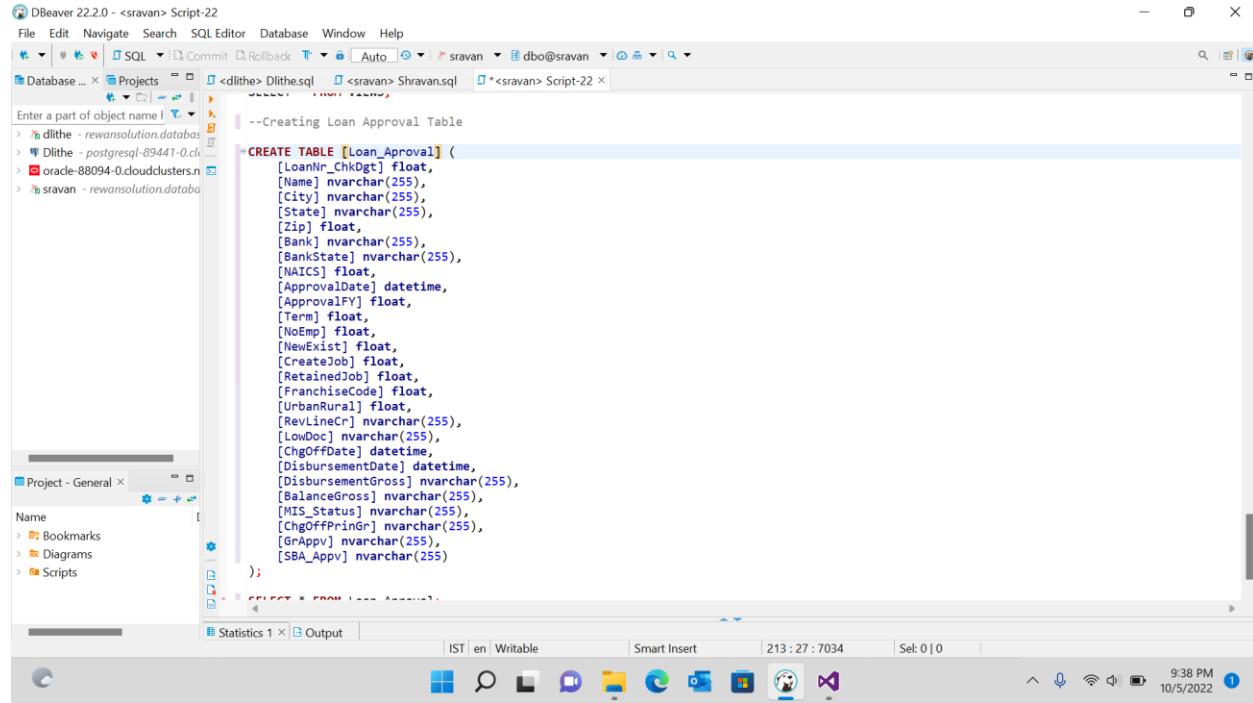


# 1. DATA WAREHOUSING

## Creating Loan Approval Destination Table



DBeaver 22.2.0 - <sravan> Script-22

--Creating Loan Approval Table

```
CREATE TABLE [dbo].[Loan_Aproval] (
    [LoanNr_ChnDgt] float,
    [Name] nvarchar(255),
    [City] nvarchar(255),
    [State] nvarchar(255),
    [Zip] float,
    [Bank] nvarchar(255),
    [BankState] nvarchar(255),
    [NAICS] float,
    [ApprovalDate] datetime,
    [ApprovalFY] float,
    [Term] float,
    [NoEmp] float,
    [NewExist] float,
    [CreateJob] float,
    [RetainedJob] float,
    [FranchiseCode] float,
    [UrbanRural] float,
    [RevLineCr] nvarchar(255),
    [LowDoc] nvarchar(255),
    [ChgOffDate] datetime,
    [DisbursementDate] datetime,
    [DisbursementGross] nvarchar(255),
    [BalanceGross] nvarchar(255),
    [MIS_Status] nvarchar(255),
    [ChgOffPrinGr] nvarchar(255),
    [GrAppv] nvarchar(255),
    [SBA_Appv] nvarchar(255)
);
```

Project - General

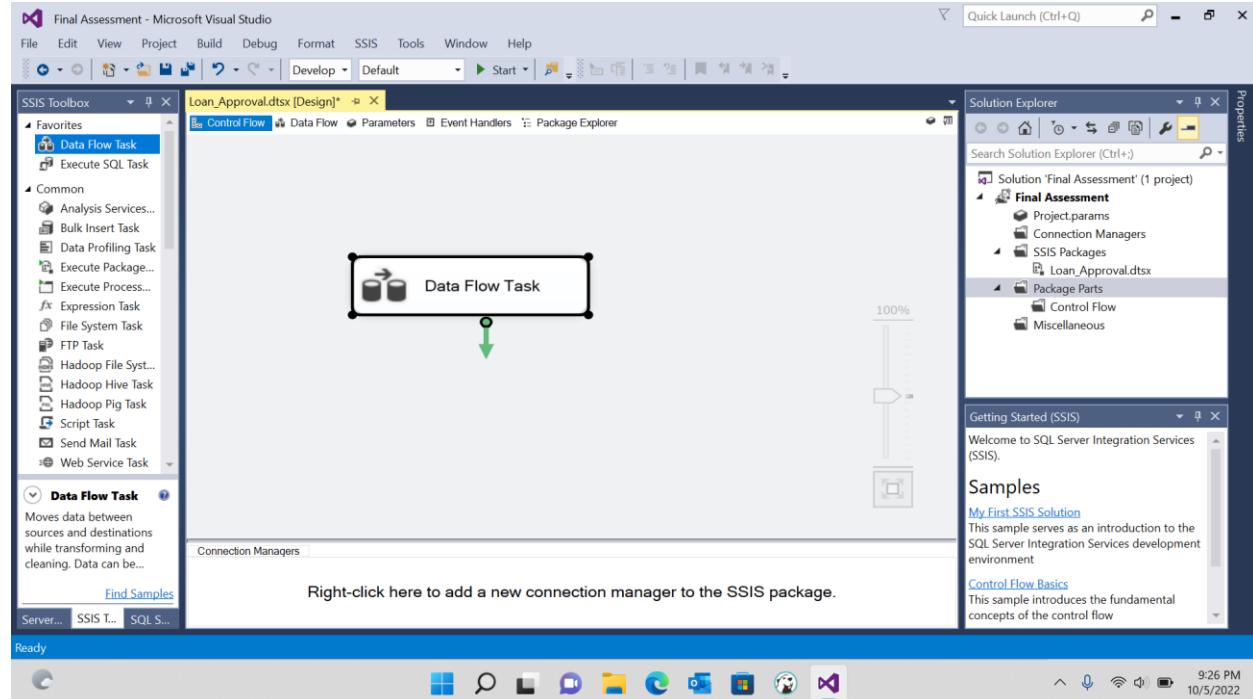
- Name
- Bookmarks
- Diagrams
- Scripts

Statistics 1 Output

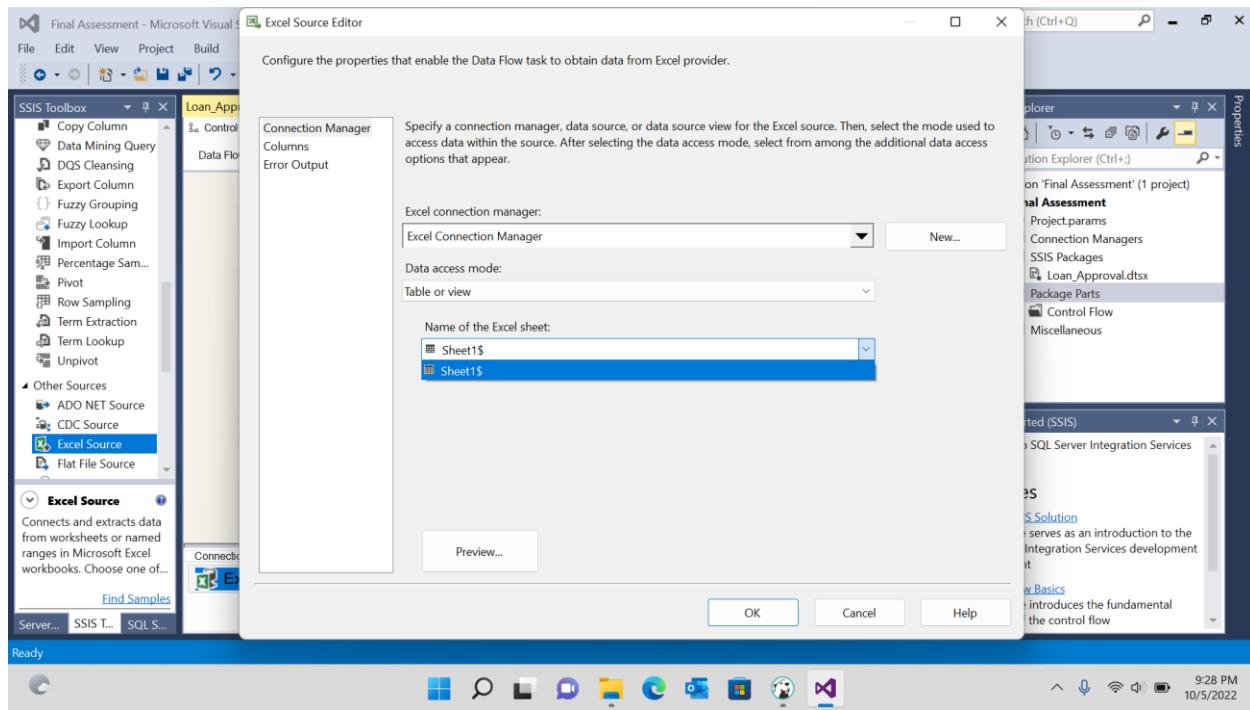
IST en Writable Smart Insert 213 : 27 : 7034 Sel: 0 | 0

9:38 PM 10/5/2022

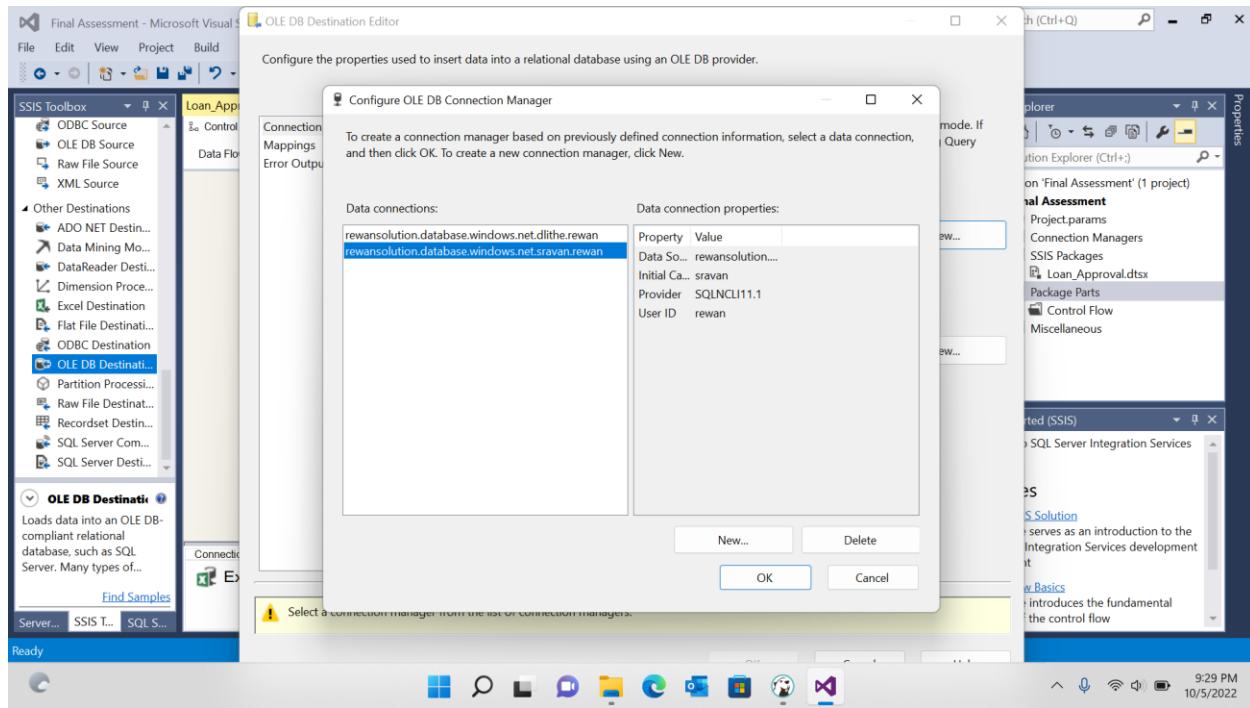
## Data Flow Task for Loan Approval Package



