

PROIECT BAZE DE DATE ȘCOALĂ DE ȘOFERI

Student: Dejan Marius-Ionuț

Grupa: 1051C

Descrierea bazei de date

**Proiectul reprezintă o bază de date ce ține evidența unei școli de șoferi.
În alcătuirea bazei de date intră 5 tabele principale și anume, tabela Instructori,
tabela Cursanți, tabela Garaj, tabela Vehicule și tabela Lecții.**

Tabela Instructori are drept câmpuri id_instructorului este cheia primară a tabelului, numele, prenumele, telefonul, email-ul, salariul și date de angajarea instructorului. La crearea tabelului am adăugat restricția de integritate NOT NULL pe toate câmpurile mai puțin asupra câmpului email la care am adăugat restricția UNIQUE. De asemenea, am adăugat pe câmpul salariu o restricție de tip CHECK ce verifică dacă acesta este mai mare ca 0. Tabela este legată de tabela Lecții printr o relație de one to many, astfel un instructor poate susține mai multe lecții, dar o lecție poate avea un singur instructor.

Tabela Cursanți are drept câmpuri id_cursant (primary key), numele, prenumele, telefonul, email-ul, vârsta, data de început și de final a școlii. Ca restricții speciale, am adăugat una de tip CHECK pentru a verifica dacă cursantul are 18 ani împliniți și restricția UNIQUE tot pe câmpul email-ului. Tabela este legată de tabela Lecții printr o relație de one to many, astfel un client poate participa la mai multe lecții, dar o lecție poate avea un singur client.

Tabela Garaj are drept câmpuri id_garaj (primary key), adresa și capacitatea sa. Aceasta tabelă este legată de tabela Vehicule printr o relație de one to many.

Tabela Vehicule are drept câmpuri id_vehicul (primary key) , id_garaj (foreign key), modelul, nr înmatriculare, kilometrajul, anul fabricației. Tabela este legată de tabela Lecții printr o legătura de one to many, astfel fiecare mașină poate fi folosită la mai multe lecții, dar o lecție poate avea doar o mașină ca mijloc de derulare a lecției.

Tabela Lecții are drept câmpuri id_lectie (primary key), data lectiei, locatia, id_instructor, id_cursant și id_vehicul ,acestea reprezentând foreign key. Aceasta este evident, legată de toate celelalte tabele prin relațiile enunțate anterior.

Crearea tabelelor

Crearea tabelii INSTRUCTORI:

```
CREATE TABLE Instructori
(id_instructor NUMBER(5) NOT NULL CONSTRAINT pk_instructor PRIMARY KEY,
nume VARCHAR2(25) NOT NULL,
prenume VARCHAR2(25) NOT NULL,
telefon VARCHAR2(20) NOT NULL,
email VARCHAR2(25) UNIQUE,
salariul NUMBER(8,2) NOT NULL CONSTRAINT SALARIUL_CK CHECK(salariul>0),
data_angajare DATE NOT NULL
);
```

Crearea tabelii CURSANȚI:

```
CREATE TABLE CURSANTI
(id_cursant NUMBER(5) NOT NULL CONSTRAINT pk_cursant PRIMARY KEY,
nume VARCHAR2(25) NOT NULL,
prenume VARCHAR2(25) NOT NULL,
telefon VARCHAR2(20),
email VARCHAR2(25) UNIQUE NOT NULL,
varsta NUMBER(4) NOT NULL CONSTRAINT VARSTA_CK check(varsta>=18),
data_incepere DATE NOT NULL,
data_finalizare DATE NOT NULL
);
```

Crearea tabelii GARAJ:

```
CREATE TABLE GARAJ
(id_garaj NUMBER(3) NOT NULL CONSTRAINT pk_garaj PRIMARY KEY,
adresa VARCHAR2(30) NOT NULL,
capacitate NUMBER(3)
);
```

Crearea tabelii VEHICULE:

```
CREATE TABLE VEHICULE
(id_vehicul NUMBER(5) NOT NULL CONSTRAINT pk_vehicul PRIMARY KEY,
model VARCHAR2(15) NOT NULL,
nr_inmatriculare VARCHAR(15) NOT NULL,
kilometraj NUMBER(6),
```

```

an_fabricatie NUMBER(4),
id_garaj NUMBER(3),
CONSTRAINT fk_vehicule FOREIGN KEY (id_garaj) REFERENCES GARAJ(id_garaj)
);

```

Crearea tabelii LECTII:

```
CREATE TABLE LECTII
```

```
(id_lectie NUMBER(5) NOT NULL CONSTRAINT pk_lectie PRIMARY KEY,
```

```
data_lectiei DATE NOT NULL,
```

```
locatie VARCHAR2(25) NOT NULL,
```

```
id_instructor NUMBER(5) ,
```

```
id_vehicul NUMBER(5),
```

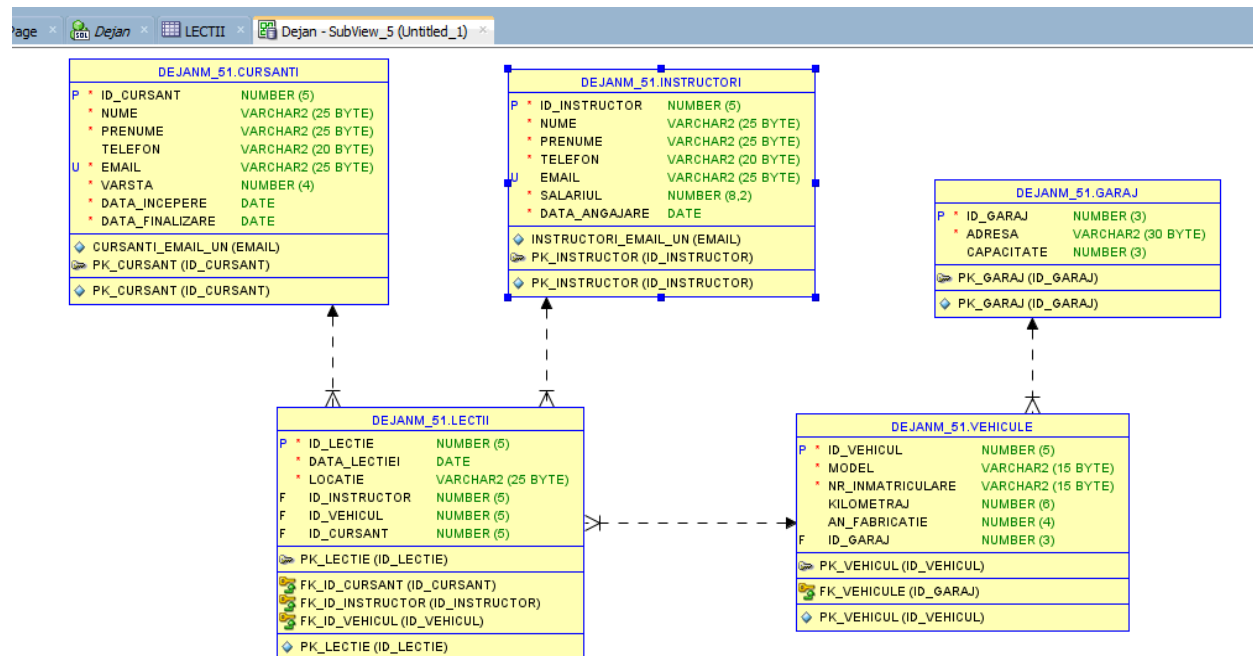
```
id_cursant NUMBER(5),
```

```
CONSTRAINT fk_id_instructor FOREIGN KEY(id_instructor) REFERENCES
Instructori(id_instructor),
```

```
CONSTRAINT fk_id_vehicul FOREIGN KEY(id_vehicul) REFERENCES
Vehicule(id_vehicul),
```

```
CONSTRAINT fk_id_cursant FOREIGN KEY(id_cursant) REFERENCES
Cursanti(id_cursant)
);
```

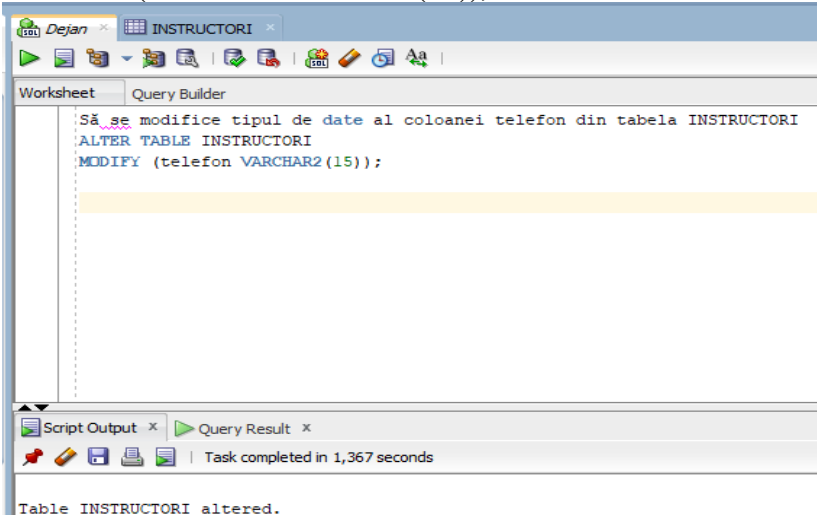
Aceasta este schema inițială a bazei de date, generată după crearea tabelelor cu comenzile
aferente de mai sus



Actualizarea structurii tabelelor și modificarea restricțiilor de integritate

Să se modifice tipul de date al coloanei telefon din tabela INSTRUCTORI:

```
ALTER TABLE INSTRUCTORI  
MODIFY (telefon VARCHAR2(15));
```



The screenshot shows the Dejan SQL editor interface with the 'Columns' tab selected. It displays the structure of the 'INSTRUCTORI' table, including column names, data types, nullability, and comments.

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_INSTRUCTOR	NUMBER (5, 0)	No	(null)	1	(null)
2	NUME	VARCHAR2 (25 BYTE)	No	(null)	2	(null)
3	PRENUME	VARCHAR2 (25 BYTE)	No	(null)	3	(null)
4	TELEFON	<u>VARCHAR2 (15 BYTE)</u>	No	(null)	4	(null)
5	EMAIL	VARCHAR2 (25 BYTE)	Yes	(null)	5	(null)
6	SALARIUL	NUMBER (8, 2)	No	(null)	6	(null)
7	DATA_ANGAJARE	DATE	No	(null)	7	(null)

Să se adauge o restricție nouă asupra coloanei data_finalizare din tabela CURSANȚI:

```
alter table CURSANTI  
add CONSTRAINT DATA_CK  
check(data_finalizare>data_incepere);
```

Dejan x CURSANTI x			
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL			
Actions...			
CONSTRAINT_...	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER
1 DATA_CK	Check	data_finalizare>data_incepere	(null)

Să se adauge o restricție pe coloana an_fabricatie din tabela VEHICULE:

ALTER TABLE VEHICULE

ADD CONSTRAINT an_fabr_ck CHECK(an_fabricatie>2005);

Dejan x INSTRUCTORI x

Worksheet | Query Builder

```
ALTER TABLE VEHICULE
ADD CONSTRAINT an_fabr_ck CHECK(an_fabricatie>2005);
```

Script Output x | Query Result x

Task completed in 0,093 seconds

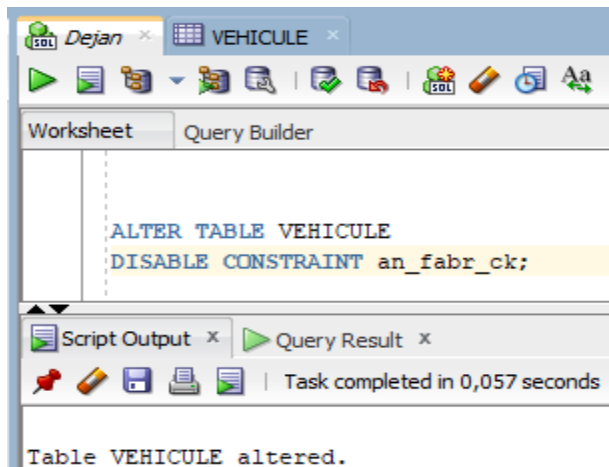
Table VEHICULE altered.

