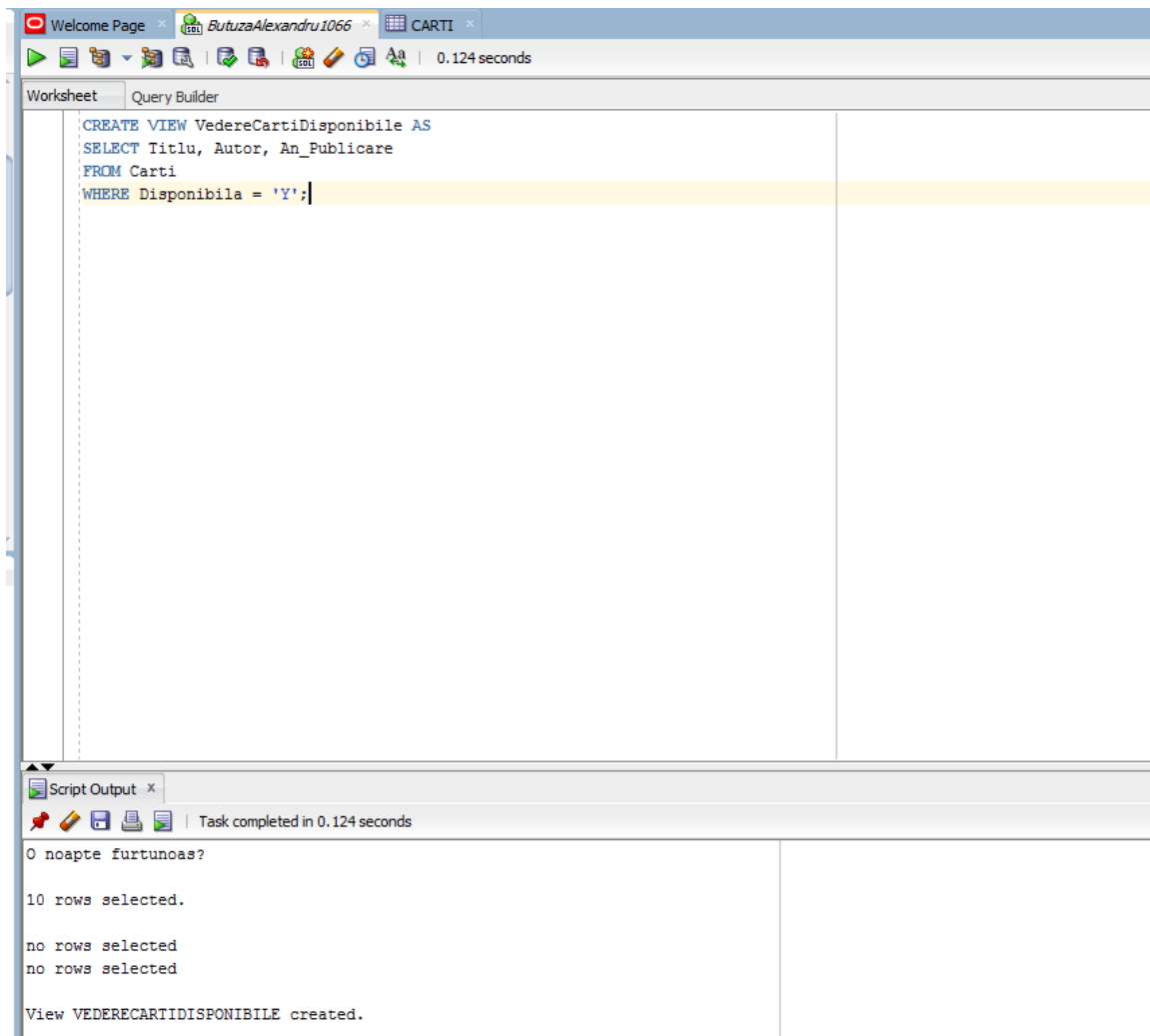
SELECT INDEX_NAME FROM ALL_INDEXES WHERE TABLE_NAME = 'Carti';



9. Gestiunea altor obiecte ale bazei de date: vederi, indecsi, sinonime, secvente.

9.1 Vedere

```
CREATE VIEW VedereCartiDisponibile AS
SELECT Titlu, Autor, An_Publicare
FROM Carti
WHERE Disponibila = 'Y';
```

9.2 Indeksi

CREATE INDEX idx_Utilizatori_Nume ON Utilizatori(Nume);

