

# TEMĂ

În această temă voi prezenta două exemple de utilizare al tipului de date **LOB** (mai precis **BLOB** și **BFILE**) și voi testa *select rowid, employee\_id from employees*.

## Exemplu pentru BLOB.

Vom creea tabelul bucătar care conține pentru fiecare bucătar un id (cheie primară care este *not null* și *unique bydefault*), numele, prenumele și țara acestuia din care provine, precum și o poză.

Pentru coloana care va conține poza vom folosi tipul de date **LOB**, mai precis **BLOB**.

Ce este **BLOB**? Este un internal **LOB**, spre deosebire de **BFILE** care este un external **LOB**.

Ce putem stoca cu ajutorul acestui tip de date? Fișiere multimedia cum ar fi imagine (JPG, PNG etc.), videoclipuri (MP4, MP3 etc.)

-- BUCATAR --

```
CREATE TABLE bucatar (
    id_bucatar NUMBER(10) PRIMARY KEY,
    nume      VARCHAR2(20),
    prenume   VARCHAR2(20),
    tara      VARCHAR2(15),
    poza      BLOB
);
```

```
commit;
```

## ***Print-Screen:***

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays the 'Regions' tree, which includes 'Connections' (selected), 'Tables Filtered' (containing 'BUCATAR'), 'Views', 'Indexes', 'Packages', 'Triggers', 'Sequences', 'Materialized Views', 'Materialized View Logs', 'Synonyms', 'Public Synonyms', 'Database Links', 'Public Database Links', 'Directories', 'Regions', 'Java', 'XML Schemas', 'Text', 'Text Analytics', 'OLAP Options', 'Analytic Views', 'RDF Semantic Graph', 'Recycle Bin', and 'Other Users'. The main workspace is a 'Worksheet' titled 'projectz' with a 'Query Builder' tab. The query editor contains the following SQL code:

```
1 -- BUCATAR --
2 CREATE TABLE bucatar (
3     id_bucatar    NUMBER(10) PRIMARY KEY,
4     nume          VARCHAR2(20),
5     prenume       VARCHAR2(20),
6     tara          VARCHAR2(15),
7     poza          BLOB
8 );
9
10 commit;
```

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

```
Table BUCATAR created.

Commit complete.
```

A red circle highlights the message 'Table BUCATAR created.' in the output window.

Observăm că tabelul nostru s-a creat cu succes, salvăm modificările aduse la nivel de sesiune cu un **commit**; (închidem o tranzacție). Acum vom insera datele în tabel.

```
insert into bucatar values (1, 'Dumitrescu', 'Florin', 'Romania', empty_blob());
```

```
insert into bucatar values (2, 'Scarlatescu', 'Catalin', 'Romania', empty_blob());
```

```
insert into bucatar values (3, 'Bontea', 'Sorin', 'Romania', empty_blob());
```

```
insert into bucatar values (4, 'Jamie', 'Oliver', 'Marea Britanie', empty_blob());
```

```
insert into bucatar values (5, 'Gordon', 'James', 'Italia', empty_blob());
```

## **Print-Screen:**

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays a tree view of database objects under the 'Connections' node, including 'Tables' and 'Views'. The main workspace contains a 'Worksheet' tab with the following SQL script:

```
11
12 insert into bucatar values (1, 'Dumitrescu', 'Florin', 'Romania', empty_blob());
13 insert into bucatar values (2, 'Scarlatescu', 'Catalin', 'Romania', empty_blob());
14 insert into bucatar values (3, 'Bontea', 'Sorin', 'Romania', empty_blob());
15 insert into bucatar values (4, 'Jamie', 'Oliver', 'Marea Britanie', empty_blob());
16 insert into bucatar values (5, 'Gordon', 'James', 'Italia', empty_blob());
17
18 select * from bucatar;
```

Below the worksheet, the 'Script Output' tab shows the results of the execution:

```
1 row inserted.
```

A red circle highlights this output. Below it, another line of output is shown:

```
1 row inserted.
```

The bottom of the screen features a toolbar with various icons and a status bar indicating 'Task completed in 0.023 seconds'.

Datele au fost inserate. Acum verificăm printr-un *select* dacă totul s-a realizat cu succes.

**Print-Screen:**

The screenshot shows the Oracle SQL Developer interface. On the left, the Connections pane shows a connection to 'lab'. The central area has a 'Worksheet' tab open with the following SQL code:

```
13 insert into bucatar values (2, 'Scarlatescu', 'Catalin', 'Romania', empty_blob());
14 insert into bucatar values (3, 'Bontea', 'Sorin', 'Romania', empty_blob());
15 insert into bucatar values (4, 'Jamie', 'Oliver', 'Marea Britanie', empty_blob());
16 insert into bucatar values (5, 'Gordon', 'James', 'Italia', empty_blob());
17
18 commit;
19
20 select * from bucatar;
```