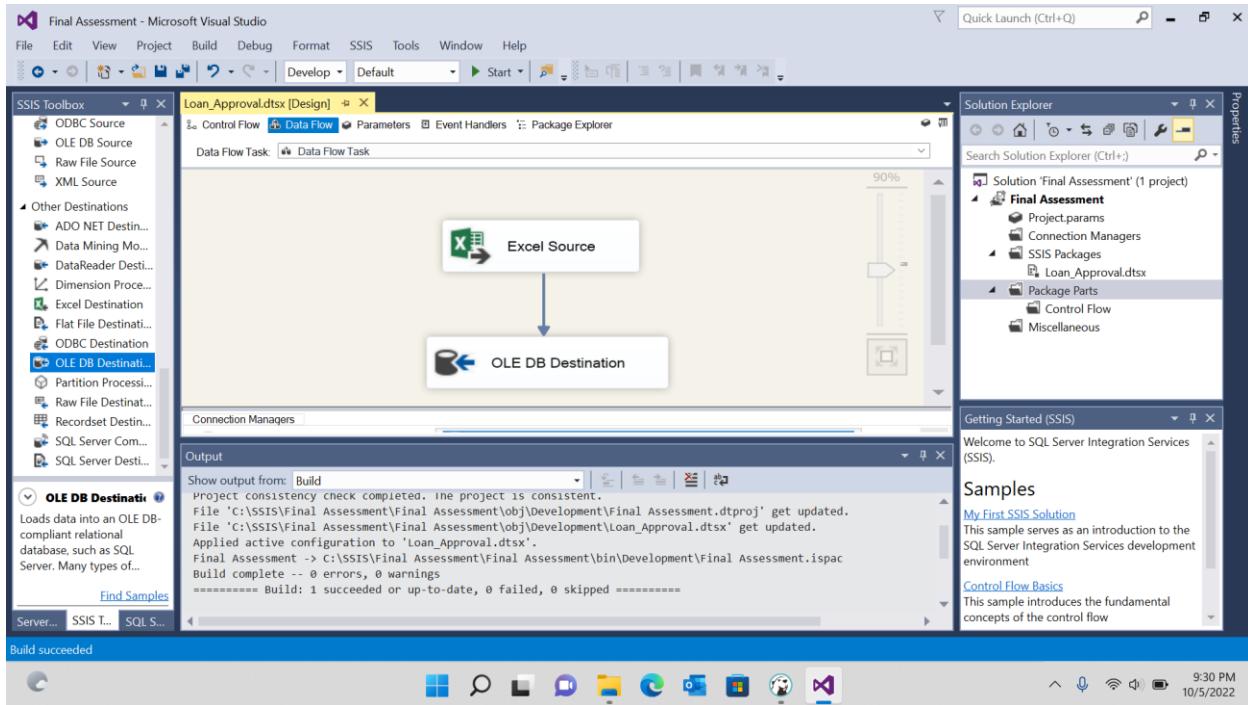
## Connecting Excel Manager



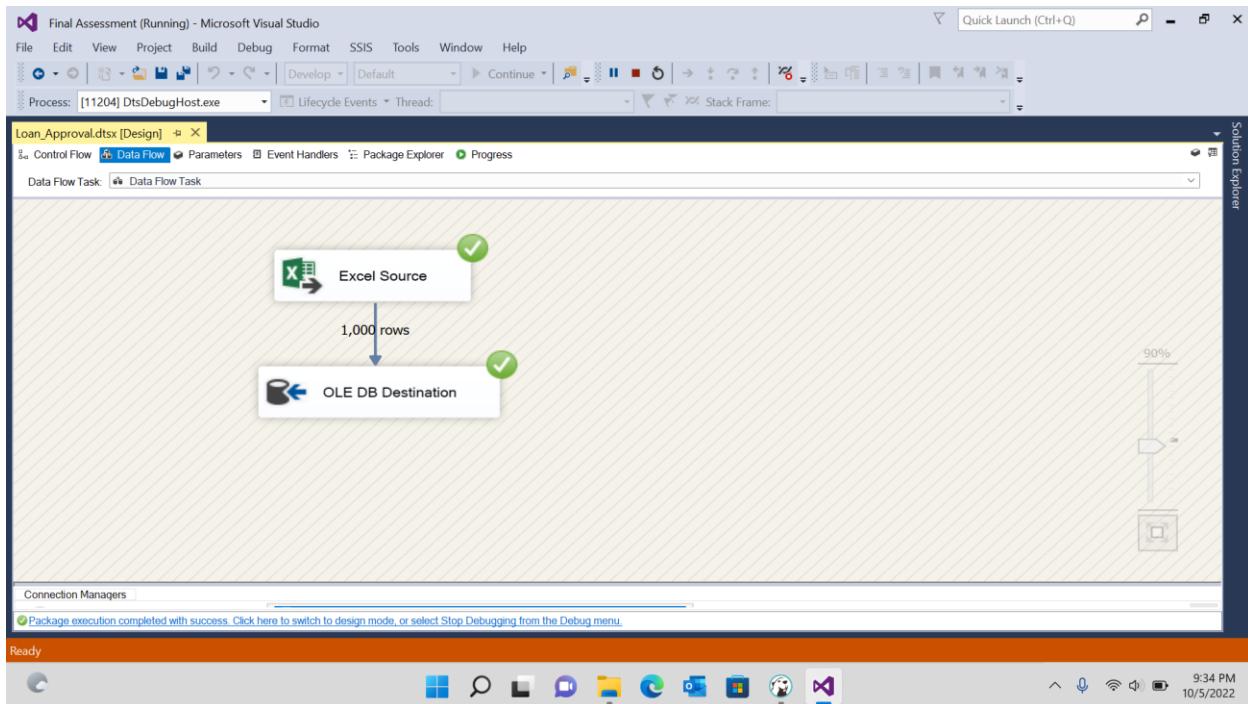
## Connecting to SQL Server



## Loan Approval Package



## Package Execution



## Data in Destination after Data Transfer

The screenshot shows the DBBeaver interface with the following details:

- Title Bar:** DBBeaver 22.2.0 - <sravan> Script-22
- Toolbar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help
- Left Panel:** Projects, Databases (dlithe, sravan), Scripts, General.
- SQL Editor:** Contains the following SQL code:
 

```

      [RevLineCr] nvarchar(255),
      [LowDoc] nvarchar(255),
      [ChgOffDate] datetime,
      [DisbursementDate] datetime,
      [DisbursementGross] nvarchar(255),
      [BalanceGross] nvarchar(255),
      [MIS_Status] nvarchar(255),
      [ChgOffPrinGr] nvarchar(255),
      [GrAppv] nvarchar(255),
      [SBA_Appv] nvarchar(255)
    );
  
```

```

  SELECT * FROM Loan_Aproval;
  
```
- Results Grid:** Shows 11 rows of data from the Loan\_Aproval table. The columns include LoanNr, ChkDgt, acrName, City, State, Zip, Bank, BankState, NAICS, ApprovalDate, ApprovalFy, and Value.
- Bottom Status Bar:** IST | en | Writable | Smart Insert | 243 : 28 : 7824 | Sel: 0 | 0
- System Tray:** Shows the date and time as 10/5/2022 9:39 PM.

## Creating Destination Table(Cinema\_Entertainment)

The screenshot shows the DBBeaver interface with the following details:

- Title Bar:** DBBeaver 22.2.0 - <sravan> Script-22
- Toolbar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help
- Left Panel:** Projects, Databases (dlithe, sravan), Scripts, General.
- SQL Editor:** Contains the following SQL code:
 

```

      [LowDoc] nvarchar(255),
      [ChgOffDate] datetime,
      [DisbursementDate] datetime,
      [DisbursementGross] nvarchar(255),
      [BalanceGross] nvarchar(255),
      [MIS_Status] nvarchar(255),
      [ChgOffPrinGr] nvarchar(255),
      [GrAppv] nvarchar(255),
      [SBA_Appv] nvarchar(255)
    );
  
```

```

  SELECT * FROM Loan_Aproval;
  
```

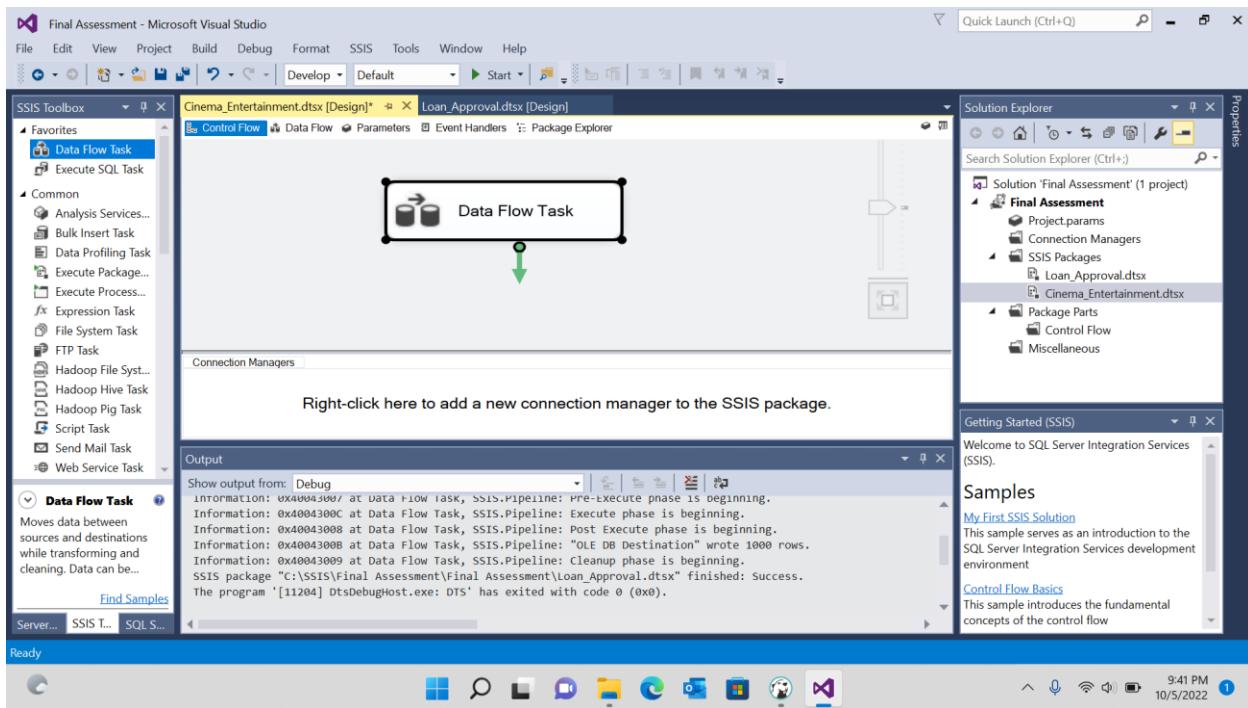
```

  --Creating Cinema Entertainment table
  
```

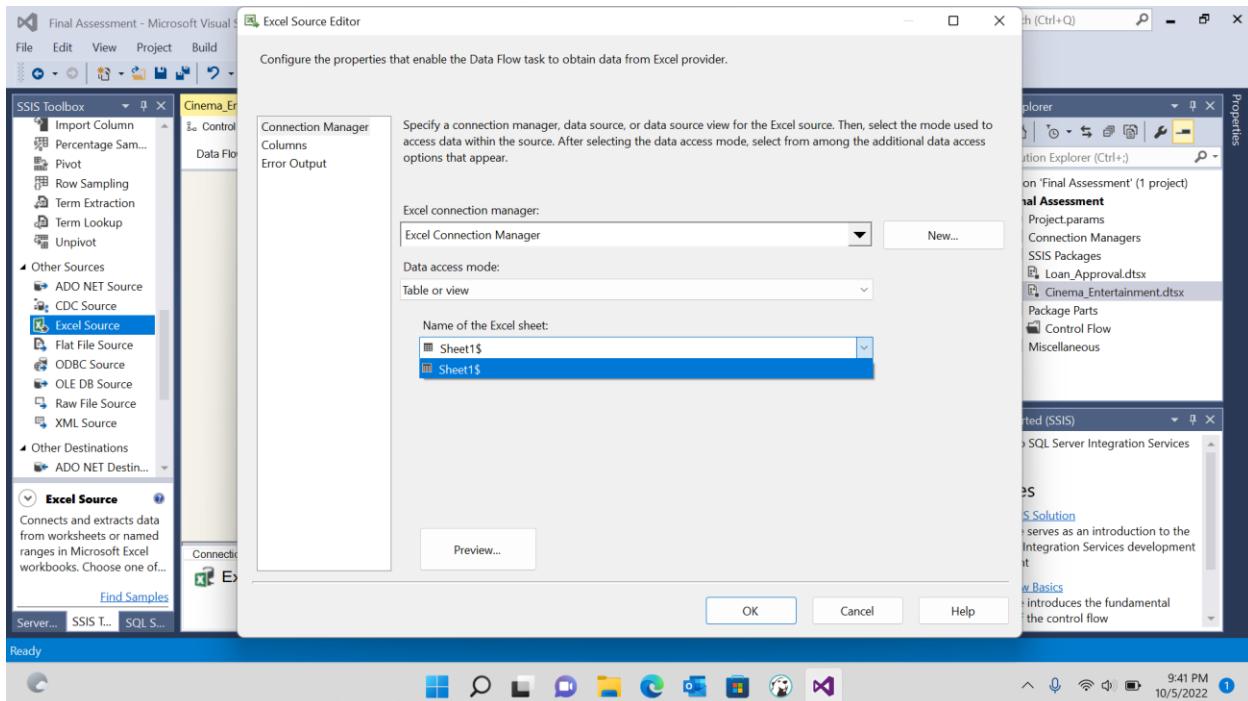
```

  CREATE TABLE [Cinema_Entertainment] (
    [Film_Type] nvarchar(255),
    [film_code] float,
    [cinema_code] float,
    [total_sales] float,
    [tickets_sold] float,
    [tickets_out] float,
    [show_time] float,
    [occu_perc] float,
    [ticket_price] float,
    [ticket_use] float,
    [capacity] float,
    [date] float,
    [month] float,
    [quarter] float,
    [day] float
  );
  
```
- Results Grid:** Shows the results of the CREATE TABLE command, indicating 1 row(s) affected.
- Bottom Status Bar:** IST | en | Writable | Smart Insert | 263 : 3 : 8272 | Sel: 0 | 0
- System Tray:** Shows the date and time as 10/5/2022 9:43 PM.