Welcome Page x ButuzaAlexandru1066 x CARTI x

0.116 seconds

Worksheet Query Builder

```
CREATE INDEX idx_Utilizatori_Nume ON Utilizatori(Nume);
```

Script Output x

Task completed in 0.116 seconds

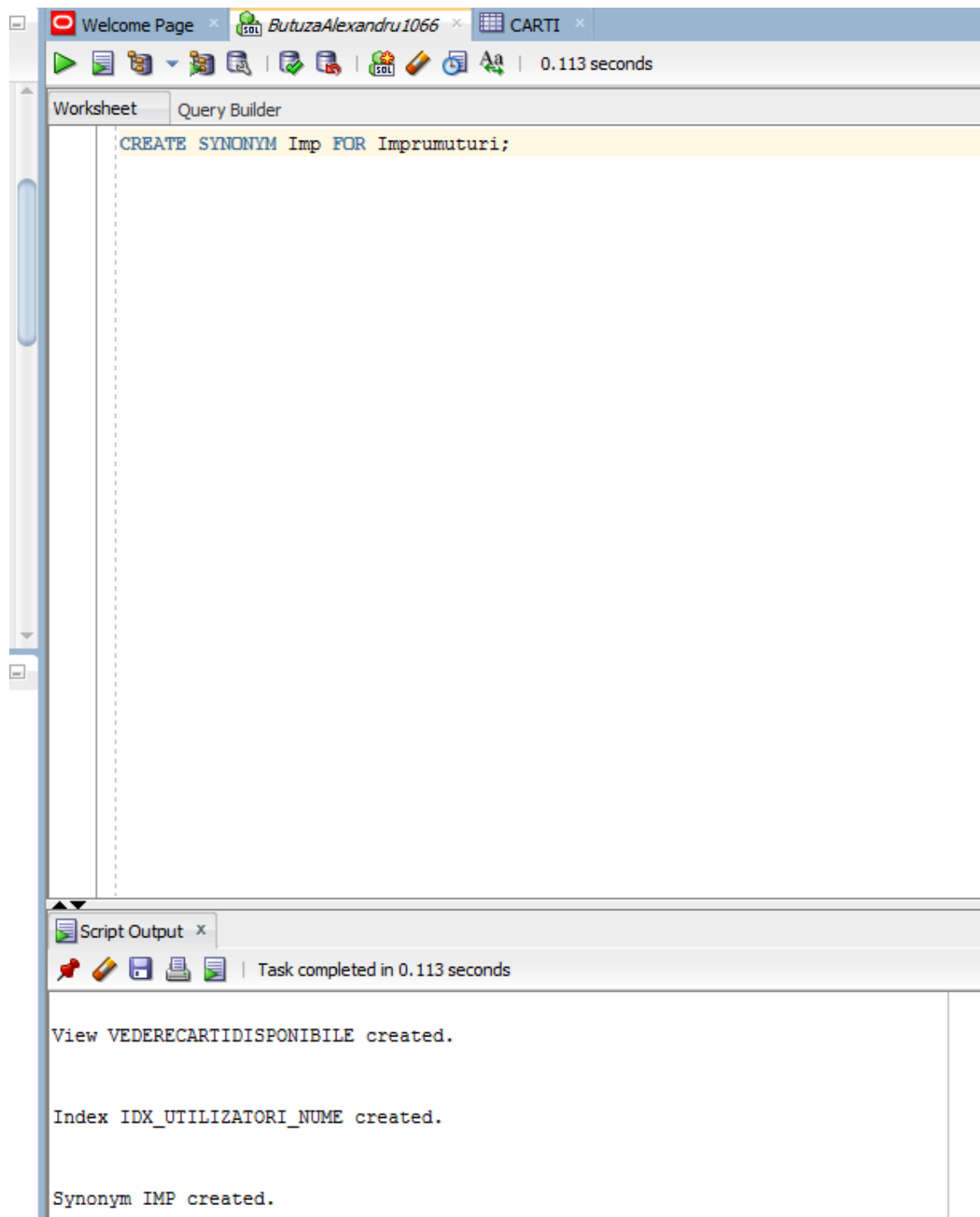
no rows selected
no rows selected

View VEDERECARTIDISPONIBILE created.

Index IDX_UTILIZATORI_NUME created.

9.3 Sinonime

CREATE SYNONYM Imp FOR Imprumuturi;



9.4 Secvente

CREATE SEQUENCE seq_rezervari START WITH 1 INCREMENT BY 1;