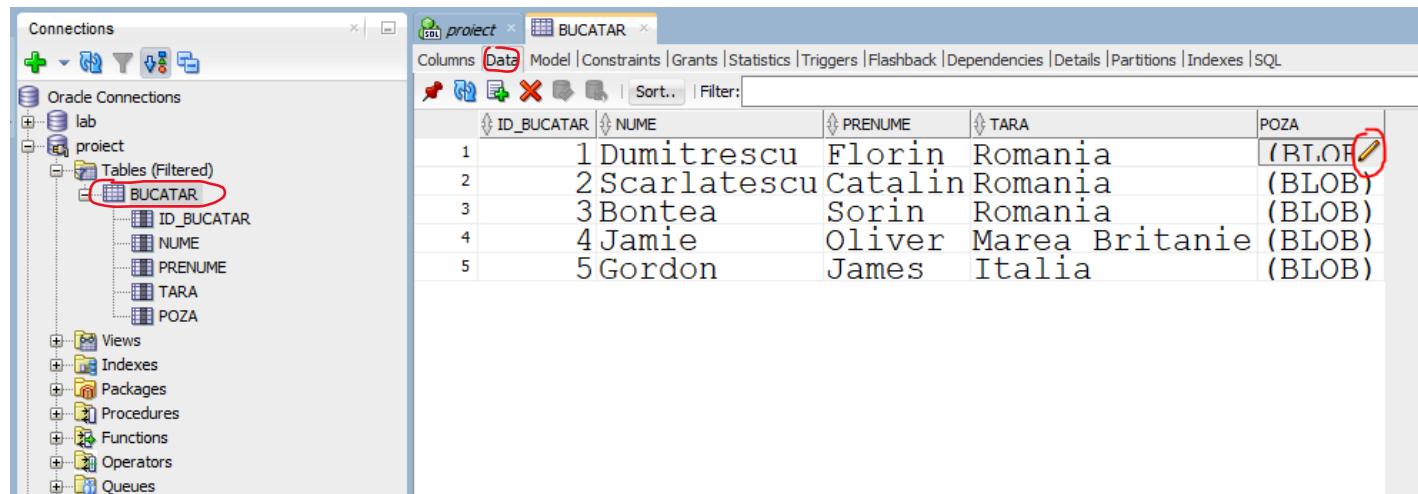
Below the code, the 'Query Result' tab displays the results of the last query:

ID_BUCATAR	NUME	PRENUME	TARA	POZA
1	Dumitrescu	Florin	Romania	(BLOB)
2	Scarlatescu	Catalin	Romania	(BLOB)
3	Bontea	Sorin	Romania	(BLOB)
4	Jamie	Oliver	Marea Britanie	(BLOB)
5	Gordon	James	Italia	(BLOB)

A red circle highlights the last row of the result set. The status bar at the bottom right indicates 'Line 20 Column 23 | Insert | Modified | Windows: 0'.

Observăm că totul a fost inserat aşa cum am dorit. Acum vom inseră pentru fiecare bucătar poza acestuia.

Cum vom realiza acest lucru? Vom deschide tabelul **BUCATAR**, după aceea vom da click pe **Data**, iar după aceea dublu click pe **(BLOB)**, observăm că apare un creionăş cu care putem edita, și dăm click pe el.



The screenshot shows the Oracle SQL Developer interface. On the left, the Connections tree is visible. In the center, a table named 'BUCATAR' is displayed with the following data:

ID_BUCATAR	NUME	PRENUME	TARA
1	Dumitrescu	Florin	Romania
2	Scarlatescu	Catalin	Romania
3	Bontea	Sorin	Romania
4	Jamie	Oliver	Marea Britanie
5	Gordon	James	Italia

An 'Edit Value' dialog is open for the first row. The 'Saved Data' section contains the value '1Dumitrescu Florin Romania'. The 'Local Data' section has a 'File Name' field with the value '1NULL'. A red circle highlights the 'Load' button at the bottom right of the dialog.

Observăm că ni s-a deschis o fereastră, unde putem edita valoare pentru coloana respectiva, de pe linia respectivă. Pentru a încărca o poză pentru bucătarul respectiv facem click pe **Load**.

The screenshot shows the 'Edit Value' dialog with the 'Local Data' section active. A file browser window is overlaid on the dialog, showing a list of files in the directory 'C:\Users\rober\Desktop\C++\SGBDfiles'. The file 'dumitrescu.jpg' is selected. A red circle highlights the 'Open' button at the bottom right of the file browser window.

Acum căutăm locația unde se află poza respectivă, după ce am găsit-o facem click pe ea pentru a selecta-o, iar după aceea facem click pe **Open**.

După ce s-a închis fereastra *Edit Value*, facem din nou dublu click pe (**BLOB**), după aceea facem click pe acel creionăș, iar după aceea bifăm căsuța **Image**.

The screenshot shows the Oracle SQL Developer interface. On the left, the Object Navigator displays a table named 'BUATAR'. In the main pane, a grid shows data for columns ID\_BUATAR, NUME, PRENUME, and TARA. The fourth row's TARA column value is highlighted with a red oval. A secondary window titled 'Edit Value' is open over the grid. It has tabs for 'Information' and 'Local Data'. Under 'Information', the 'Image' checkbox is checked. Under 'Local Data', there is a file named 'dumitrescu.jpg' located at 'C:/Users/jober/Desktop/C++/SGBD/files/dumitrescu.jpg' with a size of 54642 bytes. The 'Image' checkbox is also checked here. A red oval highlights this checkbox.

This screenshot shows the same Oracle SQL Developer environment as the previous one. The 'Edit Value' dialog is still open, and the 'Image' checkbox is checked. The preview area of the dialog shows a portrait of a man with dark hair, glasses, and a beard, wearing a white shirt. A red oval highlights the 'Image' checkbox in the dialog. The background shows the same table structure and data as the first screenshot.

Observăm că poza noastră a fost inserată cu succes. În același modalitate vom proceda și pentru ceilalți bucătari. Mai jos puteți vedea pozele cu aceștia după ce au fost inserate.

Screenshot of Oracle SQL Developer showing the successful insertion of a photo into the BUCATAR table. The table contains five rows of data, and a new row with ID 6 and NAME 'Sorin' is being inserted. A preview window shows a photo of a man with a beard giving two thumbs up.

ID_BUCATAR	NUME	PRENUME	TARA
1	Dumitrescu	Florin	Romania
2	Scarlatescu	Catalin	Romania
3	Bontea	Sorin	Romania
4	Jamie	Oliver	Mareș Britan
5	Gordon	James	Italia
6	Sorin		

Screenshot of Oracle SQL Developer showing the successful insertion of another photo into the BUCATAR table. The table now has six rows, including the previously inserted row for Sorin. A preview window shows a photo of a man with a mustache and a tattoo on his neck, wearing a dark chef's coat with 'Chef Sorin Bontea' written on it.