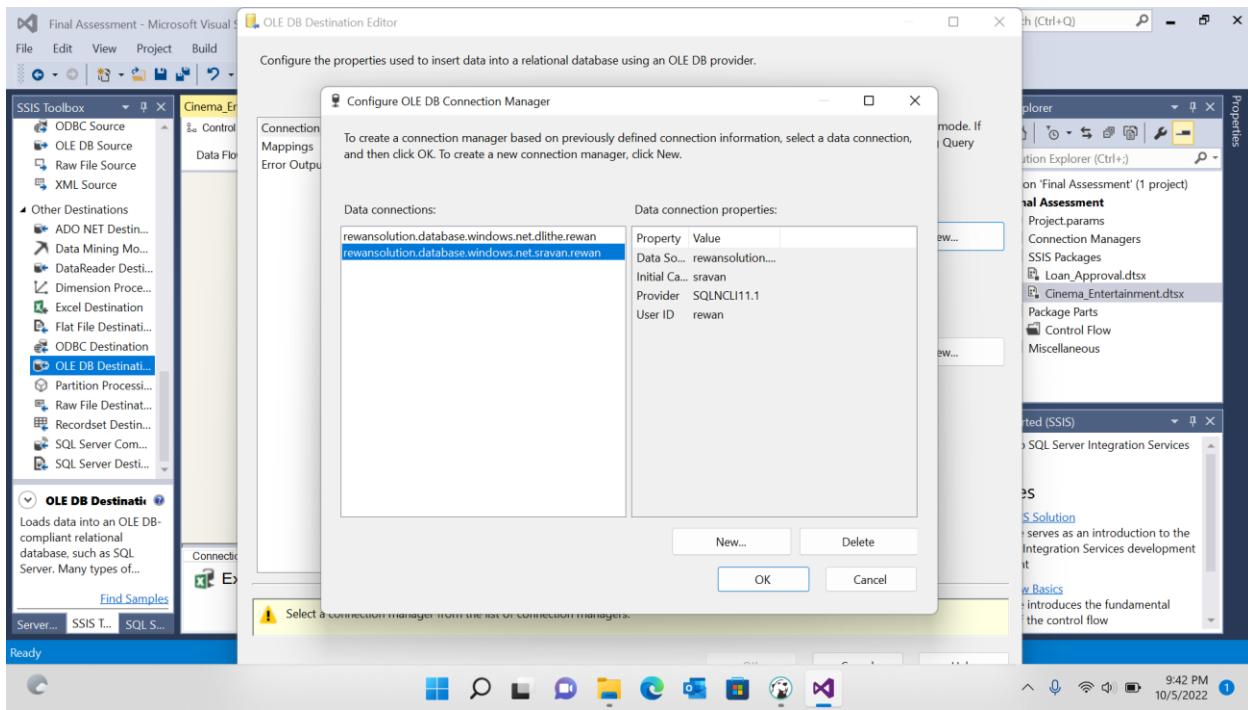
## Creating Cinema Entertainment Package



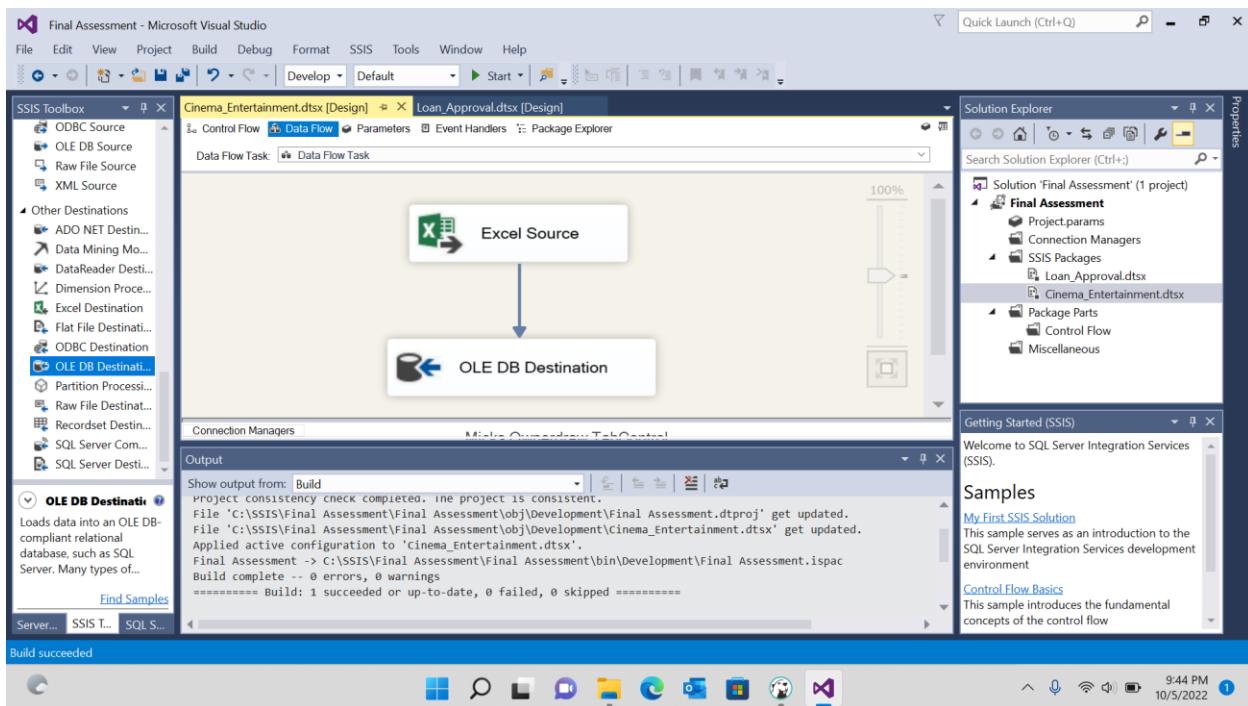
## Connecting Excel Manager and Selecting Source Table



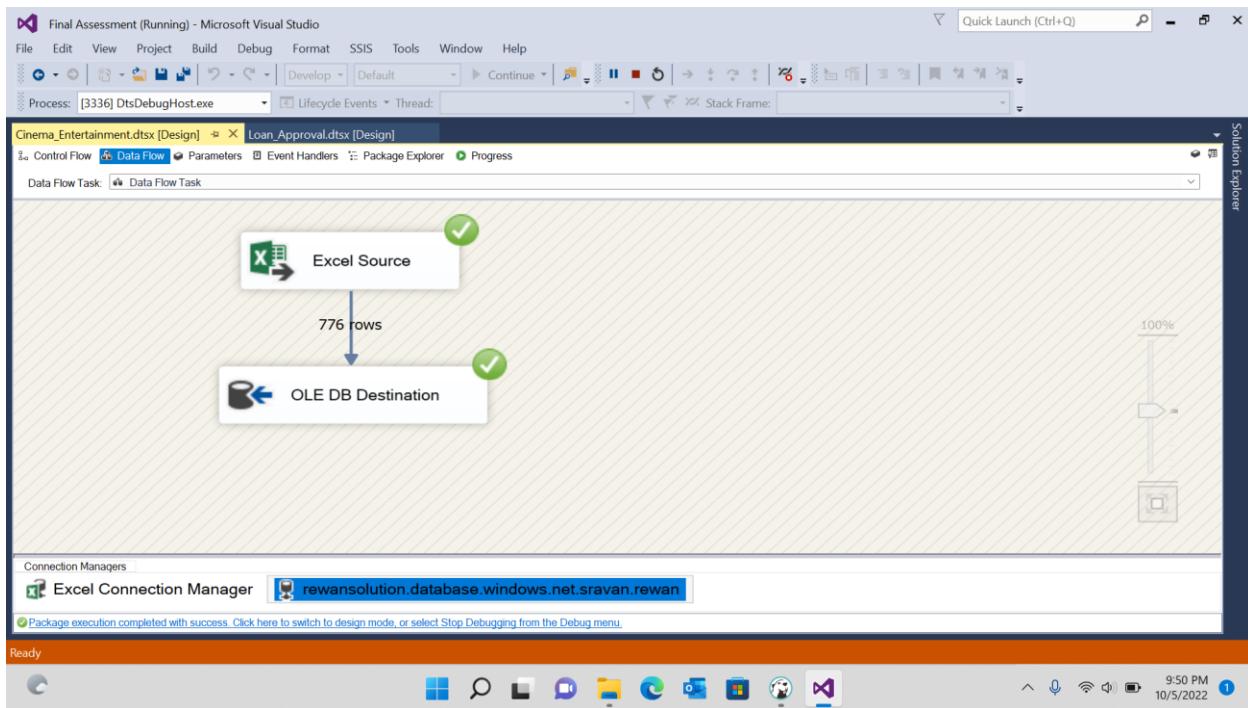
## Connecting Destination to SQL Server



## Data Flow Task



## Execution of Cinema\_Entertainment package



## Data Transfer after Execution

The screenshot shows the DBeaver 22.2.0 interface. In the SQL Editor, a CREATE TABLE statement for 'Cinema\_Entertainment' is displayed:

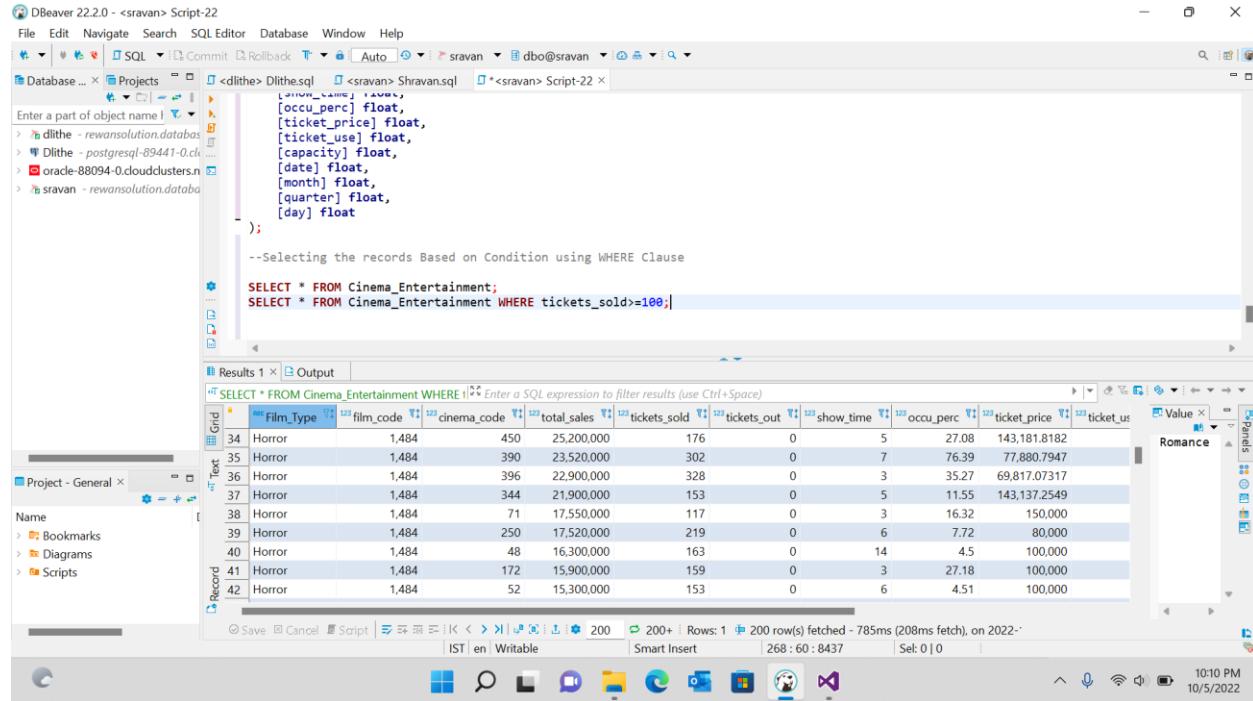
```
CREATE TABLE [Cinema_Entertainment] (
    [Film_Type] nvarchar(255),
    [film_code] float,
    [cinema_code] float,
    [total_sales] float,
    [tickets_sold] float,
    [tickets_out] float,
    [show_time] float,
    [occu_perc] float,
    [ticket_price] float,
    [ticket_use] float,
    [capacity] float,
    [date] float,
    [month] float,
    [quarter] float,
    [day] float
);
```

Below the CREATE TABLE statement, a SELECT \* FROM Cinema\_Entertainment; query is run, and the results are displayed in a grid:

Grid	Film_Type	film_code	cinema_code	total_sales	tickets_sold	tickets_out	show_time	occu_perc	ticket_price	ticket_us
1	Romance	1,492	304	3,900,000	26	0	4	4.26	150.000	
2	Romance	1,492	352	3,360,000	42	0	5	8.08	80.000	
3	Romance	1,492	489	2,560,000	32	0	4	20	80.000	
4	Romance	1,492	429	1,200,000	12	0	1	11.01	100.000	
5	Romance	1,492	524	1,200,000	15	0	3	16.67	80.000	
6	Romance	1,492	71	1,050,000	7	0	3	0.98	150.000	
7	Romance	1,492	162	1,020,000	10	0	7	7.60	100.000	

## 2. SELECT COMMAND USING WHERE CLAUSE

### Select Using Where



DBeaver 22.2.0 - <sravan> Script-22

File Edit Navigate Search SQL Editor Database Window Help

Database ... Projects <sravan> Shravansql <sravan> Script-22

Enter a part of object name |

> <sravan> - rewansolution.databas...