Dejan x VEHICULE x			
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies			
Actions...			
CONSTRAINT_...	CONSTRAINT_TYPE	SEARCH_CONDITION	
1 AN_FABR_CK	Check	an_fabricatie>2005	

Să se dezactiveze restricția creată anterior:

ALTER TABLE VEHICULE

DISABLE CONSTRAINT an_fabr_ck;

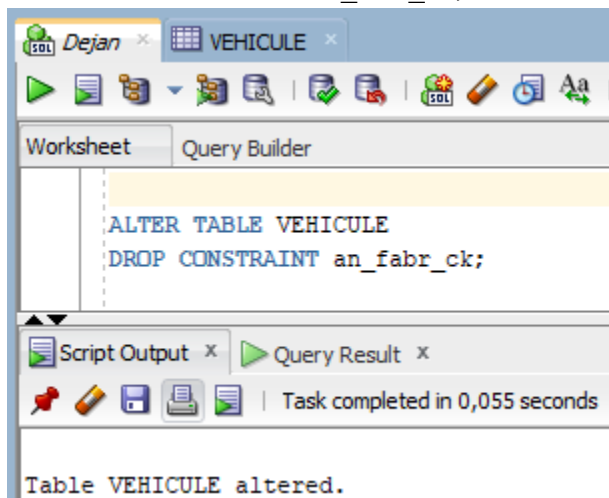


Columns	Data	Model	Constraints	Grants	Statistics	Triggers	Flashback	Dependencies	Details	Partitions	Indexes	SQL
Actions...												
1	CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_...	R_...	R_CONS...	DELET...	STATUS				
1	AN_FABR_CK	Check	an_fabricatie>2005	(n...	(n...	(null)	(null)	<u>DISABLED</u>				

Să se șteargă restricția creată anterior:

ALTER TABLE VEHICULE

DROP CONSTRAINT an_fabr_ck;



Să se adauge coloana sex în tabela CURSANȚI:

ALTER TABLE CURSANTI

ADD(SEX VARCHAR2(3));

The screenshot shows the Dejan SQL IDE interface. The 'Query Builder' tab is active, displaying the SQL command: `ALTER TABLE CURSANTI ADD (SEX VARCHAR2 (3));`. The 'Script Output' tab shows the message: 'Table CURSANTI altered.' The 'Query Result' tab displays the table structure for CURSANTI:

	COLUMN_NAME	DATA_TYPE
1	ID_CURSANT	NUMBER (5, 0)
2	NUME	VARCHAR2 (25 BYTE)
3	PRENUME	VARCHAR2 (25 BYTE)
4	TELEFON	VARCHAR2 (20 BYTE)
5	EMAIL	VARCHAR2 (25 BYTE)
6	VARSTA	NUMBER (4, 0)
7	DATA_INCEPERE	DATE
8	DATA_FINALIZARE	DATE
9	<u>SEX</u>	VARCHAR2 (3 BYTE)

Să se şteargă coloana sex din tabela CURSANTI:

`ALTER TABLE CURSANTI
ADD(SEX VARCHAR2(3));`

The screenshot shows the Dejan SQL IDE interface. The 'Query Builder' tab is active, displaying the SQL command: `ALTER TABLE CURSANTI DROP COLUMN SEX;`. The 'Script Output' tab shows the message: 'Table CURSANTI altered.' The 'Query Result' tab displays the table structure for CURSANTI:

	COLUMN_NAME	DATA_TYPE
1	ID_CURSANT	NUMBER (5, 0)
2	NUME	VARCHAR2 (25 BYTE)
3	PRENUME	VARCHAR2 (25 BYTE)
4	TELEFON	VARCHAR2 (20 BYTE)
5	EMAIL	VARCHAR2 (25 BYTE)
6	VARSTA	NUMBER (4, 0)
7	DATA_INCEPERE	DATE
8	DATA_FINALIZARE	DATE

ADĂUGAREA ÎNREGISTRĂRILOR ÎN TABELE

Adăugarea înregistrărilor în tabela INSTRUCTORI:

```
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(130,'Ursu','Ioan','0734345780','ursuioan@gmail.com',3750,TO_DATE('21-09-2020', 'dd-mm-
yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(131,'Damian','Cătălin','0747832534','catalindamian@gmail.com',3300,TO_DATE('14-10-2020',
'dd-mm-yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(132,'Manciu','Liviu','0723465583','manciuliviu13@gmail.com',3860,TO_DATE('18-10-2020', 'dd-
mm-yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(133,'Dumitru','Matei','0752420635','mateidumitru@gmail.com',3750,TO_DATE('13-03-2021', 'dd-
mm-yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(134,'Pogar','Alin','0736764332','pogaraulin@gmail.com',3354,TO_DATE('21-10-2021', 'dd-mm-
yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(135,'Sava','Cristi','0734430327','cristisava@yahoo.com',4250,TO_DATE('03-11-2021', 'dd-mm-
yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(138,'Toma','Daniel','0767343710','tomadaniel@gmail.com',3830,TO_DATE('28-08-2022', 'dd-
mm-yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(136,'Mitrea','Marian','0731342783','mitreamarian@gmail.com',3800,TO_DATE('23-04-2022', 'dd-
mm-yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(137,'Radu','Marian','0734356220','radumarian@yahoo.com',4200,TO_DATE('23-04-2022', 'dd-
mm-yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(140,'Cosa','Adi','0743909003','adicosa@gmail.com',4000,TO_DATE('18-11-2022', 'dd-mm-
yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(141,'Bibire','Bogdan','0743233003','bibirebogdan@gmail.com',4150,TO_DATE('10-12-2022', 'dd-
mm-yyyy'));
INSERT INTO INSTRUCTORI(id_instructor,nume,prenume,telefon,email,salariul,data_angajare)
values(139,'Sichim','Rares','0752620224','raessic@gmail.com',4430,TO_DATE('09-09-2022', 'dd-mm-
yyyy'));
```

Dejan x INSTRUCTORI							
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL							
Sort.. Filter:							
	ID_INSTRUCTOR	NUME	PRENUME	TELEFON	EMAIL	SALARIUL	DATA_ANGAJARE
1	139	Sichim	Rares	0752620224	raressic@gmail.com	4430	09-SEP-22
2	130	Ursu	Ioan	0734345780	ursuioan@gmail.com	3750	21-SEP-20
3	131	Damian	Catalin	0747832534	catalindamian@gmail.com	3300	14-OCT-20
4	132	Manciu	Liviu	0723465583	manciuliviu13@gmail.com	3860	18-OCT-20
5	133	Dumitru	Matei	0752420635	mateidumitru@gmail.com	3750	13-MAR-21
6	134	Pogar	Alin	0736764332	pogaraulin@gmail.com	3354	21-OCT-21
7	135	Sava	Cristi	0734430327	cristisava@yahoo.com	4250	03-NOV-21
8	138	Toma	Daniel	0767343710	tomadaniel@gmail.com	3830	28-AUG-22
9	136	Mitrea	Marian	0731342783	mitreamarian@gmail.com	3800	23-APR-22
10	137	Radu	Marian	0734356220	radumarian@yahoo.com	4200	23-APR-22
11	141	Bibire	Bogdan	0743233003	bibirebogdan@gmail.com	4150	10-DEC-12
12	140	Cosa	Adi	0743909003	adicosa@gmail.com	4000	18-NOV-12

Adăugarea înregistrărilor în tabela CURSANȚI:

INSERT INTO

CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(3220,'Vlad','Andreea','0755909003','andreea248@gmail.com',20,to_date('01-10-2020',
'dd-mm-yyyy'),to_date('15-12-2020','dd-mm-yyyy'));

INSERT INTO

CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(1345,'Popa','Marius','0753432117','mariuspopa@gmail.com',27,to_date('24-05-2021',
'dd-mm-yyyy'),to_date('27-08-2021','dd-mm-yyyy'));

INSERT INTO

CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(1482,'Moneaga','Tudor','0742155668','montudor@gmail.com',18,to_date('15-11-2020',
'dd-mm-yyyy'),to_date('05-02-2021','dd-mm-yyyy'));

INSERT INTO

CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(6521,'Toma','Sabin','0753039081','tomasabin@gmail.com',34,to_date('08-06-2022',
'dd-mm-yyyy'),to_date('11-08-2022','dd-mm-yyyy'));

INSERT INTO

CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(3473,'Popescu','Ion','0729987003','ionpopescu@gmail.com',38,to_date('13-12-2020',
'dd-mm-yyyy'),to_date('04-02-2021','dd-mm-yyyy'));

INSERT INTO

CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(2360,'Voicu','Mihai','0770408472','voicumihai@gmail.com',26,to_date('05-07-2022',
'dd-mm-yyyy'),to_date('09-10-2022','dd-mm-yyyy'));

```

INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(987,'Coman','Gabriela','0755932210','gabrielacoman@gmail.com',23,to_date('25-11-
2021', 'dd-mm-yyyy'),to_date('06-01-2022', 'dd-mm-yyyy'));
INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(3152,'Mocanu','Irina','0787672421','mocanuirina@gmail.com',40,to_date('03-01-
2021', 'dd-mm-yyyy'),to_date('13-03-2021', 'dd-mm-yyyy'));
INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(4268,'Vlad','Bianca','0740566274','biancavlad@gmail.com',44,to_date('01-09-2021',
'dd-mm-yyyy'),to_date('04-12-2021', 'dd-mm-yyyy'));
INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(4821,'Aanitei','Daria','0755620628','aaniteidaria@gmail.com',30,to_date('16-11-2020',
'dd-mm-yyyy'),to_date('12-01-2021', 'dd-mm-yyyy'));
INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(5200,'Cucu','Andrei','0741090173','cucuvlad@gmail.com',33,to_date('20-11-2022',
'dd-mm-yyyy'),to_date('06-01-2023', 'dd-mm-yyyy'));
INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(6432,'Stefan','Maria','0752583683','anastefan@gmail.com',44,to_date('10-10-2022',
'dd-mm-yyyy'),to_date('28-12-2022', 'dd-mm-yyyy'));
INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(713,'Rosu','Calin','0753383643','calinrosu@gmail.com',18,to_date('17-03-2022', 'dd-
mm-yyyy'),to_date('10-06-2022', 'dd-mm-yyyy'));
INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(520,'Preda','Andi','0723983120','andipreda@gmail.com',47,to_date('28-04-2022', 'dd-
mm-yyyy'),to_date('25-06-2022', 'dd-mm-yyyy'));
INSERT INTO
CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(1723,'Bucur','Victor','0724223003','bucurvictor@gmail.com',41,to_date('18-11-2021',
'dd-mm-yyyy'),to_date('15-02-2022', 'dd-mm-yyyy'));

```

Dejan x CURSANTI x

Worksheet Query Builder

```
INSERT INTO CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(713,'Rosu','Calin','0753383643','calinrosu@gmail.com',18,to_date('17-03-2022','dd-mm-yyyy'),to_date('10-06-2022','dd-mm-yyyy'));
INSERT INTO CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(520,'Preda','Andi','0723983120','andipreda@gmail.com',47,to_date('28-04-2022','dd-mm-yyyy'),to_date('25-06-2022','dd-mm-yyyy'));
INSERT INTO CURSANTI(id_cursant,nume,prenume,telefon,email,varsta,data_incepere,data_finalizare)
VALUES(1723,'Bucur','Victor','0724223003','bucurvictor@gmail.com',41,to_date('18-11-2021','dd-mm-yyyy'),to_date('15-02-2022','dd-mm-yyyy'));

SELECT * FROM CURSANTI;
```