ID_BUCATAR	NUME	PRENUME	TARA
1	Dumitrescu	Florin	Romania
2	Scarlatescu	Catalin	Romania
3	Bontea	Sorin	Romania
4	Jamie	Oliver	Mareș Britan
5	Gordon	James	Italia
6	Sorin		

Oracle SQL Developer : Table ROBERTO.BUCATAR@project

File Edit View Navigate Run Team Tools Window Help

Connections

- + lab
  - + project
    - + Tables (Filtered)
      - BUCATAR
        - + ID\_BUCATAR
        - + NAME
        - + PRENUME
        - + TARA
        - + POZA

BUATAR

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

ID_BUCATAR	NAME	PRENUME	TARA
#1	Dumitrescu	Florin	Romania
#2	Scarlatescu	Catalin	Romania
#3	Bontea	Sorin	Romania
#4	Jamie	Oliver	Marea Britanie
5	Gordon	James	Italia

Information Local Image

Edit Value

OK Cancel

Items Output Buffer Size:20000

Oracle SQL Developer : Table ROBERTO.BUCATAR@project

File Edit View Navigate Run Team Tools Window Help

Connections

- + lab
  - + project
    - + Tables (Filtered)
      - BUCATAR
        - + ID\_BUCATAR
        - + NAME
        - + PRENUME
        - + TARA
        - + POZA

BUATAR

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

ID_BUCATAR	NAME	PRENUME	TARA
#1	Dumitrescu	Florin	Romania
#2	Scarlatescu	Catalin	Romania
#3	Bontea	Sorin	Romania
#4	Jamie	Oliver	Marea Britanie
5	Gordon	James	Italia

Information Local Image

Edit Value

OK Cancel

Items Output Buffer Size:20000

Faptul că am adăugat poze pentru fiecare bucătar, reprezintă o tranzacție în curs de desfășurare până în momentul în care se va efectua un **commit**; (salvăm modificările aduse) sau **rollback**; (abandonăm modificările aduse).

Acum pentru a salva modificările, vom închide tabelul bucătar, după care facem click pe **Yes (commit;)**. Dacă am fi făcut click pe **No (rollback;)** atunci modificările nu s-ar fi salvat.

The screenshot shows the Oracle SQL Developer interface. On the left is the Object Navigator pane, which lists various database objects like Tables, Packages, Functions, Operators, and Types. In the center, a table named 'BUCATAR' is displayed with columns: ID\_BUCATAR, NUME, PRENUME, TARA, and POZA. The table contains 5 rows of data. Above the table, a 'Save' dialog box is open, asking 'Table ROBERTO.BUCATAR@project has been modified. Save changes?'. The 'Yes' button is highlighted with a red circle. At the bottom of the screen, a 'Messages' window is open, showing a log of SQL statements executed, followed by the message 'Commit Successful'.

ID_BUCATAR	NUME	PRENUME	TARA	POZA
1	Dumitrescu	Florin	Romania	(BLOB)
2	Scarlatescu	Catalin	Romania	(BLOB)
3	Bontea	Sorin	Romania	(BLOB)
4	Jamie	Oliver	Marea Britanie	(BLOB)
5	Gordon	James	Italia	(BLOB)

Save  
Table ROBERTO.BUCATAR@project has been modified. Save changes?  
Yes No Cancel

Messages Log

```
UPDATE "ROBERTO"."BUCATAR" SET WHERE ROWID = 'AAASQ7AHAAAAAH1AAA' AND ORA_ROWSCN = '6422974'
UPDATE "ROBERTO"."BUCATAR" SET POZA=? WHERE ROWID=:sqldevrvid AND ORA_ROWSCN=:sqldevrowscn
UPDATE "ROBERTO"."BUCATAR" SET WHERE ROWID = 'AAASQ7AHAAAAAH1AB' AND ORA_ROWSCN = '6422974'
UPDATE "ROBERTO"."BUCATAR" SET POZA=? WHERE ROWID=:sqldevrvid AND ORA_ROWSCN=:sqldevrowscn
UPDATE "ROBERTO"."BUCATAR" SET WHERE ROWID = 'AAASQ7AHAAAAAH1AC' AND ORA_ROWSCN = '6422974'
UPDATE "ROBERTO"."BUCATAR" SET POZA=? WHERE ROWID=:sqldevrvid AND ORA_ROWSCN=:sqldevrowscn
UPDATE "ROBERTO"."BUCATAR" SET WHERE ROWID = 'AAASQ7AHAAAAAH1AD' AND ORA_ROWSCN = '6422974'
UPDATE "ROBERTO"."BUCATAR" SET POZA=? WHERE ROWID=:sqldevrvid AND ORA_ROWSCN=:sqldevrowscn
UPDATE "ROBERTO"."BUCATAR" SET WHERE ROWID = 'AAASQ7AHAAAAAH1AE' AND ORA_ROWSCN = '6422974'
UPDATE "ROBERTO"."BUCATAR" SET POZA=? WHERE ROWID=:sqldevrvid AND ORA_ROWSCN=:sqldevrowscn
```

Commit Successful

Messages Logging Page Statements

Aici putem observa că s-a efectuat un **commit**; cu succes, deci pozele noastre au fost salvate!

## Exemplu pentru BFILE.

Pentru a folosi BFILE, trebuie să facem un director, pentru a lua locație fișierelor pe care le vom folosi.

```
create directory locatie_fisiere as 'C:\Users\rober\Desktop\C++\SGBDfiles';
```

### Print-Screen:

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Objects' tree view is open, showing a project named 'RESTAURANT' which contains tables like 'BUCATAR' and 'RESTAURANT'. In the center, the 'Worksheet' tab is active, displaying the following SQL code:

```
20 select * from bucatar;
21
22 create directory locatie_fisiere as 'C:\Users\rober\Desktop\C++\SGBDfiles';
23
24 -- RESTAURANT --
25 CREATE TABLE restaurant (
26     id_restaurant      NUMBER(10) PRIMARY KEY,
27     nume_restaurant    VARCHAR2(20),
28     poza                BFILE
29 );
30
31 commit;
32
```

Below the code, the message 'Directory LOCATIE\_FISIERE created.' is displayed. At the bottom, the 'Doms Output' tab is visible.