> <sravan> - postgresql-89441-0.cl...

> oracle-88094-0.cloudclusters.n...

> <sravan> - rewansolution.databa...

```
--Selecting the records Based on Condition using WHERE Clause
SELECT * FROM Cinema_Entertainment;
SELECT * FROM Cinema_Entertainment WHERE tickets_sold>=100;
```

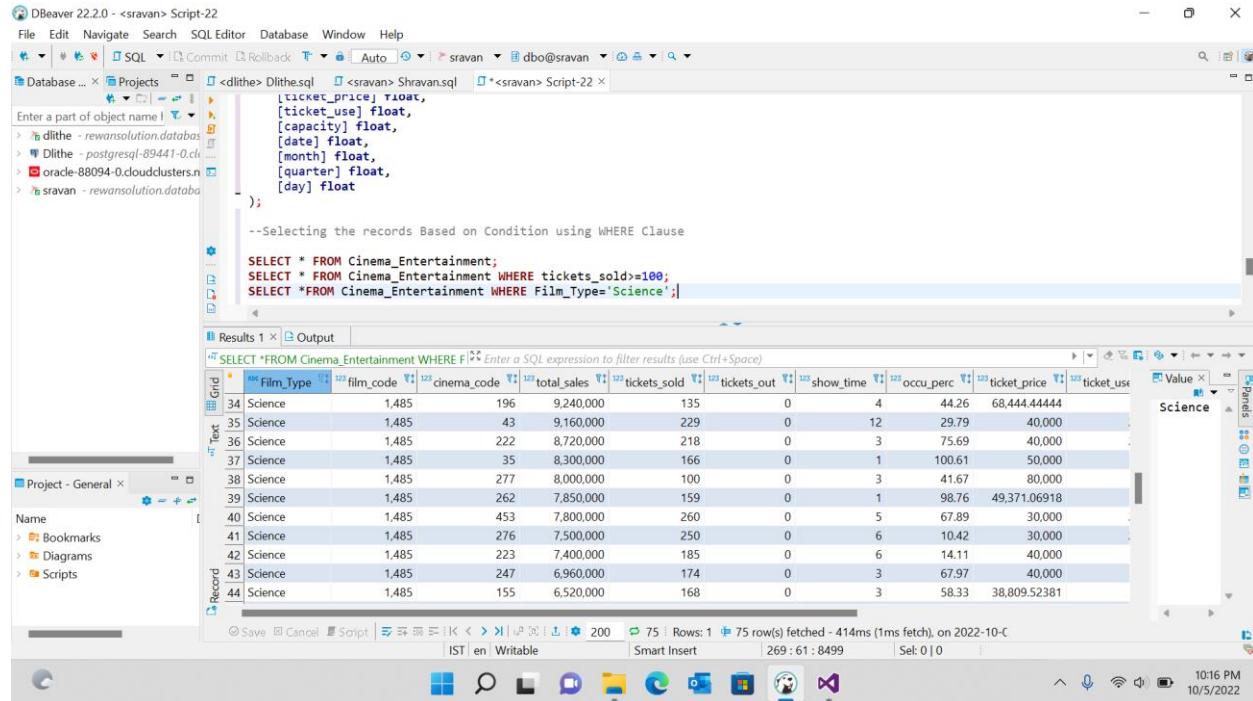
Results 1 × Output

SELECT \* FROM Cinema\_Entertainment WHERE | Enter a SQL expression to filter results (use Ctrl+Space)

Grid	film_type	film_code	cinema_code	total_sales	tickets_sold	tickets_out	show_time	occu_perc	ticket_price	ticket_us
34	Horror	1.484	450	25,200,000	176	0	5	27.08	143,181.8182	
35	Horror	1.484	390	23,520,000	302	0	7	76.39	77,880.7947	
36	Horror	1.484	396	22,900,000	328	0	3	35.27	69,817.07317	
37	Horror	1.484	344	21,900,000	153	0	5	11.55	143,137.2549	
38	Horror	1.484	71	17,550,000	117	0	3	16.32	150,000	
39	Horror	1.484	250	17,520,000	219	0	6	7.72	80,000	
40	Horror	1.484	48	16,300,000	163	0	14	4.5	100,000	
41	Horror	1.484	172	15,900,000	159	0	3	27.18	100,000	
42	Horror	1.484	52	15,300,000	153	0	6	4.51	100,000	

Save Cancel Script | IST | en | Writable | Smart Insert | 268 : 60 : 8437 | Sel: 0 | 0 | 10:10 PM 10/5/2022

### Select Using Where Clause



DBeaver 22.2.0 - <sravan> Script-22

File Edit Navigate Search SQL Editor Database Window Help

Database ... Projects <sravan> Shravansql <sravan> Script-22

Enter a part of object name |

> <sravan> - rewansolution.databas...

> <sravan> - postgresql-89441-0.cl...

> oracle-88094-0.cloudclusters.n...

> <sravan> - rewansolution.databa...

```
--Selecting the records Based on Condition using WHERE Clause
SELECT * FROM Cinema_Entertainment;
SELECT * FROM Cinema_Entertainment WHERE tickets_sold>=100;
SELECT * FROM Cinema_Entertainment WHERE Film_Type='Science';
```

Results 1 × Output

SELECT \* FROM Cinema\_Entertainment WHERE | Enter a SQL expression to filter results (use Ctrl+Space)

Grid	Film_Type	film_code	cinema_code	total_sales	tickets_sold	tickets_out	show_time	occu_perc	ticket_price	ticket_us
34	Science	1.485	196	9,240,000	135	0	4	44.26	68,444.4444	
35	Science	1.485	43	9,160,000	229	0	12	29.79	40,000	
36	Science	1.485	222	8,720,000	218	0	3	75.69	40,000	
37	Science	1.485	35	8,300,000	166	0	1	100.61	50,000	
38	Science	1.485	277	8,000,000	100	0	3	41.67	80,000	
39	Science	1.485	262	7,850,000	159	0	1	98.76	49,371.06918	
40	Science	1.485	453	7,800,000	260	0	5	67.89	30,000	
41	Science	1.485	276	7,500,000	250	0	6	10.42	30,000	
42	Science	1.485	223	7,400,000	185	0	6	14.11	40,000	
43	Science	1.485	247	6,960,000	174	0	3	67.97	40,000	
44	Science	1.485	155	6,520,000	168	0	3	58.33	38,809.52381	

Save Cancel Script | IST | en | Writable | Smart Insert | 269 : 61 : 8499 | Sel: 0 | 0 | 10:16 PM 10/5/2022

## OR Operation in Where Clause

DBeaver 22.2.0 - <sravan> Script-22

```
--Selecting the records Based on Condition using WHERE Clause
SELECT * FROM Cinema_Entertainment;
SELECT * FROM Cinema_Entertainment WHERE tickets_sold=100;
SELECT * FROM Cinema_Entertainment WHERE Film_Type='Science';
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' OR cinema_code>=300;
```

Results 1 × Output

film_type	film_code	cinema_code	total_sales	tickets_sold	tickets_out	show_time	occu_perc	ticket_price	ticket_us
Action	1.482	501	22,580,000	604	3	4	75.12	37,384.10596	
Action	1.482	415	22,330,000	322	0	3	31.94	69,347.82609	
Action	1.482	445	20,880,000	174	0	3	60.42	120,000	
Action	1.482	442	19,760,000	247	0	2	63.33	80,000	
Action	1.482	499	17,040,000	142	0	3	23.91	120,000	
Action	1.482	522	16,950,000	113	2	5	25.98	150,000	
Action	1.482	428	16,380,000	234	0	2	78	70,000	
Action	1.482	310	15,240,000	254	0	2	28.22	60,000	
Action	1.482	313	15,080,000	194	0	4	15.95	77,731.95876	
Action	1.482	430	14,500,000	145	0	2	87.35	100,000	
Action	1.482	338	14,080,000	178	0	3	25.25	79,101.1236	

533 rows fetched - 668ms (11ms fetch), on 2022-10-05 10:19 PM

## AND Operation while Selecting Records

DBeaver 22.2.0 - <sravan> Script-22

```
--Selecting the records Based on Condition using WHERE Clause
SELECT * FROM Cinema_Entertainment;
SELECT * FROM Cinema_Entertainment WHERE tickets_sold>=100;
SELECT * FROM Cinema_Entertainment WHERE Film_Type='Science';
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' OR cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' AND cinema_code>=300;
```

Results 1 × Output

film_type	film_code	cinema_code	total_sales	tickets_sold	tickets_out	show_time	occu_perc	ticket_price	ticket_us
Fiction	1.481	362	3,255,000	93	0	4	14.58	35,000	
Fiction	1.481	450	2,170,000	31	0	1	22.79	70,000	
Fiction	1.481	516	1,260,000	54	8	1	46.15	40,000	
Fiction	1.481	321	1,575,000	63	0	2	7.22	25,000	
Fiction	1.481	529	1,140,000	19	0	1	40.43	60,000	
Fiction	1.481	507	425,000	17	0	1	9.34	25,000	
Fiction	1.481	495	330,000	11	0	1	3.86	30,000	
Fiction	1.481	326	250,000	5	0	2	0.87	50,000	
Fiction	1.481	304	82,950,000	556	0	7	46.72	149,190,6475	
Fiction	1.481	448	56,680,000	380	1	5	38.78	149,157,8947	

141 rows fetched - 529ms (16ms fetch), on 2022-10-05 10:20 PM

## BETWEEN Operator

DBeaver 22.0.0 - <sravan> Script-22

```
--Selecting the records Based on Condition using WHERE Clause
SELECT * FROM Cinema_Entertainment;
SELECT * FROM Cinema_Entertainment WHERE tickets_sold>=100;
SELECT * FROM Cinema_Entertainment WHERE Film_Type='Science';
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' OR cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' AND cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE cinema_code BETWEEN 500 AND 700;
```

Results 1 | Output

film_type	film_code	cinema_code	total_sales	tickets_sold	tickets_out	show_time	occu_perc	ticket_price	ticket_us
Romance	1.492	524	1,200,000	15	0	3	16.67	80,000	
Romance	1.492	522	600,000	4	0	3	1.55	150,000	
Romance	1.492	529	480,000	4	0	3	2.96	120,000	
Romance	1.492	522	3,000,000	20	0	3	7.75	150,000	
Romance	1.492	524	1,800,000	12	0	3	13.33	150,000	
Romance	1.492	518	1,680,000	14	1	3	8.48	120,000	
Romance	1.492	529	600,000	5	0	2	5.56	120,000	
Thriller	1.486	513	21,700,000	251	0	2	51.97	86,454.18327	
Thriller	1.486	529	15,360,000	131	0	15	19.01	117,251.9084	
Thriller	1.486	524	13,560,000	113	2	7	20.25	120,000	
Thriller	1.486	518	5,520,000	47	0	6	5.28	117,446.8085	
Thriller	1.486	522	5,040,000	42	0	3	16.28	120,000	

Save Cancel Script | 200 | Rows: 0 | 116 row(s) fetched - 607ms (1ms fetch), on 2022-10-05 10:22 PM | IST | en | Writable | Smart Insert | 272 : 65 : 8735 | Sel: 0 | 0

## LIKE Function

DBeaver 22.0.0 - <sravan> Script-22

```
--Selecting the records Based on Condition using WHERE Clause
SELECT * FROM Cinema_Entertainment;
SELECT * FROM Cinema_Entertainment WHERE tickets_sold>=100;
SELECT * FROM Cinema_Entertainment WHERE Film_Type='Science';
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' OR cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' AND cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE cinema_code BETWEEN 500 AND 700;
SELECT * FROM Cinema_Entertainment WHERE Film_Type LIKE '%e%';
```

Results 1 | Output

film_type	film_code	cinema_code	total_sales	tickets_sold	tickets_out	show_time	occu_perc	ticket_price	ticket_us
Romance	1.492	431	720,000	6	0	2	11.54	120,000	
Romance	1.492	529	600,000	5	0	2	5.56	120,000	
Romance	1.492	82	160,000	2	0	6	0.21	80,000	
Romance	1.492	304	16,500,000	110	0	4	18	150,000	
Science	1.485	513	41,820,000	697	0	3	93.81	60,000	
Science	1.485	52	38,450,000	769	0	7	25.73	50,000	
Science	1.485	524	35,680,000	446	2	7	55.13	80,000	
Science	1.485	450	33,180,000	476	0	5	80.68	69,705.88235	
Science	1.485	163	31,800,000	530	0	9	69.37	60,000	
Science	1.485	207	31,230,000	1,041	0	12	33.37	30,000	1
Science	1.485	522	30,480,000	381	5	6	66.84	80,000	
Science	1.485	180	29,560,000	753	0	6	68.58	39,256.3081	

Save Cancel Script | 200 | Rows: 0 | 109 row(s) fetched - 1.39s (113ms fetch), on 2022-10-05 10:24 PM | IST | en | Writable | Smart Insert | 273 : 62 : 8807 | Sel: 0 | 0

### 3. STRING FUNCTIONS

#### TOP 10 Function

DBeaver 22.2.0 - <sravan> Script-22

```
-->Selecting the records based on condition using WHERE Clause
SELECT * FROM Cinema_Entertainment;
-->Selecting Using String Functions
SELECT TOP 10 * FROM Cinema_Entertainment ;
```