Script Output x Query Result x

SQL | All Rows Fetched: 15 in 0,013 seconds

ID_CURSANT	NUME	PRENUME	TELEFON	EMAIL	VARSTA	DATA_INCEPERE	DATA_FINALIZARE	
1	3220	Vlad	Andreea	0755909003	andreea248@gmail.com	20	01-OCT-20	15-DEC-20
2	1345	Popa	Marius	0753432117	mariuspopa@gmail.com	27	24-MAY-21	27-AUG-21
3	6521	Toma	Sabin	0753039081	tomasabin@gmail.com	34	08-JUN-22	11-AUG-22
4	3473	Popescu	Ion	0729987003	ionpopescu@gmail.com	38	13-DEC-20	04-FEB-21
5	2360	Voicu	Mihai	0770408472	voicumihai@gmail.com	26	05-JUL-22	09-OCT-22
6	987	Coman	Gabriela	0755932210	gabrielacoman@gmail.com	23	25-NOV-21	06-JAN-22
7	3152	Mocanu	Irina	0787672421	mocanuirina@gmail.com	40	03-JAN-21	13-MAR-21
8	4268	Vlad	Bianca	0740566274	biancavlad@gmail.com	44	01-SEP-21	04-DEC-21
9	4821	Aanitei	Daria	0755620628	aaniteidaria@gmail.com	30	16-NOV-20	12-JAN-21
10	5200	Cucu	Andrei	0741090173	cucuvlad@gmail.com	33	20-NOV-22	06-JAN-23
11	6432	Stefan	Maria	0752583683	anastefan@gmail.com	44	10-OCT-22	28-DEC-22
12	713	Rosu	Calin	0753383643	calinrosu@gmail.com	18	17-MAR-22	10-JUN-22
13	520	Preda	Andi	0723983120	andipreda@gmail.com	47	28-APR-22	25-JUN-22
14	1482	Moneaga	Tudor	0742155668	montudor@gmail.com	18	15-NOV-20	05-FEB-21
15	1723	Bucur	Victor	0724223003	bucurvictor@gmail.com	41	18-NOV-21	15-FEB-22

Adăugarea înregistrărilor în tabela GARAJ:

```

INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(1,'Str.Stefan cel Mare 7b',2);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(2,'Stefan cel Mare 7c',1);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(6,'Str.Mioritei 167',2);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(8,'Str.Republicii 255',1);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(5,'Str.Garii 74',2);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(3,'Str.Calea Moldovei 144',1);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(9,'Str.Venus 23',2);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(10,'Str.Mihai Viteazu 37',2);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(7,'Str.Nicolae Balcescu 18a',1);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(4,'Str.Calea Moldovei 146',2);

```

Dejan x GARAJ x

Worksheet Query Builder

```

INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(8,'Str.Republicii 255',1);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(5,'Str.Garii 74',2);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(3,'Str.Calea Moldovei 144',1);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(9,'Str.Venus 23',2);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(10,'Str.Mihai Viteazu 37',2);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(7,'Str.Nicolae Balcescu 18a',1);
INSERT INTO GARAJ(id_garaj,adresa,capacitate)
VALUES(4,'Str.Calea Moldovei 146',2);
SELECT* FROM GARAJ;

```

Script Output x Query Result x

SQL | All Rows Fetched: 10 in 0,014 seconds

ID_GARAJ	ADRESA	CAPACITATE
1	1 Str.Stefan cel Mare 7b	2
2	2 Stefan cel Mare 7c	1
3	6 Str.Mioritei 167	2
4	8 Str.Republicii 255	1
5	5 Str.Garii 74	2
6	3 Str.Calea Moldovei 144	1
7	9 Str.Venus 23	2
8	10 Str.Mihai Viteazu 37	2
9	7 Str.Nicolae Balcescu 18a	1
10	4 Str.Calea Moldovei 146	2

Adăugarea înregistrărilor în tabela VEHICULE:

```

INSERT INTO VEHICULE(id_vehicul,
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)
VALUES(602,'BMW X3','BC-75-DDS',125000,2014,8);
INSERT INTO VEHICULE(id_vehicul,
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)
VALUES(578,'VW GOLF 6','BC-13-DDS',274357,2011,1);
INSERT INTO VEHICULE(id_vehicul,
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)
VALUES(474,'RENAULT MEGANE','BC-06-DDS',78322,2019,3);

```

```
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(503,'VW GOLF 7','BC-15-DDS',84723,2019,1);  
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(524,'VW PASSAT B8','BC-23-DDS',47852,2020,7);  
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(783,'MAZDA CX3','BC-73-DDS',254328,2016,9);  
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(436,'RENAULT CLIO 4','BC-48-DDS',176455,2017,9);  
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(244,'CITROEN C3','BC-99-DDS',193452,2015,2);  
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(137,'PEUGEOT 308','BC-09-DDS',122454,2019,5);  
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(540,'VW GOLF 6','BC-19-DDS',204534,2017,4);  
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(837,'SKODA OCTAVIA','BC-30-DDS',139564,2018,6);  
INSERT INTO VEHICULE(id_vehicul,  
model,nr_inmatriculare,kilometraj,an_fabricatie,id_garaj)  
VALUES(874,'SKODA FABIA','BC-31-DDS',167520,2018,10);
```

Dejan x VEHICULE x						
Worksheet Query Builder						
<pre>VALUES (837, 'SKODA OCTAVIA', 'BC-30-DDS', 139564, 2018, 6); INSERT INTO VEHICULE(id_vehicul, model, nr_inmatriculare, kilometraj, an_fabricatie, id_garaj) VALUES (874, 'SKODA FABIA', 'BC-31-DDS', 167520, 2018, 10); SELECT * FROM VEHICULE;</pre>						
Script Output x Query Result x						
SQL All Rows Fetched: 12 in 0,011 seconds						
	ID_VEHICUL	MODEL	NR_INMATRICULARE	KILOMETRAJ	AN_FABRICATIE	ID_GARAJ
1	578	VW GOLF 6	BC-13-DDS	274357	2011	1
2	503	VW GOLF 7	BC-15-DDS	84723	2019	1
3	524	VW PASSAT B8	BC-23-DDS	47852	2020	7
4	783	MAZDA CX3	BC-73-DDS	254328	2016	9
5	436	RENAULT CLIO 4	BC-48-DDS	176455	2017	9
6	244	CITROEN C3	BC-99-DDS	193452	2015	2
7	137	PEUGEOT 308	BC-09-DDS	122454	2019	5
8	540	VW GOLF 6	BC-19-DDS	204534	2017	4
9	837	SKODA OCTAVIA	BC-30-DDS	139564	2018	6
10	874	SKODA FABIA	BC-31-DDS	167520	2018	10
11	474	RENAULT MEGANE	BC-06-DDS	78322	2019	3
12	602	BMW X3	BC-75-DDS	125000	2014	8

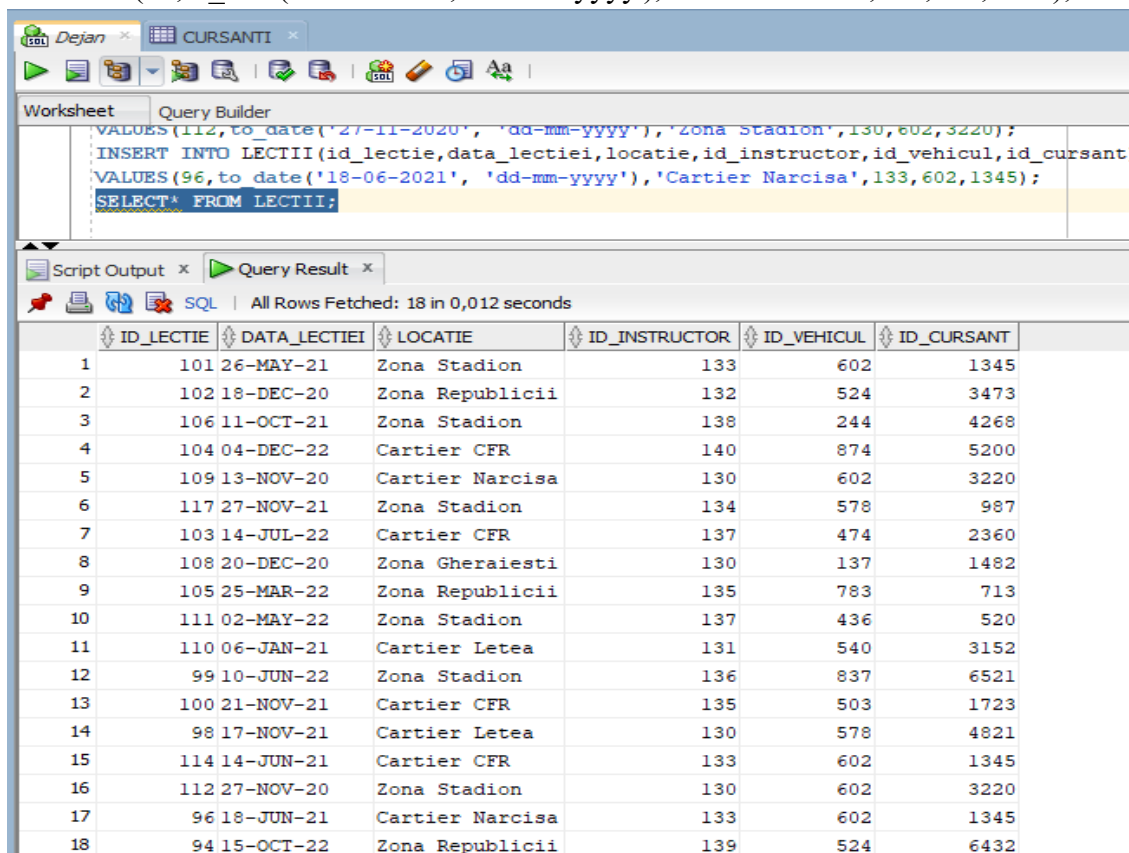
Adăugarea înregistrărilor în tabela LECTII:

```
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(101,to_date('26-05-2021', 'dd-mm-yyyy'),'Zona Stadion',133,602,1345);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(102,to_date('18-12-2020', 'dd-mm-yyyy'),'Zona Republicii',132,524,3473);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(106,to_date('11-10-2021', 'dd-mm-yyyy'),'Zona Stadion',138,244,4268);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(104,to_date('04-12-2022', 'dd-mm-yyyy'),'Cartier CFR',140,874,5200);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(109,to_date('13-11-2020', 'dd-mm-yyyy'),'Cartier Narcisa',130,602,3220);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(117,to_date('27-11-2021', 'dd-mm-yyyy'),'Zona Stadion',134,578,987);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(103,to_date('14-07-2022', 'dd-mm-yyyy'),'Cartier CFR',137,474,2360);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(108,to_date('20-12-2020', 'dd-mm-yyyy'),'Zona Gheraiesti',130,137,1482);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
```

```

VALUES(105,to_date('25-03-2022', 'dd-mm-yyyy'),'Zona Republicii',135,783,713);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(111,to_date('02-05-2022', 'dd-mm-yyyy'),'Zona Stadion',137,436,520);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(110,to_date('06-01-2021', 'dd-mm-yyyy'),'Cartier Letea',131,540,3152);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(99,to_date('10-06-2022', 'dd-mm-yyyy'),'Zona Stadion',136,837,6521);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(100,to_date('21-11-2021', 'dd-mm-yyyy'),'Cartier CFR',135,503,1723);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(98,to_date('17-11-2021', 'dd-mm-yyyy'),'Cartier Letea',130,578,4821);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(94,to_date('15-10-2022', 'dd-mm-yyyy'),'Zona Republicii',139,524,6432);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(114,to_date('14-06-2021', 'dd-mm-yyyy'),'Cartier CFR',133,602,1345);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(112,to_date('27-11-2020', 'dd-mm-yyyy'),'Zona Stadion',130,602,3220);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(96,to_date('18-06-2021', 'dd-mm-yyyy'),'Cartier Narcisa',133,602,1345);

```



The screenshot shows a SQL IDE interface with a worksheet containing the following SQL query:

```

VALUES(112,to_date('27-11-2020', 'dd-mm-yyyy'),'Zona Stadion',130,602,3220);
INSERT INTO LECTII(id_lectie,data_lectiei,locatie,id_instructor,id_vehicul,id_cursant)
VALUES(96,to_date('18-06-2021', 'dd-mm-yyyy'),'Cartier Narcisa',133,602,1345);
SELECT* FROM LECTII;

```

The Query Result tab shows 18 rows of data. The columns are ID_LECTIE, DATA_LECTIEI, LOCATIE, ID_INSTRUCTOR, ID_VEHICUL, and ID_CURSANT.