Vom crea tabelul `restaurant` care conține pentru fiecare restaurant un id (cheie primară care este *not null și unique bydefault*), numele acestuia, respectiv o poză. După care vom insera datele în el.

Pentru coloana care va conține poza vom folosi tipul de date ***LOB***, mai precis **BFILE**.

Ce este **BFILE**? Este un external ***LOB***, spre deosebire de **BLOB** care este un internal ***LOB***. Aceasta pointează către o locație specificată de noi (*director*).

Ce putem stoca cu ajutorul acestui tip de date? Fișiere multimedia cum ar fi imagine (JPG, PNG etc.), videoclipuri (MP4, MP3 etc.) la fel ca **BLOB**.

Ce dezavantaje are **BFILE** față de **BLOB**? **BFILE** are securitatea mai slabă deoarece stochează datele în afara bazei de date.

```
-- RESTAURANT --
```

```
CREATE TABLE restaurant (
    id_restaurant  NUMBER(10) PRIMARY KEY,
    nume_restaurant  VARCHAR2(20),
    poza          BFILE
);
```

```
commit;
```

```
insert into restaurant values (1, 'Gurmandul', BFILENAME('locatie_fisiere', 'restaurant1.jpg'));
```

```
insert into restaurant values (2, 'Yamas', BFILENAME('locatie_fisiere', 'restaurant2.jpg'));
```

```
insert into restaurant values (3, 'Ivans', BFILENAME('locatie_fisiere', 'restaurant3.jpg'));
```

```
insert into restaurant values (4, 'Savanna', BFILENAME('locatie_fisiere', 'restaurant4.jpg'));
```

```
insert into restaurant values (5, 'Grande appetito!', BFILENAME('locatie_fisiere',
'restaurant5.jpg'));
```

```
commit;
```

```
select * from restaurant;
```

## Print-Screen:

The screenshot shows the Oracle SQL Developer interface. In the Connections pane, a connection named 'project' is selected. In the central Worksheet tab, the following SQL script is being run:

```
23
24 -- RESTAURANT --
25 CREATE TABLE restaurant (
26     id_restaurant      NUMBER(10) PRIMARY KEY,
27     nume_restaurant    VARCHAR2(20),
28     poza               BFILE
29 );
30
31 commit;
32
33 insert into restaurant values (1, 'Gurmandul', BFILENAME('locatie_fisiere', 'restaurant1.jpg'));
34 insert into restaurant values (2, 'Yamas', BFILENAME('locatie_fisiere', 'restaurant2.jpg'));
35 insert into restaurant values (3, 'Ivans', BFILENAME('locatie_fisiere', 'restaurant3.jpg'));
```

The output pane shows the results of the execution:

```
Table RESTAURANT created.
Commit complete.
```

A red circle highlights the message "Table RESTAURANT created." and another red circle highlights the message "Commit complete."

The screenshot shows the Oracle SQL Developer interface. In the Connections pane, a connection named 'project' is selected. In the central Worksheet tab, the following SQL script is being run:

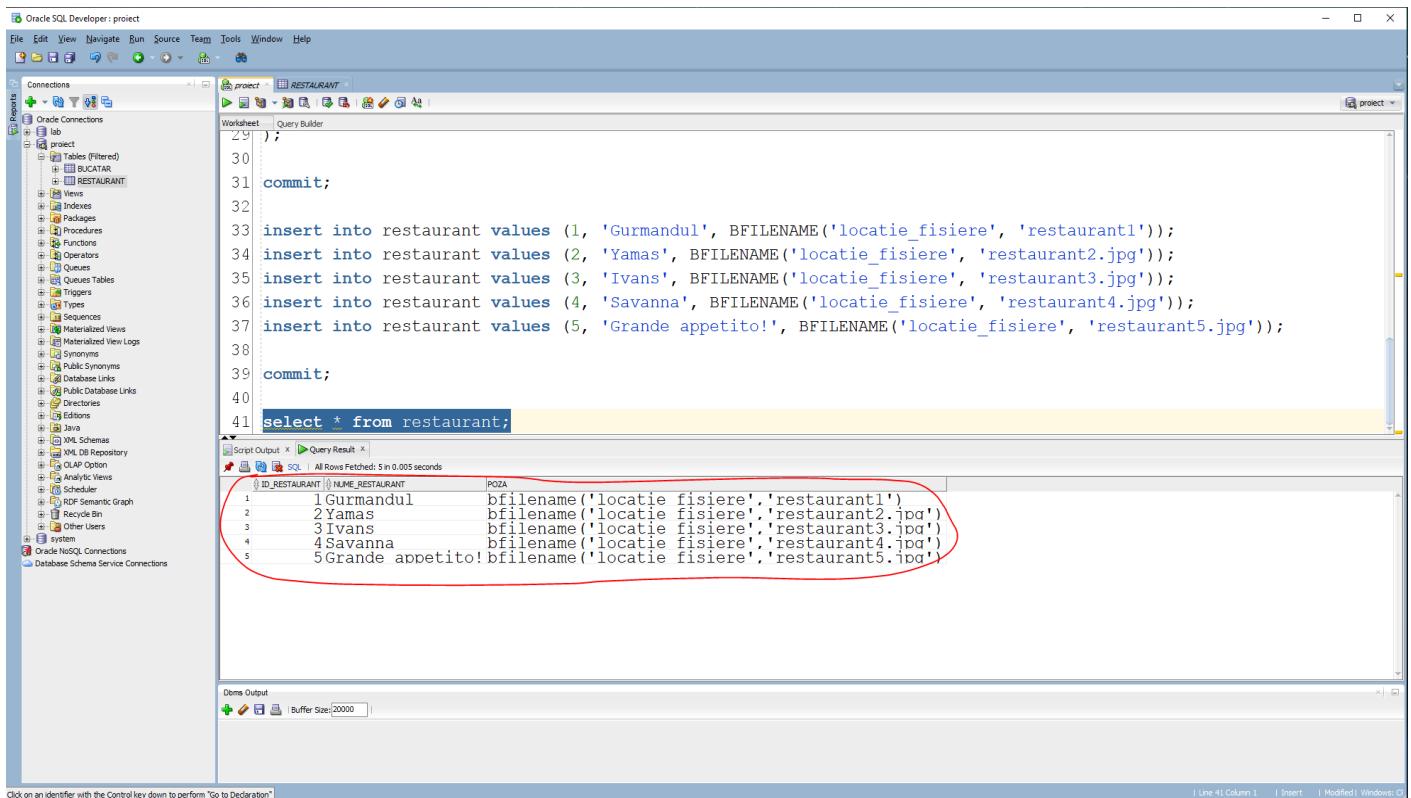
```
28     poza          BFILE
29 );
30
31 commit;
32
33 insert into restaurant values (1, 'Gurmandul', BFILENAME('locatie_fisiere', 'restaurant1.jpg'));
34 insert into restaurant values (2, 'Yamas', BFILENAME('locatie_fisiere', 'restaurant2.jpg'));
35 insert into restaurant values (3, 'Ivans', BFILENAME('locatie_fisiere', 'restaurant3.jpg'));
36 insert into restaurant values (4, 'Savanna', BFILENAME('locatie_fisiere', 'restaurant4.jpg'));
37 insert into restaurant values (5, 'Grande appetito!', BFILENAME('locatie_fisiere', 'restaurant5.jpg'));
38
39 commit;
```