Results 1 × Output

1	Film_Type	film_code	cinema_code	total_sales	tickets_sold	tickets_out	show_time	occu_perc	ticket_price	ticket_use	Value	26
1	Romance	1,492	304	3,900,000	26	0	4	4.26	150,000	26	Romance	
2	Romance	1,492	352	3,360,000	42	0	5	8.08	80,000	42		
3	Romance	1,492	489	2,560,000	32	0	4	20	80,000	32		
4	Romance	1,492	429	1,200,000	12	0	1	11.01	100,000	12		
5	Romance	1,492	524	1,200,000	15	0	3	16.67	80,000	15		
6	Romance	1,492	71	1,050,000	7	0	3	0.98	150,000	7		
7	Romance	1,492	163	1,020,000	10	0	3	7.69	102,000	10		
8	Romance	1,492	450	750,000	5	0	3	1.57	150,000	5		
9	Romance	1,492	51	750,000	11	0	2	0.95	68,181.81818	11		
10	Romance	1,492	522	600,000	4	0	3	1.55	150,000	4		

Save Cancel Script | 1K < > 200 10 Rows: 0 10 row(s) fetched - 121ms, on 2022-10-05 at 22:38:3 IST | en Writable | Smart Insert | 277 : 44 : 8893 | Sel: 0 | 0 | 10:38 PM 10/5/2022

#### UPPER Operator

DBeaver 22.2.0 - <sravan> Script-22

```
-->Selecting Using String Functions
SELECT TOP 10 * FROM Cinema_Entertainment ;
SELECT Film_Type, UPPER(Film_Type) FROM Cinema_Entertainment ;
```

Results 1 × Output

1	Film_Type	UPPER(Film_Type)
1	Romance	ROMANCE
2	Romance	ROMANCE
3	Romance	ROMANCE
4	Romance	ROMANCE
5	Romance	ROMANCE
6	Romance	ROMANCE
7	Romance	ROMANCE
8	Romance	ROMANCE
9	Romance	ROMANCE
10	Romance	ROMANCE
11	Romance	ROMANCE
12	Romance	ROMANCE
13	Romance	ROMANCE

Save Cancel Script | 1K < > 200+ Rows: 1 200 row(s) fetched - 329ms (1ms fetch), on 2022-10-05 at 22:39:10 IST | en Writable | Smart Insert | 278 : 63 : 8957 | Sel: 0 | 0 | 10:39 PM 10/5/2022

## LOWER Operator

DBeaver 22.2.0 - <sravan> Script-22

```
File Edit Navigate Search SQL Editor Database Window Help
<dlithe> Auto_ sravan dbo@sravan
Database ... Projects <dlithe> Olithe.sql <sravan> Shravan.sql <><sravan> Script-22
Enter a part of object name! <dlithe> - rewansolution.database
<dlithe> - postgresql-89441-0.cloudclusters.net
<sravan> - rewansolution.database
--Selecting Using String Functions
SELECT * FROM Cinema_Entertainment WHERE Film_Type='Science';
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' OR cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' AND cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE cinema_code BETWEEN 500 AND 700;
SELECT * FROM Cinema_Entertainment WHERE Film_Type LIKE '%e';
--Selecting Using String Functions
SELECT TOP 10 * FROM Cinema_Entertainment ;
SELECT Film_Type, UPPER(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LOWER(Film_Type) FROM Cinema_Entertainment;
```

Results 1 × Output

Film_Type
Romance

Grid Text Record

1 Romance romance

2 Romance romance

3 Romance romance

4 Romance romance

5 Romance romance

6 Romance romance

7 Romance romance

8 Romance romance

9 Romance romance

10 Romance romance

11 Romance romance

12 Romance romance

13 Romance romance

Save Cancel Script IST en Writable 200+ Rows: 1 200 row(s) fetched - 519ms, on 2022-10-05 at 22:40 279 : 62 : 9019 Set 0 | 0 10:40 PM 10/5/2022

## TRIM Operator

DBeaver 22.2.0 - <sravan> Script-22

```
File Edit Navigate Search SQL Editor Database Window Help
<dlithe> Auto_ sravan dbo@sravan
Database ... Projects <dlithe> Olithe.sql <sravan> Shravan.sql <><sravan> Script-22
Enter a part of object name! <dlithe> - rewansolution.database
<dlithe> - postgresql-89441-0.cloudclusters.net
<sravan> - rewansolution.database
--Selecting Using String Functions
SELECT * FROM Cinema_Entertainment WHERE Film_Type='Science';
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' OR cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' AND cinema_code>=300;
SELECT * FROM Cinema_Entertainment WHERE cinema_code BETWEEN 500 AND 700;
SELECT * FROM Cinema_Entertainment WHERE Film_Type LIKE '%e';
--Selecting Using String Functions
SELECT TOP 10 * FROM Cinema_Entertainment ;
SELECT Film_Type, UPPER(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LOWER(Film_Type) FROM Cinema_Entertainment;
SELECT TRIM(Film_Type), RTRIM(Film_Type) FROM Cinema_Entertainment;
```

Results 1 × Output

TRIM(Film_Type)	RTRIM(Film_Type)
Romance	Romance

Grid Text Record

1 Romance Romance

2 Romance Romance

3 Romance Romance

4 Romance Romance

5 Romance Romance

6 Romance Romance

7 Romance Romance

8 Romance Romance

9 Romance Romance

10 Romance Romance

11 Romance Romance

12 Romance Romance

13 Romance Romance

Save Cancel Script IST en Writable 200+ Rows: 1 200 row(s) fetched - 1.284s (1ms fetch), on 2022-10-05 at 22:40 280 : 68 : 9088 Set 0 | 0 10:48 PM 10/5/2022

## LEN Function

The screenshot shows the DBBeaver 22.2.0 interface. The top menu bar includes File, Edit, Navigate, Search, SQL Editor, Database, Window, Help. The title bar says "DBBeaver 22.2.0 - <sarvan> Script-22". The left sidebar has sections for Database, Projects, and a search bar "Enter a part of object name". Below these are connections: dltite (rewansolution.database), Dltite (postgresql-89441-0.cloudclusters.net), oracle-88094-0.cloudclusters.net, and sarvan (rewansolution.database). The main area has tabs for "dltite> Dltitesql", "<sarvan> Shravansql", and "<sarvan> Script-22". The current tab, "Script-22", contains the following SQL code:

```
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' OR cinema_code >= 300;
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' AND cinema_code >= 300;
SELECT * FROM Cinema_Entertainment WHERE cinema_code BETWEEN 500 AND 700;
SELECT * FROM Cinema_Entertainment WHERE Film_Type LIKE '%e';

--Selecting Using String Functions

SELECT TOP 10 * FROM Cinema_Entertainment ;
SELECT Film_Type, UPPER(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LOWER(Film_Type) FROM Cinema_Entertainment;
SELECT TRIM(Film_Type), RTRIM(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LEN(Film_Type) AS LEN FROM Cinema_Entertainment;
```

The results pane shows a table with 40 rows of data:

	Film_Type	LEN
28	Romance	7
29	Romance	7
30	Romance	7
31	Romance	7
32	Romance	7
33	Romance	7
34	Romance	7
35	Thriller	8
36	Thriller	8
37	Thriller	8
38	Thriller	8
39	Thriller	8
40	Thriller	8

The bottom status bar shows "Rows: 1 200+ 200 row(s) fetched - 726ms, on 2022-10-05 at 22:50", "Smart Insert", "281 : 67 : 9156", and "Sel: 0 | 0". The bottom right corner shows the date and time: "10:50 PM 10/5/2022".

## REPLACE Function

The screenshot shows the DBBeaver 22.2.0 application window. The top menu bar includes File, Edit, Navigate, Search, SQL Editor, Database, Window, Help. The title bar indicates the session is for 'srajan' on 'dbo@srajan'. The left sidebar shows the 'Database' tree with nodes like 'Rewardsolution', 'PostgreSQL', 'Oracle', and 'srajan'. The main area has tabs for 'SQL', 'Commit', and 'Rollback'. The 'SQL' tab contains a query editor with the following code:

```
SELECT * FROM Cinema_Entertainment WHERE Film_Type = 'Fiction' AND Cinema_Code >= 500;
SELECT * FROM Cinema_Entertainment WHERE cinema_code BETWEEN 500 AND 700;
SELECT * FROM Cinema_Entertainment WHERE Film_Type LIKE '%e';

--Selecting Using String Functions

SELECT TOP 10 * FROM Cinema_Entertainment;
SELECT Film_Type, UPPER(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LOWER(Film_Type) FROM Cinema_Entertainment;
SELECT TRIM(Film_Type), RTRIM(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LEN(Film_Type) AS LEN FROM Cinema_Entertainment;
SELECT REPLACE(Film_Type,'a','@') AS REPLACED FROM Cinema_Entertainment;
```

The 'Results' tab shows the output of the last query:

REPLACED
Rom@nce

At the bottom, there are various toolbars and status indicators.

## CONCAT Function

The screenshot shows the DBBeaver interface with the SQL Editor tab selected. The query window contains the following SQL code:

```
SELECT * FROM Cinema_Entertainment WHERE film_type = 'ROMANCE' AND cinema_code=304;
SELECT * FROM Cinema_Entertainment WHERE cinema_code BETWEEN 500 AND 700;
SELECT * FROM Cinema_Entertainment WHERE Film_Type LIKE '%e';

--Selecting Using String Functions

SELECT TOP 10 * FROM Cinema_Entertainment ;
SELECT Film_Type, UPPER(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LOWER(Film_Type) FROM Cinema_Entertainment;
SELECT TRIM(Film_Type), RTRIM(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LEN(Film_Type) AS LEN FROM Cinema_Entertainment;
SELECT REPLACE(Film_Type, 'a', '@') AS REPLACED FROM Cinema_Entertainment;
SELECT CONCAT(Film_Type, ',' ,film_code) AS cinema FROM Cinema_Entertainment ;
```

The results grid shows the output of the last query:

Grid	film_type	cinema
1	Romance	1492
2	Romance	1492
3	Romance	1492
4	Romance	1492
5	Romance	1492
6	Romance	1492
7	Romance	1492
8	Romance	1492
9	Romance	1492
10	Romance	1492
11	Romance	1492
12	Romance	1492
13	Romance	1492

At the bottom, the status bar shows "10:54 PM 10/5/2022".

## CHARINDEX Function

The screenshot shows the DBBeaver interface with the SQL Editor tab selected. The query window contains the following SQL code:

```
SELECT * FROM Cinema_Entertainment WHERE Film_Type LIKE '%e';

--Selecting Using String Functions

SELECT TOP 10 * FROM Cinema_Entertainment ;
SELECT Film_Type, UPPER(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LOWER(Film_Type) FROM Cinema_Entertainment;
SELECT TRIM(Film_Type), RTRIM(Film_Type) FROM Cinema_Entertainment;
SELECT Film_Type, LEN(Film_Type) AS LEN FROM Cinema_Entertainment;
SELECT REPLACE(Film_Type, 'a', '@') AS REPLACED FROM Cinema_Entertainment;
SELECT CONCAT(Film_Type, ',' ,film_code) AS cinema FROM Cinema_Entertainment ;
SELECT CHARINDEX('A',Film_type) AS POSITION ;
```

The results grid shows the output of the last query:

Grid	POSITION	film_code
1	4	304
2	4	352
3	4	489
4	4	429
5	4	524
6	4	71
7	4	163
8	4	450
9	4	51
10	4	522
11	4	43
12	4	529
13	4	82

At the bottom, the status bar shows "11:16 PM 10/5/2022".

## SUBSTRING Function

The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Database Selection:** <sravan> Script-22
- SQL Editor:** Contains a script with various string functions, including `SUBSTRING(Film_Type,3,6) AS SUBSTRINGS`.
- Results Grid:** Shows the output of the query, which is a single column named "SUBSTRINGS" containing the value "mance" repeated 17 times.
- System Bar:** Shows the date and time as 10/05/2022 11:17 PM.

## CASE WHEN Function

The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Database Selection:** <sravan> Script-22
- SQL Editor:** Contains a script with a `CASE WHEN` statement and other string functions.
- Results Grid:** Shows the output of the query, which is a single column named "Film\_Type" containing the value "Romance" repeated 17 times.
- System Bar:** Shows the date and time as 10/05/2022 11:50 PM.