	ID_LECTIE	DATA_LECTIEI	LOCATIE	ID_INSTRUCTOR	ID_VEHICUL	ID_CURSANT
1	101	26-MAY-21	Zona Stadion	133	602	1345
2	102	18-DEC-20	Zona Republicii	132	524	3473
3	106	11-OCT-21	Zona Stadion	138	244	4268
4	104	04-DEC-22	Cartier CFR	140	874	5200
5	109	13-NOV-20	Cartier Narcisa	130	602	3220
6	117	27-NOV-21	Zona Stadion	134	578	987
7	103	14-JUL-22	Cartier CFR	137	474	2360
8	108	20-DEC-20	Zona Gheraiesti	130	137	1482
9	105	25-MAR-22	Zona Republicii	135	783	713
10	111	02-MAY-22	Zona Stadion	137	436	520
11	110	06-JAN-21	Cartier Letea	131	540	3152
12	99	10-JUN-22	Zona Stadion	136	837	6521
13	100	21-NOV-21	Cartier CFR	135	503	1723
14	98	17-NOV-21	Cartier Letea	130	578	4821
15	114	14-JUN-21	Cartier CFR	133	602	1345
16	112	27-NOV-20	Zona Stadion	130	602	3220
17	96	18-JUN-21	Cartier Narcisa	133	602	1345
18	94	15-OCT-22	Zona Republicii	139	524	6432

ACTUALIZAREA ÎNREGISTRĂRILOR

Să se actualizeze locația lecției cu id-ul 114 cu locația lecției cu id-ul 108.

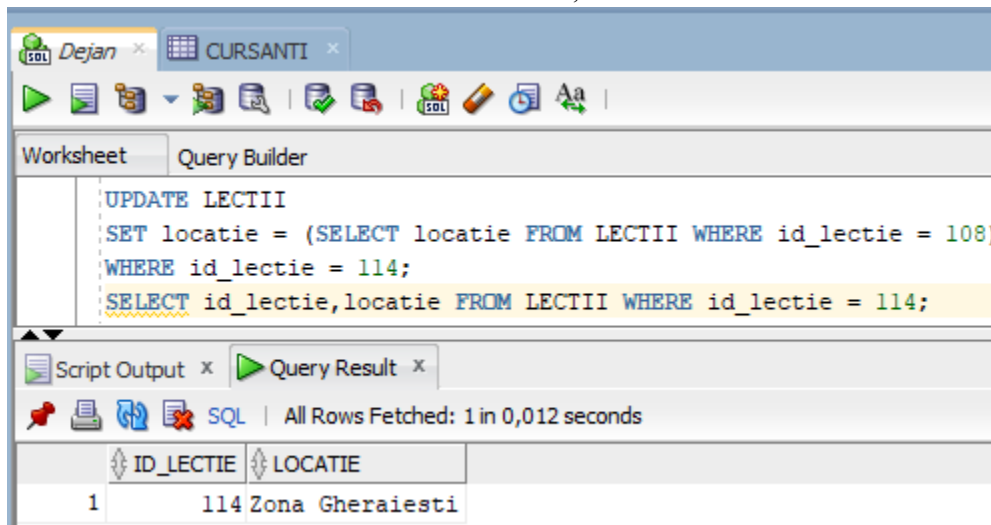
UPDATE LECTII

SET locatie = (SELECT locatie FROM LECTII WHERE id_lectie = 108)

WHERE id_lectie = 114;

SELECT id_lectie,locatie FROM LECTII WHERE id_lectie = 114;

//locatia a devenit din 'Cartier CFR' în Gherăiești//



Să se actualizeze modelul vehiculului cu id-ul 524.

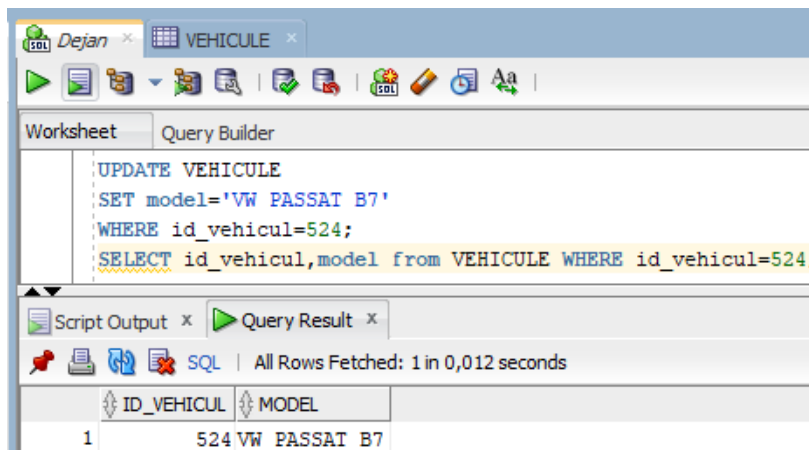
UPDATE VEHICULE

SET model='VW PASSAT B7'

WHERE id_vehicul=524;

SELECT id_vehicul,model from VEHICULE WHERE id_vehicul=524;

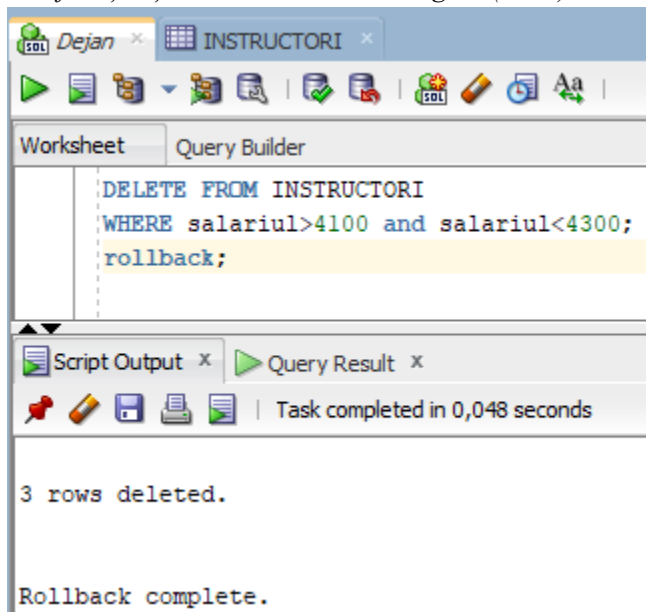
//modelul a devenit din 'VW PASSAT B8' în 'VW PASSAT B7'//



Să se ștergă înregistrările din tabela INSTRUCTORI ce au anul salariul cuprins între 4100 și 4300. Să se anuleze tranzacția.

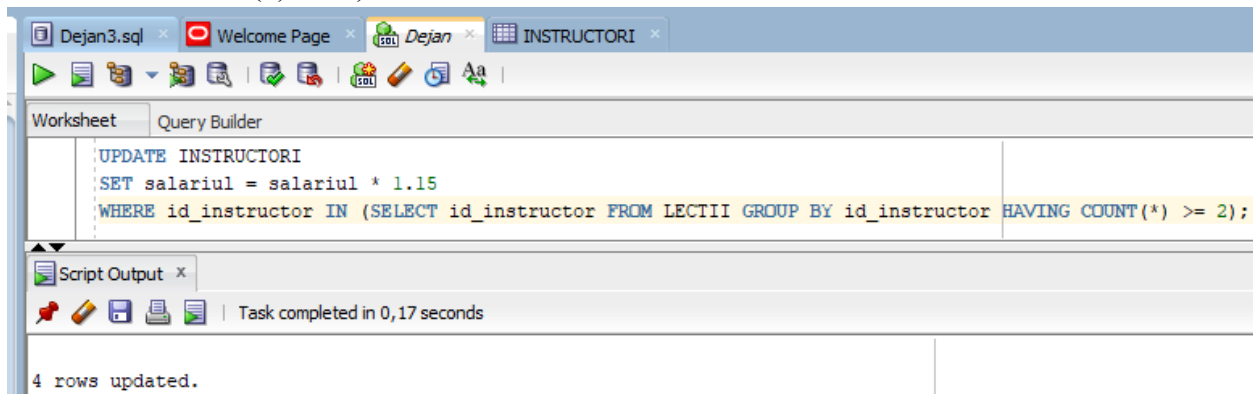
```
DELETE FROM INSTRUCTORI  
WHERE salariul>4100 and salariul<4300;  
rollback;
```

//au fost șterși instructorii Bibire Bogdan(4150), Radu Marian(4200), Sava Cristi(4250)//



Să se mărească cu 15% salariul instructorilor ce au susținut minim 2 lecții.

```
UPDATE INSTRUCTORI  
SET salariul = salariul * 1.15  
WHERE id_instructor IN (SELECT id_instructor FROM LECTII GROUP BY id_instructor  
HAVING COUNT(*) >= 2);
```



//S-a actualizat salariul instructorilor cu id-ul 130, 133, 135, 137//

The screenshot shows the SQL Developer interface with a worksheet named 'Dejan' and a query titled 'INSTRUCTORI'. The query is `select id_instructor, salariul from INSTRUCTORI;`. The 'Query Result' pane shows 12 rows of data. The status bar indicates 'All Rows Fetched: 12 in 0,02 seconds'.

	ID_INSTRUCTOR	SALARIUL
1	130	4312.5
2	131	3300
3	132	3860
4	133	4312.5
5	134	3354
6	135	4887.5
7	138	3830
8	136	3800
9	137	4830
10	140	4000
11	141	4150
12	139	4430

ȘTERGEREA ȘI RECUPERAREA UNEI TABELE

DROP TABLE INSTRUCTORI

CASCADE CONSTRAINTS;

FLASHBACK TABLE INSTRUCTORI TO BEFORE DROP;

The screenshot shows the SQL Developer interface with a worksheet named 'Dejan2.sql' and a query titled 'LECTII'. The query contains the following SQL commands: `DROP TABLE INSTRUCTORI CASCADE CONSTRAINTS;` and `FLASHBACK TABLE INSTRUCTORI TO BEFORE DROP;`. The 'Script Output' pane shows the results of the execution: 'Table INSTRUCTORI dropped.' and 'Flashback succeeded.' The status bar indicates 'Task completed in 0,081 seconds'.