The output pane shows the results of the execution:

```
1 row inserted.
Commit complete.
```

A red circle highlights the message "1 row inserted." and another red circle highlights the message "Commit complete."

Verificăm dacă datele au fost inserate cu succes.



The screenshot shows the Oracle SQL Developer interface. In the left sidebar, under the 'Connections' section, there is a tree view of the database schema, including tables like BUCATAR and RESTAURANT, and various database objects like procedures and triggers. The main workspace contains a 'Worksheet' tab with the following SQL code:

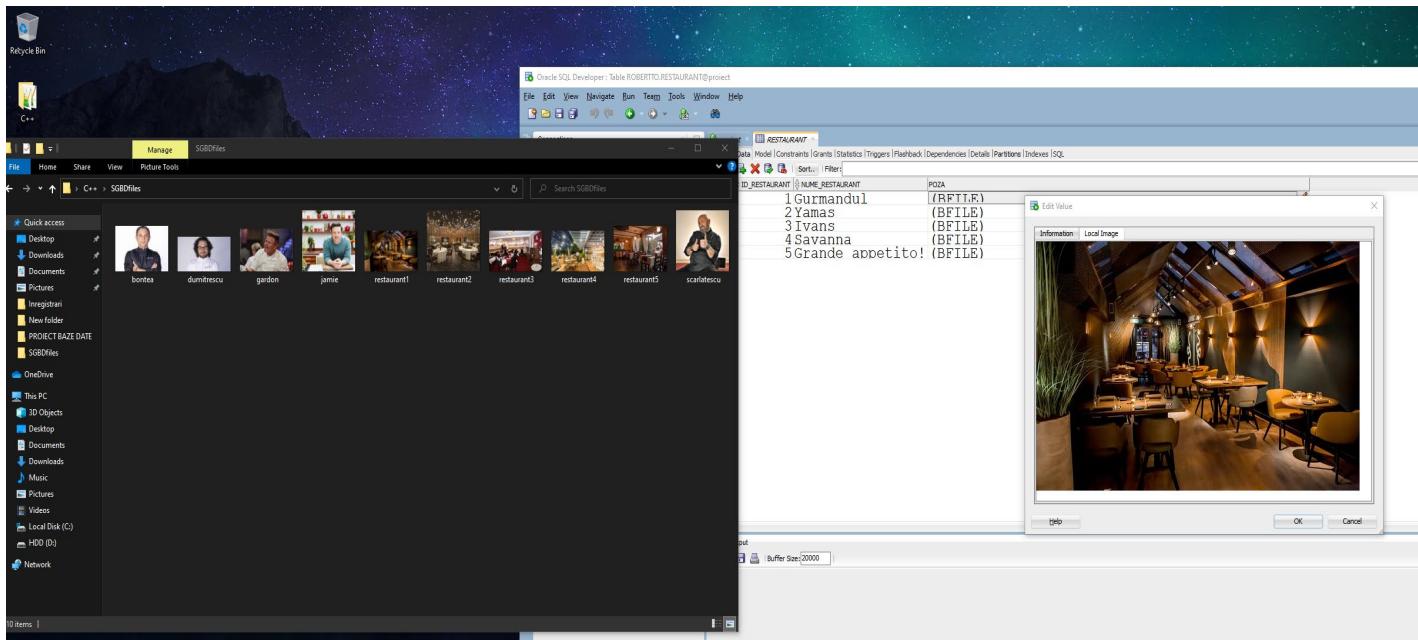
```
29    );
30
31  commit;
32
33  insert into restaurant values (1, 'Gurmandul', BFILENAME('locatie_fisiere', 'restaurant1'));
34  insert into restaurant values (2, 'Yamas', BFILENAME('locatie_fisiere', 'restaurant2.jpg'));
35  insert into restaurant values (3, 'Ivans', BFILENAME('locatie_fisiere', 'restaurant3.jpg'));
36  insert into restaurant values (4, 'Savanna', BFILENAME('locatie_fisiere', 'restaurant4.jpg'));
37  insert into restaurant values (5, 'Grande appetito!', BFILENAME('locatie_fisiere', 'restaurant5.jpg'));
38
39  commit;
40
41  select * from restaurant;
```