## 4. ANALYTICAL FUNCTIONS

### GRUOP BY AND ORDER BY CLAUSE

DBeaver 22.2.0 - <sravan> Script-22

```
File Edit Navigate Search SQL Editor Database Window Help
Database ... Projects <sravan> Shrawan.sql <dlithe> Dlithesql <sravan> Script-22
Enter a part of object name | 
dltite - rewansolution.database
dltite - postgresql-89441-0.clo ...
oracle-88094-0.cloudclusters.net
sravan - rewansolution.database

SELECT REPLACE(Film_Type, 'a', 'e') AS REPLACED FROM Cinema_Entertainment;
SELECT CONCAT(Film_Type, ' ', cinema_code) AS cinema FROM Cinema_Entertainment ;
SELECT CHARINDEX('A', Film_type) AS POSITIONS , cinema_code FROM Cinema_Entertainment ;
SELECT SUBSTRING(Film_type,3,6) AS SUBSTRINGS FROM Cinema_Entertainment ;

--SELECT Film_Type , CASE WHEN Film_Type='Romance' THEN 10
WHEN Film_Type='Thriller' THEN 5
ELSE cinema_code
END
FROM Cinema_Entertainment ;

--Analytical and Aggregate Functions

SELECT SUM(tickets_sold)AS Total,Film_Type
FROM Cinema_Entertainment
WHERE Film_Type IS NOT NULL
GROUP BY Film_Type
HAVING SUM(tickets_sold)>20000
ORDER BY Film_Type DESC;
```

Results 1 × Output

Total	Film_Type
23,650	Fiction
22,139	Family
26,733	Action

Grid Text Record

Save Cancel Script | 200 | 3 Rows: 1 | 3 row(s) fetched - 1.2s. on 2022-10-05 at 23:55:08  
IST | en | Writable | Smart Insert | 300 : 26 : 9839 | Sel: 0 | 0 | 11:55 PM 10/5/2022

### DENSE RANK Function

DBeaver 22.2.0 - <sravan> Script-22

```
File Edit Navigate Search SQL Editor Database Window Help
Database ... Projects <sravan> Shrawan.sql <dlithe> Dlithesql <sravan> Script-22
Enter a part of object name | 
dltite - rewansolution.database
dltite - postgresql-89441-0.clo ...
oracle-88094-0.cloudclusters.net
sravan - rewansolution.database

END
FROM Cinema_Entertainment ;
-- Analytical and Aggregate Functions

SELECT SUM(tickets_sold)AS Total,Film_Type
FROM Cinema_Entertainment
WHERE Film_Type IS NOT NULL
GROUP BY Film_Type
HAVING SUM(tickets_sold)>20000
ORDER BY Film_Type DESC;

SELECT *, DENSE_RANK() OVER(PARTITION BY Film_Type ORDER BY tickets_sold) AS RANK FROM Cinema_Entertainment;
```

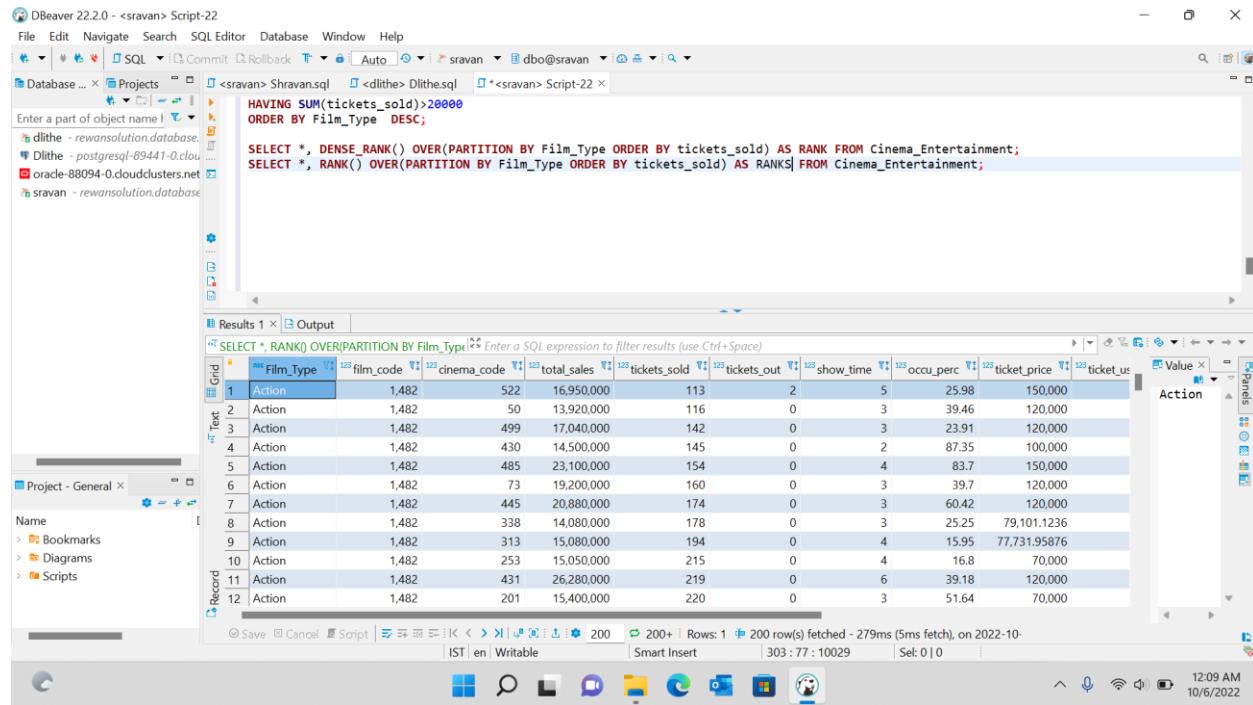
Results 1 × Output

	tickets_out	show_time	occu_perc	ticket_price	ticket_use	capacity	date	month	quarter	day	RANK	
40	1	0	14	9.49	77,250.55432	451	4,752	43,196	4	2	6	39
41	4	0	16	55.43	139,980.1587	504	909	43,196	4	2	6	40
42	6	0	5	27.18	97,430.83004	506	1,862	43,196	4	2	6	41
43	5	0	3	70.55	97,864.07767	515	730	43,196	4	2	6	42
44	8	0	3	78.48	100,000	518	660	43,196	4	2	6	43
45	5	0	5	62.21	70,000	535	860	43,196	4	2	6	44
46	1	0	2	53.99	80,000	541	1,002	43,196	4	2	6	45
47	9	0	3	78.16	53,778.55888	569	728	43,196	4	2	6	46
48	4	0	3	61.88	99,326.59933	594	960	43,196	4	2	6	47
49	5	0	6	18	91,344.53782	595	3,306	43,196	4	2	6	48
50	14	3	4	75.12	37,384.10596	601	804	43,196	4	2	6	49
51	5	0	5	30.96	97,024.79339	605	1,954	43,196	4	2	6	50

Grid Text Record

Save Cancel Script | 200+ | 200+ | Rows: 1 | 200 row(s) fetched - 293ms (6ms fetch), on 2022-10-06 at 12:05 AM  
IST | en | Writable | Smart Insert | 302 : 109 : 9951 | Sel: 0 | 0 | 12:05 AM 10/6/2022

## RANK Function



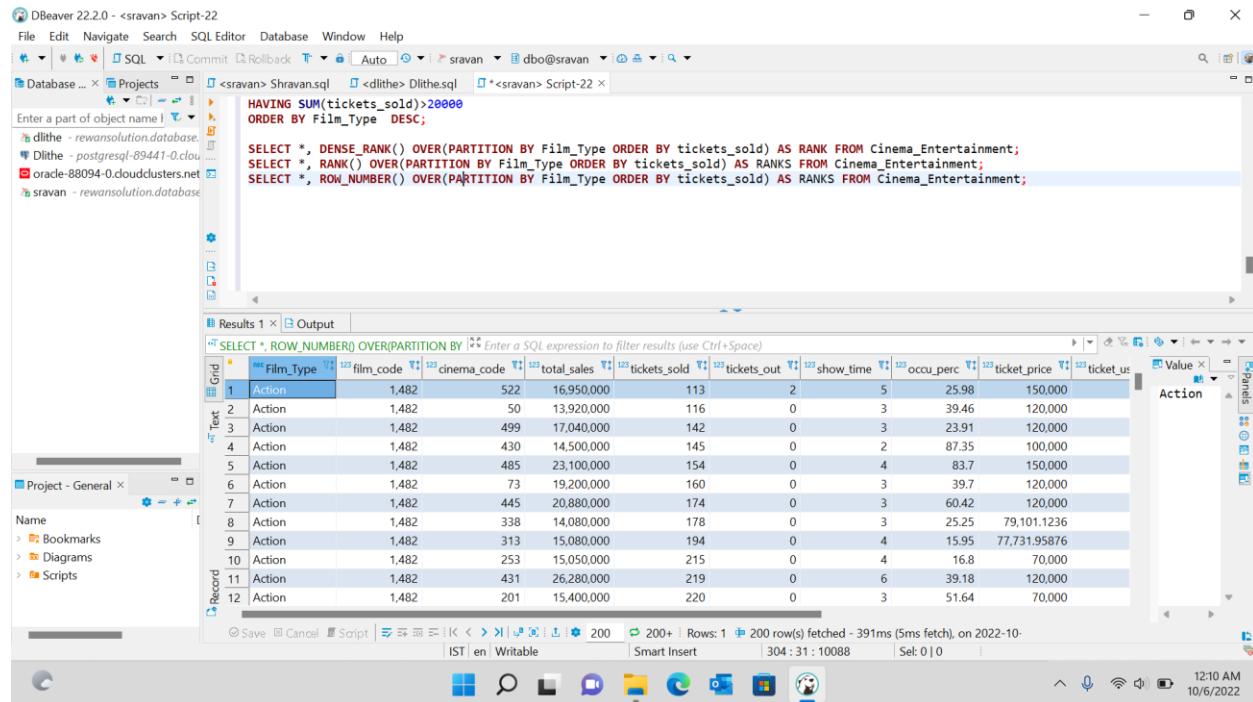
The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Toolbar:** Undo, Redo, Commit, Rollback, Auto, Projects, <sravan>, Shrawan.sql, <dlithe>, Dlithesql, <sravan> Script-22.
- Database Navigator:** Shows databases: dlithe - rewsolutions.database, Dlithesql - postgresql-89441-0.clo..., oracle-88094-0.cloudclusters.net, sravan - rewsolutions.database.
- Script Area:** Contains the following SQL code:

```
HAVING SUM(tickets_sold)>20000
ORDER BY Film_Type DESC,
```

```
SELECT *, DENSE_RANK() OVER(PARTITION BY Film_Type ORDER BY tickets_sold) AS RANK FROM Cinema_Entertainment;
SELECT *, RANK() OVER(PARTITION BY Film_Type ORDER BY tickets_sold) AS RANKS FROM Cinema_Entertainment;
```
- Results Grid:** Displays 12 rows of data for Action films, ordered by tickets\_sold descending. The columns include film\_code, cinema\_code, total\_sales, tickets\_sold, tickets\_out, show\_time, occu\_perc, ticket\_price, and ticket\_us.
- Status Bar:** Shows the query took 279ms, fetched 200 rows, and was run on 2022-10-06 at 12:09 AM.

## ROW NUMBER Function



The screenshot shows the DBBeaver interface with the following details:

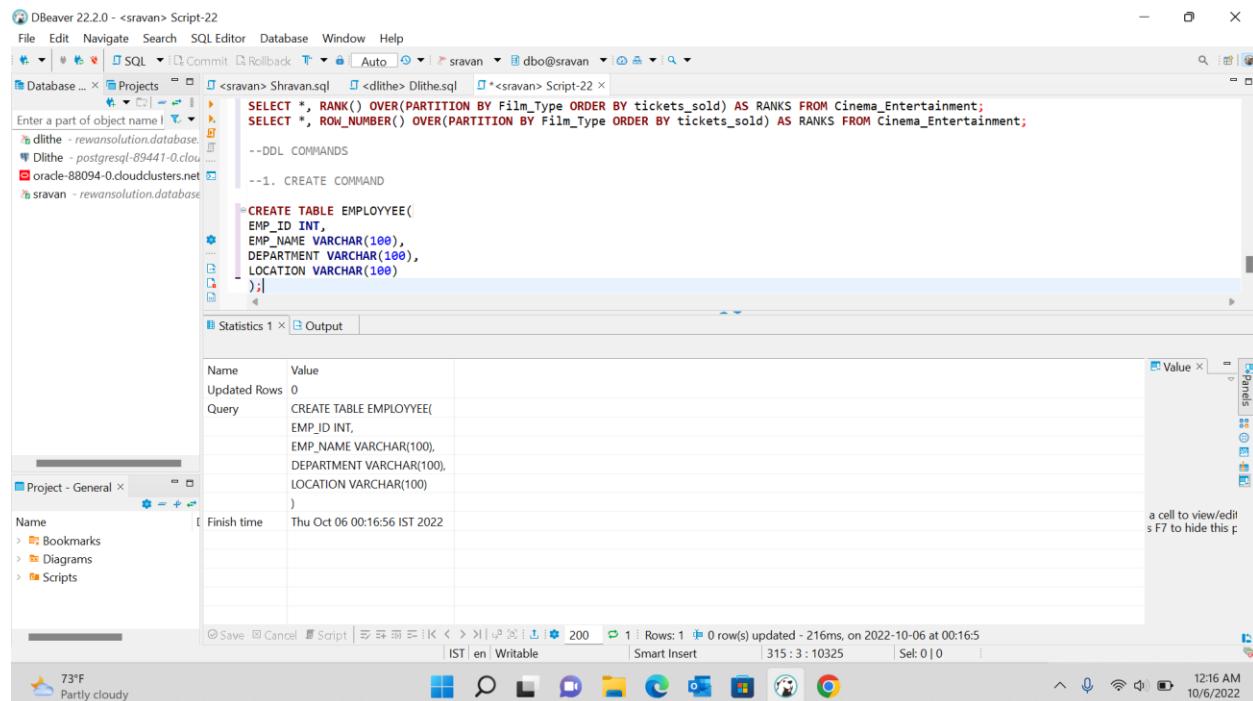
- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Toolbar:** Undo, Redo, Commit, Rollback, Auto, Projects, <sravan>, Shrawan.sql, <dlithe>, Dlithesql, <sravan> Script-22.
- Database Navigator:** Shows databases: dlithe - rewsolutions.database, Dlithesql - postgresql-89441-0.clo..., oracle-88094-0.cloudclusters.net, sravan - rewsolutions.database.
- Script Area:** Contains the following SQL code:

```
HAVING SUM(tickets_sold)>20000
ORDER BY Film_Type DESC,
```

```
SELECT *, DENSE_RANK() OVER(PARTITION BY Film_Type ORDER BY tickets_sold) AS RANK FROM Cinema_Entertainment;
SELECT *, RANK() OVER(PARTITION BY Film_Type ORDER BY tickets_sold) AS RANKS FROM Cinema_Entertainment;
SELECT *, ROW_NUMBER() OVER(PARTITION BY Film_Type ORDER BY tickets_sold) AS RANKS FROM Cinema_Entertainment;
```
- Results Grid:** Displays 12 rows of data for Action films, ordered by tickets\_sold descending. The columns include film\_code, cinema\_code, total\_sales, tickets\_sold, tickets\_out, show\_time, occu\_perc, ticket\_price, and ticket\_us.
- Status Bar:** Shows the query took 391ms, fetched 200 rows, and was run on 2022-10-06 at 12:10 AM.