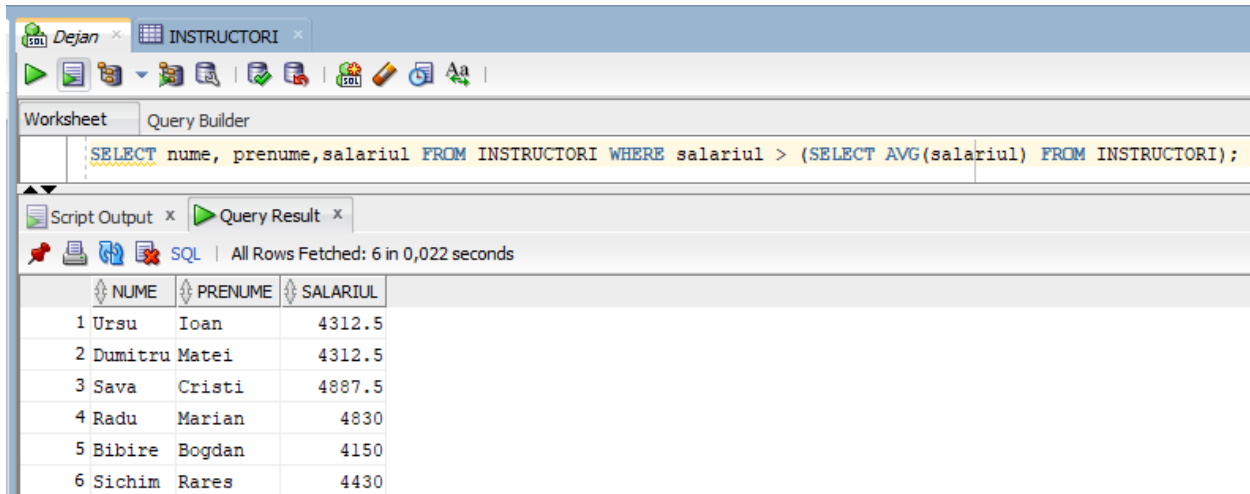
Script Output
Table INSTRUCTORI dropped.
Flashback succeeded.

Interogări variate

Utilizarea operatorilor de comparație

Să se afișeze numele, prenumele și salariul instructorilor ce au salariul mai mare decât media salariilor tuturor instructorilor.

```
SELECT nume, prenume, salariul FROM INSTRUCTORI WHERE salariul > (SELECT AVG(salariul) FROM INSTRUCTORI);
```



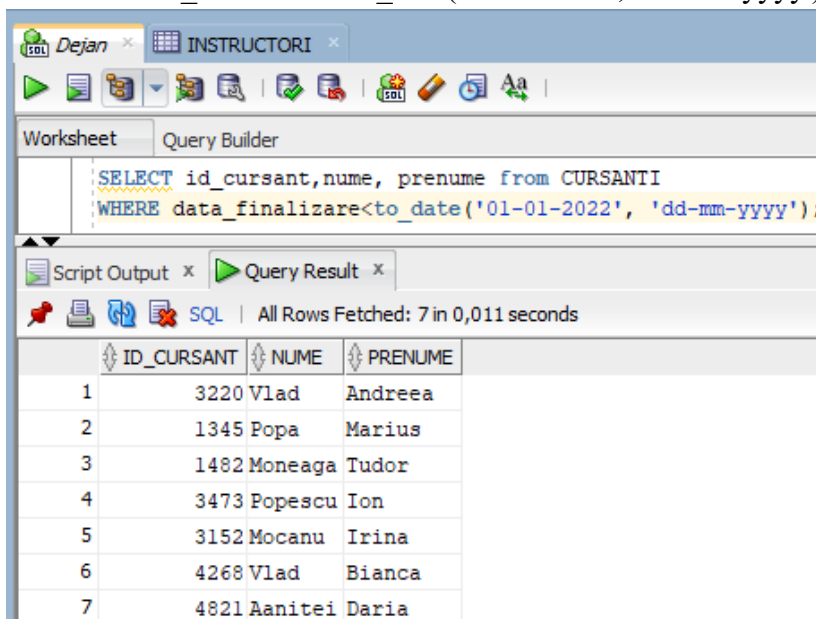
The screenshot shows the SQL Developer interface with a query executed against the INSTRUCTORI table. The query filters for instructors whose salary is greater than the average salary of all instructors. The result set contains six rows of data.

	NUME	PRENUME	SALARIUL
1	Ursu	Ioan	4312.5
2	Dumitru	Matei	4312.5
3	Sava	Cristi	4887.5
4	Radu	Marian	4830
5	Bibire	Bogdan	4150
6	Sichim	Rares	4430

//Media salariilor este 4088.875 *SELECT AVG(salariul) FROM INSTRUCTORI;*//

Să se selecteze toți cursanții care au terminat școala înainte de 01.01.2022.

```
SELECT id_cursant, nume, prenume from CURSANTII  
WHERE data_finalizare < to_date('01-01-2022', 'dd-mm-yyyy');
```



The screenshot shows the SQL Developer interface with a query executed against the CURSANTII table. The query filters for students whose completion date is before January 1, 2022. The result set contains seven rows of data.

	ID_CURSANT	NUME	PRENUME
1	3220	Vlad	Andreea
2	1345	Popa	Marius
3	1482	Moneaga	Tudor
4	3473	Popescu	Ion
5	3152	Mocanu	Irina
6	4268	Vlad	Bianca
7	4821	Aanitei	Daria

Funcții de grup

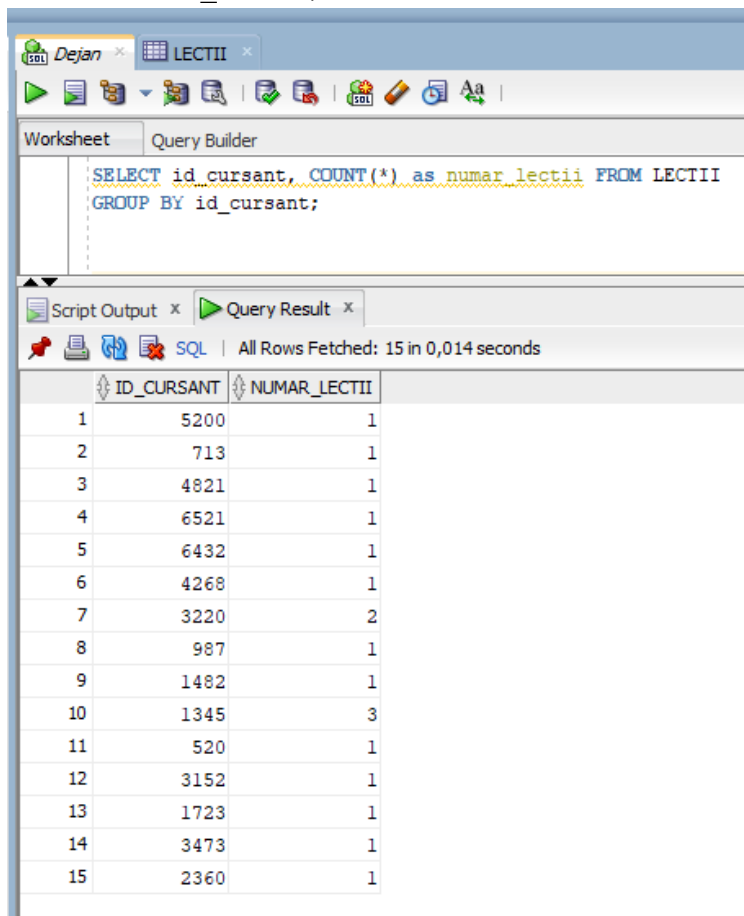
Să se afișeze numărul total de lecții susținute de fiecare instructor.

```
SELECT id_instructor, COUNT(*) as numar_lectii FROM LECTII  
GROUP BY id_instructor;
```

	ID_INSTRUCTOR	NUMAR_LECTII
1	138	1
2	134	1
3	132	1
4	133	3
5	137	2
6	135	2
7	140	1
8	130	4
9	131	1
10	136	1
11	139	1

Să se afișeze numărul total de lecții desfășurate de fiecare student.

```
SELECT id_cursant, COUNT(*) as numar_lectii FROM LECTII  
GROUP BY id_cursant;
```

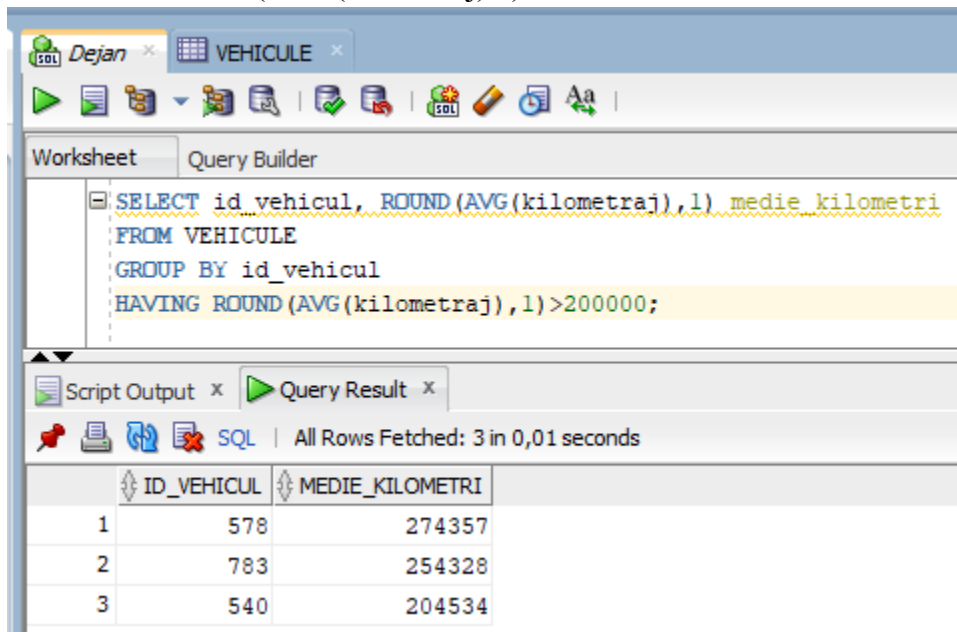


The screenshot shows a SQL query editor with a toolbar at the top. The query is entered in the main text area. Below the query, there is a 'Query Result' tab showing the results of the query. The results are displayed in a table with two columns: ID_CURSANT and NUMAR_LECTII. The table contains 15 rows of data.

ID_CURSANT	NUMAR_LECTII
1	5200
2	713
3	4821
4	6521
5	6432
6	4268
7	3220
8	987
9	1482
10	1345
11	520
12	3152
13	1723
14	3473
15	2360

Să se afișeze id-ul vehiculelor ce au kilometrajul mediu înregistrat mai mare decăt 200000.

```
SELECT id_vehicul, ROUND(AVG(kilometraj),1) medie_kilometri  
FROM VEHICULE  
GROUP BY id_vehicul  
HAVING ROUND(AVG(kilometraj),1)>200000;
```

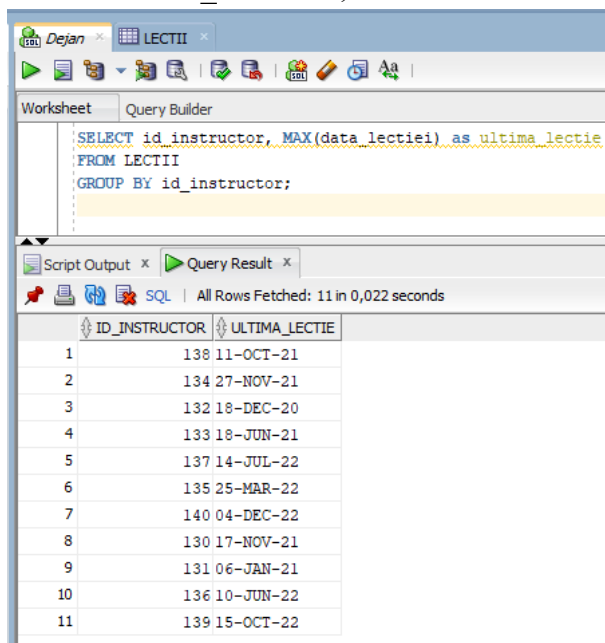


The screenshot shows the SQL Developer interface with a query window titled 'Dejan' and 'VEHICULE'. The query is: `SELECT id_vehicul, ROUND(AVG(kilometraj),1) medie_kilometri FROM VEHICULE GROUP BY id_vehicul HAVING ROUND(AVG(kilometraj),1)>200000;`. The 'Query Result' tab shows 3 rows fetched in 0.01 seconds. The results are as follows:

	ID_VEHICUL	MEDIE_KILOMETRI
1	578	274357
2	783	254328
3	540	204534

Să se afișeze data ultimei lecții predate de fiecare instructor.

```
SELECT id_instructor, MAX(data_lectiei) as ultima_lectie  
FROM LECTII  
GROUP BY id_instructor;
```



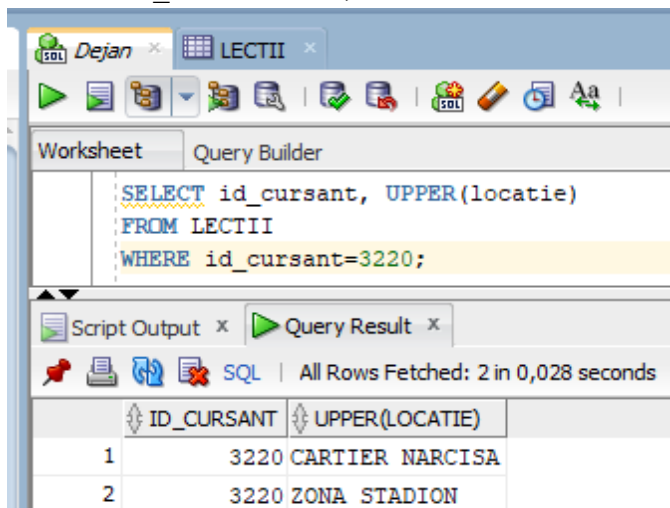
The screenshot shows the SQL Developer interface with a query window titled 'Dejan' and 'LECTII'. The query is: `SELECT id_instructor, MAX(data_lectiei) as ultima_lectie FROM LECTII GROUP BY id_instructor;`. The 'Query Result' tab shows 11 rows fetched in 0.022 seconds. The results are as follows:

	ID_INSTRUCTOR	ULTIMA_LECTIE
1	138	11-OCT-21
2	134	27-NOV-21
3	132	18-DEC-20
4	133	18-JUN-21
5	137	14-JUL-22
6	135	25-MAR-22
7	140	04-DEC-22
8	130	17-NOV-21
9	131	06-JAN-21
10	136	10-JUN-22
11	139	15-OCT-22

Funcții single-row

Să se afișeze cu litere mari locațiile unde și-a susținut lecțiile cursantul cu id-ul 3220.

```
SELECT id_cursant, UPPER(locatie)
FROM LECTII
WHERE id_cursant=3220;
```

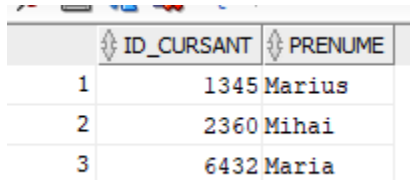


The screenshot shows a SQL query editor with a toolbar at the top. The query is entered in the main text area and is highlighted in yellow. Below the query, the 'Query Result' tab is active, displaying the results of the query in a table format. The table has two columns: 'ID_CURSANT' and 'UPPER(LOCATIE)'. There are two rows of data.

	ID_CURSANT	UPPER(LOCATIE)
1	3220	CARTIER NARCISA
2	3220	ZONA STADION

Să se afișeze toți cursanții al căror prenume începe cu litera m.

```
SELECT id_cursant, prenume FROM CURSANTI WHERE prenume LIKE UPPER('m%');
```



The screenshot shows a table with two columns: 'ID_CURSANT' and 'PRENUME'. There are three rows of data.

	ID_CURSANT	PRENUME
1	1345	Marius
2	2360	Mihai
3	6432	Maria

Să se afișeze numele, prenumele cursantului și vârsta sa.

```
SELECT INITCAP(nume) || ' ' || INITCAP(prenume) || ' are ' || varsta || ' ani.'
FROM CURSANTI;
```

	INITCAP(NUME) " " INITCAP(PRENUME) "ARE" VARSTA "ANI."
1	Vlad Andreea are 20 ani.
2	Popa Marius are 27 ani.
3	Moneaga Tudor are 18 ani.
4	Toma Sabin are 34 ani.
5	Popescu Ion are 38 ani.
6	Voicu Mihai are 26 ani.
7	Coman Gabriela are 23 ani.
8	Mocanu Irina are 40 ani.
9	Vlad Bianca are 44 ani.
10	Aanitei Daria are 30 ani.
11	Cucu Andrei are 33 ani.
12	Stefan Maria are 44 ani.
13	Rosu Calin are 18 ani.
14	Preda Andi are 47 ani.
15	Bucur Victor are 41 ani.

Să se afișeze lecțiile derulate luna trecută.

SELECT id_lectie, data_lectiei FROM LECTII

WHERE round(MONTHS_BETWEEN(sysdate, data_lectiei))=1;

The screenshot shows the SQL Developer interface. The 'Query Builder' tab is active, displaying the following SQL query:

```
SELECT id_lectie, data_lectiei FROM LECTII
WHERE round(MONTHS_BETWEEN(sysdate, data_lectiei))=1;
```

Below the query editor, the 'Query Result' tab shows the results of the query. The status bar indicates 'All Rows Fetched: 1 in 0,01 seconds'.

ID_LLECTIE	DATA_LLECTIEI
1	104 04-DEC-22

Să se afișeze instructorii angajați în anii 2020 și 2021.

SELECT id_instructor, data_angajare FROM INSTRUCTORI

WHERE EXTRACT(YEAR FROM data_angajare) IN (2020, 2021);

Dejan x INSTRUCTORI x

Worksheet Query Builder

```
SELECT id_instructor, data_angajare FROM INSTRUCTORI
WHERE EXTRACT(YEAR FROM data_angajare) IN (2020, 2021);
```

Script Output x Query Result x

SQL | All Rows Fetched: 6 in 0,017 seconds

ID_INSTRUCTOR	DATA_ANGAJARE
1	130 21-SEP-20
2	131 14-OCT-20
3	132 18-OCT-20
4	133 13-MAR-21
5	134 21-OCT-21
6	135 03-NOV-21

Să se afișeze cursanții ce au început școala între 13.03.2021 și 13.12.2021.

SELECT id_cursant,nume,prenume,data_incepere

FROM CURSANTI

WHERE data_incepere BETWEEN to_date('13-03-2021', 'dd-mm-yyyy') AND to_date('13-12-2021', 'dd-mm-yyyy');

Dejan x INSTRUCTORI x

Worksheet Query Builder

```
SELECT id_cursant,nume,prenume,data_incepere
FROM CURSANTI
WHERE data_incepere BETWEEN to_date('13-03-2021', 'dd-mm-yyyy') AND to_date('13-12-2021', 'dd-mm-yyyy');
```

Script Output x Query Result x

SQL | All Rows Fetched: 4 in 0,013 seconds

ID_CURSANT	NUME	PRENUME	DATA_INCEPERE
1	1345 Popa	Marius	24-MAY-21
2	987 Coman	Gabriela	25-NOV-21
3	4268 Vlad	Bianca	01-SEP-21
4	1723 Bucur	Victor	18-NOV-21

Expresia CASE

Să se afișeze modelul vehiculelor după repartizarea:

-în categoria “sub 100000 km”

-în categoria “între 100000km și 200000km”

-în categoria peste „200000km”

```

SELECT model,
CASE
    WHEN kilometraj<100000 THEN ' Sub 100000 km'
    WHEN kilometraj BETWEEN 100000 AND 200000 THEN ' Intre 100000 km si 200000 km'
    WHEN kilometraj >200000 THEN ' Peste 200000 km'
END AS repartizare_kilometraj
FROM VEHICULE;

```

	MODEL	REPARTIZARE_KILOMETRAJ
1	BMW X3	Intre 100000 km si 200000 km
2	VW GOLF 6	Peste 200000 km
3	RENAULT MEGANE	Sub 100000 km
4	VW GOLF 7	Sub 100000 km
5	VW PASSAT B7	Sub 100000 km
6	MAZDA CX3	Peste 200000 km
7	RENAULT CLIO 4	Intre 100000 km si 200000 km
8	CITROEN C3	Intre 100000 km si 200000 km
9	PEUGEOT 308	Intre 100000 km si 200000 km
10	VW GOLF 6	Peste 200000 km
11	SKODA OCTAVIA	Intre 100000 km si 200000 km
12	SKODA FABIA	Intre 100000 km si 200000 km

Operatorul MINUS

Să se afișeze cursanții ce au început școala în anul 2022 fără cei care au început școala în lunile iunie și iulie din același an.

```

SELECT id_cursant, nume, prenume
FROM CURSANTI
WHERE EXTRACT(YEAR FROM data_incepere) = 2022
MINUS
SELECT id_cursant, nume, prenume
FROM CURSANTI
WHERE EXTRACT(MONTH FROM data_incepere) IN ( 6, 7);

```

The screenshot shows a SQL query editor window with a toolbar and a query builder tab. The query is as follows:

```
SELECT id_cursant, nume, prenume
FROM CURSANTI
WHERE EXTRACT(YEAR FROM data_incepere) = 2022
MINUS
SELECT id_cursant, nume, prenume
FROM CURSANTI
WHERE EXTRACT(MONTH FROM data_incepere) IN ( 6, 7 );
```

Below the query editor, the 'Query Result' tab is active, showing the results of the query. The results are displayed in a table with 4 rows and 4 columns: ID_CURSANT, NUME, PRENUME, and an implicit row number.

	ID_CURSANT	NUME	PRENUME
1	520	Preda	Andi
2	713	Rosu	Calin
3	5200	Cucu	Andrei
4	6432	Stefan	Maria

Operatorul UNION

Să se afișeze numele și prenumele cursanților care au terminat școala în 2021 împreună cu cei care au terminat în 2022.

```
SELECT nume,prenume FROM CURSANTI
```

```
WHERE data_finalizare BETWEEN to_date('01-01-2021', 'dd-mm-yyyy') AND to_date('31-12-2021', 'dd-mm-yyyy')
```

```
UNION
```

```
SELECT nume,prenume FROM CURSANTI
```

```
WHERE data_finalizare BETWEEN to_date('01-01-2022', 'dd-mm-yyyy') AND to_date('31-12-2022', 'dd-mm-yyyy');
```

Dejan x LECTII x

Worksheet Query Builder

```

SELECT nume,prenume FROM CURSANTI
WHERE data_finalizare BETWEEN to_date('01-01-2021', 'dd-mm-yyyy') AND to_date('31-12-2021', 'dd-mm-yyyy')
UNION
SELECT nume,prenume FROM CURSANTI
WHERE data_finalizare BETWEEN to_date('01-01-2022', 'dd-mm-yyyy') AND to_date('31-12-2022', 'dd-mm-yyyy');

```

Script Output x Query Result x

SQL | All Rows Fetched: 13 in 0,011 seconds

	NUME	PRENUME
1	Aanitei	Daria
2	Bucur	Victor
3	Coman	Gabriela
4	Mocanu	Irina
5	Moneaga	Tudor
6	Popa	Marius
7	Popescu	Ion
8	Preda	Andi
9	Rosu	Calin
10	Stefan	Maria
11	Toma	Sabin
12	Vlad	Bianca
13	Voicu	Mihai