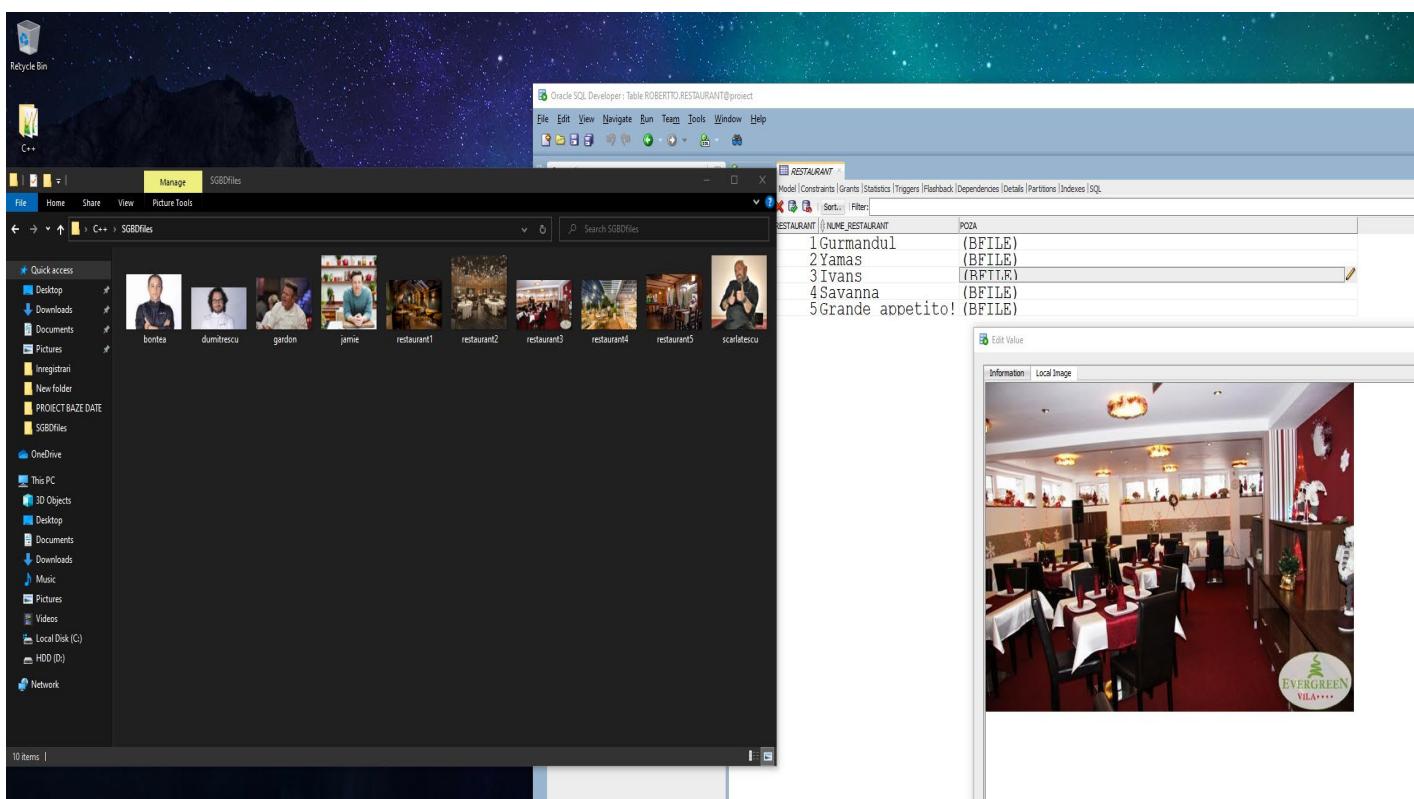
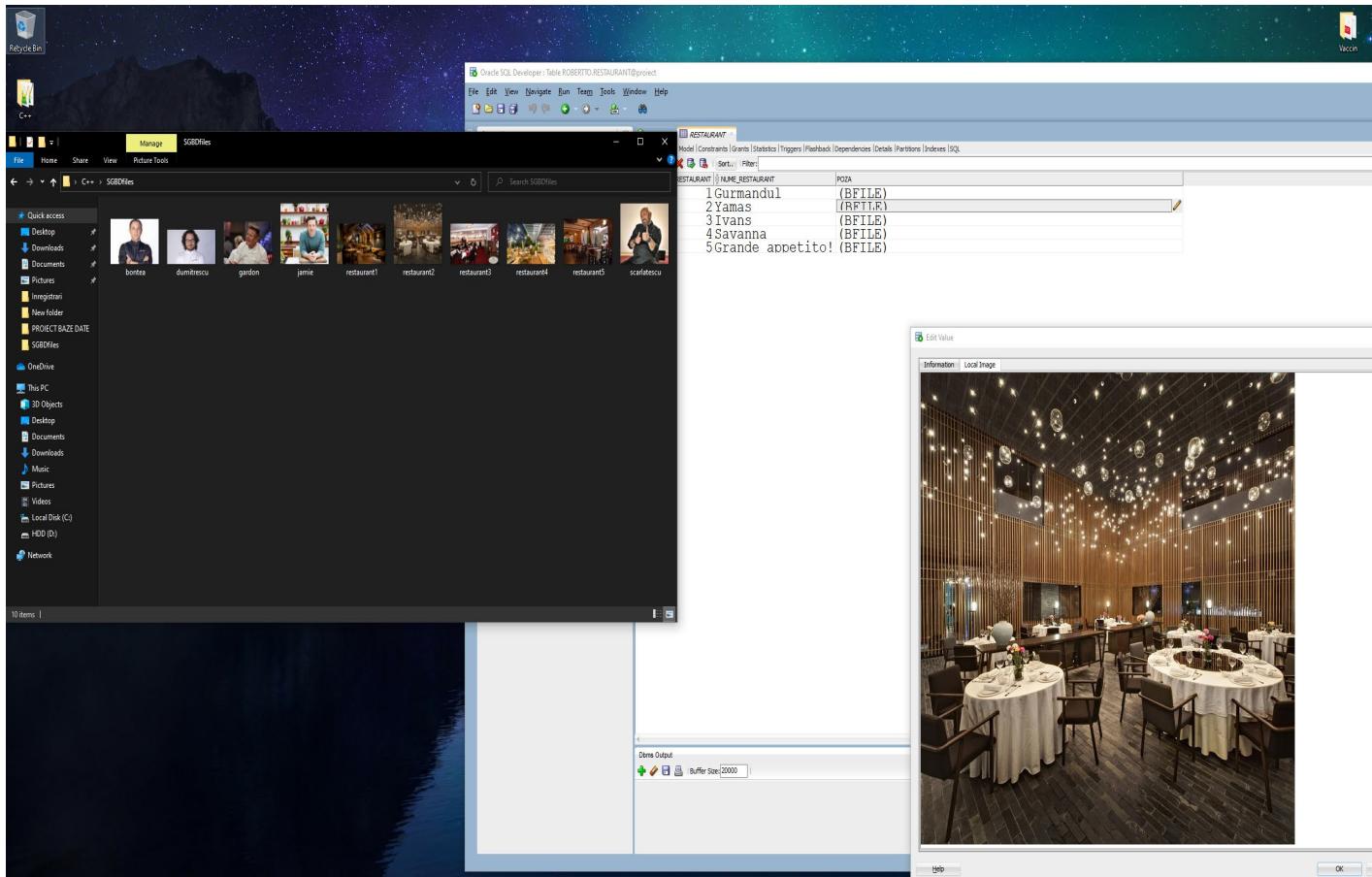
Below the worksheet is a 'Script Output' tab showing the results of the query:

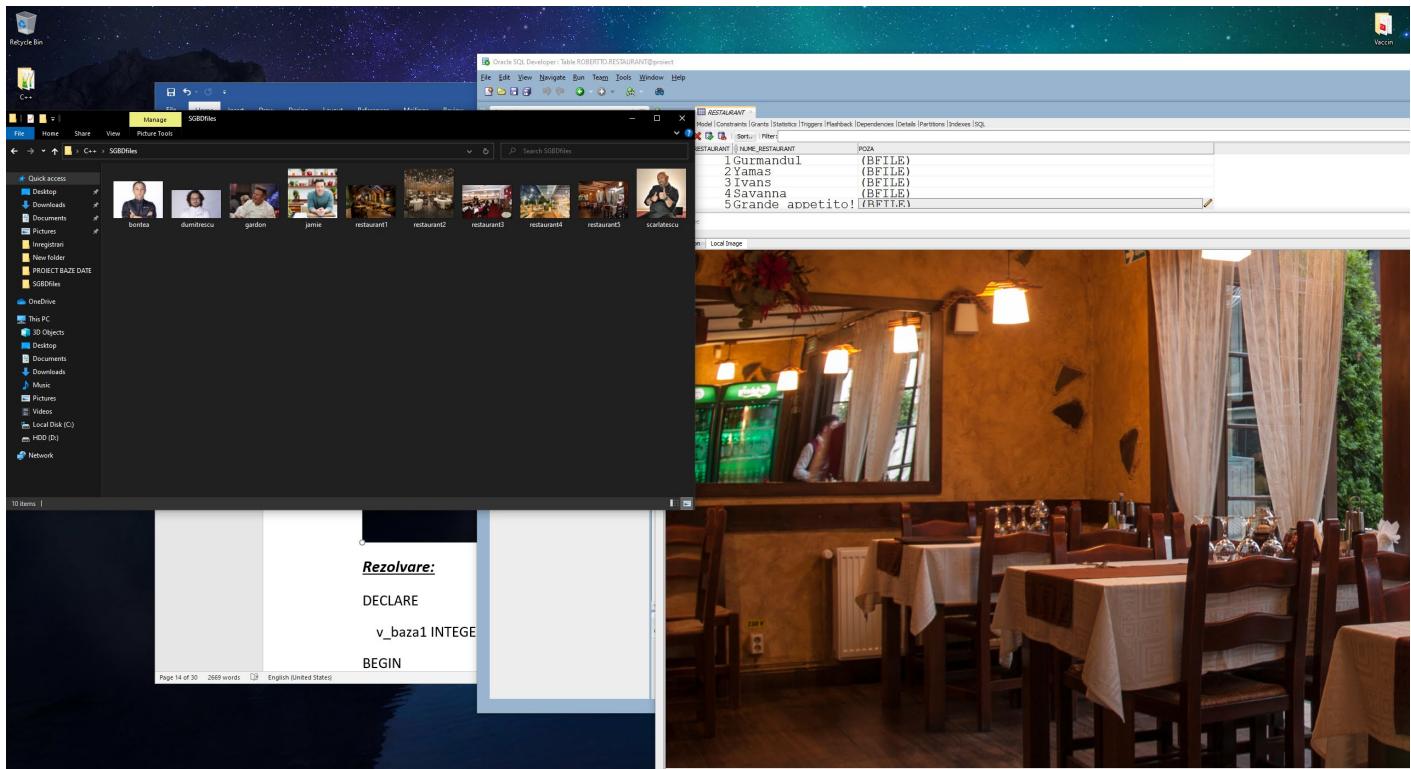
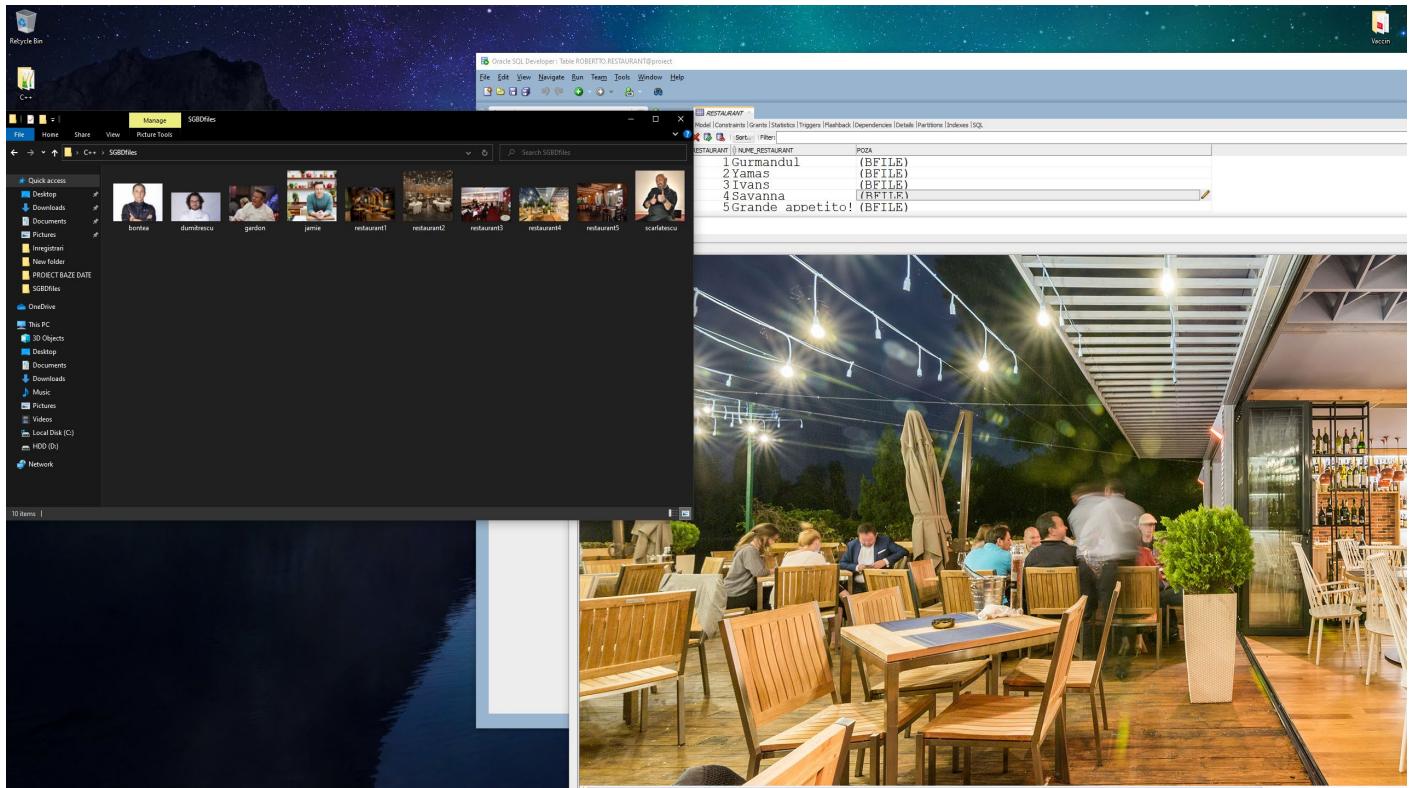
ID_RESTAURANT	NUME_RESTAURANT	POZA
1	Gurmandul	bfilename('locatie_fisiere','restaurant1')
2	Yamas	bfilename('locatie_fisiere','restaurant2.jpg')
3	Ivans	bfilename('locatie_fisiere','restaurant3.jpg')
4	Savanna	bfilename('locatie_fisiere','restaurant4.jpg')
5	Grande appetito!	bfilename('locatie_fisiere','restaurant5.jpg')