## 5. DDL AND DML COMMANDS

### CREATE Command



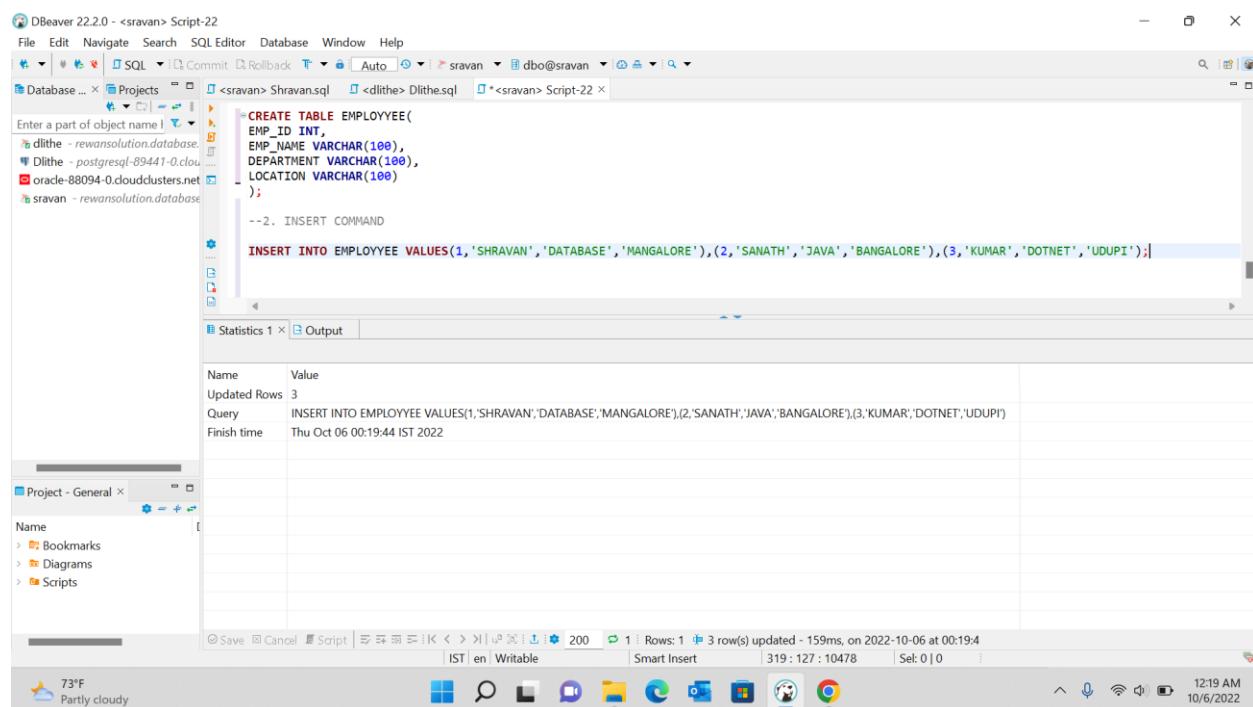
The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Toolbar:** Commit, Rollback, Auto, sravan, dbo@sravan, etc.
- Left Panel:** Project - General, showing Bookmarks, Diagrams, and Scripts.
- Central Area:** SQL tab, showing the creation of the EMPLOYEE table.

```
CREATE TABLE EMPLOYEE(
    EMP_ID INT,
    EMP_NAME VARCHAR(100),
    DEPARTMENT VARCHAR(100),
    LOCATION VARCHAR(100)
);
```
- Output Tab:** Shows the execution results:

Name	Value
Updated Rows	0
Query	CREATE TABLE EMPLOYEE( EMP_ID INT, EMP_NAME VARCHAR(100), DEPARTMENT VARCHAR(100), LOCATION VARCHAR(100) )
Finish time	Thu Oct 06 00:16:56 IST 2022
- Bottom Status Bar:** 73°F Partly cloudy, 12:16 AM 10/6/2022.

### INSERT Command



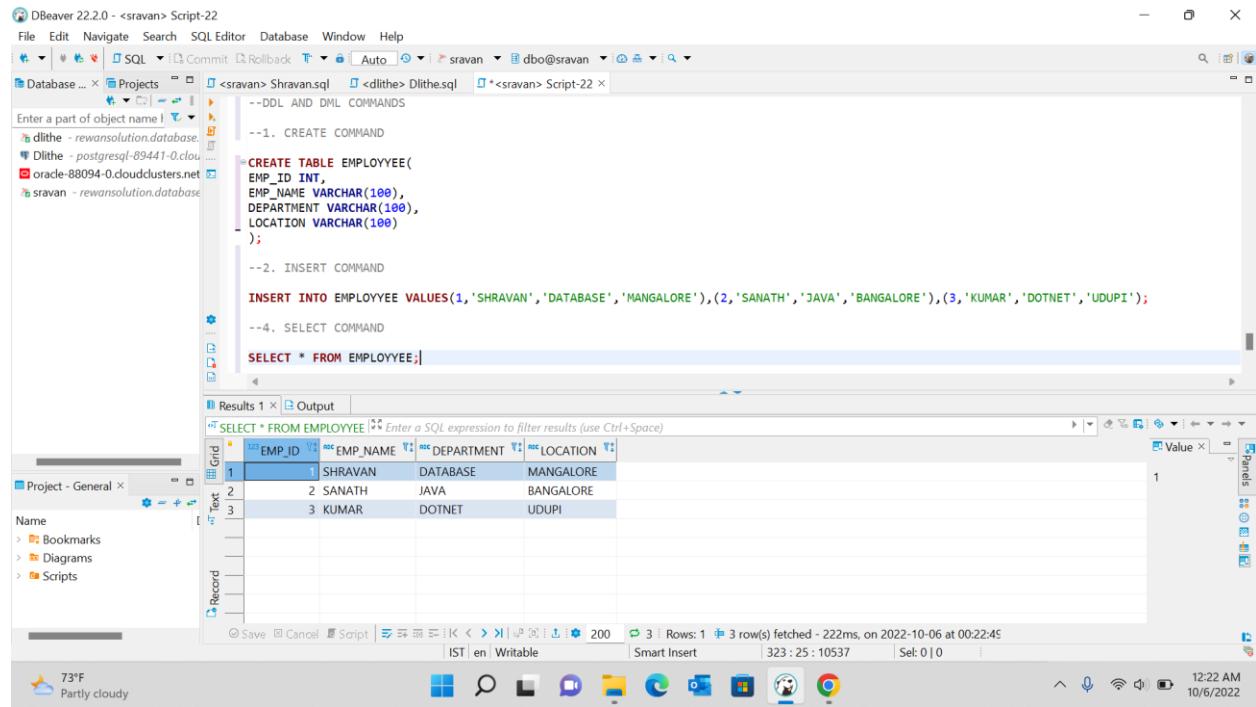
The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Toolbar:** Commit, Rollback, Auto, sravan, dbo@sravan, etc.
- Left Panel:** Project - General, showing Bookmarks, Diagrams, and Scripts.
- Central Area:** SQL tab, showing the insertion into the EMPLOYEE table.

```
INSERT INTO EMPLOYEE VALUES(1,'SHRAVAN','DATABASE','MANGALORE'),(2,'SANATH','JAVA','BANGALORE'),(3,'KUMAR','DOTNET','UDUPI');
```
- Output Tab:** Shows the execution results:

Name	Value
Updated Rows	3
Query	INSERT INTO EMPLOYEE VALUES(1,'SHRAVAN','DATABASE','MANGALORE'),(2,'SANATH','JAVA','BANGALORE'),(3,'KUMAR','DOTNET','UDUPI')
Finish time	Thu Oct 06 00:19:44 IST 2022
- Bottom Status Bar:** 73°F Partly cloudy, 12:19 AM 10/6/2022.

## SELECT Command



The screenshot shows the DBBeaver interface with the following details:

- SQL Editor:** Contains the following SQL code:

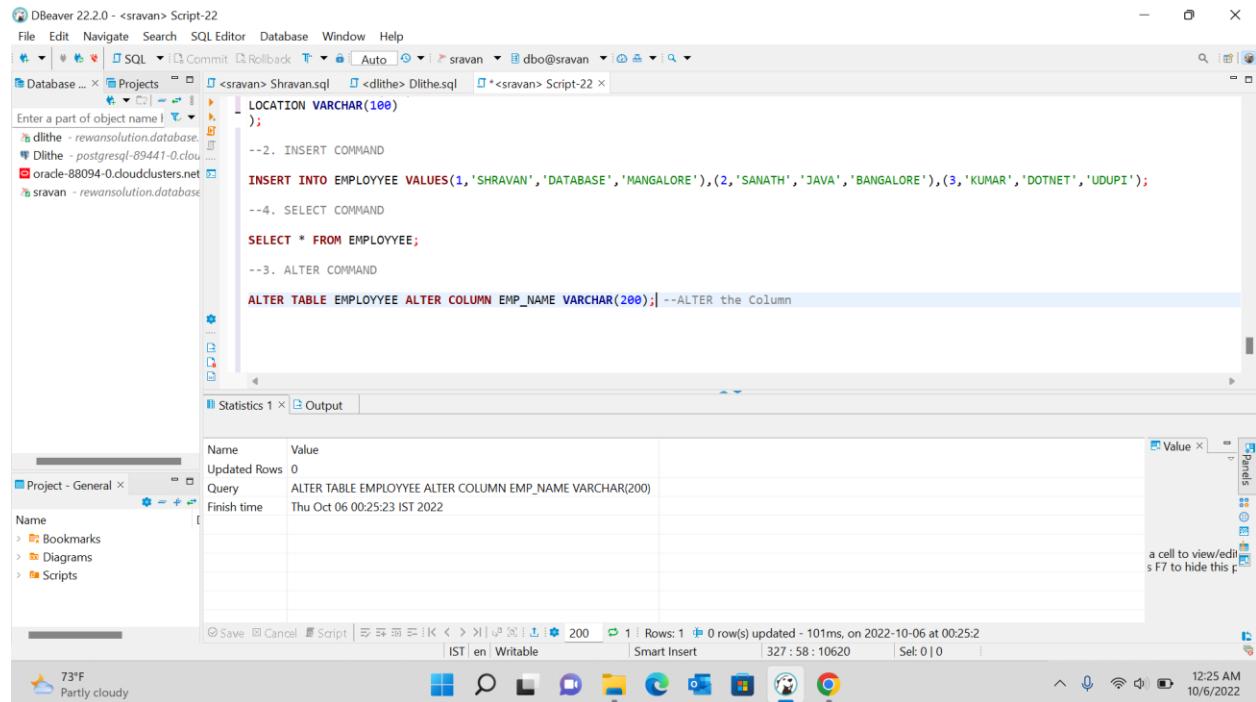
```
--1. CREATE COMMAND
CREATE TABLE EMPLOYEE(
    EMP_ID INT,
    EMP_NAME VARCHAR(100),
    DEPARTMENT VARCHAR(100),
    LOCATION VARCHAR(100)
);

--2. INSERT COMMAND
INSERT INTO EMPLOYEE VALUES(1,'SHRAVAN','DATABASE','MANGALORE'),(2,'SANATH','JAVA','BANGALORE'),(3,'KUMAR','DOTNET','UDUPI');

--4. SELECT COMMAND
SELECT * FROM EMPLOYEE;
```
- Results Grid:** Displays the results of the SELECT query:

EMP_ID	EMP_NAME	DEPARTMENT	LOCATION
1	SHRAVAN	DATABASE	MANGALORE
2	SANATH	JAVA	BANGALORE
3	KUMAR	DOTNET	UDUPI
- Statistics:** Shows 3 rows fetched in 222ms.
- System Bar:** Includes weather (73°F Partly cloudy), date (10/6/2022), and time (12:22 AM).

## Altering Column



The screenshot shows the DBBeaver interface with the following details:

- SQL Editor:** Contains the following SQL code:

```
LOCATION VARCHAR(100)
);

--2. INSERT COMMAND
INSERT INTO EMPLOYEE VALUES(1,'SHRAVAN','DATABASE','MANGALORE'),(2,'SANATH','JAVA','BANGALORE'),(3,'KUMAR','DOTNET','UDUPI');

--4. SELECT COMMAND
SELECT * FROM EMPLOYEE;

--3. ALTER COMMAND
ALTER TABLE EMPLOYEE ALTER COLUMN EMP_NAME VARCHAR(200); --ALTER the Column
```
- Statistics:** Shows 0 updated rows, Query: ALTER TABLE EMPLOYEE ALTER COLUMN EMP\_NAME VARCHAR(200), and Finish time: Thu Oct 06 00:25:23 IST 2022.
- System Bar:** Includes weather (73°F Partly cloudy), date (10/6/2022), and time (12:25 AM).