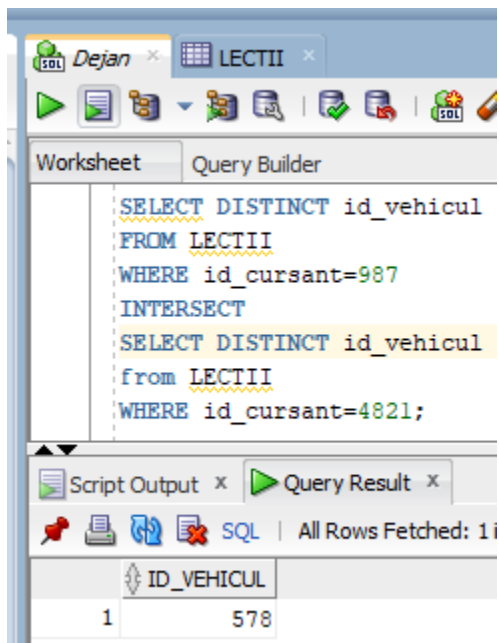
Operatorul INTERSECT

Să se afișeze id-ul vehiculului condus atât de cursantul cu id-ul 987, cât și de cursantul cu id-ul 4821.

```

SELECT DISTINCT id_vehicul
FROM LECTII
WHERE id_cursant=987
INTERSECT
SELECT DISTINCT id_vehicul
from LECTII
WHERE id_cursant=4821;

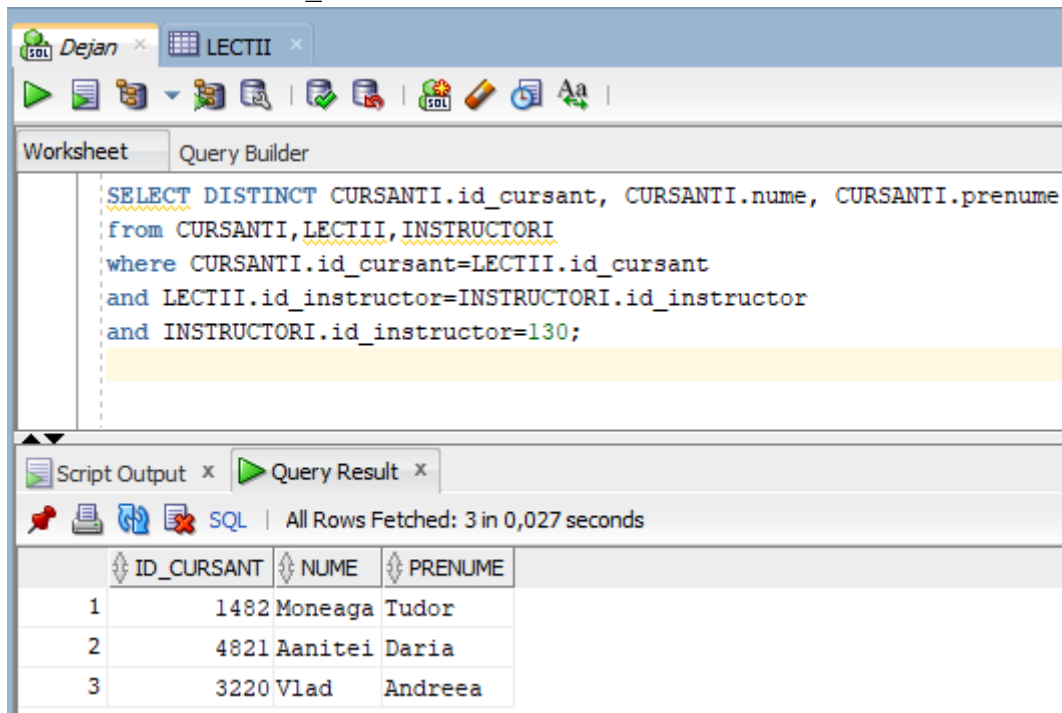
```



Joncțiuni de egalitate

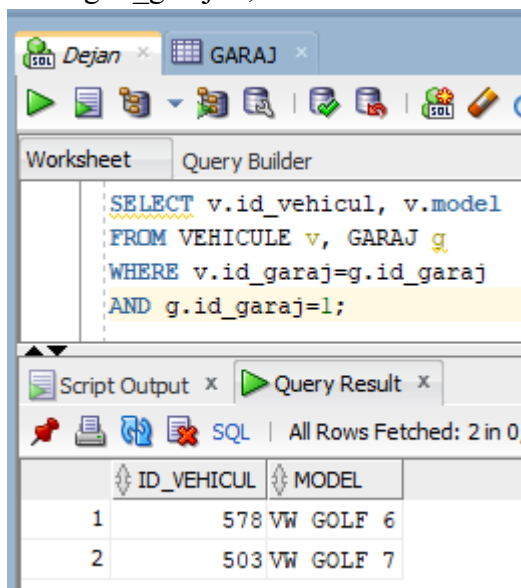
Să se afișeze cursanții care l-au avut instructor pe instructorul cu id-ul 130.

```
SELECT DISTINCT CURSANTI.id_cursant, CURSANTI.numa, CURSANTI.prenume
from CURSANTI,LECTII,INSTRUCTORI
where CURSANTI.id_cursant=LECTII.id_cursant
and LECTII.id_instructor=INSTRUCTORI.id_instructor
and INSTRUCTORI.id_instructor=130;
```



Să se afișeze id-ul și modelul vehiculelor ce aparțin garajului cu id-ul 1.

```
SELECT v.id_vehicul, v.model  
FROM VEHICULE v, GARAJ g  
WHERE v.id_garaj=g.id_garaj  
AND g.id_garaj=1;
```



The screenshot shows the SQL Dejan interface. The top bar has tabs for 'Dejan' and 'GARAJ'. Below the toolbar, the 'Query Builder' tab is active, displaying the following SQL query:

```
SELECT v.id_vehicul, v.model  
FROM VEHICULE v, GARAJ g  
WHERE v.id_garaj=g.id_garaj  
AND g.id_garaj=1;
```

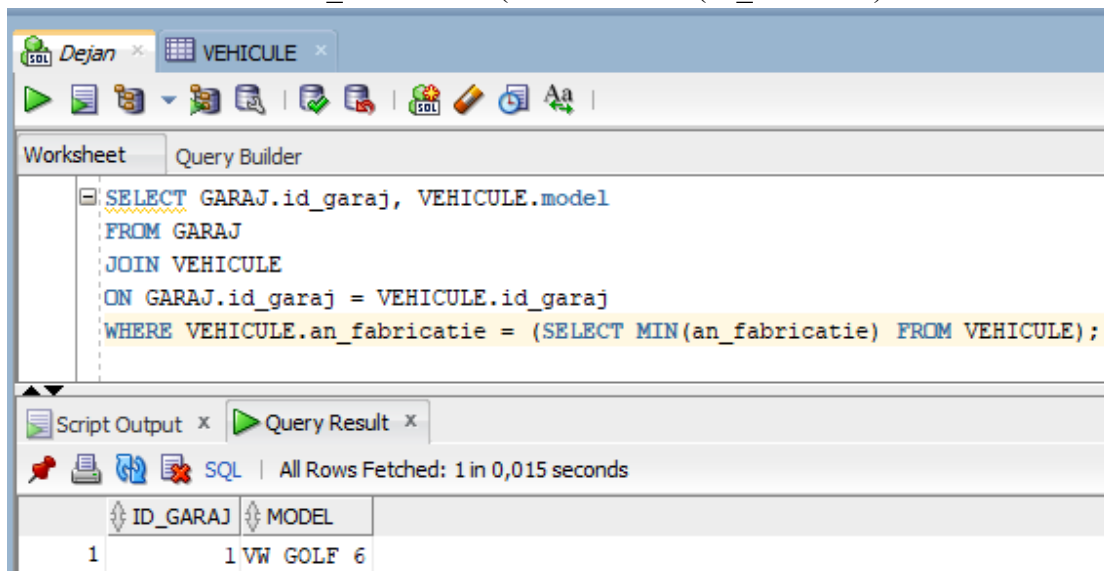
Below the query editor, the 'Query Result' tab is active, showing the results of the query. The status bar indicates 'All Rows Fetched: 2 in 0.001 seconds'.

ID_VEHICUL	MODEL
1	578 VW GOLF 6
2	503 VW GOLF 7

Cerere imbricată

Să se afișeze garajul din care face parte cel mai vechi vehicul cât și modelul acestuia.

```
SELECT GARAJ.id_garaj, VEHICULE.model  
FROM GARAJ  
JOIN VEHICULE  
ON GARAJ.id_garaj = VEHICULE.id_garaj  
WHERE VEHICULE.an_fabricatie = (SELECT MIN(an_fabricatie) FROM VEHICULE);
```



The screenshot shows the SQL Dejan interface. The top bar has tabs for 'Dejan' and 'VEHICULE'. Below the toolbar, the 'Query Builder' tab is active, displaying the following SQL query:

```
SELECT GARAJ.id_garaj, VEHICULE.model  
FROM GARAJ  
JOIN VEHICULE  
ON GARAJ.id_garaj = VEHICULE.id_garaj  
WHERE VEHICULE.an_fabricatie = (SELECT MIN(an_fabricatie) FROM VEHICULE);
```

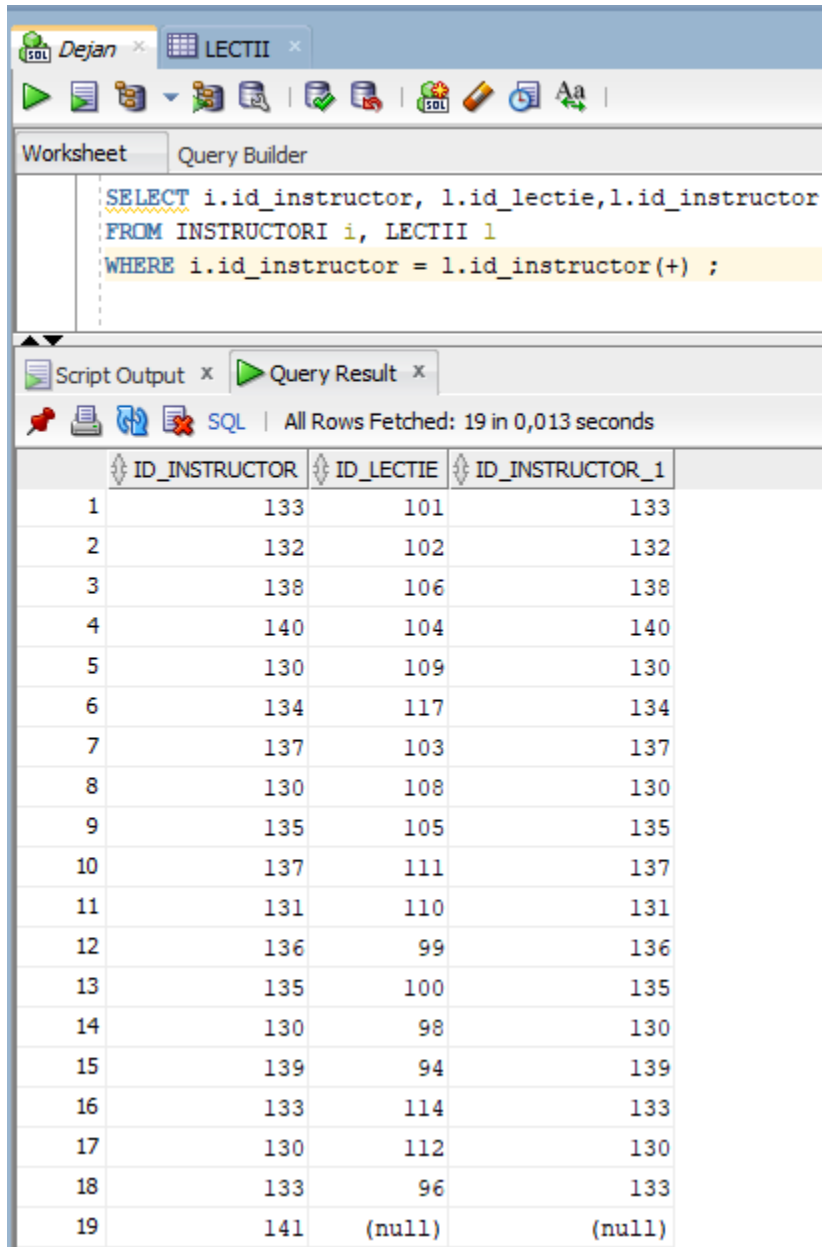
Below the query editor, the 'Query Result' tab is active, showing the results of the query. The status bar indicates 'All Rows Fetched: 1 in 0,015 seconds'.