The last row (ID 5) is highlighted with a red circle. The 'Dmls Output' tab below shows no errors.

Observăm că datele au fost inserate exact cum ne doream. Acum vom verifica dacă pozele au fost preluate.







Observăm că pozele noastre au fost preluate exact cum ne-am dorit inițial.

**Testăm select rowid, employee\_id from employees;**

**Print-Screen:**

The screenshot shows a SQL lab interface with a Worksheet tab and a Query Result tab.

In the Worksheet tab, the query is:

```
1 | select rowid, employee_id from employees;
```

In the Query Result tab, the output is:

ROWID	EMPLOYEE_ID
1 AACaYvAAEAAAARoAAA	100
2 AACaYvAAEAAAARoAAB	101
3 AACaYvAAEAAAARoAAC	102
4 AACaYvAAEAAAARoAAD	103
5 AACaYvAAEAAAARoAAE	104
6 AACaYvAAEAAAARoAAF	105
7 AACaYvAAEAAAARoAAG	106
8 AACaYvAAEAAAARoAAH	107
9 AACaYvAAEAAAARoAAI	108
10 AACaYvAAEAAAARoAAJ	109
11 AACaYvAAEAAAARoAAK	110
12 AACaYvAAEAAAARoAAL	111
13 AACaYvAAEAAAARoAAM	112
14 AACaYvAAEAAAARoAAN	113
15 AACaYvAAEAAAARoAAO	114
16 AACaYvAAEAAAARoAAP	115
17 AACaYvAAEAAAARoAAO	116
18 AACaYvAAEAAAARoAAR	117
19 AACaYvAAEAAAARoAAS	118
20 AACaYvAAEAAAARoAAT	119
21 AACaYvAAEAAAARoAAU	120
22 AACaYvAAEAAAARoAAV	121
23 AACaYvAAEAAAARoAAW	122
24 AACaYvAAEAAAARoAAX	123
25 AACaYvAAEAAAARoAAY	124
26 AACaYvAAEAAAARoAAZ	125
27 AACaYvAAEAAAARoAAa	126
28 AACaYvAAEAAAARoAAb	127
29 AACaYvAAEAAAARoAAc	128
30 AACaYvAAEAAAARoAAd	129
31 AACaYvAAEAAAARoAAe	130
32 AACaYvAAEAAAARoAAf	131
33 AACaYvAAEAAAARoAAg	132

Observăm că rowid este diferit de employee\_id. Rowid este defapt o pseudocoloană cu o valoare unică la nivel de tabel definită **bydefault**.