## Adding Column

The screenshot shows the DBBeaver interface with the following details:

- SQL Editor:** Contains the following SQL code:

```
LOCATION VARCHAR(100);
);
--2. INSERT COMMAND
INSERT INTO EMPLOYEE VALUES(1,'SHRAVAN','DATABASE','MANGALORE'),(2,'SANATH','JAVA','BANGALORE'),(3,'KUMAR','DOTNET','UDUPI');

--4. SELECT COMMAND
SELECT * FROM EMPLOYEE;

--3. ALTER COMMAND
ALTER TABLE EMPLOYEE ALTER COLUMN EMP_NAME VARCHAR(200); --ALTER the Column
ALTER TABLE EMPLOYEE ADD SALARY FLOAT;--Add new Column
```
- Results Grid:** Shows the data from the EMPLOYEE table after the column addition.

EMP_ID	EMP_NAME	DEPARTMENT	LOCATION	SALARY
1	SHRAVAN	DATABASE	MANGALORE	[NULL]
2	SANATH	JAVA	BANGALORE	[NULL]
3	KUMAR	DOTNET	UDUPI	[NULL]
- System Bar:** Displays the date and time as 10/6/2022 12:26 AM.

## Dropping Column

The screenshot shows the DBBeaver interface with the following details:

- SQL Editor:** Contains the following SQL code:

```
);
--2. INSERT COMMAND
INSERT INTO EMPLOYEE VALUES(1,'SHRAVAN','DATABASE','MANGALORE'),(2,'SANATH','JAVA','BANGALORE'),(3,'KUMAR','DOTNET','UDUPI');

--4. SELECT COMMAND
SELECT * FROM EMPLOYEE;

--3. ALTER COMMAND
ALTER TABLE EMPLOYEE ALTER COLUMN EMP_NAME VARCHAR(200); --ALTER the Column
ALTER TABLE EMPLOYEE ADD SALARY FLOAT;--Add new Column
ALTER TABLE EMPLOYEE DROP COLUMN SALARY;-- DELETE a Row from Table
```
- Results Grid:** Shows the data from the EMPLOYEE table after the column deletion.

EMP_ID	EMP_NAME	DEPARTMENT	LOCATION	SALARY
1	SHRAVAN	DATABASE	MANGALORE	
2	SANATH	JAVA	BANGALORE	
3	KUMAR	DOTNET	UDUPI	
- System Bar:** Displays the date and time as 10/6/2022 12:28 AM.

## DELETE Command

The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Toolbar:** Undo, Redo, Commit, Rollback, Auto, Projects, <sravan> Shravan.sql, <dlithe> Shravan.sql, <sravan> Script-22.
- Script Editor:** Contains the following SQL code:

```
INSERT INTO EMPLOYEE VALUES(1,'SHRAVAN','DATABASE','MANGALORE'),(2,'SANATH','JAVA','BANGALORE'),(3,'KUMAR','DOTNET','UDUPI');

--4. SELECT COMMAND

SELECT * FROM EMPLOYEE;

--3. ALTER COMMAND

ALTER TABLE EMPLOYEE ALTER COLUMN EMP_NAME VARCHAR(200); --ALTER the Column

ALTER TABLE EMPLOYEE ADD SALARY FLOAT; --Add new Column

ALTER TABLE EMPLOYEE DROP COLUMN SALARY; -- DELETE a Row from Table

--4. DELETE COMMAND
DELETE FROM EMPLOYEE WHERE EMP_ID=1; --DELETE a Record
```
- Results Grid:** Shows the initial state of the EMPLOYEE table with two rows:

EMP_ID	EMP_NAME	DEPARTMENT	LOCATION
1	SANATH	JAVA	BANGALORE
2	KUMAR	DOTNET	UDUPI
- Status Bar:** Shows the system tray, battery level (73°F), and system information (12:30 AM, 10/6/2022).

## UPDATE Command

The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Toolbar:** Undo, Redo, Commit, Rollback, Auto, Projects, <sravan> Shravan.sql, <dlithe> Shravan.sql, <sravan> Script-22.
- Script Editor:** Contains the following SQL code:

```
SELECT * FROM EMPLOYEE;

--3. ALTER COMMAND

ALTER TABLE EMPLOYEE ALTER COLUMN EMP_NAME VARCHAR(200); --ALTER the Column

ALTER TABLE EMPLOYEE ADD SALARY FLOAT; --Add new Column

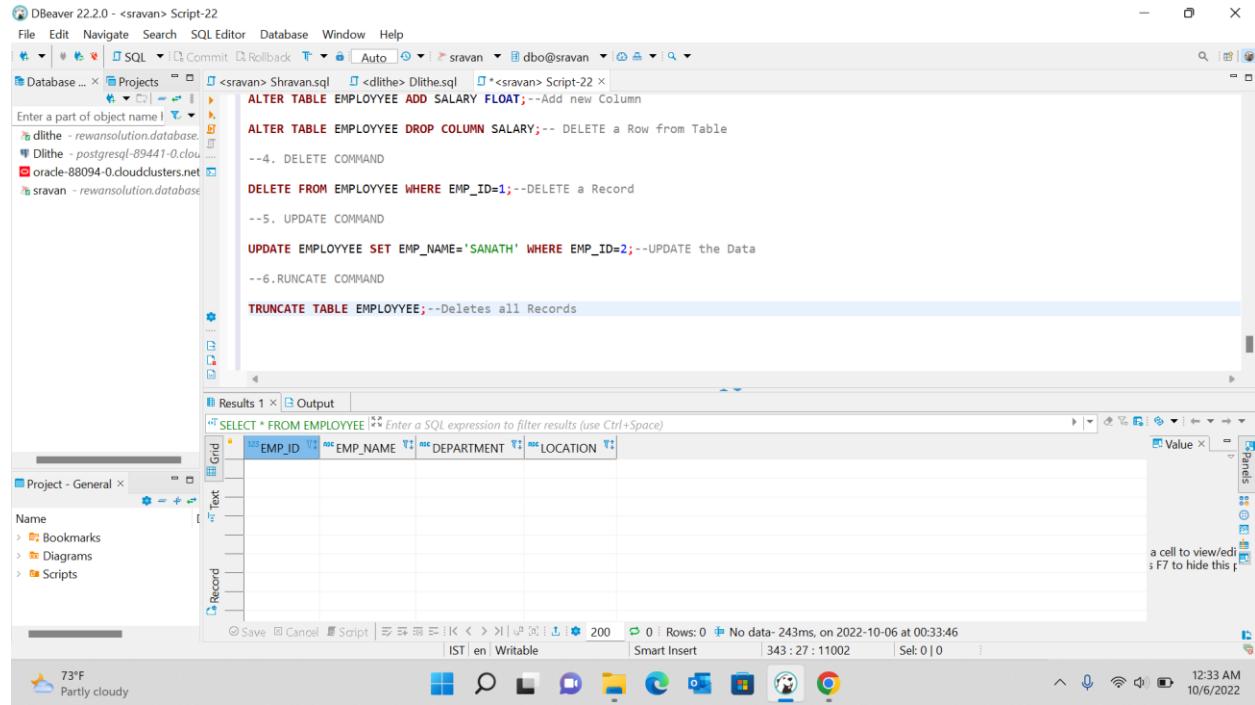
ALTER TABLE EMPLOYEE DROP COLUMN SALARY; -- DELETE a Row from Table

--4. DELETE COMMAND
DELETE FROM EMPLOYEE WHERE EMP_ID=1; --DELETE a Record

--5. UPDATE COMMAND
UPDATE EMPLOYEE SET EMP_NAME='SANATH' WHERE EMP_ID=2; --UPDATE the Data
```
- Results Grid:** Shows the state of the EMPLOYEE table after the update:

EMP_ID	EMP_NAME	DEPARTMENT	LOCATION
1	SANATH	JAVA	BANGALORE
2	SANATH	DOTNET	UDUPI
- Status Bar:** Shows the system tray, battery level (73°F), and system information (12:32 AM, 10/6/2022).

## TRUNCATE Command

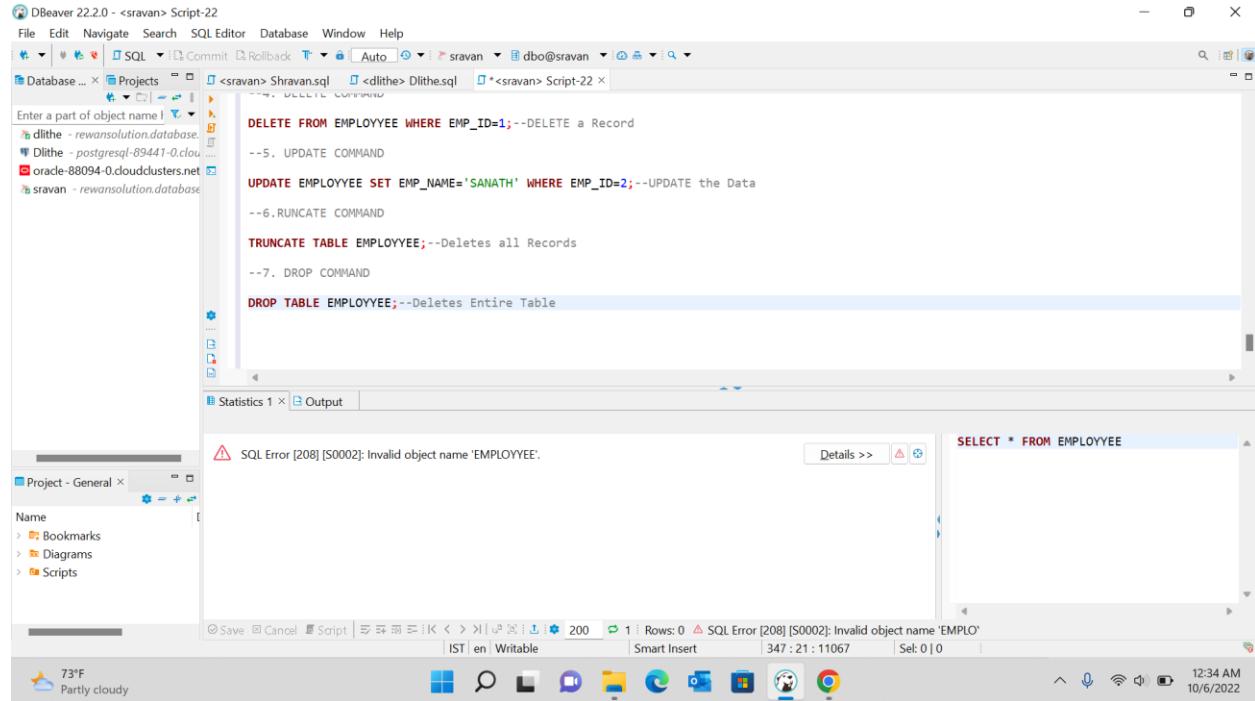


The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Toolbar:** Undo, Redo, Commit, Rollback, Auto, sravan, dbo@sravan, Search, Find, Replace.
- Project Explorer:** Shows databases like dltlhe, postgresql-89441-0, oracle-88094-0.cloudclusters.net, and sravan.
- SQL Editor:** Contains a script named "Script-22" with the following content:

```
ALTER TABLE EMPLOYEE ADD SALARY FLOAT;--Add new Column
ALTER TABLE EMPLOYEE DROP COLUMN SALARY;-- DELETE a Row from Table
--4. DELETE COMMAND
DELETE FROM EMPLOYEE WHERE EMP_ID=1;--DELETE a Record
--5. UPDATE COMMAND
UPDATE EMPLOYEE SET EMP_NAME='SANATH' WHERE EMP_ID=2;--UPDATE the Data
--6. RUNCATE COMMAND
TRUNCATE TABLE EMPLOYEE;--Deletes all Records
```
- Results Grid:** Shows a table with columns: EMP\_ID, EMP\_NAME, DEPARTMENT, LOCATION. The grid is currently empty.
- Status Bar:** Shows "Rows: 0" and "No data- 243ms, on 2022-10-06 at 00:33:46".
- System Bar:** Shows weather (73°F Partly cloudy), date (10/6/2022), and time (12:33 AM).

## DROP Command



The screenshot shows the DBBeaver interface with the following details:

- File Bar:** File, Edit, Navigate, Search, SQL Editor, Database, Window, Help.
- Toolbar:** Undo, Redo, Commit, Rollback, Auto, sravan, dbo@sravan, Search, Find, Replace.
- Project Explorer:** Shows databases like dltlhe, postgresql-89441-0, oracle-88094-0.cloudclusters.net, and sravan.
- SQL Editor:** Contains a script named "Script-22" with the following content:

```
DELETE FROM EMPLOYEE WHERE EMP_ID=1;--DELETE a Record
--5. UPDATE COMMAND
UPDATE EMPLOYEE SET EMP_NAME='SANATH' WHERE EMP_ID=2;--UPDATE the Data
--6. RUNCATE COMMAND
TRUNCATE TABLE EMPLOYEE;--Deletes all Records
--7. DROP COMMAND
DROP TABLE EMPLOYEE;--Deletes Entire Table
```
- Statistics View:** Shows an error message: "SQL Error [208] [S0002]: Invalid object name 'EMPLOYEE'." with a "Details >>" link.
- Output View:** Shows the query "SELECT \* FROM EMPLOYEE".
- Status Bar:** Shows "Rows: 0" and "SQL Error [208] [S0002]: Invalid object name 'EMPLOYEE'".
- System Bar:** Shows weather (73°F Partly cloudy), date (10/6/2022), and time (12:34 AM).