ID_GARAJ	MODEL
1	1 VW GOLF 6

Joncțiune externă

Să se afișeze instructorii care au susținut cel puțin o lecție cât și cei care nu au susținut nicio lecție, împreună cu id-ul lecției respective.

```
SELECT i.id_instructor, l.id_lectie, l.id_instructor
FROM INSTRUCTORI i, LECTII l
WHERE i.id_instructor = l.id_instructor(+);
```



The screenshot shows a SQL IDE window with a query editor and a results pane. The query editor contains the following SQL code:

```
SELECT i.id_instructor, l.id_lectie, l.id_instructor
FROM INSTRUCTORI i, LECTII l
WHERE i.id_instructor = l.id_instructor(+);
```

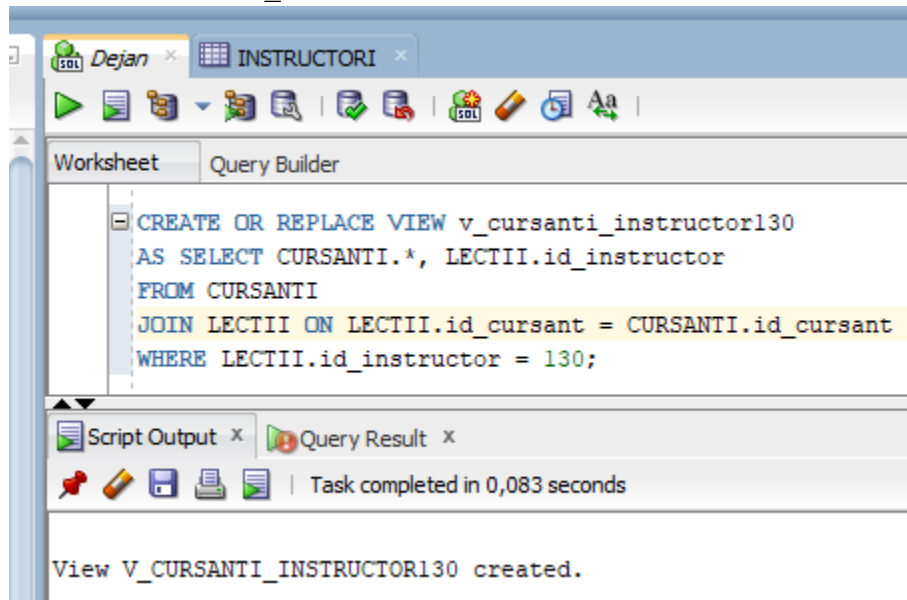
The results pane shows the output of the query, which is a table with 19 rows and 4 columns. The columns are labeled ID_INSTRUCTOR, ID_LECTIE, and ID_INSTRUCTOR_1. The first three columns are highlighted in the first row. The data is as follows:

	ID_INSTRUCTOR	ID_LECTIE	ID_INSTRUCTOR_1
1	133	101	133
2	132	102	132
3	138	106	138
4	140	104	140
5	130	109	130
6	134	117	134
7	137	103	137
8	130	108	130
9	135	105	135
10	137	111	137
11	131	110	131
12	136	99	136
13	135	100	135
14	130	98	130
15	139	94	139
16	133	114	133
17	130	112	130
18	133	96	133
19	141	(null)	(null)

Vederi, indecsi, sinonime, secvente

Creați o tabelă virtuală ce va conține toți cursanții ce au avut lecții cu instructorul cu id-ul 130.

```
CREATE OR REPLACE VIEW v_cursanti_instructor130
AS SELECT DISTINCT CURSANTI.*, LECTII.id_instructor
FROM CURSANTI
JOIN LECTII ON LECTII.id_cursant = CURSANTI.id_cursant
WHERE LECTII.id_instructor = 130;
```



<

Vizualizarea informațiilor despre tabela virtuala creată

```
SELECT VIEW_NAME, TEXT FROM USER_VIEWS;
```

The screenshot shows the Dejan SQL editor interface. The main window displays the SQL script for querying the view:

```
SELECT VIEW_NAME, TEXT FROM USER_VIEWS;
```

Below the script, the 'Script Output' pane shows the message: 'All Rows Fetched: 1 in 0,02 seconds'. The 'Query Result' pane displays the following table:

VIEW_NAME	TEXT
V_CURSANTI_INSTRUCTOR130	SELECT DISTINCT CURSANTI."ID_CURSANT",CURSANTI."NUME",CURSANTI."PRENUME",CURSANTI."TELEFON",CURSANTI."EMAIL",CURSANTI."VA

Să se șteargă tabela virtuala creată.

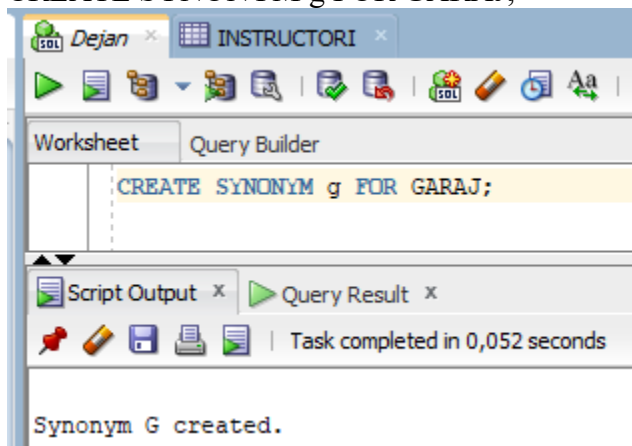
DROP VIEW v_cursanti_instructor130;

View V_CURSANTII_INSTRUCTOR130 dropped.

SINONIME

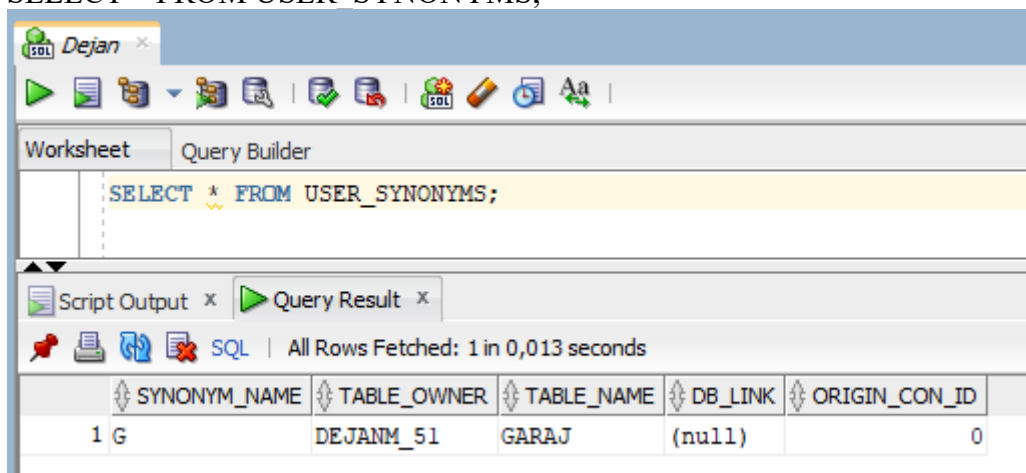
Creați un sinonim pentru tabela GARAJ

CREATE SYNONYM g FOR GARAJ;



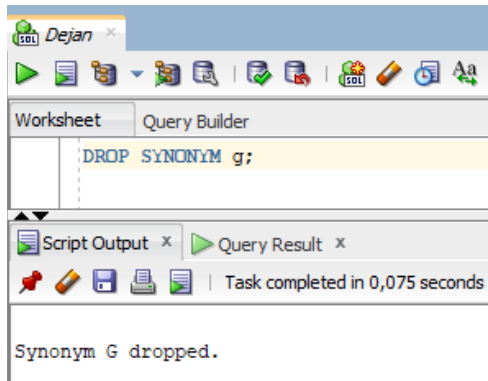
Să se vizualizeze sinonimul.

SELECT * FROM USER_SYNONYMS;



Sa se stearga sinonimul creat anterior:

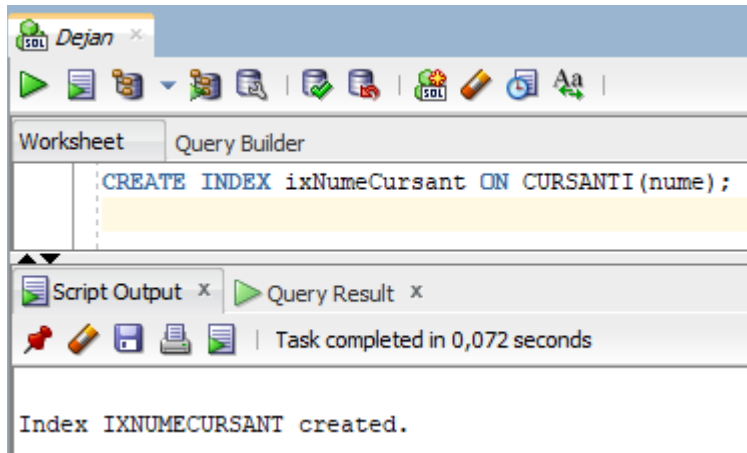
DROP SYNONYM g;



INDECSI

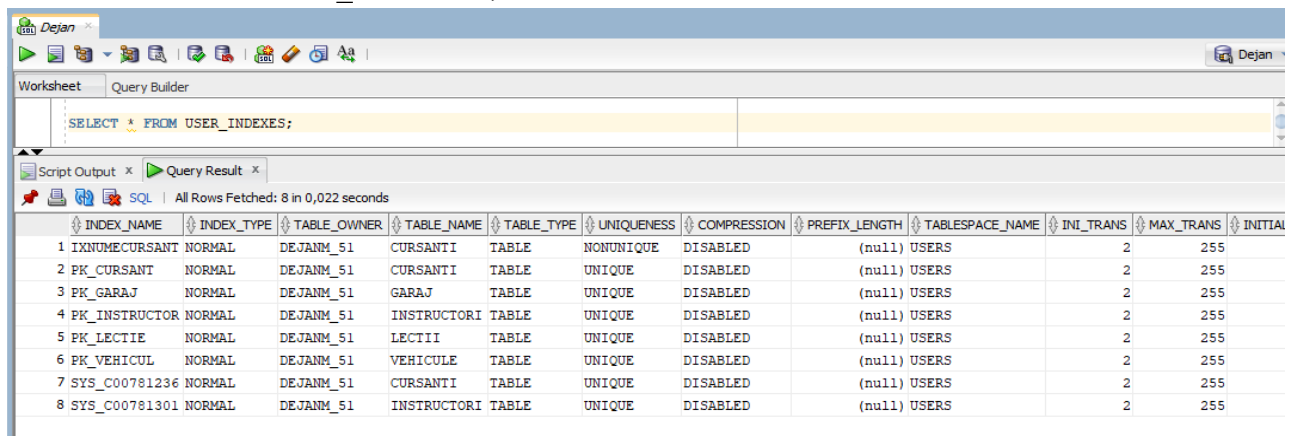
Să se creeze un index pe tabela CURSANTI pe coloana nume.

CREATE INDEX ixNumCursant ON CURSANTI(nume);



Să se vizualizeze toti indecsii utilizatorului.

SELECT * FROM USER_INDEXES;



Să se șteargă indexul creat mai sus.

DROP INDEX ixNumCursant

